

Verteilte Systeme – Übung

Aufgabe 2

Sommersemester 2024

Harald Böhm, Laura Lawniczak, Tobias Distler

Friedrich-Alexander-Universität Erlangen-Nürnberg
Lehrstuhl für Informatik 4 (Systemsoftware)

<https://sys.cs.fau.de>



Lehrstuhl für Informatik 4
Systemsoftware



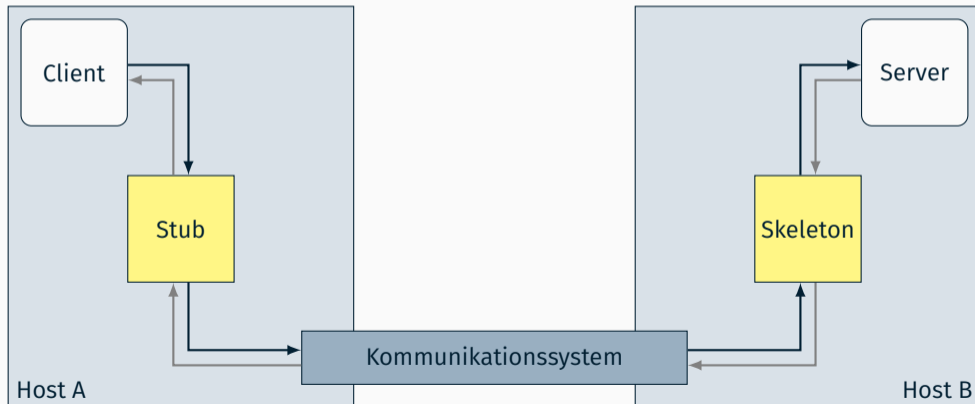
Friedrich-Alexander-Universität
Technische Fakultät

Aufgabe 2

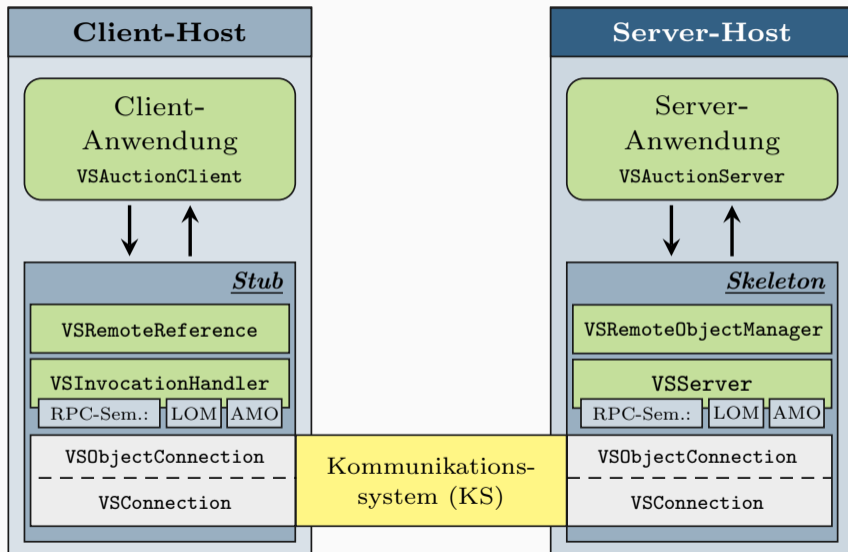
Aufgabe 2

Übungsaufgabe 2

- Dynamische Stubs und Skeletons
- Unterstützung von Rückrufen



Übungsaufgabe 2



- Ziel: Transparente Fernaufrufe
 - Normalfall: Rückgabe des Ergebnisses
 - Fehlerfall: Abbruch der Ausführung auf Server-Seite (Exception)
 - Fernaufrufsystem muss Exception zum Aufrufer propagieren

- Konsequenz für den Fehlerfall
 - Fangen der Exception beim Methodenaufruf auf Server-Seite → `InvocationTargetException`
 - Weiterleitung der Exception zur Client-Seite
 - Werfen der Exception im Stub

- Im Fernaufruf bedingte Fehler
 - Beispiele
 - Unerreichbarer Server
 - Verbindungsabbruch
 - ...
 - Fernaufrufsystem muss Fehler (soweit möglich) behandeln
[Näheres in Übungsaufgabe 3]

- Parameterübergabe (analog zu Java RMI)
 - Call-by-Value (Standard): Übertragung einer Kopie des Parameters
 - Call-by-Reference: Übertragung eines Stub für den Parameter
 - Parameter implementiert Remote-Schnittstelle
 - Parameterobjekt wurde zuvor exportiert

- Implementierung
 - Erweitertes Marshalling im Invocation-Handler des Stub
 - Analyse der Aufrufparameter
 - Unterscheidung der Parameterübergabearten
 - Beachte: Call-by-Reference ist auch relevant für Rückgabewerte

- `Class.isAssignableFrom()`: Überprüfung, ob ein Objekt `o` eine bestimmte Schnittstelle (z. B. `Serializable`) implementiert

```
Object o = [...];  
if(Serializable.class.isAssignableFrom(o.getClass())) {[...]}
```