

# Die Shell



## Einführung in die Bourne Again Shell

Marco Pfeifer

Proseminar: Linux für Umsteiger

Betreuer: Julian Martin Kunkel, Olga Mordvinova

# Übersicht



- Allgemeines
- Handhabung
  - Tasten und Tastenkombinationen
  - Wildcards
  - Ein/Ausgabeumlenkung
  - Aliase
  - Jobkontrolle
- Konfiguration
- Quellen

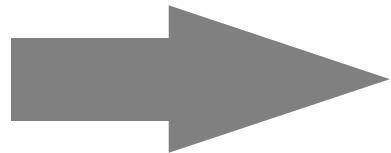
# Allgemeines



- Motivation
- Kommandovarianten
  - Eingebaute Kommandos
  - Richtige Programme
  - Funktionen
  - Skripte

# Motivation

- 
- Befehlsübermittler vom User zum OS
  - Prinzipiell gleiche Aufgabe wie GUI



Kommandozeileninterpreter

# Eingebaute Kommandos

---

- Build-In Commandos
- Werden direkt in bash ausgeführt
- Beispiele
  - alias
  - exit
  - history
  - jobs

# Richtige Programme



- Auch externe Kommandos
- Werden durch bash gestartet
- Beispiele
  - vpnclient
  - vi
  - cat

# Funktionen

---

- Gruppierungen von Befehlen
- Erzeugen und verwerfen:

```
pfeifer@TM2410:~$ hallo ()
> {
> echo 'Herzlich willkommen, '$1'!'
> }
pfeifer@TM2410:~$ hallo Kurs
Herzlich willkommen, Kurs!
pfeifer@TM2410:~$ unset hallo
pfeifer@TM2410:~$ hallo
bash: hallo: command not found
pfeifer@TM2410:~$ █
```

# Skripte



- Textdatei
- Mit Ausführungs-Berechtigung
- Kein unbeabsichtigtes Starten
- `./Testskript.sh`

# Tasten und Tastenkombinationen



- Einfachste Tastenkombinationen
- Einfache Tastenkombinationen
- Tabulator-Taste
- History

# Einfachste Tastenkombinationen



- PFEIL VOR / ZURÜCK
- BACKSPACE / ENTF
- POS1 / ENDE

# Einfache Tastenkombinationen I

---

- STRG + A
  - Zeilenanfang (wie POS1)
- STRG + E (wie ENDE)
  - Zeilenende
- ALT + f / ALT + b
  - Wort vor / zurück
- STRG + k
  - Löscht Zeile bis Ende

# Einfache Tastenkombinationen II

---

- STRG + t
  - Vertauscht die letzten beiden Buchstaben
- ALT + t
  - Vertauscht die letzten beiden Worte
- STRG + I
  - Löscht Bildschirm (clear)

# Die Tabulator-Taste I

---

- TAB
  - Ergänzt angefangene Namen
- Bei Mehrdeutigkeit: Erneut TAB
  - listet Möglichkeiten auf
- Hauptsächlich für Pfadnamen

# Die Tabulator-Taste II

---

- \$ Variable
- ~ Benutzername
- @ Hostname
- Ohne Symbol
  - Alias
  - Kommando
  - Pfadname

# Die History



- Blättern
  - PFEIL HOCH
  - PFEIL RUNTER
- Suchen
  - STRG + R
    - Enter
    - ESC / PFEIL LINKS / RECHTS
- history

# Wildcards



- \*
- Beliebige Zeichen
- ?
- Beliebiges Zeichen
- [...]
- Zeichenklasse
- [!...]
- Negierte Zeichenklasse

# Wildcard-Verwendung

---

- d[kgt]r
  - dkr
  - dgr
  - dtr
- ```
root@TM2410:~# mv *.gif Bilder/
```

  - Verschiebt alle gifs in Bilder

# Ein/Ausgabeumlenkung



- Ein/Ausgaben
- Eingabeumleitung
- Ausgabeumleitung
  - Anhängen
  - Fehler umleiten
- Ausgabe- und Eingabeumleitung
- Pipe
- Tee

# Ein/Ausgaben



- `stdin (0)`
  - Standardeingabe
- `stdout (1)`
  - Standardausgabe
- `stderr (2)`
  - Standardfehlerkanal

# Eingabeumleitung



- Eingabeumlenkung durch <

- BEFEHL < DATEI

```
_ root@TM2410:~# wc -w < test.txt  
615
```

- Zählt Worte in test.txt

# Ausgabeumleitung I

---

- Ausgabeumlenkung durch >

- BEFEHL > DATEI

- ```
– root@TM2410:~# wc -w > worte.txt
Hallo ihr da!
root@TM2410:~# cat worte.txt
3
```

- Schreibt Wortzahl in worte.txt

# Ausgabeumleitung II

---

- Ausgabeumlenkung durch >>
  - BEFEHL >> DATEI

```
root@TM2410:~# wc -w >> worte.txt
- Ja ihr da drüben!
root@TM2410:~# cat worte.txt
3
4
```

- Hängt Wortzahl an worte.txt an

# Fehler umleiten



- Dateideskriptor 2 (stderr) in Datei schreiben
  - PROGRAMM 2> error.txt
- Diesen an Datei anhängen
  - PROGRAMM 2>> error.txt
- Dateideskriptor 1 und 2 (stdout, stderr) umleiten
  - PROGRAMM > beides.txt 2>&1

# Ausgabe- und Eingabeumleitung

---

- Sowohl Eingabe als Ausgabe umleiten
  - Befehl < Eingabedatei > Ausgabedatei

```
– root@TM2410:~# wc -w < test.txt > worte.txt  
root@TM2410:~# cat worte.txt  
615
```

- Zählt in test.txt und schreibt in worte.txt

# Pipe



- Röhre
- Gepufferter Datenstrom
- Zwischen Prozessen
- FIFO
- |

# Pipe-Verwendung



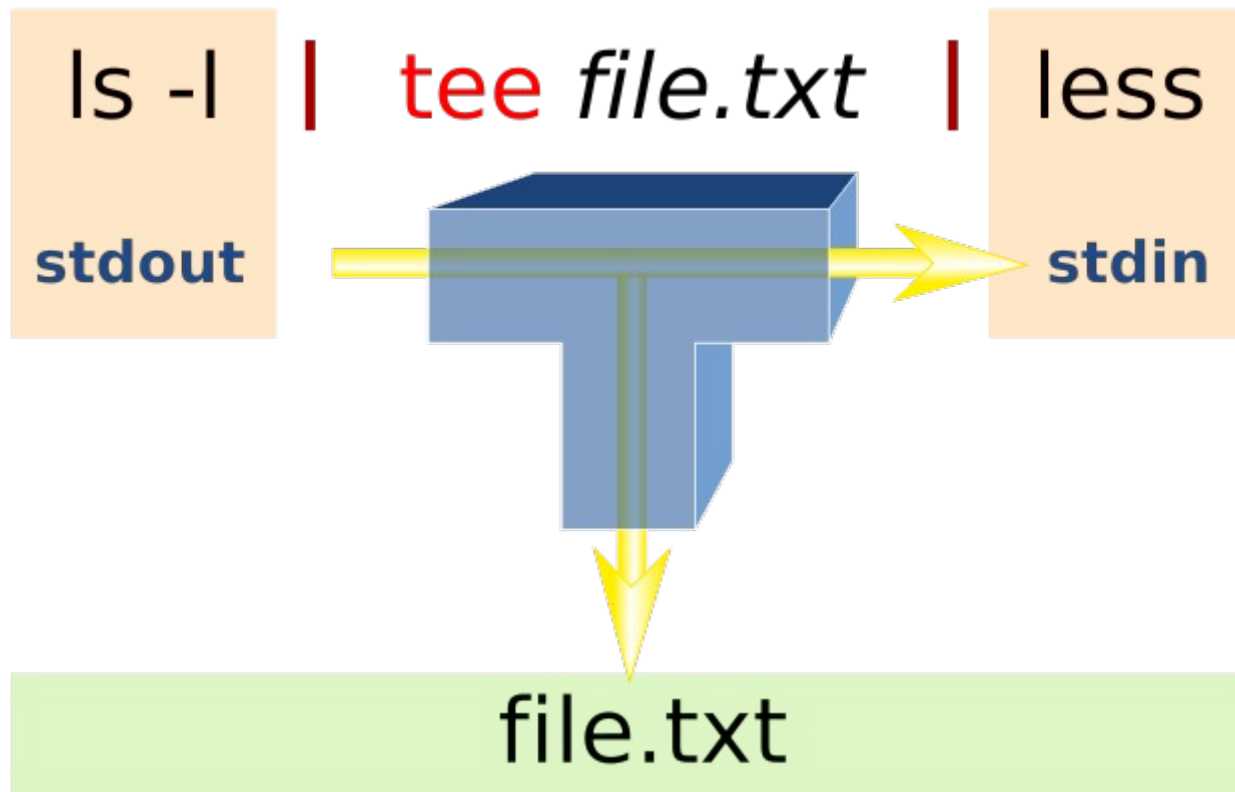
- Befehl mit Ausgabe | Befehl mit Eingabe

- ```
root@TM2410:~# ls | wc -w  
18
```

- Zählt den Verzeichnisinhalt

- Mehrere Pipes in Kombination: Pipeline

# Tee



[http://de.wikipedia.org/wiki/Tee\\_\(Unix\)](http://de.wikipedia.org/wiki/Tee_(Unix))

# Aliase



- Kommando für ein anderes
- Wozu?
- Verwaltung

# Aliase - Wozu?



- Weniger tippen
- Leichter merken
- Tippfehler korrigieren

# Aliase verwalten

---

- Anlegen

- alias NEUESKOMMANDO='KOMMANDO'

- `root@TM2410:~# alias ll='ls -l'`

- Aufheben

- unalias KOMMANDO

- `root@TM2410:~# unalias ll`

- Anzeigen

- alias

# Jobkontrolle



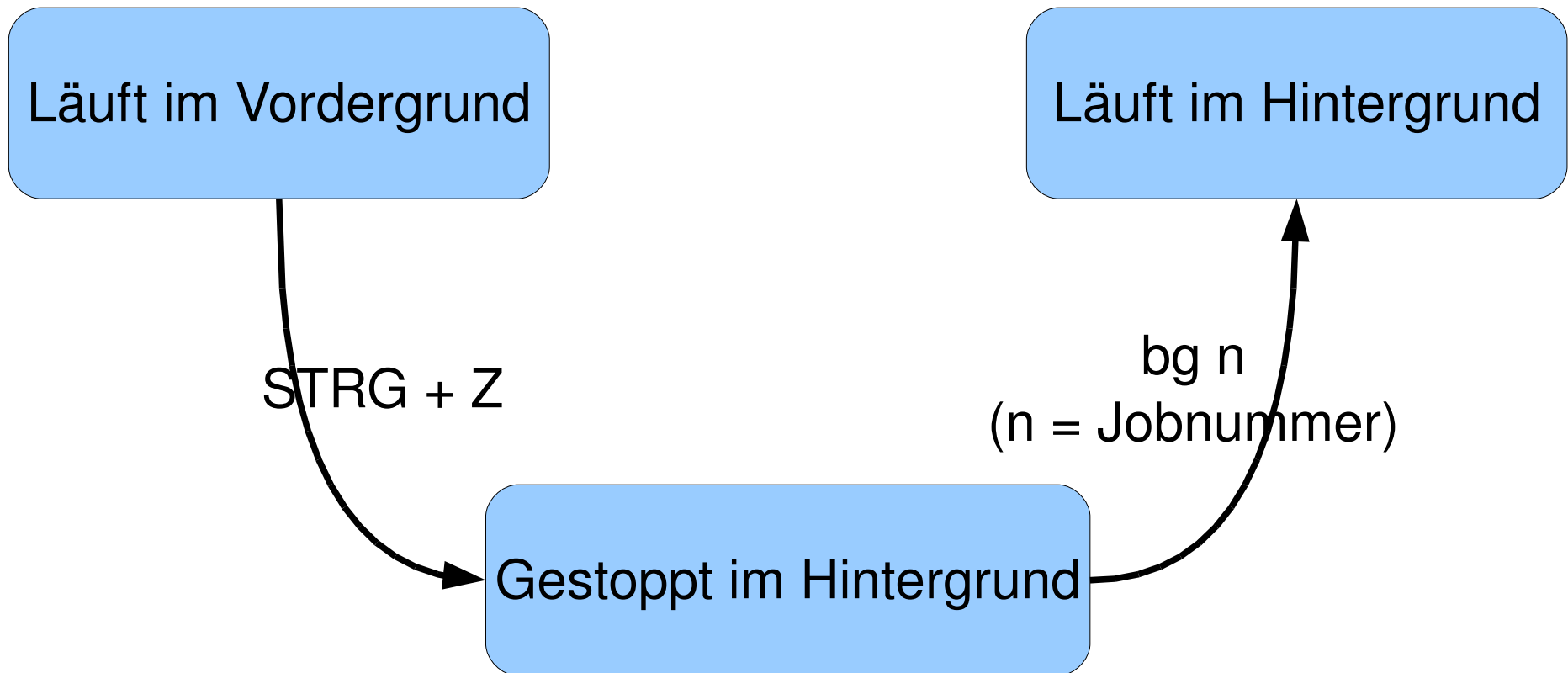
- Prozesse in den Hintergrund
  - Beim Starten
  - Nach dem Start
- Prozesse aus dem Hintergrund
- Was läuft?

# In den Hintergrund I



- Beim Start
  - BEFEHL &
  - BEFEHL&
- Während der Laufzeit
  - STRG + Z
  - bg n (n = Jobnummer)

# In den Hintergrund II



# In den Vordergrund



- %n (n Jobnummer)
- %NAME
- %?ME
- %%
- %+
- %-
- fg

# Was läuft?



- Hintergrundjobs abfragen

– jobs

```
root@TM2410:~# jobs  
[1]+  Running                  vpnclient connect uni-hd-vpn &
```

# Konfiguration



- Anmeldung
- Start der bash
  - Grundsätzlich
  - Für meinen User
- Logout

# Anmeldung

---

- /etc/passwd
  - Anmeldedaten
    - Ändern mit passwd
  - Gewünschte Shell
    - Ändern der Shell mit chsh -s [SHELL]
    - z.B.: chsh -s /bin/bash
  - Homeverzeichnis

# Start der bash



- /etc/profile wird für alle User verarbeitet
- User spezifisch:
  - ~/.bash\_profile
  - ~/.bash\_login
  - oder
  - ~/.profile

# Logout



- `~/.bash_logout`
  - Wird mit dem Logout verarbeitet

# Zusammenfassung



- Kommandovarianten
- Handhabung
  - TAB
  - History
  - Ein/Ausgabeumlenkung
  - Aliase
  - Jobkontrolle
- Startkonfiguration

# Quellen



- <http://www.digilife.be/quickreferences/QRC/Bash%20Quick%20Reference.pdf>
- <http://johannes.truschnigg.info/tut/bash-tut.sxhtml>
- <http://noone.org/shell-efficiency/shell-efficiency-dalug.web.html>
- <http://userserv.hochschule-reutlingen.de/~hug/Lehrveranstaltungsmaterial/I3/Vort>
- [http://www.selflinux.org/selflinux-devel/pdf/bash\\_basic.pdf](http://www.selflinux.org/selflinux-devel/pdf/bash_basic.pdf)
- [http://de.wikibooks.org/wiki/Linux-Kompendium:\\_Bourne\\_Again\\_Shell](http://de.wikibooks.org/wiki/Linux-Kompendium:_Bourne_Again_Shell)
- <http://www.hdm-stuttgart.de/~ms096/unix.pdf>
- [http://de.wikipedia.org/wiki/Tee\\_%28Unix%29](http://de.wikipedia.org/wiki/Tee_%28Unix%29)
- [http://de.wikipedia.org/wiki/Pipe\\_\(Informatik\)](http://de.wikipedia.org/wiki/Pipe_(Informatik))

# Die Shell



## Einführung in die Bourne Again Shell

Marco Pfeifer

Proseminar: Linux für Umsteiger

Betreuer: Julian Martin Kunkel, Olga Mordvinova

03.06.2008

1

Merke: Ich sollte unbedingt mehr Notizen addieren(!)

Also: Notiz1 + Notiz2 + Notiz3 = Notiz1-3

Vor allem auf den Übersichtsseiten. Dabei sind das doch nur Folien um den Überblick zu behalten und ohne wirklichen sachlichen Inhalt.

# Übersicht



- Allgemeines
- Handhabung
  - Tasten und Tastenkombinationen
  - Wildcards
  - Ein/Ausgabeumlenkung
  - Aliase
  - Jobkontrolle
- Konfiguration
- Quellen

03.06.2008

2

Allgemeines: Wozu eigentlich eine Shell? Was ist eine Shell? Was sind Kommandos (Build-In, Programme, Funktionen, Skripte)?

Handhabung: Wie benutze ich die bash (eine Shell)? Und das möglichst effizient? Welche Tastenkombinationen gibt es? Was sind Wildcards? Was kann man wohin umleiten? Wozu braucht man Röhren (Pipes)? Sind Aliase nur was für Faule oder lässt sich damit die Effizienz steigern? Was läuft im Hintergrund? Und wie kommt es da hin?

# Allgemeines



- Motivation
- Kommandovarianten
  - Eingebaute Kommandos
  - Richtige Programme
  - Funktionen
  - Skripte


03.06.2008

3

Wozu eine Shell?

Unterschied zwischen eingebauten Kommandos, Programmen, Funktionen und Skripten.

# Motivation

- 
- Befehlsübermittler vom User zum OS
  - Prinzipiell gleiche Aufgabe wie GUI



Kommandozeileninterpreter

Woher weiß das OS was es tun soll? Man muss es ihm sagen über ein Interface. Dieses kann graphisch oder textbasiert sein. Textbasierte Interfaces nennt man Kommandozeileninterpreter. Deshalb gibt es Shells und speziell die Linux Standardshell, die Bourne Again Shell oder bash.

# Eingebaute Kommandos



- Build-In Commandos
- Werden direkt in bash ausgeführt
- Beispiele
  - alias
  - exit
  - history
  - jobs

03.06.2008

5

Diese Kommandos sind direkt in der bash integriert.  
Sozusagen zur Verfügung gestellte Befehle.

# Richtige Programme



- Auch externe Kommandos
- Werden durch bash gestartet
- Beispiele
  - vpnclient
  - vi
  - cat

Ausführbare nicht integrierte Dateien die von der Shell gestartet werden.

# Funktionen

- Gruppierungen von Befehlen
- Erzeugen und verwerfen:

```
pfeifer@TM2410:~$ hallo ()  
> {  
> echo 'Herzlich willkommen, '$1'!'  
> }  
pfeifer@TM2410:~$ hallo Kurs  
Herzlich willkommen, Kurs!  
pfeifer@TM2410:~$ unset hallo  
pfeifer@TM2410:~$ hallo  
bash: hallo: command not found  
pfeifer@TM2410:~$ █
```

03.06.2008

7

Funktionen kann man sich schnell selbst erzeugen.  
Sozusagen integrierte meist kleinere Skripte.

# Skripte



- Textdatei
- Mit Ausführungs-Berechtigung
- Kein unbeabsichtigtes Starten
- `./Testskript.sh`

03.06.2008

8

Ausführbare nicht integrierte Textdateien Um nicht versehentlich ein Skript zu starten muss es aus Sicherheitsgründen absichtlich mit `./` gestartet werden.

# Tasten und Tastenkombinationen



- Einfachste Tastenkombinationen
- Einfache Tastenkombinationen
- Tabulator-Taste
- History

03.06.2008

9

Hier werden einfache Tastenkombinationen erklärt.  
Und auch mächtigere Tastenfunktionen wie TAB  
und die History.

## Einfachste Tastenkombinationen



- PFEIL VOR / ZURÜCK
- BACKSPACE / ENTF
- POS1 / ENDE

03.06.2008

10

Wie bewege ich den Cursor? Wie entferne ich einzelne Zeichen vor oder hinter dem Cursor?

# Einfache Tastenkombinationen I



- STRG + A
  - Zeilenanfang (wie POS1)
- STRG + E (wie ENDE)
  - Zeilenende
- ALT + f / ALT + b
  - Wort vor / zurück
- STRG + k
  - Löscht Zeile bis Ende

03.06.2008

11

Cursor-Bewegung:

STRG + A: Zeilenanfang

STRG + E: Zeilenende

ALT + b / ALT + f : Wort zurück / vor

STRG + k: löscht Zeilenrest

## Einfache Tastenkombinationen II



- STRG + t
  - Vertauscht die letzten beiden Buchstaben
- ALT + t
  - Vertauscht die letzten beiden Worte
- STRG + I
  - Löscht Bildschirm (clear)

03.06.2008

12

Tippfehler/Syntaxfehler ausgleichen:

STRG + t: tauscht Buchstaben

ALT + t: tauscht Worte

Neugierige Menschen?

STRG + I: löscht Bildschirm

# Die Tabulator-Taste I



- TAB
  - Ergänzt angefangene Namen
- Bei Mehrdeutigkeit: Erneut TAB
  - listet Möglichkeiten auf
- Hauptsächlich für Pfadnamen

03.06.2008

13

Tab ergänzt, falls kein Ergebnis erscheint nochmals drücken für Liste der Möglichkeiten. In den allermeisten Fällen werden Pfadnamen ergänzt.

## Die Tabulator-Taste II



- \$ Variable
- ~ Benutzername
- @ Hostname
- Ohne Symbol
  - Alias
  - Kommando
  - Pfadname

03.06.2008

14

Falls Symbol davor wird nur das entsprechende geprüft.

Falls kein Symbol davor, wird erst zuletzt auf Pfadname geprüft.

# Die History



- Blättern
  - PFEIL HOCH
  - PFEIL RUNTER
- Suchen
  - STRG + R
    - Enter
    - ESC / PFEIL LINKS / RECHTS
- history

03.06.2008

15

Hoch / Runter zum Blättern in der History. Normales Editieren möglich.

Bei STRG + R muss, falls Editierung des Befehls gewünscht ESC (PFEIL LINKS / RECHTS) gedrückt werden.

# Wildcards



- \*
  - Beliebige Zeichen
- ?
  - Beliebiges Zeichen
- [...]
  - Zeichenklasse
- [!...]
  - Negierte Zeichenklasse

03.06.2008

16

\* Beliebige Anzahl beliebiger Zeichen

? Ein beliebiges Zeichen

[...]: Eines der Zeichen in der Klammer

[!...]: Keines der Zeichen in der Klammer

# Wildcard-Verwendung



- d[kgt]r
  - dkr
  - dgr
  - dtr
- `root@TM2410:~# mv *.gif Bilder/`
  - Verschiebt alle gifs in Bilder

03.06.2008

17

Erstes steht für, die angegebenen Möglichkeiten.  
Und ein konkretes Beispiel.

## Ein/Ausgabeumlenkung



- Ein/Ausgaben
- Eingabeumleitung
- Ausgabeumleitung
  - Anhängen
  - Fehler umleiten
- Ausgabe- und Eingabeumleitung
- Pipe
- Tee

03.06.2008

18

Was gibt es für Standardein- und Ausgaben?  
Anderer Eingang als Tastatur. Andere Ausgabe als Bildschirm. In Datei ausgeben/anhängen.  
Fehlerausgabe umleiten. Ausgabe und Fehlerausgabe kombinieren. Pipe rein, Pipe raus.  
Was tut tee? Tee als Beispiel für eine Pipeline.

# Ein/Ausgaben



- `stdin` (0)
  - Standardeingabe
- `stdout` (1)
  - Standardausgabe
- `stderr` (2)
  - Standardfehlerkanal

03.06.2008

19

Röhren.

Geöhnlich `stdin`:Tastatur. `Stdout`, `stderr`: Monitor. Alle sind wie Dateizugriffe. Und können daher auch auf Dateien oder Programme gelegt werden. Zahlen sind die Deskriptoren dieser Dateien.

# Eingabeumleitung



- Eingabeumlenkung durch <

- BEFEHL < DATEI

- ```
root@TM2410:~# wc -w < test.txt
615
```

- Zählt Worte in test.txt

Ohne die Eingabeumlenkung müsste man erst den Text abschreiben. Und mit STRG + d beenden.

# Ausgabeumleitung I



- Ausgabeumlenkung durch >

– BEFEHL > DATEI

```
– root@TM2410:~# wc -w > worte.txt  
Hallo ihr da!  
root@TM2410:~# cat worte.txt  
3
```

- Schreibt Wortzahl in worte.txt

03.06.2008

21

Wc starten, Worte eingeben, mit STRG + d beenden.  
Mit cat worte.txt überprüfen. Datei worte.txt wird  
erzeugt oder gegebenenfalls überschrieben.

# Ausgabeumleitung II



- Ausgabeumlenkung durch >>

- BEFEHL >> DATEI

```
root@TM2410:~# wc -w >> worte.txt
- Ja ihr da drüben!
root@TM2410:~# cat worte.txt
3
4
```

- Hängt Wortzahl an worte.txt an

03.06.2008

22

Wc starten, Worte eingeben, mit STRG + d beenden.  
Mit cat worte.txt überprüfen. Datei worte.txt wird  
erzeugt oder gegebenenfalls erweitert.

# Fehler umleiten



- Dateideskriptor 2 (stderr) in Datei schreiben
  - PROGRAMM 2> error.txt
- Diesen an Datei anhängen
  - PROGRAMM 2>> error.txt
- Dateideskriptor 1 und 2 (stdout, stderr) umleiten
  - PROGRAMM > beides.txt 2>&1

03.06.2008

23

Letztes: Normale Ausgabe in Datei umleiten.  
Fehlerausgabe mit Normalausgabe verknüpfen ->  
Beides in der Datei.

# Ausgabe- und Eingabeumleitung



- Sowohl Eingabe als Ausgabe umleiten

- Befehl < Eingabedatei > Ausgabedatei

```
root@TM2410:~# wc -w < test.txt > worte.txt  
root@TM2410:~# cat worte.txt  
615
```

- Zählt in test.txt und schreibt in worte.txt

Klar kommt jetzt auch das Zählen in einer Datei und das Schreiben in eine solche.

# Pipe



- Röhre
- Gepufferter Datenstrom
- Zwischen Prozessen
- FIFO
- |

03.06.2008

25

Pipe??? Ein gepufferter FIFO-Datenstrom zwischen Prozessen. Wird durch | dargestellt.

# Pipe-Verwendung



- Befehl mit Ausgabe | Befehl mit Eingabe

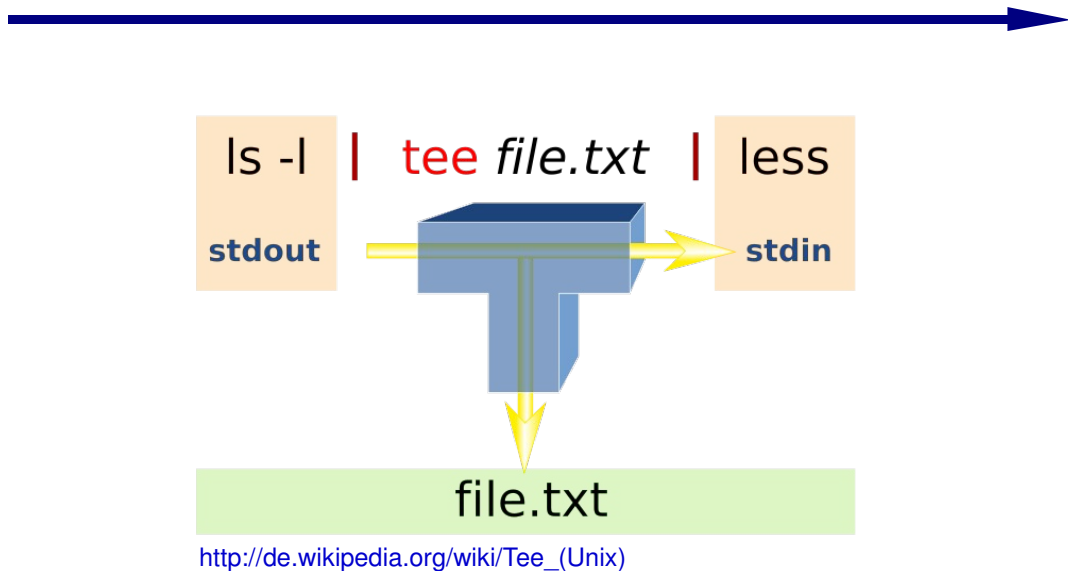
- ```
root@TM2410:~# ls | wc -w  
18
```

- Zählt den Verzeichnisinhalt

- Mehrere Pipes in Kombination: Pipeline

Ausgeb von ls erscheint nicht, sondern wird auf wc als Eingabe weitergereicht.

# Tee



03.06.2008

27

Ausgabe von ls wird zu tee gepiped, von dort wird es zu less gepiped. Tee ist einfach nur Kommando mit Parameter, dessen Aufgabe es ist in eine Datei zu schreiben und das Erhaltene weiter zu reichen. Damit Anwendungsbeispiel für Pipeline.

# Aliase



- Kommando für ein anderes
- Wozu?
- Verwaltung

03.06.2008

28

Was ist ein Alias? Was verbirgt er? Und warum?  
Handhabung.

## Aliase - Wozu?



- Weniger tippen
- Leichter merken
- Tippfehler korrigieren

03.06.2008

29

Mögliche Gründe. Alle für Bequemlichkeit.

# Aliase verwalten



- Anlegen
  - alias NEUESKOMMANDO='KOMMANDO'
  - `root@TM2410:~# alias ll='ls -l'`
- Aufheben
  - unalias KOMMANDO
  - `root@TM2410:~# unalias ll`
- Anzeigen
  - alias

03.06.2008

30

Umgang mit Aliasen.

# Jobkontrolle



- Prozesse in den Hintergrund
  - Beim Starten
  - Nach dem Start
- Prozesse aus dem Hintergrund
- Was läuft?

03.06.2008

31

Wie werde ich lange Prozesse los damit ich weiter arbeiten kann? Und wie hole ich sie zurück? Was läuft eigentlich im Hintergrund?

## In den Hintergrund I



- Beim Start
  - BEFEHL &
  - BEFEHL&
- Während der Laufzeit
  - STRG + Z
  - bg n (n = Jobnummer)

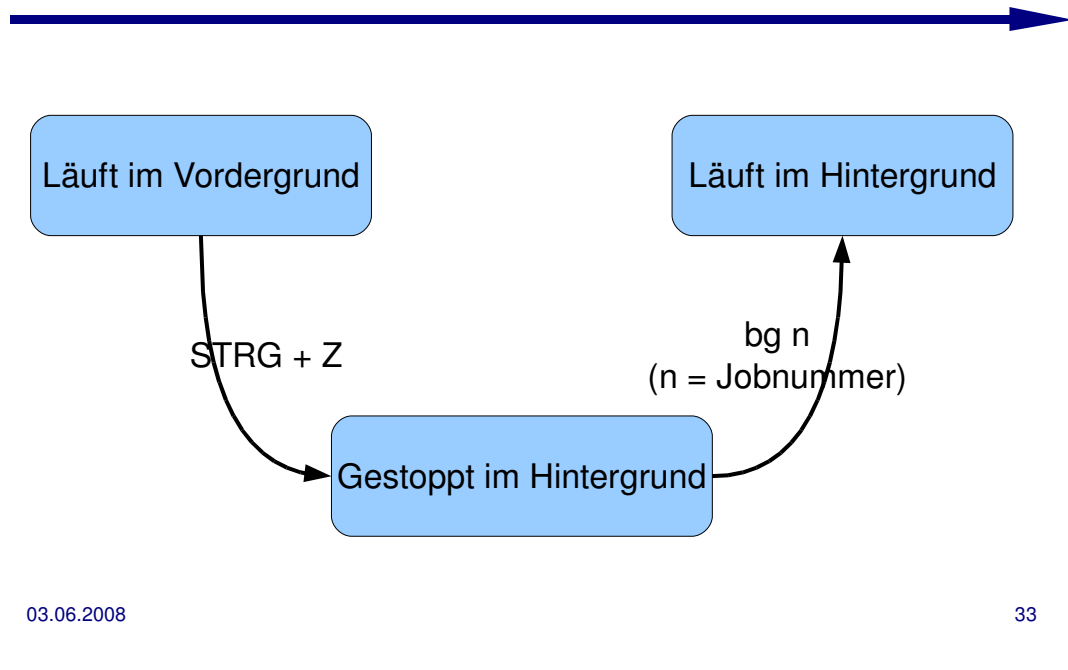
03.06.2008

32

& mit oder ohne Leerzeichen direkt in den Hintergrund..

Laufend Prozesse in den Hintergrund und dort laufen lassen.

## In den Hintergrund II



Schema um Prozesse nach dem Start im Hintergrund laufen zu lassen.

## In den Vordergrund



- %n (n Jobnummer)
- %NAME
- %?ME
- %%
- %+
- %-
- fg

03.06.2008

34

%n: Jobnummer

%NAME: Name des Aufrufs oder dessen Beginn wenn eindeutig.

%?ME: Name mit Platzhalter. ? Auch für mehrere Zeichen.

%%, %+ , fg (wenn ohne Parameter): zuletzt gestarteter Job.

%-: vorletzter Job

# Was läuft?



- Hintergrundjobs abfragen

– jobs

```
root@TM2410:~# jobs  
[1]+  Running                  vpnclient connect uni-hd-vpn &
```

03.06.2008

35

## Anzeige der Hintergrundjobs

# Konfiguration



- Anmeldung
- Start der bash
  - Grundsätzlich
  - Für meinen User
- Logout

03.06.2008

36

Wo werden Einstellungen für Login/Logout und den bash-Start gemacht?

# Anmeldung



- /etc/passwd
  - Anmeldedaten
    - Ändern mit passwd
  - Gewünschte Shell
    - Ändern der Shell mit chsh -s [SHELL]
    - z.B.: chsh -s /bin/bash
  - Homeverzeichnis

03.06.2008

37

In /etc/passwd stehen alle Userdaten inkl. Shell und Homeverzeichnis. Das Passwort liegt dort verschlüsselt.

# Start der bash



- /etc/profile wird für alle User verarbeitet
- User spezifisch:
  - ~/.bash\_profile
  - ~/.bash\_login
  - oder
  - ~/.profile

03.06.2008

38

Globale Einstellungen in /etc/profile können falls nicht explizit verboten von User-Einstellungen überschrieben werden. Bei den User-Files wird nur das Erst-Gefundene ausgeführt.

# Logout



- `~/.bash_logout`
  - Wird mit dem Logout verarbeitet

User Befehle in `~/.bash_logout` werden bei seinem Logout verarbeitet.

# Zusammenfassung



- Kommandovarianten
- Handhabung
  - TAB
  - History
  - Ein/Ausgabeumlenkung
  - Aliase
  - Jobkontrolle
- Startkonfiguration

03.06.2008

40

Wir haben die Kommandovarianten, komplexere Tastenbefehle, die Histor, Ein/Ausgabeumlenkung, Aliase, Jobkontrolle und Bash Konfigurationsmöglichkeiten kennen gelernt.

# Quellen



- <http://www.digilife.be/quickreferences/QRC/Bash%20Quick%20Reference.pdf>
- <http://johannes.truschnigg.info/tut/bash-tut.sxhtml>
- <http://noone.org/shell-efficiency/shell-efficiency-dalug.web.html>
- <http://userserv.hochschule-reutlingen.de/~hug/Lehrveranstaltungsmaterial/I3/Vort>
- [http://www.selflinux.org/selflinux-devel/pdf/bash\\_basic.pdf](http://www.selflinux.org/selflinux-devel/pdf/bash_basic.pdf)
- [http://de.wikibooks.org/wiki/Linux-Kompendium:\\_Bourne\\_Again\\_Shell](http://de.wikibooks.org/wiki/Linux-Kompendium:_Bourne_Again_Shell)
- <http://www.hdm-stuttgart.de/~ms096/unix.pdf>
- [http://de.wikipedia.org/wiki/Tee\\_%28Unix%29](http://de.wikipedia.org/wiki/Tee_%28Unix%29)
- [http://de.wikipedia.org/wiki/Pipe\\_\(Informatik\)](http://de.wikipedia.org/wiki/Pipe_(Informatik))

03.06.2008

41

Notiz, Notiz, ...???  
Dort habe ich beschrieben!