

PI.SWA.SA.VO Software Architekturen und Web Technologien

Schriftliche Einzelprüfung zur Vorlesung des Wintersemesters 2008/09
Vierter und letzter Prüfungstermin
Fr 5.6.2009

Prüfer: Dr. Michael Derntl

Anweisungen:

- **Arbeitszeit: 90 Minuten**, es sind **max. 90 Punkte** zu erreichen (die Anzahl der Punkte, die pro Aufgabe zu erreichen ist, finden Sie jeweils in eckiger Klammer). Kalkulieren Sie also für jeden erreichbaren Punkt ca. eine Minute Arbeitszeit.
- Halten Sie einen **Lichtbildausweis** (Studentenausweis, Führerschein, Pass, etc.) bereit.
- Lösen Sie die Aufgaben direkt auf den Angabeblättern und schreiben Sie **leserlich**.
- Falls Sie zusätzliche Blätter benötigen, fragen Sie die Prüfungsaufsicht.
- Es sind **keine Unterlagen**, Handies, PDA, Laptops, Taschenrechner, Headsets, oder sonstige Hilfsmittel erlaubt!

Hinweise:

- **Ergebnisse** werden bis spätestens **Mo 22.6.** im PISWI verfügbar sein.
- **Einsicht** ist **nach Terminvereinbarung** per E-Mail in der Woche von 22.-26.6.
- Dies ist der **letzte Prüfungstermin** für das Stoffsemester WS 2008/09.

Aufgabe 1: XML Technologien**[40]**

Gegeben ist folgendes XML Schema. (Hinweis: die Zahlen am Beginn jeder Zeile sind Zeilennummern, auf die Sie sich bei der Lösung der Aufgabe beziehen können.)

```
1 <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
2
3   <xsd:complexType name="banken_t">
4     <xsd:sequence maxOccurs="1">
5       <xsd:element name="bank" type="bank_t" maxOccurs="unbounded"/>
6       <xsd:element name="CEO" type="xsd:string"/>
7     </xsd:sequence>
8     <xsd:attribute name="holding" type="xsd:string" use="required"/>
9   </xsd:complexType>
10
11  <xsd:complexType name="bank_t">
12    <xsd:sequence>
13      <xsd:element name="chef" type="chef_t"/>
14      <xsd:element name="gründungsjahr" type="xsd:gYear"/>
15      <xsd:element name="kunden" type="kunden_t"/>
16      <xsd:element name="wertpapierkennung" type="wpk_t" minOccurs="0"/>
17    </xsd:sequence>
18    <xsd:attribute name="name" type="xsd:string" use="required"/>
19    <xsd:attribute name="ort" type="xsd:string" use="required"/>
20    <xsd:attribute name="land" type="xsd:string" use="optional"/>
21  </xsd:complexType>
22
23  <xsd:simpleType name="chef_t">
24    <xsd:restriction base="xsd:string">
25      <xsd:maxLength value="10"/>
26    </xsd:restriction>
27  </xsd:simpleType>
28
29  <xsd:complexType name="kunden_t">
30    <xsd:attribute name="firmen" type="xsd:nonNegativeInteger" use="required"/>
31    <xsd:attribute name="privat" type="xsd:nonNegativeInteger" use="required"/>
32  </xsd:complexType>
33
34  <xsd:simpleType name="wpk_t">
35    <xsd:restriction base="xsd:string">
36      <xsd:length value="12"/>
37    </xsd:restriction>
38  </xsd:simpleType>
39
40  <xsd:element name="banken" type="banken_t">
41    <xsd:unique name="xxx">
42      <xsd:selector xpath="bank"/>
43      <xsd:field xpath="chef"/>
44    </xsd:unique>
45  </xsd:element>
46
47 </xsd:schema>
```

- 1a. Schema-Validierung.** Markieren Sie jene Stellen im folgenden XML Dokument, die in Bezug auf das gegebene XML Schema *ungültig* sind. Begründen Sie jeweils kurz, warum die Stelle ungültig ist. [15]

```

1  <?xml version="1.0" encoding="utf-8"?>
2
3  <banken holding="UniCredit Group">
4
5      <CEO>Gennaro Gattuso</CEO>
6
7      <bank name="Bank Pekao SA" ort="Warschau" land="Polen">
8          <chef>«VORNAME» «NACHNAME»</chef>
9          <gründungsjahr>1929</gründungsjahr>
10         <kunden privat="299576" firmen="2434">
11             </kunden>
12         <wertpapierkennung>123456789012</wertpapierkennung>
13     </bank>
14
15     <bank name="HypoVereinsbank" ort="München" land="">
16         <chef>K. Amel</chef>
17         <kunden firmen="344654" privat=""/>
18         <gründungsjahr>-</gründungsjahr>
19         <wertpapierkennung/>
20     </bank>
21
22     <bank name="Bank Ruptcy, Inc." ort="Boston, USA">
23     </bank>
24
25     <bank name="Zagrebacka banka" land="Kroatien" ort="Zagreb">
26         <chef>K. Amel</chef>
27         <kunden firmen="keine" privat="398476"/>
28         <gründungsjahr></gründungsjahr>
29     </bank>
30
31 </banken>

```

1b. Schema vs. DTD. Erklären Sie, welche in dem oben gegebenen XML Schema definierten Einschränkungen Sie mit einer DTD nicht definieren könnten (beziehen Sie sich auf Zeilennummern im Schema und geben Sie jeweils eine kurze Begründung!) [8]

1c. XPath. Nehmen Sie an, es existiert ein XML Dokument, das gültig für das gegebene XML Schema ist. Erstellen Sie nun je einen XPath Ausdruck für folgende Abfragen:

i. Die Anzahl der Banken im Land Österreich mit mindestens 2500 Firmenkunden? [3]

ii. Die Orte aller Banken, deren Chef “«VORNAME» «NACHNAME»” heißt? [2]

iii. Die Jahre, in denen mindestens zwei Banken in Frankreich gegründet wurden? [4]

1d. XSL Transformation. Notieren Sie die Ausgabe bei der Anwendung des XSL Stylesheet (rechts) auf die gegebene XML Datei (links). [8]

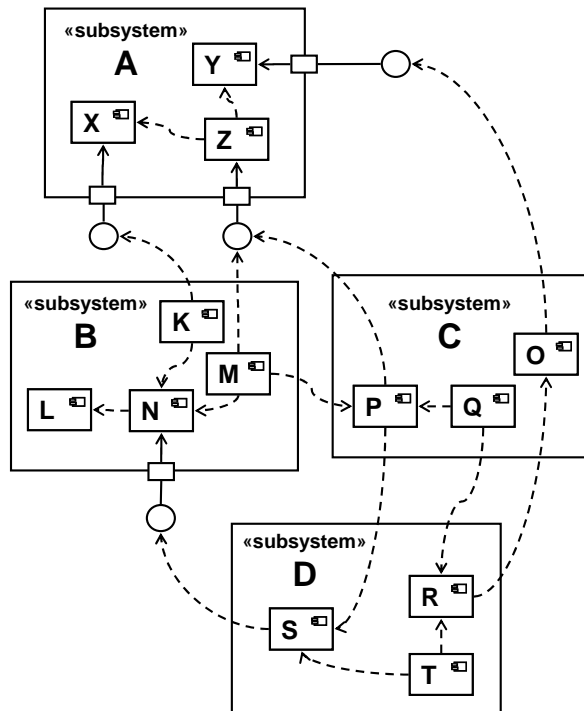
| | |
|---|--|
| <p><i>XML Datei:</i></p> <pre> <?xml version="1.0"?> <X> <Y a="«NACHNAME»"> <R>Kupfer</R> <R n="1" m="3">Eisen</R> </Y> <R>Holz</R> <Y x="47" y="«MATNR»"> <Z>Gold</Z> <Z><R>Gummi</R><R/></Z> <Z t="woa">Nickel</Z> </Y> </X> </pre> | <p><i>XSL Stylesheet:</i></p> <pre> <?xml version="1.0"?> <xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"> <xsl:template match="Y[@*]"> Y: <xsl:for-each select="@*"> <xsl:value-of select="."/> / </xsl:for-each> Mehr: <xsl:apply-templates select="*/*" /> </xsl:template> <xsl:template match="R"> R = <xsl:value-of select="text()" /> !!! </xsl:template> </xsl:stylesheet> </pre> |
|---|--|

Ausgabe:

Aufgabe 2: Software Architektur

[31]

- 2a. **Architekturprinzipien.** Beurteilen Sie das Ausmaß von Kohäsion, Kopplung und der Verwendung von Schnittstellen der Subsysteme A, B, C und D im folgenden Architekturmodell (mit kurzer Begründung jeweils). [10]



2b. Architekturpatterns.

i. Was versteht man allgemein unter einem Architekturpattern? [2]

ii. Erläutern Sie kurz Zweck und Vorteile des MVC Pattern. [3]

iii. Skizzieren Sie die MVC Pattern Komponenten und deren Beziehungen, und erklären Sie welche Aufgaben die Komponenten jeweils erfüllen. [8]

- 2c.** Was sind die Charakteristika und Unterschiede bei einer *holistischen* und einer *reduktionistischen* Herangehensweise bei der Betrachtung/Erstellung des Architekturmodells eines Systems? Erläutern Sie anhand eines konkreten selbstgewählten Beispiels. [8]

Aufgabe 3: Webservices

[19]

3a. Die Operationen eines Flugbuchungs-Webservice sind wie folgt definiert:

```
flugBuchen (  
    flug: Flug,  
    passagiere: Passagier[]) : Boolean  
  
getAlleFlügeVonKunde (  
    kundenNr: String) : Flug[]
```

Das Webservice ist unter `http://flug.bu/chung.php` verfügbar und muss mittels SOAP angesprochen werden.

Erklären Sie verbal, was beim Erstellen der WSDL Datei für dieses Webservice zu tun wäre. Gehen Sie dabei konkret auf alle Abschnitte der WSDL Datei ein! [12]

3b. Was ist UDDI und wozu wird es im Zusammenhang mit Webservices verwendet? [3]

3c. Wozu wird ein SOAP Envelope verwendet und was ist darin definiert? [4]