

## Netzwerk - Programmierung

# Vorbesprechung, Administrativa, Einführung

Jan Krüger

[jkrueger@cebitec.uni-bielefeld.de](mailto:jkrueger@cebitec.uni-bielefeld.de)

Alexander Sczyrba

[asczyrba@cebitec.uni-bielefeld.de](mailto:asczyrba@cebitec.uni-bielefeld.de)

# Kommunikation in Netzwerken

- einige typische Anwendungen:
  - ssh
  - Thunderbird, Outlook
  - ftp
  - Mozilla Firefox, Safari, Opera, Chrome
  - finger, whois, talk, . . .
- Client-Server-Modell
- Gegenstücke: sshd, ftpd, httpd, . . .

## Wann? Wo? Wer?

- Di, 10-12 Uhr, c.t. vom 18.4. - 18.7.2017  
außer : **18.4 und 27.6**
- Grundstudiumszentrum Informatik (GZI), V2-221/222
- Praktikum / Übung
- Aktive Teilnahme, 1 LP, Modul Netzwerkprogrammierung
- Vorkenntnisse in Python (Skriptsprachen!)  
→ kleine Auffrischung am Ende der Einführung

## Was?

- Interprozess-Kommunikation
- Programmierung mit sockets
- „Python“ statt „C“
  - (Fast) identisches Interface zu Netzwerk-Funktionen
  - Leichter zu erlernen
  - Weniger Code
  - Konzentration auf das Wesentliche

## •Abschlussprojekt

- Einzelprojekt (ggf. Gruppenprojekt)
- Benotet (falls gewünscht)
- Umfang ca. 30h  
Zeitraum 13.6 – 18.7
- Abgabe & Vorstellung der Projekte am 25.7
- Interessante Vorschläge ? → Alternative: Chatclient
  - Protokollspezifikation
  - Implementierung (API) in Python 3
  - Anwendung

## Literatur

- *UNIX Network Programming*, Volume 1, Second Edition  
W. Richard Stevens
- *Internetworking with TCP/IP*, Volume 3  
D.E. Comer, D.L. Stevens
- *Foundations of Python Network Programming*  
B. Rhodes, J. Goerzen
- *Advanced Programming In The UNIX Environment*  
W. Richard Stevens
- *TCP/IP Illustrated*  
W. Richard Stevens

# Fahrplan

- IPC – interprocess communication (pipes, signals)
- Netzwerk-Protokolle (TCP/IP)
- Client-Server-Design, sockets-API
- Client-Design/Programming
- Server-Design/Programming
- . . .
- Modulabschluss-Projekt

# Python

- Python: 2.7.X oder 3.X
- Überprüfen mit  
    `> python --version`
- Pythonskripte:

```
#!/usr/bin/env python  
print("hello world\n")
```

- Setzen des execute-Bits

Interaktiv:   `> ipython`  
              `> ipython2.7`  
              `> ipython3`



## Selbsttest Python

Schreibe ein Python-Skript, das eine Text-Datei zeilenweise einliest und das Ergebnis in eine zweite Datei ausgibt. Die Namen der beiden Dateien sollen als Kommandozeilen-Parameter übergeben werden.

Trennzeichen: **Leerzeichen**, Tab & co können wegfallen

**Einfache Aufgabe:** Kehre jede Zeile wortweise um:

Eingabe: The quick brown fox

Ausgabe: fox brown quick The

**Fortgeschrittenere Aufgabe:** Vereinheitliche Groß-/Kleinschreibung:

Eingabe: jumped oVER THE Lazy DoGs

Ausgabe: Jumped Over The Lazy Dogs

**Expertenmodus:** Kombination beider Aufgaben.