

6. Übungsblatt

Aufgabe 1 (10 Punkte) CORBA

Pro Spiegelstrich je maximal einen Punkt.

- a. Was ist ein verteiltes System?
- b. Was versteht man unter Middleware?
- c. Was ist CORBA?
- d. Wofür ist der Object Transaction Service bei CORBA gedacht?
- e. Warum verwendet CORBA die interface definition language (IDL)?
- f. Worin liegt der Unterschied zwischen Stub und Skeleton?
- g. Was stellt die Interoperable Object Reference (IOR) dar und was soll durch die Verwendung von IORs erreicht werden?
- h. Worin liegt der Unterschied zwischen lokalen Objekten in Java und CORBA-Objekten?
- i. Was macht die IDL Operation narrow? Und was macht is_a?
- j. Wann liegen CORBA.SystemExceptions und wann CORBA.UserExceptions vor? Worin liegt der Unterschied zwischen beiden Exceptions?

Aufgabe 2 (10 Punkte) Hello World

- a. Erstellt die IDL-Datei HelloWorld.idl
 - Das verwendete Modul soll uebung lauten
 - Es soll ein Interface HelloWorldServer geben
 - Der HelloWorldServer soll die Methode helloWorld mit dem string-Parameter name haben

Hinweis: Um sowohl Client- als auch Server-seitigen Code zu generieren, muss idlj mit dem Parameter -fall aufgerufen werden.

Überprüfe bitte ein erfolgreiches Kompilieren der HelloWorld.idl. Es sollten die Dateien
HelloWorldServer.java
HelloWorldServerHelper.java
HelloWorldServerHolder.java
HelloWorldServerOperations.java
HelloWorldServerPOA.java
_HelloWorldServerStub.java
erstellt worden sein.

- b. Erstelle eine Implementierung des Server-Objekts.
 - Erstelle zuerst die Klasse HelloWorldImpl für die eigentliche Funktionalität der CORBA-Methode. Sie muss vom automatisch generierten Typen uebung.HelloWorldServerPOA erben. (Vgl. Folie 12)
 - Die Methode helloWorld soll den übergebenen Namen plus der Zeichenkette „Hallo Welt an “ zurückgeben
 - Als nächstes ist analog zu Folie 14 der HelloWorldServer zu erstellen.
 - i. OfferServer ist durch HelloWorldServer und OfferImpl ist durch HelloWorldImpl zu ersetzen

c. Anschließend ist der HelloWorldClient zu erstellen

- Es kann analog zu Folie 16 verfahren werden. Aus dem OfferHelper und der Offer sind natürlich HelloWorldServerHelper und HelloWorldServer zu machen.
- Der Methodenaufruf muss natürlich auch angepasst werden.

Abgegeben werden muss der Quelltext als .jar File oder ZIP/RAR Archiv, jeweils mit Dokumentation

Aufgabe 3 (10 Punkte) Referenzen

Gegeben sei folgende IDL-Datei

module uebung

```
{
  interface Reference
  {
    long count(in long addMe);
  };
};
```

- a. Die Implementierung verwende eine private Klassenvariable `_count`, um den übergebenen Parameter aufzuaddieren und das Resultat dann in der Methode `count` zurückzugeben.
- b. Implementiere nun den ReferenceClient, welcher wiederholt Werte an den Server übergibt und das Resultat darstellt. Starte zwei Instanzen des Clients.
- c. Was passiert? Warum?
- d. Stoppe den Server, während mindestens ein Client das Server-Objekt noch benutzt. Was passiert?

Packe den Code samt Dokumentation in eine Jar / Zip / RAR Datei und gebe ihn ab.

Hinweis: neben den einschlägigen Wikipedia-Artikeln ist auch das Buch Wirtschaftsinformatik 2 von Hansen / Neumann hilfreich (wenn auch nicht notwendig).

Hinweis zur Abgabe: Erwartet werden zwei Archiv-/Jar-Dateien mit dem Code und eine Datei im MS Office, txt- oder PDF-Format mit der Ausarbeitung zu den Fragen.

Die Abgabe dieses Arbeitsblattes erfolgt per Email an artus@ismll.uni-hildesheim.de bis zum 24.06.2008 um 10:00 Uhr GMT+2.