

## Aufgabe 3: Broker und Balancer

**Lernziele** Kenntnisse spezieller Client/Server-Server Strukturen vertiefen, Abläufe beim Zugriff auf Server-Dienste bei vorgeschalteten Brokern und Load-Balancern verstehen.

**Aufgabe** Schreiben Sie ein Programm, das einen Broker-Dienst zur Verfügung stellt. Der Broker hat folgende Funktionalität:

- Server können bei dem Broker einen Dienst registrieren. Dazu wird der Name des Dienstes und die Port-Nummer, unter der der Dienst erreichbar ist, dem Broker mitgeteilt. Die Daten werden in einer Tabelle verwaltet.

Protokoll: `register <service name> <port>`

- Clients können Server, die spezielle Dienste anbieten, beim Broker erfragen. Dazu schicken sie dem Broker den Namen des gewünschten Dienstes, der Broker schickt eine Liste aller Server/Port-Paare zurück, unter denen der Dienst angeboten wird. Der Client muss dann selbst zu einem der Dienste Kontakt herstellen (handle driven broker).

Protokoll: `request <service name> liefert <ip1:port1> <ip2:port2> ...`

- optional: Der Broker testet in regelmäßigen Abständen, ob die Server noch erreichbar sind. Ist dies nicht der Fall, wird der entsprechende Dienst aus der Tabelle entfernt, der Server muss sich ggf. erneut beim Broker registrieren.

Schreiben Sie außerdem ein Programm, das einen Load-Balancer implementiert. Der Balancer hat folgende Funktionalität:

- Server können sich beim Balancer registrieren (siehe Broker).
- Der Balancer registriert sich beim Broker als Service-Anbieter, stellvertretend für die Server, die den eigentlichen Dienst zur Verfügung stellen. Die Registrierung erfolgt, nachdem der erste Server angemeldet ist.
- Ein Client erhält vom Broker evtl. die Adresse eines Balancers, der wiederum die eintreffenden Anfragen nach einer gegebenen Strategie an die Server weiterleitet. Eintreffende Anfragen sollen nach dem Round-Robin-Prinzip verteilt werden.

Zum Testen der Applikation soll ein Client implementiert werden, der einen Dienst beim Broker erfragt, anschließend eine Anfrage an einen der vom Broker gelieferten Server schickt und die Antwort ausgibt. Weiterhin sind mindestens zwei Server zu implementieren, die verschiedene Dienste bereitstellen.

**Testat** Zum Testat müssen Sie obige Programme vorlegen und erklären können.