

Teil I

Emacs - Der König der Editoren

1 Prolog

Motivation für dieses Skript ist die Tatsache das die Familie der Emacs-Editoren eine bedeutende Rolle auf Unix - und Unix-Like Systemen innehat und der Einstieg für Anfänger eine beträchtliche Hürde trotz umfangreicher Dokumentation darstellt. Ich will mit diesem Beitrag versuchen dem Neuling einen leicht verständlichen und übersichtlichen Einstieg in die Anwendung der Emacsen zu ermöglichen. Auf Unix und Unix-Like Betriebs-Systemen gibt es folgende Kategorien von Editoren:

- 1. Die Familie der Vís (z.B.: Ed, Vim, Elvis, Ex, etc....)
- 2. Die Familie der Emacsen´s (z.B.: XEmacs, Jove, Jed, em, etc....)
- 3. Die Wald - und Wiesen-Editoren (z.B.: Notepad-Clones, ee, etc....)
- 4. Die Spezial-Editoren (z.B.: Binary.(hex), HTML.(NVU), Java.(Eclipse), etc....)

Weil der Aufbau und die Funktionsweise bei allen Emacsen prinzipiell gleich ist, kann man von einer Editoren-Familie reden. Die Unterschiede liegen vor allem auf den Schwerpunkten der einzelnen Editoren. So ist z.B. der em besonders schnell (weil in Assembler geschrieben), der Jove besonders simple, etc.... Bei den Emacsen gibt es zwei große Editoren die gegeneinander antreten. Das eine ist der GNU-Emacs -der von Richard Stallmann geschriebene Ur-Emacs- und der andere ist der von Lucid geschriebene XEmacs. Der Name GNU-Emacs ist allerdings etwas irreführend da beide der GPL unterstehen. Der Unterschied zwischen den Beiden ist das XEmacsen über ein netteres widget verfügt d.h. Menü und Scrollleisten sind schöner anzuschauen und er ist etwas einfacher zu Bedienen. Allerdings fand ich persönlich ihn etwas weniger stabil zumindest unter X-Window. Beide unterscheiden sich vor allem von der Konfigurationsdatei. Das wars dann aber auch schon. Für welchen man sich entscheidet ist persönliche Geschmackssache. Ich benutze den GNU-Emacs und bin Happy. Will aber niemand beeinflussen ...

2 Grundkonzepte

Emacs ist ein

- ANZEIGBARER heißt das der eingegebene Text angezeigt wird.
- ECHTZEIT heißt das Veränderungen sofort durchgeführt werden und nicht erst im Kopf "zwischengespeichert" werden müssen.
- FORTGESCHRITTENER heißt das Funktionen unterstützt werden die weit über das einfache editieren von Texten hinausgeht. Z.B.: Kontrolle von Sub-Prozessen, automatisches Einrücken von Text, gleichzeitige Ansehen von zwei oder mehr Dateien, editieren von formatiertem Text, fähigkeit zum Einbinden von Ausdrücken und Programmiersprachen (bzw. skripten), etc....
- SELBST-DOKUMENTIERENDER heißt das die online-hilfe im Editor eingebaut ist und mittels Tastenkürzel aufgerufen werden kann.

- KONFIGURIERBARER heißt das die Möglichkeit zur vollen Konfiguration von der Oberfläche, dem Kommandoset, den Keybindings, den eingebauten Ausdrücken und was sonst noch denkbar ist, gegeben ist.
- ERWEITERBARER heißt das über die Konfiguration weit hinausgehende Veränderungen möglich sind. Z.B. können völlig neue Kommandos selbstgeschrieben und eingebunden werden. Erweiterungen werden im Allgemeinen in Form von Modulen eingebunden. Als Programmiersprache kommen Lisp und C zum Einsatz.

Editor.

3 Aufrufen von Emacs

Emacs kann wenn er installiert ist in der Console einfach mit dem Kommando "emacs" aufgerufen werden. Dabei ist zu beachten das "emacs" in der Regel nur ein Link auf das eigentliche Binary ist. Das Binary ist i.d.R. mit der Versionsnummer versehen um mehrere Emacsen die gleichzeitig installiert sind auseinanderhalten zu können. Der Link zeigt im allgeminen auf die neuste Emacs-Version. Unter X-Window ist zu Beachten das Emacsen nicht als root in der Console gestartet werden kann (wie jede andere Anwendung auch nicht). Root ist in der Default-Einstellung ein Unprivilegierter User unter X-Window. Umgehen kann man das mit sudo (siehe man sudo) oder der -nw (no window) Option bei der Emacsen in der Console verbleibt und nicht das X-window berührt. Das gleiche gilt selbstverständlich auch für XEmacs außer das der Aufruf xemacs lautet.

4 Aufbau eines Frames

Definition: Frames sind die Fenster des Emacs. Unter X können beliebig viele Frames geöffnet werden. Wird der Emacs geschlossen werden alle Frames die zu diesem spezifischen Emacs gehören mit geschlossen. Deswegen ist darauf zu achten ob ein Emacs (mit seinen Frames) geschlossen wird oder nur ein einzelner Frame. In der Console gibt es nur einen Frame.

Die Leiste am oberen Rand ist die Menu-Bar, dicht gefolgt von der Tool-Bar. Das gesamte editierbare Feld ist das Fenster eines Frames was vertikal und horizontal geteilt werden kann. Die inverse Leiste am unteren Rand ist die Mode-Line die Informationen über den Inhalt des Fensters gibt. Der unterste Fensterrand ist der Mini-Buffer (bufferdefinition siehe unten) oder "echo area" in dem Rückmeldungen über den Betrieb Emacsens gegeben werden. Rechts, Links und Oben oder Unten können sich Scrollbars befinden. Der Aufbau und die Funktionsweise eines Frames sind umfangreich konfigurierbar.

Frames	
C+X 5 2	neuer frame
C+X 5 O	löscht aktuellen frame (funktioniert nur wenn mehrere frames geöffnet sind)
C+X,C+C	beendet emacsen (löscht alle frames) (null)
C+Z	frame in den hintergrund schicken (nur console -kann mit "fg" wieder in den vordergrund geholt werden)
C+X 5 O	springt in einen anderen frame ("Oh")
C+X 5 1	löscht alle frames außer dem, der den fokus hat
Windows	
C+X,O	springt zum nächsten fenster ("Oh")

C+X,0	löscht das aktuelle fenster ("null")
C+X,1	löscht alle fenster außer dem, indem der cursor steht
C+X,2	teilt das aktuelle fenster in zwei horizontale bereiche
C+X,3	teilt das aktuelle fenster in zwei vertikale bereiche
C+X,	verschiebt den fensterinhalt nach links
C+X,	verschiebt den fensterinhalt nach rechts

Cursor

M+X hl-line-mode	hebt die horizontale linie farbig hervor, minor-mode
M+X blink-cursor-mode	schaltet blinken des cursors ein/aus, minor-mode
C+F	cursor ein zeichen nach rechts (forward)
C+B	cursor ein zeichen nach links (backwards)
C+P	cursor ein zeichen nach oben (previous)
C+N	cursor ein zeichen nach unten (next)
C+V	eine seite nach unten
M+V	eine seite nach oben
C+X,U	letzte änderung wiederrufen (undo)
M+F	cursor ein wort vor (forward)
M+B	cursor ein wort zurück (backward)
C+A	cursor an den beginn der zeile (advance)
C+E	cursor an das ende der zeile (end)
M+A	cursor an den beginn des absatzes (advance)
M+E	cursor an das ende des absatzes (end)
M+	cursor an den beginn des textes
M+Shift+	cursor an ende des textes
C+X,R,Leertaste "zeichen"return	speichert aktuelle cursorposition im Register "zechen"
C+X,R,J "zeichen"return	springt zu der im register "zeichen"gespeicherten cursorposition
C+L	cursor nach textmitte scrollen
M+G n return	cursor in zeile n bewegen

Modeline

M+X customize-group return modeline	konfigurationsbuffer für die modeline
M+X line-number-mode	schaltet zeilennummerierung an/aus

Mini-Buffer

M+X customize-group return minibuffer	konfigurationsbuffer für den minibuffer
---------------------------------------	---

5 Die 3 Grund-Prinzipien

5.1 Buffer

Buffer sind Speicherbereiche in denen Dateien, Funktionen, Konfigurationen, Anwendungen und Informationen jeglicher Art gespeichert sind. Sie sind von Fenstern unabhängig. Wird ein Fenster geschlossen geht die Information nicht verloren weil sie im Buffer lebt. Sie geht erst verloren wenn der Buffer gelöscht wird. Durch das Konzept der Buffer ist es möglich gleichzeitig mehrere Texte zu bearbeiten, Hilfsfunktionen aufzurufen, Anwendungen zu benutzen (die im Emacs integriert sind) oder Kommandos abzusetzen. Die Buffer sind das Rückgrat Emacsen. I.d.R hat man ein oder zwei Fenster geöffnet in denen zwischen mehreren Buffern gewechselt wird. Buffer sind Speicherbereiche in denen Informationen gelagert werden. Prinzipiell gilt: Mehrere Fenster können auf einen Buffer zugreifen, mehrere Buffer (genauer: ihr Inhalt) können aber nicht in einem Fenster dargestellt werden.

5.2 Tastenkürzel

Emacs lebt von Tastenkürzeln. Ohne Tastenkürzel kein Leben. Gerade als Anfänger sieht man den Wald vor lauter Bäumen nicht. Dabei ist alles schön logisch. Emacs unterscheidet zwischen:

- 1. Der Kontroll-Taste, an die bestimmte Funktionen in den einzelnen Modi gebunden werden. D.h die Kontroll-Taste verändert ihre Bedeutung je nach Mode bzw. Buffer. Im C-Mode sind andere Funktionen an sie gebunden als z.B im Latex-Mode. Weil jeder Mode seine eigenen Kontroll-Sequenzen kennt ist es nicht möglich alle Funktionen in allen Modi zu kennen (wieviel Modi es genau gibt weiß ich nicht, es sind aber sehr viele). Aus diesem Grund ist es wichtig sich im klaren zu sein was man mit Emacs machen möchte um sich in die jeweiligen Modi einzuarbeiten die man benötigt. Die Kontrolltaste ist mit "C" für "Control" gekennzeichnet (Strg bei deutscher Tastaturbelegung), ich gebe alle Tastenkürzel Prinzipiell in English an. Kontroll-Sequenzen werden mit C+C (Strg und Buchstabe C gleichzeitig drücken, steht für "Control-Control") oder C+X (Strg und Buchstabe X gleichzeitig drücken, steht für "Control-Extend/Extensible") eingeleitet. C+C ist an einen Mode-Gebunden, C+X führt "Meta Kontroll Funktionen" aus, z.B Window - oder Buffer-Operationen)
- 2. Der Meta-Taste, an die auch Mode-Funktionen gebunden sind, führt Funktionen durch die jederzeit zugänglich sein müssen (z.B. den Wechsel in einen anderen Mode), Meta-Funktionen eben. Modi werden mit M+X (Meta und Buchstabe X) geladen. Die Meta-Taste stammt ursprünglich von Unix-Keyboards (Sparc, Alpha) und ist auf i386 Maschinen nicht vorhanden. Sie wird je nach Tastaturbelegung durch die Window oder Alt oder Esc Taste ersetzt. Die Meta-Tasten Konfiguration wird weiter unten beschrieben.

5.3 Modi

Der Emacs unterscheidet zwischen Major-Modi (Haupt-Modi) und Minor-Modi (Neben-Modi). Als Major-Mode werden Modi bezeichnet die eine Benutzer-Umgebung für eine Anwendung (z.B C-Programmierung) oder ein funktionales Frontend (z.B Algebra-Programme wie Octave) zur Verfügung stellen. Beim Beispiel C-Programmierung wird z.B. syntax-highlighting, debugging, etc, zur Verfügung gestellt. Major-Modi zeichnen sich dadurch aus das immer nur einer in einem Buffer aktiviert sein darf und das sie für die spezifische Anwendung ein spezifisches Set an Tasten-Kürzeln mitbringen. Minor-Modi dürfen dagegen beliebig viele in einem Buffer aktiviert sein. Sie stellen i.d.R nur eine Funktion zur Verfügung (z.B.: zeilennummerierung). Die Modi machen Emacs zur EIERLEGENDEN-WOLL-MILCH-SAU.

Wichtige Major-Modi	
M+X fundamental-mode return	standardmodus (Defaulteinstellung)
M+X text-mode return	modus zur einrückung von text
M+X c-mode return	programmierungsumgebung für c
M+X c++-mode return	programmierungsumgebung für c++
M+X html-mode return	umgebung zum erstellen von html
M+X sh-mode return	mode zur bearbeitung von shell-scripts
M+X latex-mode return	umgebung für latex
M+X mail-mode return	mode zum editieren und senden von mail
M+X rmail-mode return	mode zum editieren und lesen von mail
M+X gnus return	ein vielseitiger news-reader
M+X doctor return	für die nerven
M+X perl-mode return	programmierungsumgebung für perl
Wichtige Minor-Modi	
M+X auto-fill-mode return	fließtextmode (automatischer wortumbruch)
M+X font-lock-mode return	syntax-highlighting
M+X abbrev-mode return	abkürzungsmodus (expansion von abkürzungen)
M+X iso-accents-mode return	eingabe fremdsprachiger sonderzeichen
M+X line-number-mode return	zeilennummerierung (erscheint in der modeline)

6 Der Fundamental-Mode

Weil der Fundamental-Mode der wichtigste Mode Emacsens ist, beschreibe ich ihn hier. Er ist unverzichtbar zum Bearbeiten von Texten. Auch wenn man sich hauptsächlich für einen anderen Mode interessiert, sollte man den Fundamental-Mode beherrschen. Er stellt grundlegende Funktionen zur Verfügung.

Dateien laden und speichern, Emacs beenden	
C+X,C+F datei return	Datei laden (find)
C+X,I	Datei in einen vorhandenen Text einfügen (insert)
C+X,C+S	Datei speichern (save)
C+X,S,!	Alle offenen Dateien speichern
C+X,C+W datei return	Datei unter neuem namen speichern (write)
C+X,S	Alle Dateien mit Rückfrage speichern
C+X,C+C	Emacs schließen
Text löschen und wieder einfügen	
M+D	löscht das nächste Wort bzw. den Beginn des Wortes bis zur Cursorposition
M+backspace	löscht das vorige Wort bzw. den Beginn des Wortes bis zur Cursorposition

C+K	löscht das Zeilenende ab der Cursorposition
M+0(null),C+K	löscht den Zeilenanfang vor der Cursorposition
M+M	löscht den nächsten Absatz
M+Z, x	löscht alle Zeichen bis zum nächsten Auftreten von x (das Zeichen x wird mit gelöscht)
C+Y	fügt den zuletzt gelöschten Text an der aktuellen Cursorposition wieder ein
Text markieren	
C+Leertaste	setzt einen unsichtbaren Markierungspunkt
C+W	löscht den Text zwischen dem Markierungspunkt und der aktuellen Cursorposition
C+Y	fügt den gelöschten Text wieder ein
C+X,C+X	vertauscht Cursorposition und Markierungspunkt
Text überschreiben	
M+X overwrite-mode	Text wird ab Cursorposition überschrieben
Text ein- und ausrücken	
C+Leertaste	Markierungspunkt setzen
C+X,Tab	Text zwischen Markierungspunkt und Cursorposition um ein Zeichen einrücken
Esc,-,C+X,Tab	Text um ein Zeichen ausrücken
M+n,C+X,Tab	Text um n Zeichen einrücken
Esc,-,M+n,C+X,Tab	Text um n Zeichen ausrücken
Inkrementelle Suche	
C+S	inkrementelle Suche vorwärts
C+R	inkrementelle Suche rückwärts
C+G	Abbruch (dies ist ein META-KOMMANDO)
C+X,C+X	vertauscht Markierungspunkt (beginn der Suche) und aktuelle Cursorposition
M+P	wählt einen früher verwendeten Suchtext aus (previous)
M+N	wählt einen früher verwendeten Suchtext aus (next)
Suchen und ersetzen	
M+%	Kommando zum Suchen und ersetzen ohne Muster
M+X query-replace-r return	Suchen und ersetzen mit Muster
Leertaste oder Y	ersetzen, Suche fortsetzen
, ersetzen, aber Cursor stehen lassen (zur Ergebniskontrolle)	
backspace oder N	nicht ersetzen, Suche fortsetzen
Esc	nicht ersetzen, Kommando abbrechen
!	alle weiteren Ersetzungen ohne Rückfrage durchführen (Vorsicht!)
C+R	Kommando unterbrechen und rekursive Suche durchführen
C+M+R	Kommando wieder aufnehmen

7 Das integrierte Hilfe-System

Emacsen verfügt über ein integriertes Hilfe-System das mit F1 eingeleitet werden kann.

F1,F1	Übersicht über vorhandene Hilfefunktionen
F1,A text return	Apropos
F1,B	Übersicht über alle Tastenkürzel (Bindings)
F1,C tastenkürzel	Kurzbeschreibung des zugeordneten Kommandos (Command)
F1,F kommando return	Kurzbeschreibung des Kommandos (Function)
F1,Shift+F	Emacs-FAQ
F1,I	Info
F1,N	Zusammenfassung eines Vergleichs zwischen früheren und der aktuellen Emacs-Version
F1,T	Emacs-Tutorial
F1,C+F name	Zeigt Info zu Kommando an
F1,C+P	Stallmanns Crusade against closed source (stichwort: "Free as in Freedom")
M+X man	Man

8 Konfiguration

Das Konfigurations-System ist weitgehend selbsterklärend.

M+X customize-browse	Konfigurations-Baum zum Browsen (stichwort:"versatile")
M+X customize-group	Konfigurations-Dialog für spezifische Gruppen
M+X customize	Konfigurations-Dialog für die Emacs-Gruppe

9 Interessante Anwendungen

Emacsen verfügt über eine große Zahl an Anwendungen die entweder ausschließlich für Emacsen geschrieben wurde oder in Form eines Frontends zu externen Anwendungen existieren. Hier möchte ich eine Anregung zu interessanten Anwendungen geben. Jede Anwendung wird als Mode (meistens Major) zur Verfügung gestellt. Der Einarbeitungsaufwand unterscheidet sich je nach Anwendung, für den man allerdings meist mehr als entschädigt wird durch die Erleichterungen die Emacsen einem zukommen läßt. Anwendungen werden meist in Form von Modulen an den Emacs angebunden. Es kann sein das die jeweiligen Module die gewünscht sind, nicht gerade auf dem System vorhanden sind. Sie können dann aus dem Internet besorgt werden (www.emacs.org). Das ist wahrscheinlich das was Sie wollen.

M+X gnus	Ein umfangreicher News-Reader. Vermutlich einer der Mächtigsten seiner Art.
M+X octave	Frontend zum Linear-Algebra-Prozessor
M+X r	Frontend zur R Statistical Computer Language (GNU's Synonym für S)
M+X doctor	Psychotherapeut. Alle Kassen. Spezialisiert auf Sexuelle und Emacs-Spezifische Störungen
M+X calendar	Damit der User up to date ist
M+X shell	Startet Shell im Emacs
M+X latex-mode	Latex-Mode je nachdem was installiert ist (ich selbst empfehle auctex)
M+X ps-mode	Postscript Mode
M+X sql-mode	Selbsterklärend
M+X sql-postgres	difo
M+X sql-mysql	as well
M+X bdb	Insidious Big Brother Database (sexy hexy)
M+X dictionary	Mode um öffentlich zugängliche Wörterbücher im Internet effizient zu durchstöbern
M+X w3m	html browser
M+X psgml-mode	Mode für XML und SGML

Und noch viele mehr...

www.gnu.org/software/emacs/emacs.html gnu.emacs.sources (newsgroup)

Fröhliches Emacsen!

©by Oliver Kölsch, 2005 made with the help of \LaTeX