

Skriptprogrammierung Teil II

Themen

- ▶ Verwendete Shell ist grundsätzlich die **bash**
Andere Shells können sich teilweise gravierend unterscheiden!
- ▶ Themen
 - Skripte
 - Kontrollstrukturen
 - Auswahlanweisungen
 - Schleifen
 - Shell-Funktionen
 - Reguläre Ausdrücke

Ressourcen

- ▶ wie Teil I, außerdem:
- ▶ Friedl, J.E.F.:
Reguläre Ausdrücke
O'Reilly & Associates, Köln 1997
ISBN 3-930673-62-2 (DM 59,-)

Skripte

- ▶ Ein Shellskript ist eine reguläre Textdatei, die
 - ausführbar sein sollte
 - in der ersten Zeile eine Angabe enthält, mit welcher Shell der Text interpretiert werden soll: `#!/bin/bash`
- ▶ Kommentarzeichen `#`
 - schließt den Rest einer Zeile von der Bearbeitung durch die Shell aus
 - Die Kombination `#!` in der ersten Zeile ist ein Pseudo-Kommentar

Parameter

- ▶ Jedem Skript können Parameter übergeben werden
 - Variable `$#` enthält die Anzahl Parameter
 - nullter Parameter (`$0`) Skriptname
 - erster (`$1`), zweiter (`$2`) bis neunter (`$9`) Parameter
 - weitere Parameter über Umweg (`shift`)
 - ähnlich wie bei Feldern enthält `$*` alle Parameter in einer Zeichenkette, `$@` jeden Parameter in einer einzelnen Zeichenkette

Fallunterscheidung

- ▶ `if kommandoliste; then`
 `anweisungen1`
`else`
 `anweisungen2`
`fi`
- ▶ Betrachtet wird der Rückgabestatus des letzten Kommandos aus `kommandoliste`. Rückgabestatus 0 bedeutet: Bedingung erfüllt
- ▶ Auch `true` und `false` sind Kommandos!

case

```
▶ case wort in
  muster1)
    anweisungen
    ;;
  muster2)
    anweisungen
    ;;
*)
  anweisungen
  ;;
esac
```

Schleifen

```
▶ while kommandoliste; do
  anweisungen
done

▶ for i in liste; do
  anweisungen
done

▶ for (( expr1; expr2; expr3 )); do
  anweisungen
done
```

Funktionen

```
▶ function funktionsname() {  
    kommandoliste;  
}
```

- ▶ Aufruf: `funktionsname parameter`
- ▶ Rückgabewerte nur `true` und `false`
- ▶ lokale Variable möglich:
 `local localvar="lokale Variable"`
 - globale Variable gleichen Namens wird nicht geändert

reguläre Ausdrücke

- ▶ Metazeichen
 - `$` Zeilenende-Anker
 - `^` Zeilenanfangsanker
 - `\x` Aufheben der Sonderbedeutung von `x`
 - `[x]` Zeichenklasse:
 Einzelzeichen oder Zeichenbereich
 - `[^x]` Zeichenklasse aller Zeichen
 mit Ausnahme der in `x` beschriebenen
 - `x*` der Ausdruck `x` kommt 0-mal oder öfter vor
 - `x+` der Ausdruck `x` kommt mehrmalig vor
 - `x|y` Alternative
 - `.` genau ein beliebiges Zeichen