**Internet Relay Chat**

Viktor Popov, 336389

[Technische Hochschule Lübeck,

Informatik/Softwaretechnik, 3. Semester]

Verteilte Systeme Praktikum – Projektaufgabe: IRC – Internet Relay Chat Basics, Multiple Clients

**Internet Relay Chat**

Der IRC Server habe ich mithilfe der früher im Praktikum implementierten funktionalen Klassen TCPSocket,Writer, ActorSystem u.a. gebaut. In seinem Grund steht die Verwendung das Aktor Models, sowie der funktionalle TCP-Transceiver.

# Aktor Modell

Für die Umsetzung des IRC Protokolls habe ich das schon im Praktikum benutzte Aktor Modell angenommen. Der Server ist dann ein Aktor, der Nachrichten empfängt und sendet. Dafür habe ich in seiner IRCActor Klasse sein Verhalten beim Empfangen implementiert, und mithilfe des Strategy-Pattern sein Verhalten für die spezifischen Befehle abgebildet. Auf diese Weise bei jeder empfangenen Nachricht wird die jeweilige executeStategy-Methode aufgerufen, wo man dann die entsprechende Antwort sendet (IRC-Reply).

# CommandHandler(Strategy-Pattern)

Für die Implementierung der verschiedenen Kommandos habe ich das Stategy-Pattern am sinnvollsten gefunden und mich dafür entschieden. Mithilfe dessen wird die Behandlung einer empfangenen Nachricht in einer separaten Klassen implementiert und es wird zur Laufzeit das richtige Strategy-Objekt ausgefählt.

# AktorSystem

Die Klasse AktorSystem aus dem Praktikum habe ich wenig geändert. Das Publizieren eines Severs gibt dann eine erweiterung von Runnable, die noch eine getWriter-Methode hat. In der endlosen run-Schleife wartet der Server auf neue Verbindungen, und bei solchen wird eine neue IRCConnection instanziiert, die dann in die Liste der bevorstehenden Verbindungen hinzugefügt wird.

# Parsing

Der IRCParser analysiert Kommandos mit ihren Argumente. Dann packt er den Befehl, die Argumente und das ganze Kommando in einem IRCMessage, wo man auch den Typ (USER, NICK, QUIT, PRIVMSG, PING , PONG ,ERR) zuweist. Damit kann man dann später nur Argumente einer Nachricht, oder nur den Befehl holen.

# IRCConnection

Die Klasse IRCConnection handelt die Verbindung von einem Client zum Server. Sie behält Infomrationen wie das Socket des verbundenen Clients, den Server und den Writer (Aktor, der dann den TCPTransceiver vorstellt).

# IRCServer

Die Klasse IRCServer startet einen TCPTransceiver, der ständig auf neue Verbindungen am gegebenen Port wartet (mithilfe von ActorSystem). Hier werden auch alle Verbindungen gespeichert und behandelt (bei Erzeugen oder schließen). Außerdem speichert man hier auch eine Liste von belegten Nicknames, damit man dann schneller später über Nicknames Clients holen kann.

# Client

Die Client Klasse repräsentiert ein Client, der dann Nickname, Username und Realname besitzt.