Kommandozeilen-Interpreter

Am Beispiel der Bourne-Again Shell

Gliederung

- Was ist ein Kommandozeileninterpreter?
- Geschichte
- CLI-Beispiele
- Bash
- Aufbau eines Befehls
  - Parameter
  - Logische Verknüpfung von Befehlen
- Datenstrom-Umleitung
- Scripting
- DEMO

Was ist ein Kommandozeilen-Interpreter?

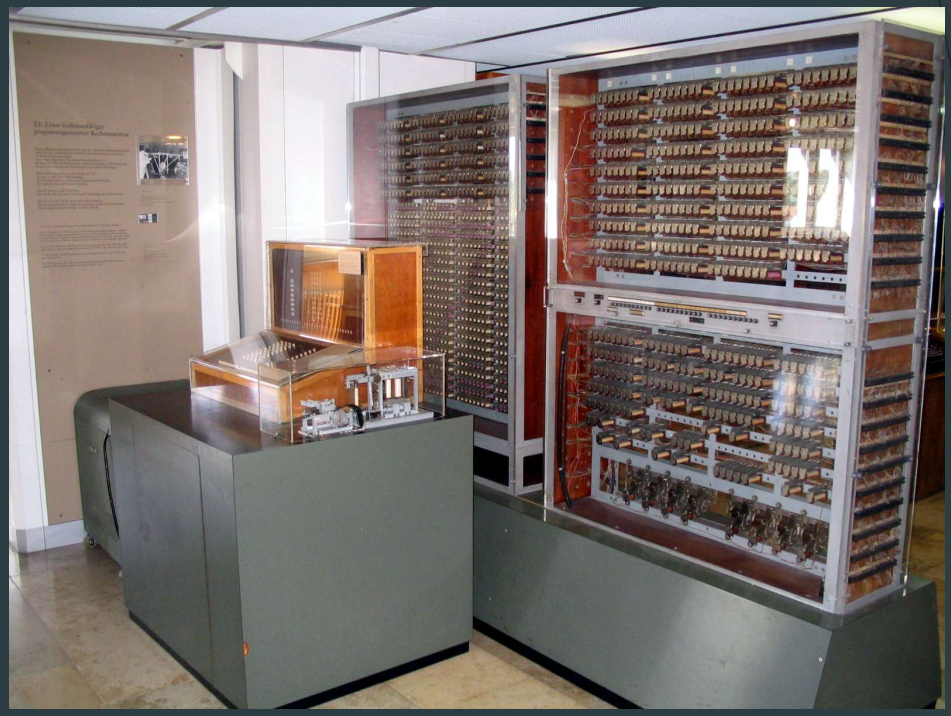
Was ist ein Kommandozeileninterpreter?

- Ein zeilenorientiertes Computerprogramm
  - Liest einzelne Zeilen (Kommandozeilen) als Befehle ein
  - Führt diese Befehle aus
  - Präsentiert das Ergebnis
- Auch Kommandointerpreter oder CLI (command-line interpreter) genannt
- Mensch-Maschine-Schnittstelle
  - Shell (Schale) um den Kernel (innerster Kern) eines Betriebssystems
- Ermöglicht Automatisierung durch Scripting
- Steuerung und Überwachung von Funktionen des OS

Geschichte

Zuse Z3 (1941)

- Von Konrad Zuse entwickelt
- 200 Byte Arbeitsspeicher
- Neuen verschiedene Befehle
- Programme wurden über Lochstreifen eingelesen
- Zahlen wurden über
  - Eine **Dezimaltastatur** eingegeben
  - Ein **Lampenfeld** ausgegeben



https://upload.wikimedia.org/wikipedia/commons/4/4c/Z3_Deutsches_Museum.JPG

Fernschreiber (1960) / TTY

- Bestehen aus elektrischen Schreibmaschinen
- In sternförmigem Netz zu Großrechnern verbunden
- Vom Computer getrennte Schnittstelle
- Oft mit Lochstreifenschreiber/-leser ausgestattet
- Ausgabe über elektrischen Schreibautomat auf Papier
- Eingabe über elektrische Tastatur
  - Signale werden herausgeleitet
  - Ermöglicht Vernetzung über mehrere hundert Meter (Telex-Netz)

http://ddr-fernmeldetechnik.de/bilder_fst/fst_rft2000_b01kl.png

Fernschreiber (1960)

- Ursprung des ASCII
  - Einheitlicher Standard
  - Ermöglicht Kommunikation zwischen Geräten verschiedener Hersteller
  - 7 Bit werden zur Kodierung eines Zeichens verwendet
  - Beinhaltet spezielle Steuerzeichen
    - CR-Wagenrücklauf (\r)
    - LF-Zeilenvorschub (\n)
    - BEL-Glocke

Glas-Fernschreiber

- Steigende Popularität von Röhren-Bildschirmen in den 1960er Jahren
- Werden **glass-tty** oder ***dumb terminal*** genannt
- Erlaubt Cursorbewegung
- Alte Standards zur Vernetzung wurden durch Ethernet und IP-basierte Protokolle ersetzt
  - Telnet
  - SSH

https://upload.wikimedia.org/wikipedia/commons/8/87/Televideo925Terminal.jpg

CLI-Beispiele

CLI-Beispiele

- Bourne shell (sh | 1977)
- C Shell (csh | 1979)
- Bourne-again Shell (bash | 1989)
- Z Shell (zsh | 1990)
- COMMAND.COM (DOS)
- cmd.exe (Windows)
- PowerShell (Windows)
- Python
- MySQL
- G-Code (CNC-Maschinen)

Bash

Bash

- Bourne-Again Shell
  - *Wieder eine Bourne Shell*
  - *Wiedergeborene Shell*
- Kommandointerpreter des GNU Betriebssystems
- Ursprünglich von Brian Fox entwickelt
- Stammt direkt von der Bourne Shell ab
- Vereint Eigenschaften ihrer Vorgänger (z.B. ksh, csh)
  - Kompatibel zu einigen Shells
- Standard-Shell in vielen Linux Distributionen

Aufbau eines Befehls

Aufbau eines Befehls

$ *<Befehlsname>* *<optionale Parameter>*

- Werden mit ihrem Namen aufgerufen
  - Können auch über Aliase aufgerufen werden
- Optionale Parameter
  - Durch Leerzeichen getrennt
- Können mit einem „&“ verkettet werden
  - Befehle werden asynchron ausgeführt

Parameter

- Optionen (Flags)

```
$ Befehl --verbose
$ Befehl -v
$ Befehl --input /Pfad/zur/Input-Datei
$ Befehl -i /Pfad/zur/Input-Datei
$
$ ls -la
$
```

- Argumente

```
$ ls /tmp/
tmpfile
$ rm /tmp/tmpfile
$
```

Logische Verknüpfung von Befehlen

- **Logisches UND**

  $ Befehl1 && Befehl2

  - Befehl2 wird ausgeführt, wenn Befehl1 erfolgreich ausgeführt werden konnte

- **Logisches ODER**

  $ Befehl1 || Befehl2

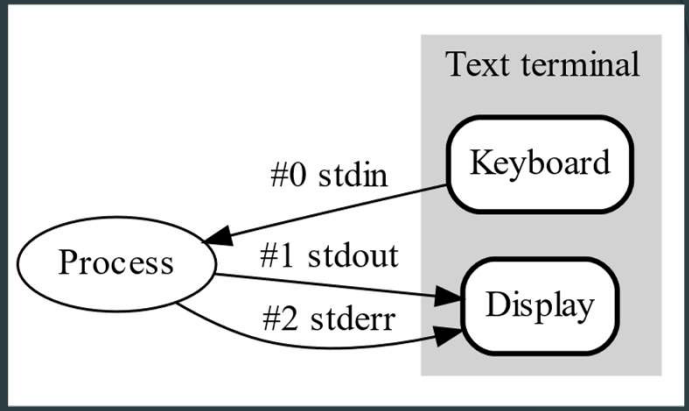
  - Befehl2 wird ausgeführt, wenn Befehl1 **nicht** erfolgreich ausgeführt werden konnte

```
$ compile_program && run_program || show_log
```

Datenstrom-Umleitung

Die Standard-Datenströme

- Standardeingabe (stdin | 0)
  - Üblicherweise über eine Tastatur
  - Liest Befehle und Daten ein
- Standardausgabe (stdout | 1)
  - Üblicherweise über einen Bildschirm
  - Gibt Ergebnisse von Befehlen aus
- Standardfehlerausgabe (stderr | 2)
  - Üblicherweise über einen Bildschirm
  - Gibt möglicherweise Fehler aus



https://upload.wikimedia.org/wikipedia/commons/7/70/Stdstreams-notitle.svg

Ausgabeumleitung

- Ergebnis eines Befehls in eine Datei umleiten

```
$ Befehl > Ausgabe.txt
$ ls > result.txt
```

- Fehlermeldungen in eine Datei umleiten

```
$ FehlerhafterBefehl 2> Fehlermeldung.txt
$ rm -rf / 2> errors.log
```

- Ausgaben einer Datei anhängen

```
$ Befehl >> Ausgabe.txt
```

```
$ FehlerhafterBefehl >> Fehlermeldungen.txt
```

Eingabeumleitung

- Inhalte einer Datei als Eingabe für einen Befehl

```
$ Befehl < Datei.txt
$ cat < Datei.txt
Das ist die erste Zeile der Datei.
Das ist die zweite Zeile.
Und hier ist noch ne Zeile.
$
```

Piping

- Ausgabe eines Befehls als Eingabe in einen anderen Befehl umleiten

```
$ Befehl1 | Befehl2
$ cat Logfile.log | grep SEVERE
```

- Ausgabe eines Befehls als Argument eines anderen Befehls

```
$ Befehl1 $(Befehl2)
$ echo $(date)
Tue Dec 4 19:58:43 UTC 2018
$
```

DEMO