

Institut für Betriebssysteme und Rechnerverbund
Übungen zur Vorlesung “Verteilte Systeme”, WS 02/03

<http://www.ibr.cs.tu-bs.de/lehre/ws0203/vs/>

Dozent: Prof. Dr. Stefan Fischer <fischer@ibr.cs.tu-bs.de> · Übungsleiter: Frank Strauß <strauss@ibr.cs.tu-bs.de>

Zu RMI wollen wir keine weiteren Programmieraufgaben durchführen. Stattdessen beschäftigen wir uns dieses Mal mit einigen Technologien, die für WebServices von Bedeutung sind. Ferner wollen wir uns einen kleinen Überblick über existierende WebServices verschaffen und (optional) einmal versuchen, einen Client mit Hilfe einer existierenden Java API zu entwickeln.

Die meisten in diesen Aufgaben als URLs angegebenen Dateien sind auch in dem beiliegenden ZIP-File (uebung-05.zip) enthalten.

5 Middleware

Übung am 04.12.2002

5.1 XML — Wohlgeformtheit

Im allgemeinen werden alle Daten, mit denen man es bei WebServices zu tun hat, in Form von XML Dokumenten repräsentiert: SOAP Nachrichten, Service Beschreibungen in WSDL und selbst Grammatiken in XML Schema, alles basiert auf XML. Daher ist ein grundlegendes Verständnis von XML wichtig.

Eine entscheidende notwendige Eigenschaft von XML-Dokumenten ist ihre sog. “Wohlgeformtheit”, die zum Parsen eines Dokumentes unerlässlich ist. XML-Parser können leicht feststellen, ob ein Dokument wohlgeformt ist.

- (a) Recherchieren Sie selbst nach einem XML Parser, den Sie im folgenden benutzen möchten. Achten darauf, dass Sie einen Parser verwenden, der in der Lage ist, die Gültigkeit eines XML Dokumentes anhand eines XML Schema zu prüfen; diese Fähigkeit brauchen wir für die folgende Aufgabe. Beim W3C empfiehlt man eine Google-Suche¹. In der Übung werden wir `xmlLint` (nicht-validierend) bzw. den Parser aus Xerces (genauer: ein kleines Demoprogramm `SAX2Count`, validierend) benutzen.
- (b) Betrachten Sie einige XML Dateien zunächst selbst und versuchen Sie zu entscheiden, ob sie wohlgeformt sind bzw. welche Fehler sie aufweisen. Verwenden sie anschließend den XML-Parser, um die Wohlgeformtheit zu prüfen.
 - <http://www.ibr.cs.tu-bs.de/index.xml>
 - <http://www.ibr.cs.tu-bs.de/users/strauss/ibr/newsgroups.xml>
 - <http://www.ibr.cs.tu-bs.de/users/strauss/dataformat.xml>

¹<http://www.google.com/search?q=XML+parser>

5.2 XML Schema — Validität

Die Benutzung von XML allein sorgt noch nicht dafür, dass spezifische XML-basierte Applikation auch mit Daten, die in XML-Form vorliegen, etwas anfangen können. In diesem Sinne ist XML vergleichbar mit dem lateinischen Alphabet, dass viele menschliche Sprachen gemeinsam haben. Dennoch können Spanier, Franzosen und Italiener sich nicht ohne weiteres verständigen, da sie nur verschiedene Grammatiken und Semantiken kennen. In der Informatik spricht man ebenfalls von Grammatiken, die die gültigen Worte beschreiben. Man kann überprüfen, ob ein Wort (ein XML-Dokument) einer Grammatik (einer XML Schema Definition) genügt. Ist dies der Fall, spricht man von einem validen Dokument bezüglich einer gegebenen XML Schema Definition. Für diese Überprüfung verwenden wir einen validierenden Parser.

Untersuchen Sie das Dokument <http://www.ibr.cs.tu-bs.de/users/strauss/IF-MIB.xml> auf Validität. Die zugrunde liegende XML Schema Definition ist im Kopf des Dokumentes selbst angegeben.

5.3 Real existierende WebServices

Eine Übersicht öffentlich verfügbarer WebServices gibt es unter <http://www.xmethods.org/>. Blättern Sie doch mal ein wenig in der dortigen Liste und notieren Sie sich ein paar WebServices, die Sie für interessant oder zukunftssträftig halten. Wieviel Vertrauen würden Sie diesen Diensten entgegen bringen?

5.4 Google WebService

Wenn Sie einmal etwas Erfahrung mit der Nutzung eines Webservice machen wollen, bietet sich z.B. der Webservice von Google an. Er ist unter <http://www.google.com/apis/> beschrieben. Dort ist auch - nach einer Registrierung - der Download eines Paketes möglich, mit dem Sie relativ leicht Google-WebService Clients basierend auf Microsofts .NET oder einer Java "Adapter-Klassen-Bibliothek" erstellen können.

Wenn Sie Spaß und Interesse daran haben, dann können Sie sich dort registrieren und versuchen, einen eigenen Client hierfür zu erstellen. Wir werden in der Übung die Realisierung einer IBR-Suchmaschine (<http://www.ibr.cs.tu-bs.de/ibrsearch/servlet/SearchServlet>) basierend auf dem Google Java Interface besprechen und einen Blick auf die WSDL Beschreibung dieses Webservice werfen.