

How to VIM

Eine kurze Einführung in den Texteditor vim

Marcel Schneider

marcel.schneider@stud.uni-goettingen.de

<https://t.me/teleschneider>

Georg-August-Universität Göttingen

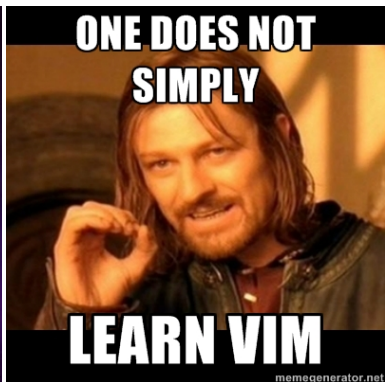
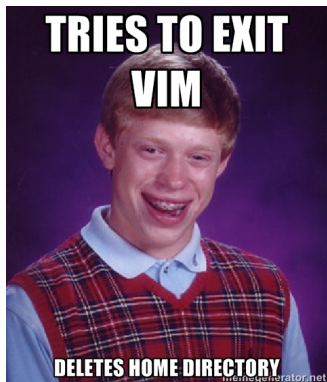
Info-O-Phase 2019

Was ist das?

Vi Improved

- ▶ 1991 erstmals veröffentlicht
- ▶ Erweiterung von *vi* (1976), welcher auf *ed* (1971) aufbaute
- ▶ Üblicherweise kein GUI, arbeitet im Terminal
- ▶ Kenntnisse in der Shell vorteilhaft
- ▶ Ungewöhnliche, aber hocheffiziente Eingabemöglichkeiten
- ▶ Bietet Vielzahl an Funktionen, per Plugins noch erweiterbar
- ▶ Quelle vieler Memes

Mememes



Memes



I Am Devloper

@iamdeveloper

Follow



I've been using Vim for about 2 years now, mostly because I can't figure out how to exit it.

3:26 PM - 17 Feb 2014

14,015 Retweets 8,698 Likes









314

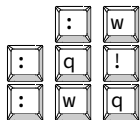
14K

8.7K

Vim beenden

Mit folgendem Kommando wird Vim beendet  
(Oder    )

Weitere Wichtige Kommandos



Speichern

Schließen ohne speichern

Speichern und schließen

Modi in Vim

Es gibt im Wesentlichen drei verschiedene Modi:





1. Normal mode
2. Insert mode
3. Visual mode

Normal mode

- ▶ Standardmodus, Vim startet mit diesem
- ▶ Alle Eingaben werden als Kommandos interpretiert
- ▶ Wird per `[Esc]` aus anderen Modi erreicht

```
1 \documentclass[xetex]{beamer}-  
2 -  
3 \usepackage[ngerman]{babel}-  
4 \usepackage[hyperref]-  
5 \usepackage[default]{sourcesanspro}-  
6 \usepackage[graphicx]-  
7 \usepackage[keystroke]-  
8 -  
9 %\setmainfont{Source Serif Pro}-  
10 -  
11 %Information to be included in the title page:-  
12 \title{How to Vim}-  
13 \subtitle{Eine kurze Einführung in den Texteditor vim}-  
14 \author{Marcel Schneider\(\marcel@webschneider.org)}-  
15 \institute{Georg-August-Universität Göttingen}-  
16 \date{Info-0-Phase 2018}-  
17 -  
18 \begin{document}-  
19 -  
20 \frame{\titlepage}-  
21 -  
22 \begin{frame}-  
23   \frametitle{Was ist das?}-  
24   \emph{V}i \emph{I}mproved-  
25   \begin{itemize}-  
26     \item 1991 erstmals veröffentlicht-  
27     \item Erweiterung von \emph{vi} (1976), welcher auf \emph{ed} (1971)-  
28       aufbaute-  
29     \item Üblicherweise kein GUI, arbeitet im Terminal-  
30     \item Ungewöhnliche, aber hocheffiziente Eingabemöglichkeiten-  
31     \item Bietet Vielzahl an Funktionen, per Plugins noch erweiterbar-  
NORMAL > SPELL [DE] > how-to-vim.tex          tex > utf-8[unix] 145 words < 31% 22/69 : 1  
~/git/unl/vim-workshop/how-to-vim.tex* 69L, 1930C  
main0 v 1 ..-workshop/img 2 ../vim-workshop          57% 19:57 26 Sep klappschneider
```




Insert mode

- ▶ Eingabemodus, „normales“ Verhalten, wie aus anderen Editoren bekannt
- ▶ Wird im Normal mode per  /  oder  /  erreicht

```
53 \begin{frame}{Modi in Vim}-  
54   Es gibt im Wesentlichen drei verschiedene Modi:-  
55   \begin{enumerate}-  
56     \item Normal mode-  
57     \item Insert mode-  
58     \item Visual mode-  
59   \end{enumerate}-  
60 \end{frame}-  
61 -  
62 \begin{frame}{Normal mode}-  
63   \begin{itemize}-  
64     \item Standardmodus, Vim startet mit diesem-  
65     \item Alle Eingaben werden als Kommandos interpretiert-  
66     \item Wird per \Esc_ aus anderen Modi erreicht-  
67   \end{itemize}-  
68 -  
69   \begin{center}-  
70     \includegraphics[width=0.9\textwidth]{img/normal-mode.png}-  
71   \end{center}-  
72 \end{frame}-  
73 -  
74 \begin{frame}{Insert mode}-  
75   \begin{itemize}-  
76     \item Eingabemodus, \enquote(normales) Verhalten, wie aus anderen-  
77       Editoren bekannt-  
78     \item Wird im Normal mode per \keystroke(i)/\keystroke(I) oder-  
79       \keystroke(a)/\keystroke(A) erreicht-  
80   \end{itemize}-  
81 \end{frame}-  
82 \end{document}-
```

INSERT > SPELL [DE] how-to-vim.tex[*] tex > [c] utf-8[unix] 181 words < 96% 79/82 : 46 W:1(L66)
-- INSERT --
main@ v 1 ...workshop/lng 2 ../vim-workshop 56% 20:06 26 Sep klappsneider



Visual Mode

- ▶ In diesem Modus kann Text selektiert werden
- ▶ Selektion kann bearbeitet/gelöscht werden, oder als Argument für ein Kommando dienen
- ▶ Funktioniert Zeilen- und Blockweise
- ▶ Aufruf per  / 

```
70 ~
71   \begin{center}~
72     \includegraphics[width=0.9\textwidth]{img/normal-mode.png}~
73   \end{center}~
74 \end{frame}~
75 ~
76 \begin{frame}[Insert mode]~
77   \begin{itemize}~
78     \item Eingabenodus, \enquote{normales} Verhalten, wie aus anderen~
79         Editoren bekannt~
80     \item Wird in Normal mode per \kbd{I}/\kbd{I} oder~
81         \kbd{a}/\kbd{A} erreicht~
82   \end{itemize}~
83 ~
84   \begin{center}~
85     \includegraphics[width=0.9\textwidth]{img/insert-mode.png}~
86   \end{center}~
87 \end{frame}~
88 ~
89 \begin{frame}[Visual Mode]~
90   \begin{itemize}~
91     \item In diesem Modus kann Text selektiert werden~
92     \item Selektion kann bearbeitet/gelöscht werden, oder als Argument für~
93         ein Kommando dienen~
94     \item Funktioniert Zeilen- und Blockweise~
95   \end{itemize}~
96 ~
97   \begin{center}~
98     \includegraphics[width=0.9\textwidth]{img/insert-mode.png}~
99   \end{center}~
100 \end{frame}~
```

V-LINE > SPELL [DE] > how-to-vim.tex[+] < tex > {c} utf-8[unix] 25 words < 90% 91/101 : 49 W:1(L68) 4
-- VISUAL LINE --
main0 v 1 ...workshop/Img 2 ../vim-workshop 54% 20:17 26 Sep klappschnelder

Protipp: Esc-Taste

Tauscht die Belegung von  und , z.B. mit folgendem Eintrag in `~/ .profile` oder `~/ .bashrc`:

```
xmodmap -e 'clear Lock' -e 'keycode 0x42 = Escape'
```

Oder per dconf-editor in Unity, unter `org/gnome/desktop/input-sources` bei `xkb-options` `'caps:swapescape'` einfügen.

Bewegung im Normal Mode

Der Cursor wird nicht mit den Pfeiltasten bewegt, sondern:



Nach links



Nach unten



Nach oben

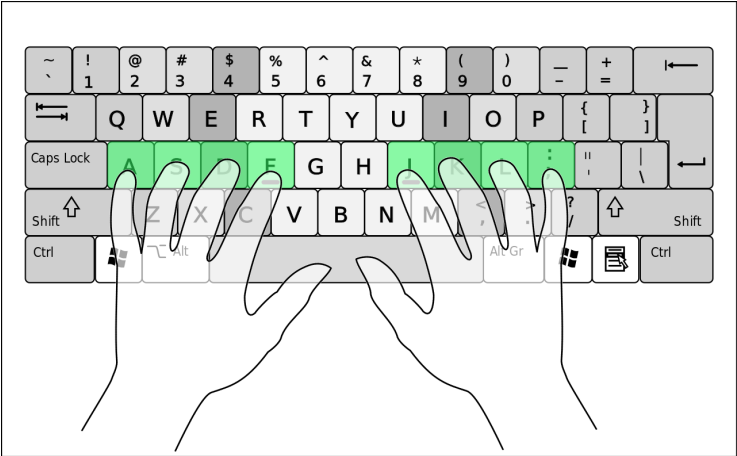


Nach rechts

Herkunft der hjkl-Tasten

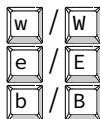


Homerow



Text Objects — Bewegung

Dateiinhalte wird nicht als Abfolge einzelner Zeichen betrachtet, sondern als „Wörter“:



Anfang eines Wortes/WORTES

Ende eines Wortes/WORTES





Vorheriges Wort/WORT

Vim-Spiel



<https://vim-adventures.com/>

Text Objects — Selektion




Operatoren:

	Löschen (delete)
	Kopieren (yank)
	Ändern (change)
	Markieren (visual)

















Modifikatoren:

	Alles, mit Leerzeichen (all)
	Inhalt, ohne Leerzeichen (inner)


Bewegung (motion):

 / 	Wort (ohne Satzzeichen)/WORT (mit Satzzeichen)
	Satz

Text Objects — Selektion — Beispiele

- ▶    : change all word
- ▶    : change all WORD
- ▶    : delete in {}
- ▶    : visual all ()
- ▶     : 3 × delete all word

Wiederholung von Kommandos

Per  kann das letzte Kommando wiederholt werden

Anpassung/Einstellungen

- ▶ Vim nutzt `~/.vimrc`
- ▶ Neovim nutzt `~/.config/nvim/init.vim`
- ▶ Beide nutzen selbe, simple Syntax

Unter

<https://git.schneider-hosting.de/uni/vim-workshop>
findet ihr eine Beispieldatei für den Anfang