

Socket-Programmierung in Java

Dieses mal müssen wir Linux als Arbeitsumgebung verwenden, da das in den Rechenzentren installierte Windows durch die aktivierte Firewall Netzwerkverbindungen von Außen unterbindet und wir so keine Kommunikation zwischen den Rechenzentrums-Computern aufbauen können.

Mit Hilfe des `telnet`-Kommandos (`telnet IP-Adresse Portnummer`) lassen sich TCP-Verbindungen interaktiv testen. Das ist insbesondere hilfreich, um das HTTP-Protokoll¹ interaktiv zu verwenden z.B.:

```
telnet www.fh-wedel.de 80
GET / HTTP/1.1
Host: www.fh-wedel.de
Leerzeile
```

5. Socket-Client

- (a) Programmieren Sie bitte einen einfachen Client, der die primitiven Socket-Möglichkeiten von Java verwendet und der an einen Server eine einfache HTTP-Anfrage (z.B. `GET /index.html`) stellen kann. Versuchen Sie eine Web-Server im Intranet und im Internet zu kontaktieren.
- (b) Wie würde der Client alternativ unter Verwendung der Klasse `java.net.URL` aussehen?
- (c) Wie geht man mit einem Proxy um?

6. Socket-Server

- (a) Programmieren Sie bitte einen einfachen Server, der einfache HTTP-Anfragen akzeptiert und bearbeitet, indem er etwa als Antwort lokale Files überträgt. Erzeugen Sie für jede einkommende Anfrage einen neuen Thread. Sprechen Sie den Server über einen Web-Browser an.
- (b) Testen Sie bitte Ihren Server mit `telnet`.
- (c) Verwenden Sie bitte den Client aus Aufgabe 5, um Anfragen an ihren Server zu stellen.
- (d) Versuchen Sie die Server Ihrer Kommilitonen anzusprechen.

¹Genauere Informationen zum HTTP-Protokoll finden sich in der Wikipedia unter http://de.wikipedia.org/wiki/Hypertext_Transfer_Protocol.