

Paradigmen der Programmierung

Übung 2
grep und sed

Jobs

- Während Programme laufen, kann man auf der Shell nichts anderes machen
- Manche Programme sollen die ganze Zeit im Hintergrund laufen
- Lösung: Prozesse in den Hintergrund schieben mit &:
`find *.pdf > pipe1 &`
- Laufenden Prozess suspenden: Strg+Z

Jobs

jobs - list the current jobs

fg - resume the job that's next in the queue

fg %[number] - resume job [number]

bg - Push the next job in the queue into the background

bg %[number] - Push the job [number] into the background

kill %[number] - Kill the job numbered [number]

kill -[signal] %[number] - Send the signal [signal] to job number [number]

Aufgabe

- Führen Sie folgendes aus:
`find *.pdf > pipe1`
- Suspenden Sie den aktuellen Prozess
- Was wird angezeigt, wenn Sie `jobs` ausführen?
- Starten Sie den Prozess wieder, der Prozess soll aber im Hintergrund bleiben
- Lesen Sie in einem anderen Terminal von `pipe1`
- Was wird jetzt angezeigt, wenn Sie `jobs` ausführen?

Grep

- `grep`: finde Zeichenkette in einer Datei
- **Beispiel:** `grep "http://" index.html`

Reguläre Ausdrücke

- Problem bei vorheriger Aufgabe: Was ist mit ftp – Adressen?
- Lösung: Reguläre Ausdrücke
- Spezifizieren Mengen von Zeichenketten
- Gehen weit über einfache Wildcards, wie *, hinaus

Reguläre Ausdrücke

- \ – Escape Code
- [AaBcCc] – Genau ein Zeichen davon
- [a-z] – Genau ein Kleinbuchstabe
- [^a-z] – Beliebiges Zeichen, das nicht [a-z] ist
- . – Ein beliebiges Zeichen
- * – Zeichen links davon beliebig oft wiederholt
Achtung: keine Wildcard
- a | b – a oder b

Reguläre Ausdrücke (2)

- (...) – Gruppenbildung
- \1 – Referenz auf erste Gruppe, z.B.
([0-9])[0-9]*\1 – Zahl, die mit gleicher Ziffer anfängt und endet
- \<, \> – Wortbeginn und -ende
\<Kai\> – Findet “Kai”, nicht “Kaiserslautern”
- ^ \$ – Nachfolgendes Zeichen muss am Anfang bzw. Ende der Zeile stehen, z.B.
^1\.11\.2011 – findet “1.11.2011, nicht
“11.11.2011”

Reguläre Ausdrücke (3)

- Gutes Tutorial:
http://www.id.uzh.ch/cl/dl/bs/unix/einf_15.html

SED

- Stream Editor, Werkzeug zum manipulieren von Zeichenketten
- Wichtigste (und einzige hier besprochene) Anwendung: suchen und ersetzen

SED

- `sed 's/<regexp>/<replacement>/<flags>'`
Ersetze überall <regexp> durch <replacement>
- `sed '/<rowregexp>/s/<regexp>/<replacement>/'`
Ersetze <regexp> durch <replacement>, aber nur in Zeilen, die <rowregexp> enthalten
- `sed '/<start>/,/<end>/s/<regexp>/<replacement>/' file`
Ersetze <regexp> durch <replacement>, aber nur in Zeilen zwischen Zeilen, die <start> und <end> enthalten

SED Skripte

- SED kann auch Abfolge von Befehlen aus einer Textdatei ausführen:

```
sed -f skript.sed datei.txt
```

- Inhalt der .sed-Datei sieht z.B. so aus:

```
/bad/d
```

```
s/pessimist/optimist/
```

```
...
```