

Gimme Shell_(ter)

Mein Umzug in die Shell

markus schnalke <meillo@marmaro.de>

Vorwort

Dass meine Zukunft in der Shell stattfinden soll, war mir schon lange klar.

Bis ich jedoch soweit war, war es ein langer Weg.

Und keinesfalls möchte ich behaupten, dass ich am Ziel wäre, denn es gibt kein Ziel. Es gibt nur vorwärts.

Ziele

Dieser Vortrag soll zeigen, dass man auch ohne X komfortabel arbeiten kann.

Ich möchte die “Angst” vor der textbasierten Benutzerschnittstelle nehmen. Auch will ich hierzu Einstiegshilfen geben.

Ich möchte motivieren, mal Urlaub in der Shell zu machen.

Es soll klar werden, dass und warum Kommandozeilenprogramme oft besser als ihre grafischen Pendants sind.

Und als zentrales Ziel soll dieser Vortrag Gründe darstellen, weshalb es sich lohnt, in die Shell umzuziehen.

Motivation

- ▶ Weil es cool ist **hehe**
- ▶ effektiveres Arbeiten möglich
- ▶ konfigurierbarer
- ▶ ressourcenschonender

- ▶ Vorteile von screen
- ▶ ssh von überall
- ▶ auch ohne X daheim

Konzentration auf das Wesentliche!

Näher an der UNIX-Philosophie

UNIX

UNIX is a registered trademark of The Open Group.

Die UNIX-Philosophie

DER Grund UNIX zu verwenden
"Pro UNIX" statt "Anti Windows"

Welche wirklichen Vorteile bietet Gnome gegenüber der
Windows-Oberfläche?

Richtig, keine!

An der Basis seid ihr zu UNIX gewechselt, weshalb denn nicht auch
an der Oberfläche?

Oder: *An der Oberfläche seid ihr zu UNIX gewechselt, weshalb
denn nicht auch an der Basis?*

UNIX ist mehr als Ubuntu!

Um UNIX zu verstehen, muss man seine Philosophie kennen!

Die UNIX-Philosophie

- ▶ klein ist schön
- ▶ genau eine Sache gut machen
- ▶ früher Prototyp
- ▶ Portabilität vor Effizienz
- ▶ Daten in ASCII-Files speichern
- ▶ Code borgen
- ▶ alles automatisieren
- ▶ CLI statt GUI
- ▶ Programme als Filter

siehe auch

- ▶ Mike Gancarz
- ▶ Rob Pike
- ▶ <http://harmful.cat-v.org/cat-v/>
- ▶ Plan9

GUI vs. CLI

GUI-Programme ...

- ▶ haben hässlichen Code
- ▶ sind aufgebläht
- ▶ schränken ein
- ▶ laufen nur mit X

CLI-Programme ...

- ▶ sind kleiner
- ▶ sind besser konfigurierbar
- ▶ sind älter und ausgereifter
- ▶ sind portabler (und damit auch öfter portiert)

(Natürlich gilt das nicht zwangsläufig, die Wahrscheinlichkeit ist aber groß.)

CURSES

CLI, TUI, GUI

- ▶ CLI: `cat cli.txt | mail -s CLI lugu`
- ▶ TUI: `mutt`
- ▶ GUI: `Menu > Applications > Evolution`

CURSES

- ▶ CURSES ist *nicht* CLI
- ▶ sondern Text User Interface
- ▶ Ich bin hier nicht immer ganz korrekt
- ▶ CURSES gehört aber auch zur Shell

GUIs als Frontend

GUIs sollen laut UNIX-Philosophie nur Frontends sein.

Beispiele

- ▶ Programm: `nmap`
- ▶ Frontends: `nmapfe` (GTK), `knmap` (Qt)

- ▶ Programm: `libparted`
- ▶ Frontends: `parted` (curses), `gparted` (GTK), `qtparted` (Qt)

Größe von User-Interfaces

- ▶ Package: **libparted1.7-1**
- ▶ Size: 198,554
- ▶ Depends: libc6 (>= 2.3.6-6), libuuid1

- ▶ Package: **parted**
- ▶ Size: 56,268
- ▶ Depends: libc6 (>= 2.3.6-6), libncurses5 (>= 5.4-5), libparted1.7-1 (>= 1.7.1-1), libreadline5 (>= 5.2)

- ▶ Package: **gparted**
- ▶ Size: 452,616
- ▶ Depends: libatk1.0-0 (>= 1.12.2), libc6 (>= 2.3.6-6), libcairo2 (>= 1.2.4), libfontconfig1 (>= 2.3.0), libfreetype6 (>= 2.2), libgcc1 (>= 1:4.1.1-12), libglib2.0-0 (>= 2.10.0), libglibmm-2.4-1c2a, libgtk2.0-0 (>= 2.8.0), libgtkmm-2.4-1c2a, libice6 (>= 1:1.0.0), libpango1.0-0 (>= 1.12.3), libparted1.7-1 (>= 1.7.1-1), libpng12-0 (>= 1.2.8rel), libsigc++-2.0-0c2a (>= 2.0.2), libsm6, libstdc++6 (>= 4.1.1-12), libuuid1, libx11-6, libxcursor1 (>= 1.1.2), libxext6, libxfixed3, libxi6, libxinerama1, libxrandr2, libxrender1, zlib1g (>= 1:1.2.1), gksu, menu

Programme vergleichen

Weshalb die Programmgröße beachten?

- ▶ Kleine Programme sind: flexibler, portabler, einfacher zu verstehen, bugfreier (vgl. UNIX-Philosophie)
- ▶ Die Paketgröße gibt indirekt Aufschluss auf den Codeumfang
- ▶ Der Codeumfang im Verhältnis zu den Features wirft Überlegungen zur Qualität des Codes auf

Dies ist nur als Anhaltspunkt zu sehen!

Messmethoden

Paketgröße

- ▶ `apt-cache show <paket> | grep ^Size`
- ▶ einfach auszulesen
- ▶ Problem: Dokumentation ist oft auch im Paket enthalten

Source-Filesizes

- ▶ `du -c *. [ch]` (je nach Sprache)
- ▶ Quellcode muss vorliegen (`apt-get source`)
- ▶ Problem: { vs. `begin`

Source Lines Of Code

- ▶ `sloccount` von David A. Wheeler
- ▶ Sprachen werden “gleicher bewertet”

Die Shell und ich

Mein Weg

2005

Fluxbox, Firefox, Kate, Sylpheed Claws, gFTP, Gaim, vlc, k3b, gThumb, OOo, Grip, XMMS, ...

2006

Auslöser:

Artikel "Mein Umzug in die Shell" von Gregor Oelze auf linuxlog

2007

dwm + screen, Iceweasel + *w3m*, *vim*, *mutt*, *ncftp*, *weechat* + *bitlbee*, *mplayer*, *wodim* + *growisofs*, *gqview* + *qiv*, OOo, *jack*, *cplay*, ...

Nachzulesen auf <http://marmaro.de/lue>

Editor

vi(m)

- ▶ Der Editor schlechthin
- ▶ Auf jedem UNIX-System vorhanden
- ▶ meist in Form eines Klons
- ▶ darf ruhig angepasst werden: `.vimrc`
- ▶ Die meisten User verwenden viel zu wenige seiner Möglichkeiten!

Alternativen

- ▶ vim (7350k) Der bekannteste (und beste?) vi-Klon.
- ▶ nvi (290k) Reimplementierung des original vi. Weniger Features; u.a. kein Syntax-Highlighting.
- ▶ emacs (5.89e37) ...

Mailclient

mutt

- ▶ “The mail client, that sucks less”
- ▶ sehr konfigurierbar
- ▶ Kombination mit `fetchmail`, `procmail`, `masqmail`
- ▶ Debian-Paket: 1810k

Alternativen

- ▶ `mailx` (155k) einfaches Mailprogramm
- ▶ `pine` (4M) (nur als Source-Paket verfügbar)
- ▶ `mutt-ng` (1790k) Neue Features für mutt (tot?)

Browser

w3m

- ▶ Textbrowser
- ▶ hat Caret-Navigation
- ▶ kein JavaScript
- ▶ Inline-Bilder mit `w3m-img`
- ▶ Debian-Paket: 1100k

Alternativen

- ▶ lynx (1860k) der Dinosaurier
- ▶ links (380k) lynx-Klon
- ▶ elinks (1190k) hat auch JavaScript-Unterstützung
- ▶ links2 (2080k) bietet textuellen und grafischen Modus
- ▶ dillo (360k) rein grafisch, layoutet aber wie Textbrowser

Musikplayer

cplay

- ▶ Frontend für mpg123, ogg123, ...
- ▶ sehr simpel, aber meist ausreichend
- ▶ vi-like Bedienung
- ▶ Remote-Control via FIFO
- ▶ Debian-Paket: 41k

Alternativen

- ▶ moc (190k) sehr einfach, etwas instabil
- ▶ cmus (124k) sehr featurereich
- ▶ mpd+ncmpc (127k+85k) Client-Server-Architektur

Filemanager

coreutils

- ▶ `cd`, `mv`, `cp`, `rm`, `mkdir`, `rmdir`, ...
- ▶ man braucht eine gewisse Übersicht im Filesystem
- ▶ letztendlich aber einfach am schnellsten
- ▶ überall vorhanden

Alternativen

- ▶ `mc` (2140k) Alleskönner mit schrecklicher Bedienung
- ▶ `vifm` (63k) `mc` mit `vi`-Bedienung; noch zu wenige Features

FTP-Client

ncftp

- ▶ toller FTP-Client
- ▶ hatte nur früher ein curses-Interface (`ncftp2`)
- ▶ `ncftpget`, `ncftpput`, ...
- ▶ Debian-Paket: 500k

Alternativen

- ▶ `ftp` (50k) wenig Komfort
- ▶ `lftp` (540k) (kenne ich leider nicht)
- ▶ `mc` (2140k) er kann auch `ftp`

Chatten

weechat-curses

- ▶ ein IRC-Client
- ▶ “irssi von Grund auf neu”
- ▶ Debian-Paket: 580k

Alternativen

- ▶ irssi (1060k) populärer IRC-Client
- ▶ sic (6k) liest stdin, schreibt stdout
- ▶ centericq (1260k) sehr bekannter ICQ-Client

bitlbee

- ▶ bitlbee ist ein IM-to-IRC-Gateway
- ▶ IRC-Client für ICQ und Jabber verwenden
- ▶ Debian-Paket: 315k

Framebuffer

fbi

- ▶ fbi zeigt Bilder im Framebuffer
- ▶ fbgs zeigt PDF- und PS-Dokumente im Framebuffer
- ▶ Debian-Paket: 54k (enthält fbgs)

mplayer

- ▶ `mplayer -vo fbdev` zeigt Filme im Framebuffer
- ▶ kann auch über libaa, libcaca und Ähnliches ausgeben
- ▶ Debian-Paket: 4400k

screen

screen

- ▶ Terminal-Multiplexer
- ▶ ein `must-have` für Shelluser
- ▶ Nur noch ein Terminal nötig (ssh)
- ▶ Arbeit an beliebiger Stelle unterbrechen und später genau dort fortsetzen
- ▶ Wer einmal damit gearbeitet hat kommt nicht mehr davon los!
- ▶ Debian-Paket: 590k

Alternativen

- ▶ `dtach` bietet `detach`-Feature von `screen`

Remote arbeiten

Server

- ▶ Ein Server ist etwas sehr Tolles!
- ▶ per dyndns.org aus dem Internet erreichbar
- ▶ Upload-Bandbreite sollte nicht zu dünn sein
- ▶ (64k sind etwas wenig *heul*)
- ▶ Zugang per ssh

Client

- ▶ nur Internet und offener Port nötig
- ▶ für Windows: PuTTY (gibt's auch für den USB-Stick)

Remote arbeiten

Vorteile

- ▶ weniger Wartungs- und Konfigurationsaufwand
- ▶ einfache Datensicherung
- ▶ von überall im eigenen System arbeiten

Nachteile

- ▶ Dateitransfers zum Client nötig
- ▶ u.U. etwas lahm
- ▶ nicht überall ist ssh erlaubt

dwm

Window-Manager

dwm

- ▶ dynamic window manager
- ▶ seit Juli 2006
- ▶ von Anselm R. Garbe (wmii)
- ▶ <http://dwm.suckless.org>
- ▶ Debian-Paket: 20k

Alternativen

- ▶ `larswm` (110k) hatte Einfluss auf `dwm`
- ▶ `ratpoison` (170k) soll zu `screen` passen
- ▶ `ion2` (330k) bekannt für seine Tastatursteuerung
- ▶ `icewm` (700k) wird oft als sehr kleiner “herkömmlicher” WM aufgeführt

dwm

Features

- ▶ minimalistisch, nur das Nötig(st)e
- ▶ Tagging statt Workspaces
- ▶ Konfiguration durch Quellcode editieren und neu kompilieren
- ▶ Quellcode kann an einem Abend verstanden werden
- ▶ aktive Entwicklung

Ziele

- ▶ 2kLOC
- ▶ portables C
- ▶ guter Code (Refactoring)
- ▶ sehr standardkonform (aber: broken apps)
- ▶ Erweiterung mit Patches

suckless-Projekt

Community

- ▶ <http://suckless.org>
- ▶ kleine aktive Community
- ▶ Patches als fester Bestandteil
- ▶ Mailingliste ist Pflicht
- ▶ einfach mitmachen!

Mein dwm

- ▶ ein Patchset
- ▶ <http://prog.marmaro.de/dwm-meillo>

Abschluss

Soll erfüllt?

Es sollte klar geworden sein, dass man auch ohne X komfortabel arbeiten kann.

Ich habe Einstiegshilfen gegeben und hoffentlich die “Angst” vor der textbasierten Benutzerschnittstelle genommen.

Jetzt sollte eigentlich jeder von euch dabei sein, einen Ausflug in die Shell zu planen.

Weshalb die Shell wirklich besser ist, als die grafische Oberfläche, das kann man nur für sich selbst herausfinden . . .

Zusammenfassung

Die Shell ist die grundlegende Kommunikationsschnittstelle zum Rechner. Keine GUI ist so mächtig wie die Shell.

UNIX und die Shell sind wie Yin und Yang.

Man versteht die Shell erst, wenn man UNIX versteht — UNIX zu verstehen, heißt seine Philosophie zu verstehen.

Der entscheidende Teil meines Vortrags ist die UNIX-Philosophie. Versteht man diese, so versteht man was ich meine. Findet man aber nicht seinen Weg zu ihr, so wird man kaum verstehen, um was es hier geht.

Die Konzepte hinter UNIX haben mich vollkommen überzeugt. . . . und es war eine der besten Entscheidungen meines Lebens, nach ihnen zu leben!

Links auf einen Blick

- ▶ <http://dwm.suckless.org> (Website des dwm)
- ▶ <http://harmful.cat-v.org/cat-v/> (Website von Uriel)
- ▶ <http://marmaro.de/lue> (Mein Computer-Tagebuch)
- ▶ <http://pimpmyshell.de> (Projekt von Gregor Oelze)
- ▶ <http://prog.marmaro.de/dwm-meillo> (Mein dwm-Patchset)
- ▶ <http://suckless.org> (Software which sucks less)
- ▶ <http://www.cs.bell-labs.com/plan9/> (Das bessere UNIX!?)

unbedingt lesen!

The Command Line — The Best Newbie Interface

<http://osnews.com/story.php/6282/The-Command-Line-The-Best-Newbie-Interface>

Verwendete Software

- ▶ Debian GNU/Linux
- ▶ \LaTeX -Beamer und pdf \LaTeX
- ▶ Vim
- ▶ qiv und ImageMagick
- ▶ Mercurial

Danke für eure Aufmerksamkeit

Den Vortrag gibt's online auf <http://marmaro.de/docs>

2007-12-11