

## Fortgeschrittene Programmiertechniken

Prof. Dr. J. Pauli, Dipl.-Inform. J. Hoefinghoff, J. Kapitza, M. Cherubim

### Übungsblatt 6

Seite 1 von 2

Das letzte Thema der Übungen ist RMI. In der letzten Übung haben sie gesehen, dass die Netzwerkkommunikation auf mehreren Wegen realisierbar ist. RMI stellt eine weitere Möglichkeit bereit, mit verteilten Systemen zu arbeiten. Sie sollten ein kleines Schaubild skizzieren, um sich die Kommunikationswege zu veranschaulichen. Ergänzen Sie die verbindenden Kanten mit Notizen über die Reihenfolge, sowie über die Eigenschaften dieser Verbindung (z.b. lookup oder rebind).

### Aufgabe 1

### Theorie

Beantworten Sie die nachfolgenden Fragen und fangen Sie bitte erst danach mit der Programmieraufgabe an. Die Fragen können auf einem Ausdruck oder mündlich beantwortet werden. Grundlegend für die Aufgabe sind auch Threads und Netzwerkkommunikation. Den Zettel mit Ihren Antworten müssen Sie nicht abgeben. Sie dürfen diesen aber mit in die Abnahme nehmen.

- Welche Methoden sind blockierende Aufrufe bei der Netzwerkkommunikation?  
D.h. der Aufrufer wartet auf die angefragten Daten bzw. auf das Beenden der Methode.
  - `public void wetteMichGibtsNicht() {}`
  - `public void removeSomeData(Socket s) {  
    InputStreamReader r = new InputStreamReader(s.getInputStream());  
    BufferedReader in = new BufferedReader(r);  
    in.readLine();  
}`
  - `socket.read()`
  - `socket.write(integerZahl)`
- Wie sieht eine gültige Methodensignatur in RMI aus?
  - `public void someMethod() {}`
  - `public void someMethod() throws Exception {}`
  - `public void someMethod() throws RemoteException {}`
- Was ist der Unterschied zu bisherigen Methodensignaturen (ohne RMI)?
- Wie wird ein Objekt bei der Registry angemeldet?
- Wie kann eine Referenz auf ein Objekt der Registry geholt werden?
- Was ist der Unterschied zwischen bind und rebind?
- Wofür ist das Remote Interface?

## Fortgeschrittene Programmieretechniken

Prof. Dr. J. Pauli, Dipl.-Inform. J. Hoefinghoff, J. Kapitza, M. Cherubim

### Übungsblatt 6

Seite 2 von 2

#### Aufgabe 2

### Programmierung

Entwickeln Sie ein Programm, welches Usern das Chatten ermöglicht. Für die Kommunikation soll RMI verwendet werden. Bei dem gestalten der GUI, stellen Sie sicher, dass mindestens ein Textfenster und eine Texteingabemöglichkeit über STRG-Enter vorhanden ist. Sie benötigen zur Realisierung mindestens 4 Klassen. **ChatServer** implementiert das Interface **ChatService** und **ChatClient** implementiert das Interface **ClientService**. Für die Volle Punktzahl sollten Sie diesen Service in den HardwareShop integrieren.

Der ChatService besitzt die Methoden:

- login(String)
- logout(String)
- send(String)
- List<String> getUserList()

Implementieren Sie oben genannten Methoden im ChatServer und geben Sie die korrekte Signatur in dem beschriebenen Interface an.

Der ClientService besitzt die Methoden:

- send(String)
- String getName()

Diese Methoden müssen durch den ChatClient implementiert werden.

Server sowie Clients müssen sich hierbei bei der Registry anmelden/abmelden, damit die Kommunikation funktioniert. Ein beispielhaftes Szenario zum Senden einer Nachricht:

- Registry starten
- Server starten und an der Registry anmelden.
- Client starten
- Client an der Registry unter BENUTZERNAMEN anmelden
- Dem Server den Namen des Client mitteilen.
- Client sendet eine Nachricht an den Server.
- Server holt sich aus der Registry die ihm bekannten Clients (bekannte Namen), um die erhaltene Nachricht an diese zu verteilen.
- Die eingehenden Nachrichten in der GUI anzeigen.

Kein Client stellt seine Nachricht direkt in der GUI dar, sondern empfängt, wie bei Order (letzte Blatt U5) die eigene Nachricht auch vom Server.