



Einführung in die Skriptprogrammierung

Markus Dahms
<mad@automagically.de>



Motivation – Aufgaben

- Automatisierung (Dateiverwaltung, Backup, Konvertierung...)
- parametrisierter Programmstart, bedingte Optionen
- „Rapid Prototyping“



Shell – was ist das?

- interaktive Skriptsprache
- `ash`, `bash`, `cs``h`, `dash` . . . `zsh`
- `sh` (Bourne Shell) De-facto-Standard
- `bash` Standard auf vielen Linux-Systemen
- wenige eingebaute Kommandos
- starke Nutzung der kleinen Hilfsprogramme im Suchpfad (`/usr/bin`, `/bin` . . .)



Shell – wann nicht?

- bei wenig Ein-/Ausgabe
- wenn es schnell gehen soll
- fast jeder Befehlsaufruf (z.B. Vergleich mit test) erzeugt neuen Prozess → kostet Zeit
- Ausweichmöglichkeiten sind schnellere Skriptsprachen (Perl...) oder „echte“ Programmiersprachen (C...)



Shell – Konzepte

- Rückgabewert eines Prozesses
- Standard-Eingabe und -Ausgabe, Fehlerausgabe
- Zeilenorientierung
- Vorder-/Hintergrundprozesse
- Shebang (*#!interpreter*)



Shell – Hilfe bekommen

- `man bash`
- `help` und `help <builtin>`
- `man <command>` oder `info <command>`
- `<command> (--help|-help|-h)`
- andere Skripte lesen (z.B. `/etc/init.d/*`)
- Debugging aktivieren (`set -x`)
- ein Buch lesen, jemanden fragen...



Shell – Variablen

```
# Zuweisung  
var1=test  
var2="value with space..."  
var3='ls'
```

```
# Expansion  
echo $var1  
echo ${var2}bla
```

```
$var3
```



Shell – Spezialvariablen

```
true ; echo $?
```

```
nc -l -p 12345 & pid=$!  
sleep 5 ; kill $pid
```

```
echo $PWD
```

```
cd $HOME
```

```
touch tmp-$$-$RANDOM
```

```
set | grep BASH_
```




Shell – Anführungszeichen

```
dir=/etc
```

```
echo ls -1 $dir
```

```
echo "ls -1 $dir"
```

```
echo 'ls -1 $dir'
```

```
echo `ls -1 $dir`
```

```
echo `ls -1 "$dir" `
```

```
echo `ls -1 '$dir' `
```



Shell – E/A-Umleitungen

```
grep -r foobar /etc 2>/dev/null
```

```
ls -1 > files
```

```
sort < /etc/passwd
```

```
du -s */ | sort -nr
```

```
find . -name '*.[ch]' | xargs du -b | \  
sort -nr | head -10 2>/dev/null >top10.txt
```



Shell – Bedingungen

```
if [ ! -d /tmp/bla ]
then
    echo "Verzeichnis existiert nicht"
fi

# andere Schreibweise
if test ! -d "/tmp/bla" ; then \
echo "Verzeichnis existiert nicht" ; fi
```



Shell – bedingte Schleifen

```
while ! ping -c 1 hostname >/dev/null ; do
    echo -n "."
    sleep 1
done

i=0
until [ $i = 20 ] ; do i='expr $i + 1'; \
echo $i; done
```



Shell – Zählschleifen u.Ä.

```
IFS=""  
"  
for f in `ls -1 *.jpg`; do  
    convert "$f" -geometry 1024x1024 "s/$f"  
done  
  
# bash only!  
for (( i = 0 ; i < 10 ; i ++ )); do  
    echo $i  
done
```



Shell – Funktionen

```
bla_blubb () {  
    bla=$1  
    blubb=$2  
  
    echo "$bla$blubb"  
}  
  
bla_blubb "foo" "bar"
```



Shell – Links

- http://www-user.tu-chemnitz.de/~hot/unix_linux_werkzeugkasten/bash.html
- <http://tldp.org/HOWTO/Bash-Prog-Intro-HOWTO.html>