

Praktikum Technische Grundlagen der Informatik IV

Aufgabenblatt Th2

— Transparenz, RPC und Web-Services —

Abgabe: 6./7.5.2013

Zusätzliche Literaturempfehlung zu Web-Services: G.Coulouris, J. Dollimore, T. Kindberg: *Distributed Systems Concepts and Design*, fourth edition, Addison Wesley 2005, Lehrbuchsammlung der TUB: ST200 14

Aufgabe 1:

Sie haben in der Vorlesung gelernt, dass eine wesentliche Eigenschaft eines verteilten Systems das Verbergen der Verteiltheit ist. In diesem Zusammenhang haben Sie den Begriff der Transparenz kennengelernt. Welche Formen der Transparenz werden bei folgenden Beispielen (a und c) realisiert? Erläutern Sie diese bitte jeweils kurz!

- a) Beim NFS (Network Filesystem) wird ein auf einem Server befindliches Dateisystem beim Client in den lokalen Verzeichnisbaum eingehängt und erscheint wie ein lokales Verzeichnis mit den entsprechenden Dateien und Unterverzeichnissen des NFS Volumes. Programme greifen auf Dateien und Verzeichnisse des NFS Volumes mit denselben Systemaufrufen und Bibliotheksfunktionen zu wie auf lokale Dateien/Verzeichnisse. Bezeichnet ein Name eine Datei auf dem NFS Volume, so werden Zugriffe auf die Datei automatisch durch das Betriebssystem mittels RPCs (Remote Procedure Calls) über das Netzwerk zum Server übertragen und ggf. der entsprechende Teil der Datei geändert (Schreiboperation) oder an den Client übertragen (Leseoperation). Ist dies nicht erfolgreich, wird der RPC ggf. mehrfach wiederholt. Mehrere Clients können gleichzeitig lesend und schreibend auf verschiedene Dateien des Volumes oder nur lesend auf die selben Dateien zugreifen. Wird schreibend zugegriffen, so erscheint die Datei ggf. als hätte sie sich von selbst verändert, oder eine Datei kann explizit gegen gleichzeitigen Zugriff geschützt werden. Wird ein NFS Volume auf einen anderen Server verschoben, so kann nun dieses neue Volume unter dem bisher bekannten Verzeichnis eingehängt werden.
- b) Ist vollständige Failure Transparency realisierbar? Wie sieht das insbesondere in Hinblick auf das vorherige Beispiel aus?
- c) Ein Drucker wird über seinen Namen angesprochen. Zum Ausdrucken werden Druckjobs als Postscript-Dateien und unter Angabe des Druckernamens an den Druck-Server geschickt. Dieser konvertiert den Druckjob gegebenenfalls in die Sprache des Druckers, wenn der Drucker nicht Postscript-fähig ist. Mehrere gleichzeitige Druckjobs werden automatisch in eine Warteschlange eingereiht. Ist der Drucker überlastet (die Warteschlange zu lang) oder gar defekt, wird der Job automatisch an einen anderen Drucker weitergeleitet.

- d) Sind alle beim vorherigen Beispiel auftretenden Formen der Transparenz überhaupt sinnvoll? Begründen Sie Ihre Antwort.

	7
--	---

Aufgabe 2:

- Stellen Sie kurz die Übergabesemantiken *call-by-value*, *call-by-reference* und *call-by-copy/restore* des klassischen Prozeduraufrufs gegenüber.
- Sie möchten “Remote Procedure Call” (RPC) mit den zuvor genannten Übergabesemantiken verwenden. Bei welcher Semantik sehen sie die größten Probleme bezüglich der Umsetzbarkeit und wie lösen sie diese?
- Welche Form der Transparenz stand bei der Entwicklung des RPC-Konzeptes im Vordergrund?
- Grenzen Sie RPC gegenüber “Remote Method Invocation” (RMI) ab! Gehen Sie dabei insbesondere darauf ein, welches Ziel jeweils verfolgt wird und wo Zustand gehalten wird!

	4
--	---

Aufgabe 3:

Tafelaufgabe - Wird im Termin vorgerechnet, ist also nicht zu Hause zu lösen. Mitarbeit und damit Vorbereitung werden aber vorausgesetzt.

Sie möchten aus Ihrem Programm, das von einem 32-Bit-Prozessor mit Little-Endian-Darstellung ausgeführt wird, einen RPC aufrufen, der als Argument ein Paar aus ID und Namen verlangt. Das Programm speichert das Paar intern wie in Listing 1 dargestellt.

- Worauf müssen Sie beim (De-)Marshalling dieses Werte-Paares achten? 1P
- Ein weiterer RPC benötigt eine Liste dieser Werte-Paare. Die Liste wird intern gemäß Listing 2 gespeichert. Welche zwei zusätzlichen Probleme können hier auftreten? 2P
- Treten alle Probleme aus b) auch bei einem binären Baum wie in Listing 3 auf? (Tipp: Welche speziellen Eigenschaften besitzt ein Baum in der Informatik?) 1P
- Erläutern Sie bitte im Kontext dieses Beispiels die Begriffe *Shallow Copy* und *Deep Copy* 2P

	5
--	---

Listing 1: Werte-Paar

```

1 typedef struct ValueType {
2     int      id ;
3     const char* name ;
4 } ValueType ;

```

Listing 2: Doppelt-verkettete Liste

```

1  typedef struct ListNode {
2      ValueType  value;
3      ListNode*  prev;
4      ListNode*  next;
5  } ListNode;
6
7  ListNode*  myListHead;
8  ListNode*  myListTail;

```

Listing 3: Binärer Baum

```

1  typedef struct TreeNode {
2      ValueType  value;
3      TreeNode*  left;
4      TreeNode*  right;
5  } TreeNode;
6
7  TreeNode*  myTreeRoot;

```

Aufgabe 4:

Nennen Sie die wesentlichen Unterschiede zwischen HTML, XML und XHTML!

	1
--	---

Aufgabe 5:

Laden Sie die WSDL-Datei des TKN-Sensornetzwerk-Gateways von der ISIS-Seite herunter.

a) Interpretieren Sie knapp die allgemeine Bedeutung der folgenden Elemente¹:

- <types>
- <message>
- <interface>/<portType>²
- <binding>
- <service>

5P

b) Welche Adresse hat der in der WSDL-Datei beschriebene Web-Service?

1P

c) In welcher Programmiersprache ist der Web-Service implementiert? In welcher Programmiersprache ist der Web-Service-Client zu implementieren?

1P

	7
--	---

¹Hilfreich ist dabei die z.B. die WSDL-Spezifikation: <http://www.w3.org/TR/wsdl>

²Beide Bezeichner werden genutzt; im konkreten Beispiel <portType>

Aufgabe 6:

Kann ein Web-Service auch ohne “Universal Description, Discovery and Integration” (UDDI)-Eintrag angeboten werden? Ergibt das ggf. Sinn?

	1
--	---