

Übungsaufgabe 4.2 (Sockets) *Bei der Erstellung einer Client-/Server-Anwendung mit Sockets wird bei der Erstellung der Sockets nur für den Server eine Portnummer angegeben. Warum muss die Portnummer beim Server angegeben werden und beim Client nicht? Wie wird die Portnummer für den Client vergeben?*

Sockets beschränken sich auf die Übertragung von Daten. Die Semantik der Daten wird dabei nicht berücksichtigt. Daraus folgt, dass Protokolle zwischen Client und Server zur Verfügung gestellt werden müssen, welche die Daten semantisch korrekt interpretieren. Die Entwicklung derartiger Protokolle ist aufwendig und fehlerträchtig. Insbesondere entsteht zusätzlicher Aufwand bei der objektorientierten Modellierung für die Protokollprüfung, die Parameter- und Ergebnisübergabe sowie die Fehlerbehandlung. Durch das notwendige Protokoll zur Kommunikation zwischen Client und Server wird deutlich, dass durch die Verwendung von Sockets keine Zugriffstransparenz erreicht wird. Da der Client außerdem IP-Adresse und Port des Servers kennen muss, wird durch die Verwendung von Sockets die Ortstransparenz nicht unterstützt. Sockets haben aber den Vorteil, dass sie schnell sowie generell verfügbar sind, insbesondere sind auch keine clientseitigen Zusatzvoraussetzungen zu erfüllen.

In objektorientierten Anwendungssystemen auf einem Rechner kommunizieren Objekte mit anderen Objekten durch entsprechende Methodenaufrufe. Die Methoden sind dabei sowohl durch die Typen der Aufrufparameter als auch die Typen der Rückgabewerte mit Semantik angereichert. In verteilten objektorientierten Anwendungssystemen ist ein analoger Kommunikationsmechanismus erforderlich, der insbesondere auch entfernte Methodenaufrufe ermöglicht. Eine derartige Funktionalität wird allgemein, wie in Abschnitt 4.2.1 erläutert, durch Verteilungsplattformen angeboten.

Die Vorgänger entfernter Methodenaufrufe sind **Remote-Procedure-Calls (RPC)**. RPC ist eine Technologie zum Aufruf von Prozeduren auf entfernten Rechnern. Dadurch wird insbesondere auch der Aufruf von Prozeduren möglich, die sich in einem anderen Adressraum befinden. Wie bereits bei der Darstellung der Funktionen einer Verteilungsplattform in Abschnitt 4.2.1 erläutert, müssen referenzierte Daten im verteilten Fall als Kopien übergeben werden, da Referenzen in einem anderen Adressraum bedeutungslos sind. Aufgrund unterschiedlicher Rechnerarchitekturen ist möglicherweise auch die Darstellung der Daten auf unterschiedlichen Rechnern verschieden. Aus diesem Grund müssen plattformunabhängige Datenformate verwendet werden. Auf dem Quellsystem sind dazu die zu übertragenden Daten zunächst in ein unabhängiges Datenformat umzuwandeln, um dann auf dem Zielsystem wieder in die interne Darstellung übertragen zu werden.

Das RPC-Prinzip kann nicht unmittelbar in die objektorientierte Welt übertragen werden. Dafür sind die nachfolgenden Gründe zu nennen [3]:

1. Referenzen werden in objektorientierten Sprachen nicht als physische, sondern als logische Adressen behandelt.

Übertragung
von RPC in
OO-Welt