

Netzwerkprogrammierung Client/Server, Sockets

Von Tuba Ciftci

...Definitionen

Netzwerk

- Eine Verbindung, zwischen Computern
- man kann damit Daten miteinander austauschen

Client(Diensabnehmer)

- auf deutsch->>Kunde
- kann den Dienst nutzen
- kann einen Dienst beim Server anfordern

Server(Dienstnehmer,)

- bietet einen Dienst an
- kommuniziert mit Client (stellt Service zur Verfügung)
- verschafft ihm Zugang zu einem Dienst
- Kommunikation mit Client muss zuverlässig sein

Dienst(Service)

- bestimmt welche Daten ausgetauscht werden

Client/Server Prinzip

- Netzwerkverbindung muss zwischen zwei(mindestens)Endpunkten bestehen.
- danach kann die Kommunikation beginne

Sockets

- auf deutsch->> Sockel, Steckverbindung
- Erstellung:s=socket(domain, type, protocol)
- Endpunkt einer Verbindung zwischen zwei Prozessen
- Schnittstelle zwischen Anwendung und Netzwerkverbindung

... Socket Klassen

Client und Server sind Socket-Klassen. Sie können Daten verschicken oder auch empfangen. Verwaltet werden sie vom Betriebssystem. Außerdem unterscheidet man zwischen Stream Sockets und Datagram Sockets.

Stream Sockets

Client:

- Socket erstellen
- diese mit der Serveradresse verbinden
- Daten->senden oder empfangen
- Socket herunterfahren
- Verbindung trennen und schließen

Server:

- Server-Socket erstellen
- Socket an die Adresse binden
- auf Anfrage warten
- neues Socketpaar für Client erstellen
- Bearbeitung der Clientsocket-Schließen

Datagram Sockets

Client:

- Socket erstellen
- an Adresse senden

Server:

- Socket erstellen
- Socket binden-auf Pakete warten

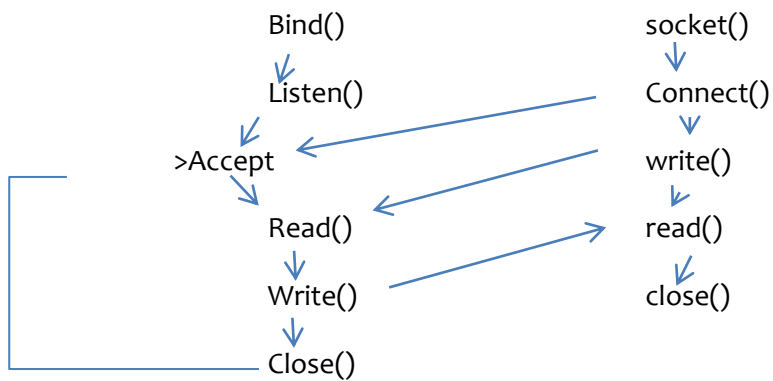
Nochmal zusammengefasst für *Client*: Verbindung erstellen (open), Daten senden(send), Daten empfangen(read) und danach einfach Verbindung schließen mit close(). *Server* muss noch zusätzlich IP-Ports binden(bind) und Anforderung annehmen also akzeptieren(accept).

Schaubild

Server

Client

Socket()
↓



Aufbau einer Kommunikationsverbindung

1. Zunächst müssen wir Socket anfordern (Funktion: `socket()`) mit

```
int socket(domain, type, protocol)
```

↓
Adressfamilie

Übertragungsarten -> `SOCK_STREAM` (Stream, TCP) und `SOCK_DGRAM` (Datagram, UDP)

2. Verbindung aufbauen mit `connect()`

```
int connect
```

```
(
```

```
Socket s,
```

```
SOCKADDR* name,
```

```
int namelen
```

```
);
```

.Übergibt den Socket den man verbinden möchte.

.Zeiger auf `SOCKADDR` Struktur

.Größe von `sockaddr`, `sizeof`

3. senden von Daten `send()`

4. empfangen von Daten `recv()`

5. Schließen mit `int close(int fd);`

Quellen:

http://de.wikipedia.org/wiki/Socket_%28Software%29

http://users.informatik.haw-hamburg.de/~schmidt/rn/socket_prog.pdf

<http://robsite.net/files/0000/0012/sockdoc.pdf>

<http://www.google.de/url?sa=t&rct=j&q=was+sind+netzwerk+sockets&source=web&cd=9&ved=0CIBEBYwCA&url=http%3A%2F%2Frobsite.net%2Ffiles%2F0000%2F0012%2Fsockdoc.pdf&ei=dzzDT4P4OsXlswaZyM2aCg&usg=AFQjCNHRfzclPDv7X9ohWM8GcVCdu2locw>

<http://www.hausarbeiten.de/faecher/vorschau/109556.html>