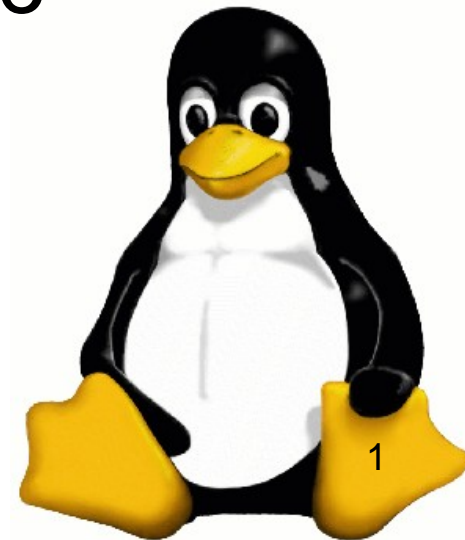


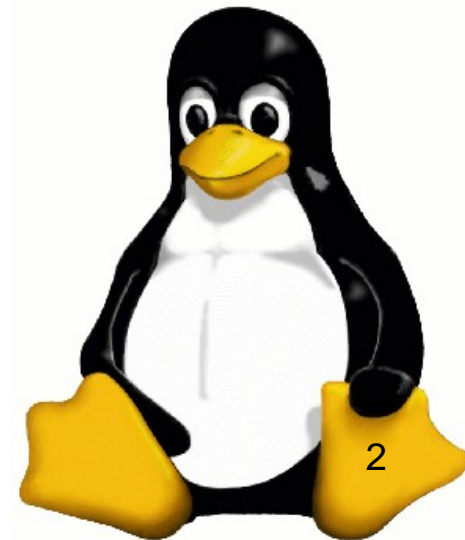
# Shell

Manuel Blechschmidt  
&  
Volker Grabsch  
CdE Sommerakademie 2006  
Kirchheim



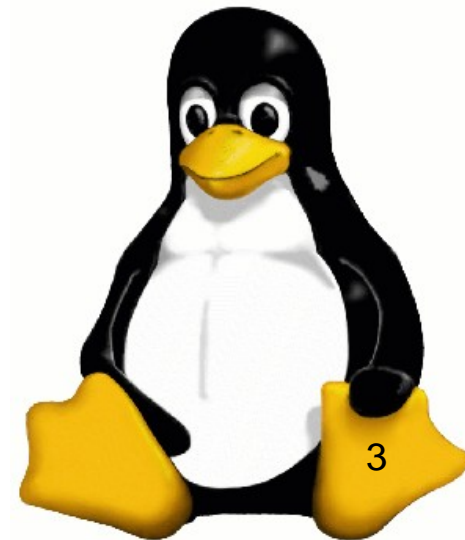
# Gliederung

- Unsere Shell die Bourne Again Shell kurz bash
  - Geschichte
- Funktionsweise
  - Programme starten
  - Argumente
  - Ausgabeumleitung
- Prozessverwaltung
- Wichtige Kommandos
- Tipps & Trick



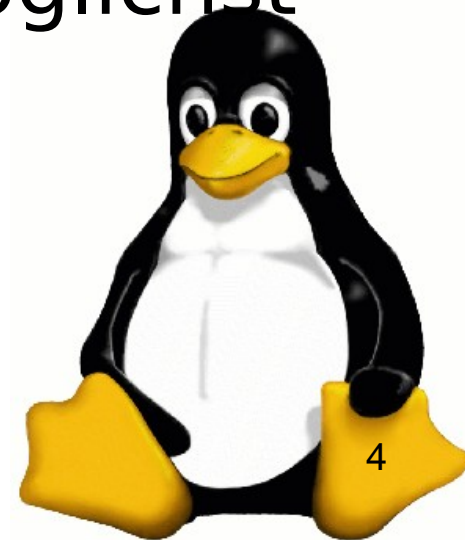
# Geschichte

- Die Bash
  - Geschrieben von Brian Fox und Chet Ramey  
Ende der 80er Jahre
- Meistverwendete Shell auf Linux Systemen



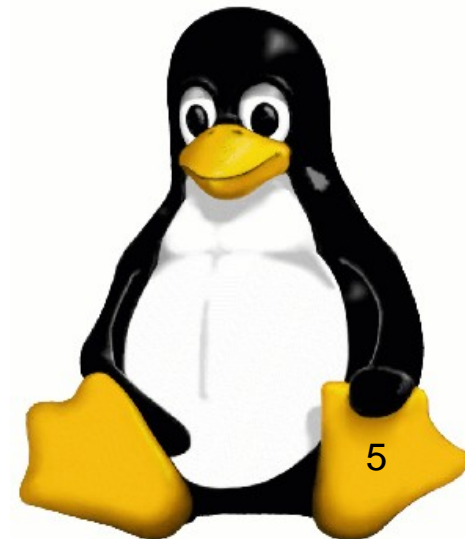
# Funktionsweise

- Die Shell startet weitere Prozesse für uns
  - Wir geben in der Shell ein Programm ein
  - Die Shell startet das Programm
  - Die Shell berichtet uns, ob das Programm ohne Fehler abgelaufen ist
- Die Shell hilft uns Programme möglichst effizient zu starten und zu überwachen



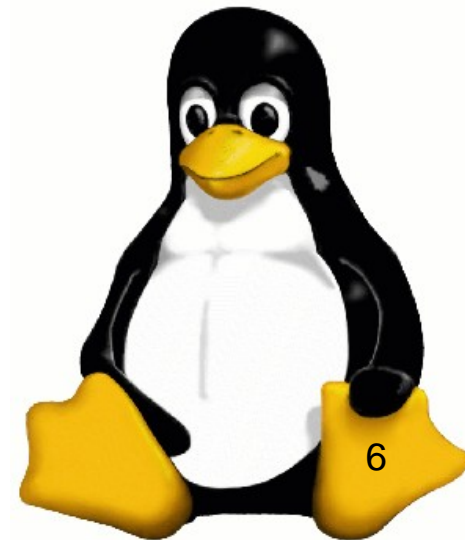
# Kommandos

- ls – Listet alle Dateien und Ordner im Verzeichnis aus
  - \$ ls
- cd – Wechselt in ein anderes Verzeichnis
  - \$ cd ~
- cat – Gibt eine Datei auf der Standardausgabe aus
  - \$ cat /etc/passwd
- echo – Gibt das Argument aus
  - \$ echo „Hallo Welt“



# Programm starten

- `$ programmname <Eingabe>`
- Beispiel:
  - `$ eject`
- Programm abbrechen mit `Ctrl+C`
- Programm im Hintergrund starten
- `$programmname & <Eingabe>`
- Beispiel
  - `$ emacs &`
  - Zurückkehren mit `fg`



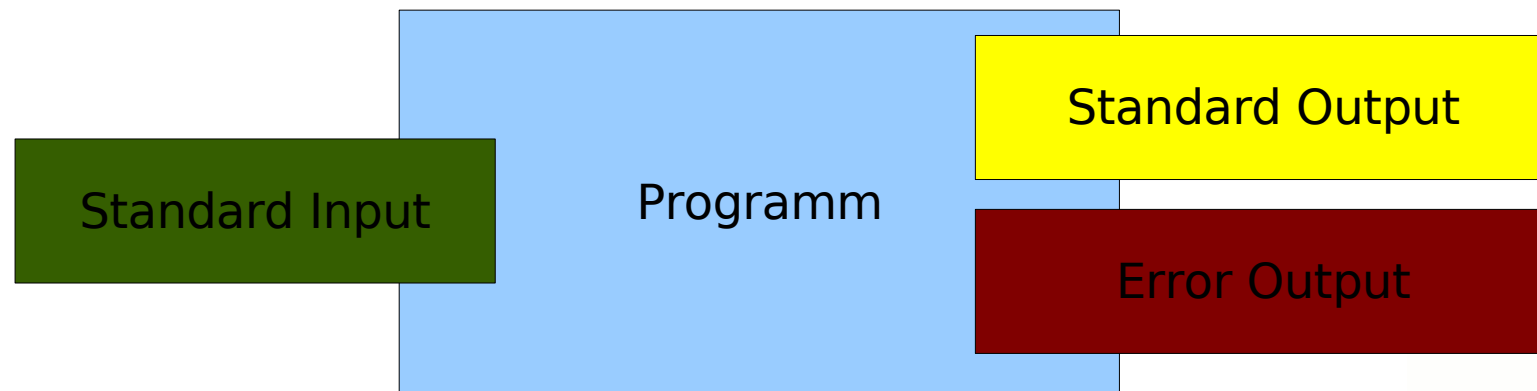
# Argumente

- Die meisten Programme machen alleine keinen Sinn, man muss Argumente übergeben
- Beispiel:
  - \$ sleep 10
- Liste der möglichen Argumente erhalten
  - \$ programname --help
  - \$ programname -help
  - \$ man programname



# Ausgabe umleiten

- Programme verfügen über 3 Streams



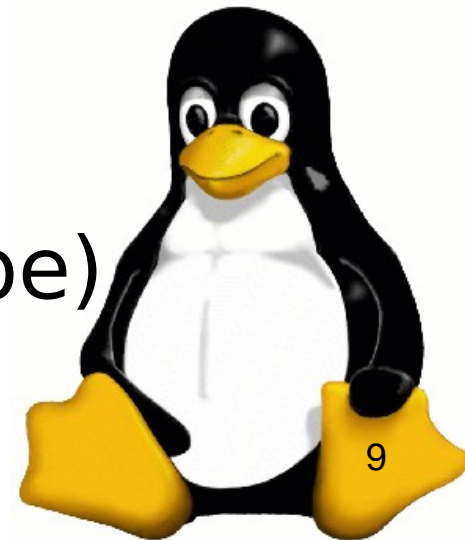
- Diese Ausgaben kann man umleiten





# Standardoutput umleiten

- Standard Output in eine Datei umleiten
  - `$ programmname > meineAusgabe`
  - `$ programmname >> anhangDatei`
  - `$ ls > meinOrdner`
- Datei in Programm umleiten
  - `$ programmname < datei`
  - `$ cat < /etc/passwd`
- Ausgabe in Eingabe umleiten (Pipe)
  - `$ programmname | programmname`
  - `$ ls -l | grep test`



# Aufgaben

- Legt eine Datei mit dem Inhalt „Hallo Welt“ mit einer Umleitung und dem Befehl echo an
- Hängt Sie an die Datei die Datei passwd an
- Filtern Sie alle Einträge aus der neuen Datei, die das Wort root beinhalten



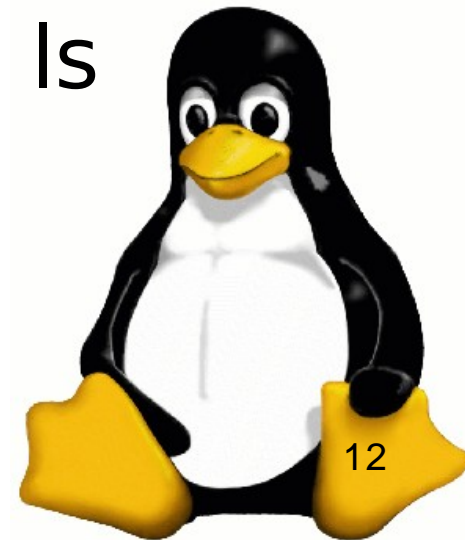
# Prozessverwaltung

- Prozesse sind Programme, die momentan auf dem Computersystem ausgeführt werden
  - \$ ps fax # Zeigt Prozessbaum an
  - \$ top # Interaktive Prozessanzeige
- kill <pid> tötet einen Prozess
  - \$ kill 1456
  - \$ kill -9 1567 # Töten mit viel Gewalt
- killall <programmname> tötet
  - \$ killall yes



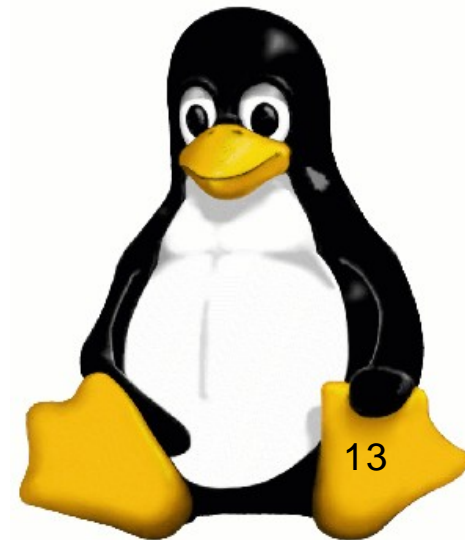
# Aufgaben

- Startet einen yes Prozess im Hintergrund, der die gesamte Zeit hallo ausgibt
- Leitet die Ausgabe dabei in eine Datei aus
- Findet die pid dieses Prozesses raus
- Killt den Prozess
- Wie groß ist die Datei? Tipp: man ls
- Löscht die Datei



# Wichtige Kommandos

- Datei verwaltung  
touch, mkdir, mv, cp, ln, rm, rmdir
- Prozessverwaltung  
top, ps, kill, killall
- Anzeige  
less, more
- Editoren  
vi, emacs
- Pipes  
tr, grep, sed



# Tipps und Tricks

- Autovervollständigung mit Tab
- Mit der Pfeiltaste nach oben, werden alte Kommandos angezeigt

