

Einführung in Emacs

Uwe Ziegenhagen

12. Juli 2017

Inhaltsverzeichnis

1	Geschichtliches	1
2	Grundlagen	3
2.1	Installation unter Windows	3
2.2	Installation unter Linux	5
2.3	Installation unter Mac OS X	5
2.4	Deinstallation	5
3	Grundlagen der Bedienung	7
3.1	Von Buffern und anderen Dingen...	7
3.2	Starten und Beenden	7
3.3	Laden und Speichern	11
3.4	Bewegen in der Datei und zwischen Buffern	12
3.5	Editieren von Text	13
3.6	Suchen und Ersetzen	13
3.7	Das eingebaute Hilfesystem	13
3.8	Zusammenfassung	13
4	Konfiguration	15
4.1	Manuelle Konfiguration von Paketen	15
4.2	Das Emacs-Paketsystem	15
4.3	use-package	15
5	Org Mode	17
6	AucT_EX	19
7	Was sonst noch geht...	21
7.1	ccrypt	21
7.2	GraphViz	21
7.3	Emacs als E-Mail-Programm	21
7.3.1	Emacs als RSS-Reader	21
7.4	Sweave	21
7.5	Spielen im Emacs	21

8	Programmierung	23
8.0.1	Das „Hello Emacs“-Beispiel	23
9	Weitere nützliche Software-Tools	25
9.1	TEX/L ^A T _E X	25
9.1.1	Das erste L ^A T _E X-Dokument	25
9.1.2	Eine Musterpräsentation mit Beamer	25
9.1.3	Ein Musterbrief mit <code>scrlttr2</code>	26
9.1.4	Wer mehr wissen möchte...	26
9.2	sed und awk	26
9.3	VI(M)	26

Abbildungsverzeichnis

3.1	Emacs 25.2 nach dem Start	10
3.2	Emacs 25.2 nach dem Start im Kommandozeilenmodus	11

Tabellenverzeichnis

3.1	Befehle zum Bewegen innerhalb des Buffers	13
-----	---	----

Liste der noch zu erledigenden Punkte

Link	4
ausprobieren auf dem Macbook	5
Link	25

Vorwort

Der vorliegende Text ist aus der Erkenntnis entstanden, dass es anscheinend kein aktuelles Emacs-Buch gibt, das die Entwicklungen der letzten Jahre im Emacs-Universum behandelt. Beschreibungen von `use-package` und anderen Paketen finden sich oft nur online und meist auch nur auf Englisch.

Das Skript soll daher eine Einführung in Emacs und nützliche Emacs-Pakete geben. Der Fokus liegt dabei auf Windows, da ich persönlich mit Emacs hauptsächlich unter Windows arbeite, sofern möglich werden aber auch die entsprechenden Hinweise für Linux und Mac OS X gegeben.

Es können nicht alle möglichen Befehle und Befehlskombinationen vorgestellt werden, die Emacs und die verschiedenen Emacs-Pakete bieten, dazu sind es schlichtweg zu viele. Vielmehr möchte ich die wichtigsten Befehle zeigen, die man im Alltag benötigt.

Köln, den 12. Juli 2017

Uwe Ziegenhagen

Konventionen

Folgende Konventionen gelten für den Text:

Kursiv Neue Begriffe, Dateinamen und Dateierweiterungen werden *kursiv* dargestellt.

`nichtproportional` Nichtproportionale Schrift, also „Schreibmaschinenschrift“, wird für Code-Listings benutzt.

nichtproportional fett Nichtproportionale fette Schrift wird für Befehle genutzt, die der Leser über die Tastatur eingibt.

nichtproportional kursiv Nichtproportionale kursive Schrift wird für Werte genutzt, die vom Nutzer vorzugeben sind oder für Werte, die sich aus dem Kontext ergeben.

Mittels `[Strg]` + `[X]` werden Tastensequenzen dargestellt, das „+“ bedeutet hier, dass die `[Strg]`-Taste gedrückt sein muss, während man „X“ drückt.

Zusatzinformationen, die mit Emacs eher am Rand zu tun haben, werden in grauen Boxen dargestellt.

Die Technik hinter dem Text

Dieses Buch ist in \LaTeX gesetzt worden, dem freien Satzprogramm, das ich jedem ans Herz legen möchte, der effizient komplexe oder längere Texte verfässt. Zur Verwaltung der Dateien nutze ich github (<https://www.github.com>), das ich mittels *TortoiseSVN* bediene.

Als Schriftart wird die freie Linux Libertine-Schrift¹ genutzt, für die Darstellung der Tastenkombinationen die Biolinum Keycaps.

¹<http://www.linuxlibertine.org/>

1 Geschichtliches

Emacs, die Abkürzung steht für „Editor MACroS“¹, war ursprünglich kein eigenständiger Editor, sondern lediglich eine Sammlung von Makros für TECO. TECO, was ursprünglich „Tape Editor and COrrector“ stand, später jedoch zu „Text Editor and COrrector“ wurde, wurde 1962/63 als Editor für Lochstreifen entwickelt. Die Bedienung von TECO war dabei nicht auf Benutzerfreundlichkeit ausgelegt, es galt viel mehr, mit möglichst wenigen Tastendrücken den Editor zu steuern. Ebenso wie Emacs gibt es auch TECO noch heute, unter <http://almy.us/teco.html> kann der interessierte Leser Binaries für Windows, Linux und Mac OS X herunterladen.

1972 begann dann die Entwicklung von Emacs² am MIT³, als Carl Mikkelson TECO um Funktionen zur Anzeige von Textänderungen auf dem Bildschirm erweiterte. Was heute als selbstverständlich gilt, war damals keineswegs so: die übliche Art, editierte Text zu sehen zu bekommen, war die Ausgabe auf einem angeschlossenen Drucker.

Im Jahr 1974 schuf Richard Stallman dann die Möglichkeit, Makros in TECO auszuführen, diese Möglichkeit wurde von den TECO-Nutzern am MIT auch rege genutzt. Die erstellten Makros wurden von Richard Stallman 1976 gesammelt und um Funktionen zur Selbstdokumentation und Erweiterbarkeit ergänzt. TECOEmacs wurde anschließend zum Standard-Editor auf den ITS-Systemen⁴ Maschinen des MIT.

In den folgenden Jahren entstanden verschiedene Emacs-Derivate, von denen MulticsEmacs von Bernard Greenberg erwähnenswert ist. MultiEmacs war in LISP geschrieben, auch Erweiterungen waren in LISP geschrieben. Die Wahl von LISP bot einfachere Erweiterungsmöglichkeiten als je zuvor und wurde daher von den meisten folgenden Emacs-Generationen genutzt.

1984 erblickte GNU Emacs das Licht der Welt, als erstes Produkt der GNU Software Foundation. Geschrieben von Richard Stallman in C nutzte er EmacsLisp als Sprache für Erweiterungen. Die erste weithin verfügbare Version war dann Emacs 15.34, die 1985 erschien und bald der Standard für Emacs unter Unix war.

¹nicht für „Escape-Meta-Alt-Control-Shift“

²siehe <https://www.emacswiki.org/emacs/EmacsHistory>

³Massachusetts Institute of Technology

⁴„Incompatible TimeSharing System“

Wissen: Richard Stallman, GNU, und die Free Software Foundation

Richard Stallman (* 16.03.1953) ist ein US-amerikanischer Programmierer und Aktivist, der sich für die Freiheit von Software ein. „Freiheit“ bedeutet dabei, dass die Nutzer die Software ausführen, analysieren, verbreiten und abändern zu dürfen.

Stallman hat neben der Arbeit an Emacs das GNU Projekt ins Leben gerufen, den GNU C Compiler, den GNU Debugger und andere Werkzeuge entwickelt.

Die FSF, die „Free Software Foundation“, ist die Organisation, die dem GNU Projekt den logistischen, finanziellen und rechtlichen Rahmen gibt. Daneben kümmert sich die FSF um die Beratung, Berichterstattung und Aufklärung rund um das Thema „freie Software“. Die FSF besitzt seit 2001 auch einen europäischen Ableger, die FSFE („Free Software Foundation Europe“), die das Thema „Freie Software“ in Europa koordiniert.

Heute, im Jahr 2017, ist Emacs in Version 25.2 angekommen, diese Version ist auch die Basis für dieses Skript.

Einige Worte noch zum immerwährenden „Kampf“⁵ zwischen Emacs und VI/VIM: Wenn-gleich ich auch Emacs jedem VI bzw. VIM vorziehe (sonst würde dieses Skript ja VIM und nicht Emacs behandeln), so sind Grundkenntnisse in VI oder ED (dem Vorgänger von VI) sehr ratsam. Im Appendix findet sich daher ein kurzer Abriss zu den wichtigsten Funktionen.

⁵siehe https://de.wikipedia.org/wiki/Editor_War

2 Grundlagen

2.1 Installation unter Windows

Die Installation von Emacs ist recht einfach. Man besucht die Downloadseite von Emacs unter <https://www.gnu.org/software/emacs/download.html>, wo verschiedene Links auf GNU Spiegelserver, u. a. zum Beispiel <https://www.gnu.org/software/emacs/download.html> angegeben sind. Beim Besuch einer dieser Seiten wird das entsprechende Verzeichnis des Webservers aufgelistet, hier wählt man dann einfach die Version mit der höchsten Versionsnummer für das eigene Betriebssystem. Nutzer der 32-Bit-Version von Windows wählen die ZIP-Datei, die auf „i686“ endet, Nutzer der 64-Bit-Version die ZIP-Datei, die auf „64“ endet. Diese 64-Bit-Version aus der `emacs-25.2-i686.zip` bildet die Basis für dieses Buch.

Emacs ist komplett portabel, nach dem Download der ZIP-Datei entpackt man diese in ein Verzeichnis der Wahl, das sich auch auf einem USB-Stick oder Netzlaufwerk befinden kann. Der übliche Installationsweg wird jedoch die lokale Installation sein, persönlich installiere ich üblicherweise nach `C : \`. Nach dem Entpacken enthält das Verzeichnis dann die folgenden Unterordner:

bin Dieses Verzeichnis enthält die ausführbaren Dateien von Emacs:

addpm.exe fügt Emacs dem Startmenü unter Windows hinzu

ctags.exe Werkzeug, um sogenannte „Tag“-Dateien für eine Reihe von Programmiersprachen zu erstellen. Dies sind Index-Dateien, die Klassen- und Variablennamen der Quellcodes enthalten und durch die man recht einfach browsen kann.

ebrowse.exe Werkzeug, um durch C++ Klassenhierarchien browsen zu können.

emacs.exe die eigentliche ausführbare Datei, die sowohl im Textmodus (mittels Kommandozeilenoption `-nw`) als auch im Fenstermodus läuft.

emacsclient.exe Ein Kommandozeilen-Client, um mit einem Emacs-Prozess zu kommunizieren.

emacsclientw.exe Eine Client-Version, die kein Kommandozeilen-Fenster beim Start

öffnet.

etags

runemacs Ein Wrapper für die `emacs . exe`, die kein Kommandozeilen-Fenster beim Start öffnet.

libexec Enthält eine Verzeichnisstruktur mit weiteren ausführbaren Dateien.

cmdproxy.exe Internes Emacs-Werkzeug, um Probleme mit den Shells der verschiedenen Windows-Version zu umgehen.

ddeclient.exe Werkzeug, um mit DDE-Servern zu interagieren. Wer den Begriff nicht kennt, wird dies vermutlich auch nie müssen.

hexl.exe Werkzeug zum Erstellen von Hex-Dumps, also Abbildern von Binärdateien.

movemail.exe Hilfswerkzeug zum Verschieben von E-Mails von einem Server in eine lokale Mailbox.

profile.exe Hilfsprogramm zum Profilen von Lisp-Code

update-game-score.exe Werkzeug zum Update von Emacs-Spielstanddateien.

share Enthält die folgenden Unterordner:

appdata ist leer

applications ist leer

emacs Beinhaltet weitere Unterordner u. a. mit den zentralen Lisp-Verzeichnissen.

icons Emacs Icon Dateien in verschiedenen Größen und Formaten.

info Enthält die Info-Seiten für das Emacs-eigene Hypertext-Hilfesystem. Siehe

man Enthält die Emacs-spezifischen Man-Pages, also Hilfe- und Dokumentationsseiten.

var Enthält weitere Unterordner mit den Spielstand-Dateien für Tetris und Snake, siehe Abschnitt 7.5 auf Seite 21.

Zusätzlich zu Emacs selbst empfiehlt es sich, die zusätzlichen Dateien aus der Dependencies-ZIP-Datei zu installieren. Da die Basis für dieses Buch Emacs 25.2 bildet, muss auch die passende Dependencies-Datei, `emacs-25-i686-deps . zip`, gewählt werden und in das Installationsverzeichnis zu entpacken. Sollten Dateien bereits vorhanden sein – was bei mir

Link

bei einer Datei der Fall war – so habe ich die neuere Datei behalten. Hat man die Dateien korrekt installiert, verfügt Emacs über eine Reihe weiterer Funktionen:

- verschiedene Grafik-Typen – PNG, JPEG, GIF, TIFF, SVG – können jetzt in Emacs angezeigt werden
- Netzwerk-Funktionen können jetzt SSL/TLS nutzen. Dies ist insbesondere für den sicheren Zugriff auf MELPA wichtig
- gepackter Text kann jetzt entpackt werden
- HTML und XML können geparkt werden, was beispielsweise für den in Emacs eingebauten EWW Browser wichtig ist

2.2 Installation unter Linux

Emacs ist in den Paketquellen der gängigen Distributionen enthalten, die Installation über den jeweiligen Paketmanager daher auch die empfohlene Option. Persönlich nutze ich überwiegend Xubuntu, hier reichen die folgenden Befehle:

1. `sudo apt-get update`
2. `sudo apt-get install emacs`

2.3 Installation unter Mac OS X

Eine Emacs-Distribution findet man beispielsweise unter <https://emacsformacosx.com>. Man lädt die DMG-Datei herunter und

ausprobieren auf dem Macbook

2.4 Deinstallation

Sollte der Leser jemals in die Verlegenheit kommen, Emacs löschen zu müssen, so reicht es aus, das komplette Installationsverzeichnis zu löschen. Da Emacs wie oben erwähnt komplett portabel ist, werden keine Informationen in irgendwelche System-Verzeichnisse geschrieben.

Hat man mit `addpm.exe` Einträge zum Startmenü hinzugefügt, so lassen sich diese über

`regedit` im `Software/GNU/Emacs` Ast der Registry löschen. Je nach Art der Installation (ob diese als Admin erfolgte oder nicht), findet man diesen Ast in `HKEY_LOCAL_MACHINE` oder `HKEY_CURRENT_USER`.

3 Grundlagen der Bedienung

In diesem Kapitel wird die grundlegende Bedienung von Emacs vorgestellt. Anpassungen, die die „User Experience“ mit Emacs deutlich verbessern, werden in den folgenden Kapiteln behandelt, dieses Kapitel soll daher die „reine Lehre“ vermitteln.

3.1 Von Buffern und anderen Dingen...

Im Rahmen dieses Textes wird häufiger von „Buffern“ und anderen Dingen die Rede sein, daher ist es sinnvoll, diese an zentraler Stelle (also hier) einzuführen.

Text, den man in Emacs editiert, landet in einem sogenannten „Buffer“. Ein Buffer ist keine Datei, sondern ein Stück Speicher, das Emacs zum Arbeiten nutzt. Zu jedem Zeitpunkt ist nur ein Buffer aktuell, diesen werden wir im folgenden auch als den „aktuellen Buffer“ bezeichnen. Befehle zur Manipulation von Buffern werden in den nächsten Abschnitten vorgestellt.

Die maximale Größe eines Buffers hängt von der Architektur des Rechners ab: bei 32-Bit-Rechnern beträgt sie $2^{29} - 2$ Byte, also rund 215 MB, bei 64-Bit-Rechnern hingegen $2^{61} - 2$ Byte, also mehr als 2300 Petabyte. Zusätzlich limitiert natürlich auch das verfügbare RAM die Buffer-Größe.

3.2 Starten und Beenden

Emacs kann man auf verschiedene Arten starten:

- per Mausklick auf `runemacs . exe` im `emacs/bin` Verzeichnis, im Gegensatz zum Mausklick auf `emacs . exe` wird hier kein Interpreter-Fenster (früher „DOS“-Fenster) erzeugt.
- aus der Kommandozeile, wenn man vorher per `cd` in das `bin` Verzeichnis der Emacs-Installation gewechselt ist.
- aus der Kommandozeile aus einem beliebigen Verzeichnis, wenn man vorher die `PATH`-

Variable der Windows-Installation angepasst hat. Siehe die folgende Box „Anpassung der PATH-Variablen“

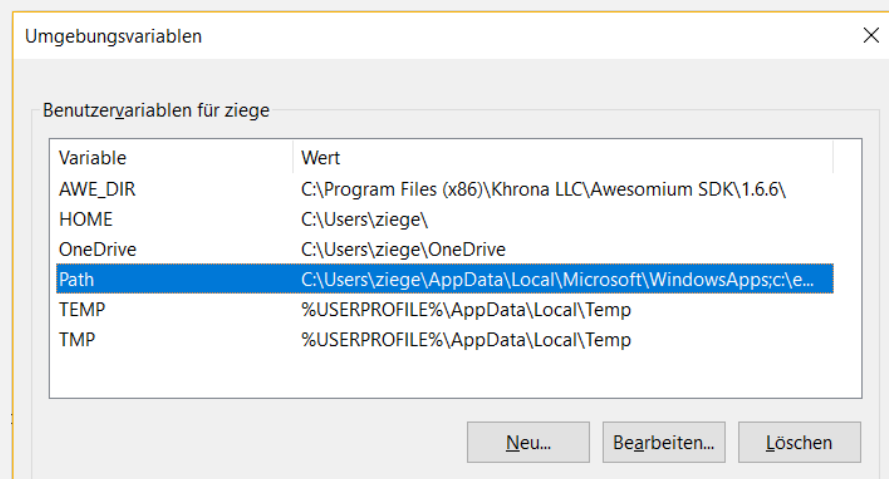
Anpassung der PATH-Variablen unter Windows

Unter Windows 10 kann man am einfachsten die Pfad-Variable, die für die Suche nach ausführbaren Dateien genutzt wird, wie folgt editieren:

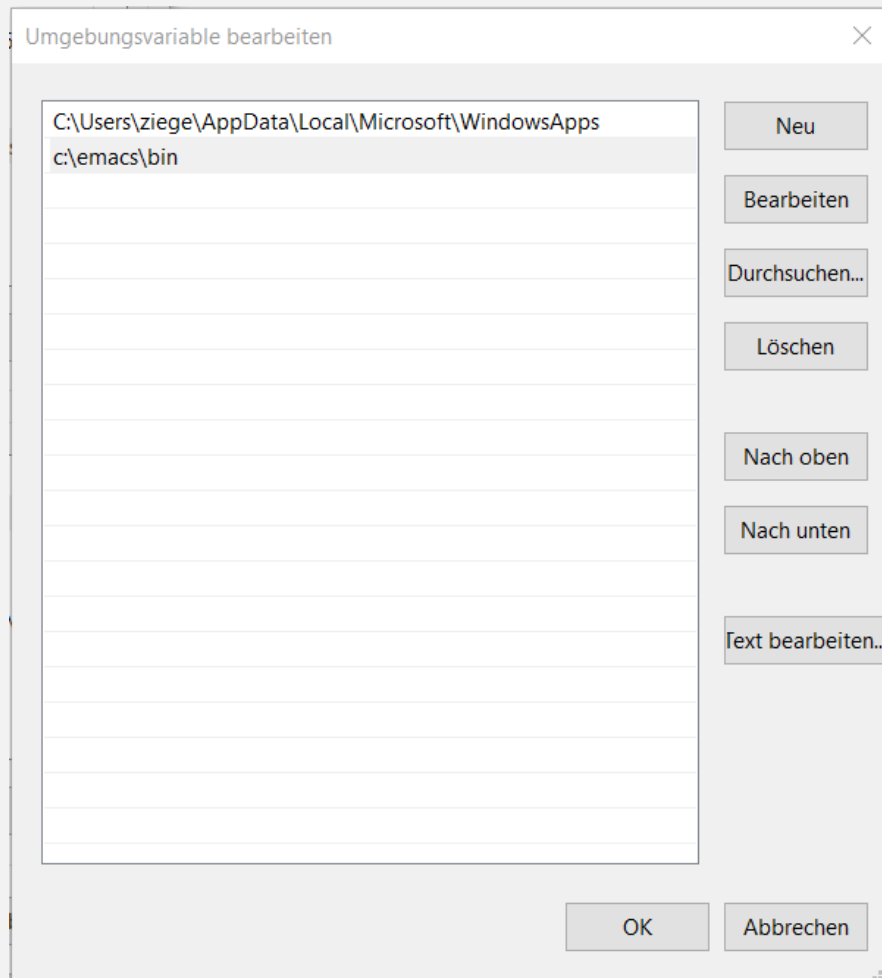
1. Man öffnet die Systemsteuerung und gibt im Suchfenster „Umgebung“ ein.



2. Im sich öffnenden Dropdown-Menü wählt man entweder den Eintrag „Systemumgebungsvariablen bearbeiten“ aus, wenn der Pfad für alle Nutzer des Rechners angepasst werden soll oder „Umgebungsvariablen für dieses Konto bearbeiten“, wenn der Pfad nur für den aktuellen Nutzer angepasst werden soll



3. Im „Umgebungsvariablen“ Menü wählt man den Eintrag „Path“ und geht auf „Bearbeiten“
4. Im folgenden Menü klickt man auf „Neu“ und fügt den Pfad zur Emacs .exe – bei mir `c:\emacs\bin` hinzu



Startet man jetzt eine neue Eingabeaufforderung (aka „DOS-Box“), so kann man Emacs jetzt aus jedem Verzeichnis heraus starten, ein manueller Wechsel nach `c:\emacs\bin` ist nicht mehr notwendig.

Beenden lässt sich Emacs ebenfalls auf verschiedene Arten:

- Durch Drücken des Schließen-Knopfes
- Durch Ausführen des „Quit“-Menü-Eintrags im `File`-Menü

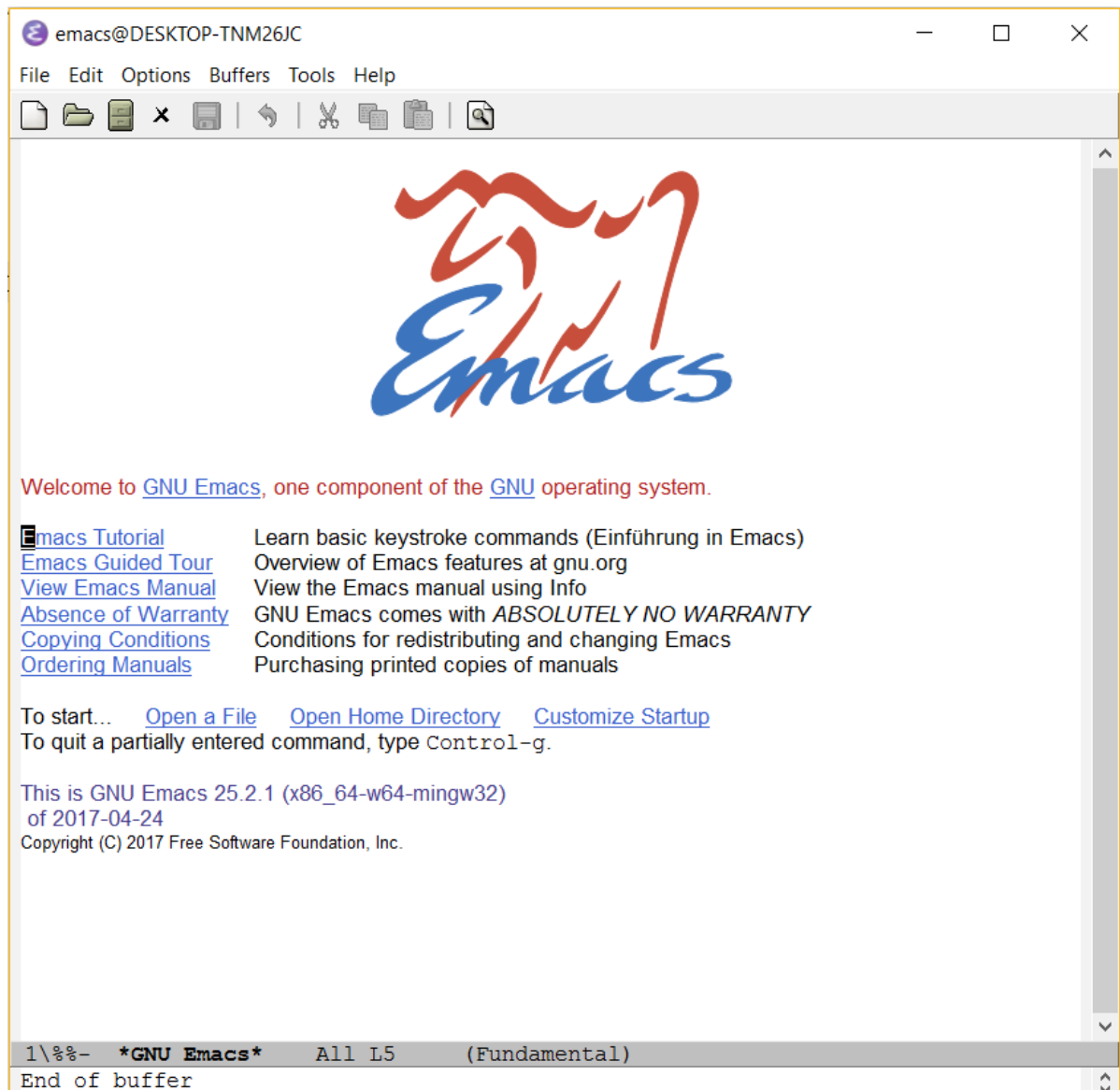


Abbildung 3.1: Emacs 25.2 nach dem Start

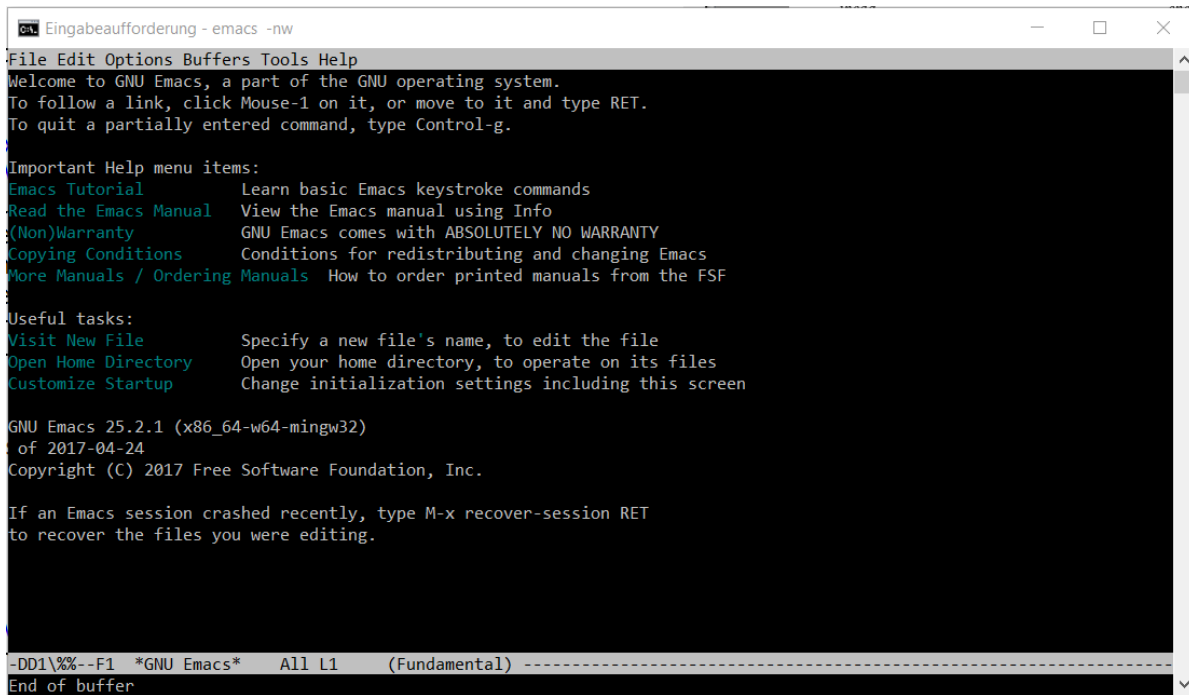


Abbildung 3.2: Emacs 25.2 nach dem Start im Kommandozeilenmodus

- Durch Ausführen des Befehls `[Alt]+[x] save-buffers-kill-terminal`
- Durch Ausführen der Tastensequenz `[Strg]+[x] [Strg]+[c]`; dies ist die vordefinierte Tastenkombination für `[Alt]+[x] save-buffers-kill-terminal`

In jedem dieser Fälle fragt Emacs nach, falls es noch Buffer gibt, die noch nicht gespeichert wurden. Wer in keinem Fall gefragt werden möchte, selbst wenn es noch ungesicherte Buffer gibt, kann `[Alt]+[x] kill-emacs` ausführen oder sogar – was nicht empfehlenswert ist – die Standardtastenkombination `[Strg]+[x] [Strg]+[c]` mit diesem Befehl überschreiben.

3.3 Laden und Speichern

Mittels Maus lädt man Dateien entweder über den Menüeintrag „Open File“ im File-Menü oder durch Drücken des Öffnen-Symbols in der Symbolleiste, siehe folgendes Bild. Startet man Emacs von der Kommandozeile aus, so lässt sich der Dateiname auch als Parameter übergeben, wobei gilt: Ist die Datei nicht vorhanden, so wird ein entsprechender Buffer erzeugt, der beim Speichern eben diese Datei schreibt, wenn der User Schreibrechte hat.

Der eigentliche Emacs-Weg für das Öffnen liegt aber nicht darin, die Maus von A nach B zu schubsen, wie bei allen weiteren Aktionen gibt es auch hier entsprechende Tastenkombinationen.

binationen. Im Fall des Öffnens bzw. Erstellens einer Datei lautet diese Tastenkombination **Strg** + **x** **Strg** + **f**, was ein Shortcut für den Befehl `find-file` ist, der auch über **Alt** + **x** `find-file` ausgeführt werden kann.

Zusätzlich zum Laden bzw. Erstellen eines Buffers über `find-file` gibt es noch die Möglichkeit, über **Strg** + **x** **b** `<Buffername>` einen Buffer neu zu erstellen, wenn noch kein Buffer dieses Namens im Speicher existiert.

Für das Speichern von Dateien bietet Emacs die folgenden Möglichkeiten an

- die beiden Menü-Einträge im File-Menü, „Save“ und „Save As“, um Dateien unter dem bisherigen Namen bzw. einem neuen Namen zu speichern
- das entsprechende Symbol in der Symbolleiste
- **Strg** + **x** **Strg** + **s** für den Shortcut zum Befehl `save-buffer`
- **Strg** + **x** **s** für den Shortcut zum Befehl `save-some-buffers`, was einem „Speichere alle Buffer“ entspricht
- **Strg** + **x** **Strg** + **w** für den „Speichern als“-Befehl `write-buffer`
-
-
-

3.4 Bewegen in der Datei und zwischen Buffern

Befindet man sich in einem Buffer, kann man sich mit den Befehlen aus der folgenden Tabelle innerhalb des Buffers bewegen:

Tabelle 3.1: Befehle zum Bewegen innerhalb des Buffers

Bewegung um	vor	zurück
Zeichen	Strg + b	Strg + f
Wort	Alt + b	Alt + f
Zeile	Strg + p	Strg + n
Satz	Alt + a	Alt + e
Absatz	Alt + {	Alt + }
Seite	Strg + x [Strg + x]

3.5 Editieren von Text

3.6 Suchen und Ersetzen

3.7 Das eingebaute Hilfesystem

3.8 Zusammenfassung

Hier die wichtigsten Erkenntnisse des

4 Konfiguration

4.1 Manuelle Konfiguration von Paketen

4.2 Das Emacs-Paketsystem

4.3 use-package

5 Org Mode

6 AucT_EX

7 Was sonst noch geht...

7.1 ccrypt

7.2 GraphViz

7.3 Emacs als E-Mail-Programm

7.3.1 Emacs als RSS-Reader

7.4 Sweave

7.5 Spielen im Emacs

8 Programmierung

8.0.1 Das „Hello Emacs“-Beispiel

9 Weitere nützliche Software-Tools

In diesem Kapitel sollen weitere nützliche Software-Tools vorgestellt werden, die im Zusammenhang mit Emacs interessant sind und von denen man zumindest gehört haben sollte. Denn „Hat man nur einen Hammer, so sieht alles wie ein Nagel aus!“

Ergänzen oder ersetzen

9.1 T_EX/L_AT_EX

T_EX/L_AT_EX ist ein professionelles Textsatzsystem, mit dem auch dieses Buch gesetzt wurde. Ähnlich alt wie Emacs nutzt L_AT_EX Befehle, um den Text auszuzeichnen. Ein Compiler verwandelt dann den Text in PDF. Zur Installation empfiehlt es sich, die T_EX Live Distribution von <http://www.tug.org> zu installieren. Ebenso wie den Pfad zur `emacs.exe` trägt man am besten auch den Pfad zu den `*latex.exe` Dateien in die PATH-Variable ein.

9.1.1 Das erste L_AT_EX-Dokument

Das folgende Listing zeigt ein einfaches Beispiel für ein L_AT_EX-Dokument. Gibt man den Text so mittels Emacs ein und übersetzt dann über AucT_EX mit beispielsweise `pdflatex`, so sollte das PDF aus Abbildung [dabei entstehen](#).

Link

9.1.2 Eine Musterpräsentation mit Beamer

L_AT_EX lässt sich auch vortrefflich für die Erstellung von Präsentationen nutzen. Das vermutlich fortschrittlichste Paket, Beamer, ist von Till Tantau.

9.1.3 Ein Musterbrief mit **scr1ttr2**

9.1.4 Wer mehr wissen möchte...

Tabellensatz eigenes Buch, daher nur kurzer Überblick

Installiere TeX Live oder MikTeX, stelle sicher dass im Pfad vorhanden

Probiere das folgende Dokument aus und übersetze es auf der Kommandozeile.

Schnapp Dir ein Buch.

9.2 sed und awk

9.3 VI(M)

Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld. Der Text gibt lediglich den Grauwert der Schrift an. Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburn“? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen. An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft. Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein. Er muss keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln.

```
1 ;; https://nickhigham.wordpress.com/2016/01/14/distracti  
  -free-editing-with-emacs/  
2 (scroll-bar-mode 0) ; Turn off scrollbars.  
3 (tool-bar-mode 0) ; Turn off toolbars.  
4 (fringe-mode 0) ; Turn off left and right fringe cols.  
5 (menu-bar-mode 0) ; Turn off menus.  
6 ;; bind fullscreen toggle to f9 key  
7 (global-set-key (kbd "<f9>") 'toggle-frame-fullscreen)  
8  
9 ;; http://emacs.stackexchange.com/questions/2999/how-to-  
  maximize-my-emacs-frame-on-start-up  
10 ;; Start fullscreen (cross-platf)  
11 (add-hook 'window-setup-hook 'toggle-frame-fullscreen t)  
12 ; emacs-doctor.com/emacs-strip-tease.html  
13 ;; Prevent the cursor from blinking  
14 (blink-cursor-mode 0)  
15 ;; Don't use messages that you don't read  
16 (setq initial-scratch-message "")  
17 (setq inhibit-startup-message t)
```


Literatur

- Cameron, Debra, Bill Rosenblatt und Eric Raymond. *Learning GNU Emacs*. 2. Aufl. 1996. ISBN: 1565921526.
- Dominik, Carsten. *The Org Mode 9 Reference Manual*. ARTPOWER INTL PUB, 21. Nov. 2016. 300 S. ISBN: 9789888406852.
- Glickstein, Bob. *Writing GNU Emacs Extensions*. O'Reilly Media, 1997. ISBN: 1565922611.
- Guenther, Karsten. *Emacs Ge-Packt*. mitp, 2003. ISBN: 3826613090.
- Stallman, Richard M. „Emacs: The Extensible, Customizable Display Editor“. In: (1981). URL: <https://www.gnu.org/software/emacs/emacs-paper.html>.
- *GNU Emacs Manual*. Gnu Press, 2000. ISBN: 1882114078.
- Winston, Patrick Henry und B.K.P. Horn. *Lisp*. Addison-Wesley, 1980. ISBN: 0201083299.