

Übungsblatt 2

zur Vorlesung

Verteilte Systeme/Ubiquitous Computing

Wintersemester 2007/2008

Achtung: Besprechung in der Übung am **Mittwoch, den 31. Oktober 2007 um 14 Uhr** (c.t.)

Bitte beachten Sie auch die Web-Site der Vorlesung:

<http://www.mobile.ifi.lmu.de/Vorlesungen/ws0708/vs/>

Aufgabe 4: (H) RPC

Das C/S-Modell basiert auf dem Paradigma der Ein- und Ausgabe von Daten. Verteilte Berechnungen sind in diesem Modell nur schwer so zu verwirklichen, dass der Benutzer die Verteiltheit nicht bemerkt, sondern an eine lokale Ausführung seiner Befehle glaubt. Um die Verteilung von Aufgaben für den Benutzer transparent zu machen, ersannen Birell und Nelson den Remote Procedure Call (RPC).

- a. Bitte erläutern Sie den Ablauf eines RPCs und gehen Sie dabei auf das Konzept des kritischen Pfads (critical path) ein.
- b. Remote Method Invocation (RMI) kann als die objekt-orientierte Variante des RPCs aufgefasst werden. Java RMI ist eine Implementierung der RMI. Benutzen Sie das Java.RMI-Paket, um sämtliche Objekte zu implementieren, die es einem Klienten ermöglichen, eine entfernte Methode zur Ermittlung der Zeit aufzurufen. Eine Anleitung, wie RMI verwendet wird, finden Sie unter <http://java.sun.com/j2se/1.3/docs/guide/rmi/getstart.doc.html>.
 - (i) Beginnen Sie zunächst damit, das Interface zu schreiben, das der Stub des Klienten benötigt, um das entfernte Objekt zu referenzieren. Dieses muss die Signatur der aufzurufenden Methode beinhalten.
 - (ii) Implementieren Sie anschließend den Server. Dieser erbt vom Standard-Remote-Object "UnicastRemoteObject" und verwendet das zuvor geschriebene Interface!
 - (iii) Kompilieren Sie den Server und das Interface. Erzeugen Sie Stub und Skeleton durch die Verwendung des Befehls "rmic"!
 - (iv) Implementieren und kompilieren Sie den Client! Die Class-Datei des Interfaces und der Stub müssen auf dieselbe Maschine kopiert werden, auf der der Client residieren soll.

- (v) Starten Sie auf der Serverseite den Remote Registry Server. Unter Windows geschieht dies durch den Befehl „start rmiregistry“. Melden Sie das entfernte Objekt dort an. Schenken Sie dabei der Sicherheitspolitik des RMI-Paketes besondere Aufmerksamkeit. Näheres hierzu erfahren Sie unter <http://java.sun.com/j2se/1.3/docs/guide/security>. Falls Sie nicht mehr weiter kommen, kopieren Sie einfach die erhaltene Fehlermeldung als Ganzes ins Suchfenster von www.google.com und suchen Sie nach einer der vielen Seiten, die das Problem behandeln.
- (vi) Starten Sie anschließend den Klienten und vergewissern Sie sich, dass Ihr Programm korrekt operiert!

Aufgabe 5: (H) Das World Wide Web als großes Verteiltes System

- a. Was ist HTML?
- b. Wie ist ein URL aufgebaut?
- c. Welche Arten aktiver Inhalte kennen Sie?

Aufgabe 6: (H) Gruppenkommunikation

- a. Wodurch unterscheidet sich die Gruppen- von der Zwei-Parteien-Kommunikation?
- b. Was ist der Unterschied zwischen Unicast, Multicast und Broadcast?
- c. IP-Multicast ist eine bekannte Implementierung der Gruppenkommunikation. Benutzen Sie das java.net-Paket und implementieren Sie das Szenario der Aufgabe 4, das wie folgt erweitert wird:

Mehrere Zeit-Server, die auf verschiedenen Maschinen installiert sind, bilden eine Gruppe. Ein Klient wendet sich an jene und erfragt die Zeit. Aufgrund der Antworten der Gruppenmitglieder können etwaige Unterschiede zwischen den lokalen Zeiten identifiziert werden.

- (i) Beginnen Sie damit eine Klasse für die Zeit-Server zu implementieren. Schreiben Sie den Code so, dass Sie mehrere Instanzen dieser Klasse unterscheidbar erzeugen können. Der Server öffnet zunächst eine neue Socket und meldet sich bei der Gruppe an. Dies geschieht über eine IP-Multicast-Adresse, die zwischen 224.0.0.1 und 239.255.255.255 gewählt werden kann. Anschließend wartet der Server auf eine Anfrage und antwortet bei Verlangen mit der lokalen Zeit.
- (ii) Schreiben Sie im zweiten Schritt den Klienten. Dieser muss ebenfalls eine Socket öffnen und der Gruppe beitreten. Eine Anfrage wird an die Gruppe versendet. Die eingehenden Antworten werden einfach auf den Bildschirm ausgegeben.
- (iii) Erzeugen Sie mehrere Instanzen des Zeit-Servers auf unterschiedlichen Computern und starten Sie Ihren Klienten. Wie Sie sehen, ist die Synchronisation der Uhren in einem verteilten System ein reales Problem.