2G1523
Programming Web Services
Home works

Course leader:
Professor Mihhail Matskin
misha@imit.kth.se

Teaching assistant:
Hamid Reza Mizani
hrmizani@imit.kth.se

KTH/ICT/ECS , VT07

Administration

- 3 Home works + 1 Project
- Groups of 2 students, or individual
- 6 Bonus points for home works ***delivered in time***
- 6 Bonus points for project ***delivered in time***
- Bonus points are valid ONLY on the first exam!
- How to deliver? Send your reports to BOTH course leader and teaching assistant:
  (misha@imit.kth.se , hrmizani@imit.kth.se)

Homework #1 - Deadline

- Start date: 2007-01-25

- Due date: 2007-02-04

- Deliverable: Report (See course web)

Homework #1 - Aims

- Aims:

  - To understand the concepts of XML schema

  - To learn the XML processing method via JAXB

Homework #1 – J2EE

- Download J2EE from http://java.sun.com/j2ee/.
  Follow the instructions to install.
- Suggestions:
  - Read the QuickStart guide:
    "c:\sun\AppServer\QuickStart.html"
  - Try to Start, Stop and Verify the server
  - Try packaging and deploying applications
  - Try using admin console
  - Try using Deploytool utility
  - Try using command-line tools

Homework #1 - JWSDP

- Download Java Web Service Developer Pack (JWSDP) from:
  http://java.sun.com/webservices/jwsdp/index.jsp.
  The current version is 2.0.

- Follow the instructions to install.

- Read documents in:
  "c:\sun\jwsdp-2.0\jaxb\docs\index.html"

1

Homework #1 - JWSDP

- JWSDP includes:
  - Sun Java Streaming XML Parser Version 1.0 EA
  - XML Digital Signature Version 1.0 EA2
  - XML and Web Services Security Version 1.0
  - WS-I Attachments Sample Application Version 1.0 EA3
  - JAXB Version 1.0.4 (Java Architecture for XML Binding)
  - JAXP Version 1.2.6_01 (Java API for XML Processing)
  - JAXR Version 1.0.7 (Java API for XML Registries)
  - JAX-RPC Version 1.1.2_01 (Java API for XML-based RPC)
  - SAAJ Version 1.2.1 (SOAP with Attachments API for Java)
  - JSTL Version 1.1.1_01 (JavaServer Pages Standard Tag Library)
  - Java WSDP Registry Server Version 1.0_08
  - Ant build tool

Homework #1 - JWSDP

- JAXB:
  - Java Architecture for XML Binding
  - API and tools that automate the mapping between XML documents and Java objects

Homework #1 – Java IDE

- Get familiar with one of Java Development Environment, such as Eclipse, JBuilder, VisualCafe or Jakarta Ant.

- The manual of Jakarta Ant can be found at: http://ant.apache.org/manual/index.html

Homework #1 – Jakarta Ant

- A Java-based build tool

- Make: shell-based, limited to the OS.

- Ant is extended using Java classes.

- XML-based configuration files

Homework #1 – Jakarta Ant

- **Example Build file**

```xml
<project name="MyProject" default="dist" basedir=".">
    <description>
            simple example build file
    </description>

    <!-- set global properties for this build -->
    <property name="src" location="src"/>
    <property name="build" location="build"/>
    <property name="dist" location="dist"/>

    <target name="init">
            <!-- Create the time stamp -->
            <tstamp/>
            <!-- Create the build directory structure used by compile -->
            <mkdir dir="${build}"/>
    </target>
```

Homework #1 – Jakarta Ant

- **Example Build file**

```xml
<target name="compile" depends="init" description="compile the source ">
        <!-- Compile the java code from ${src} into ${build} -->
        <javac srcdir="${src}" destdir="${build}"/>
</target>

<target name="dist" depends="compile" description="generate the distribution" >
        <!-- Create the distribution directory -->
        <mkdir dir="${dist}/lib"/>
        <!-- Put everything in ${build} into the MyProject-${DSTAMP}.jar file -->
        <jar jarfile="${dist}/lib/MyProject-${DSTAMP}.jar" basedir="${build}"/>
</target>
```

Homework #1 – Jakarta Ant

- **Example Build file**

```
<target name="clean" description="clean
up" >
    <!-- Delete the ${build} and ${dist}
    directory trees -->
    <delete dir="${build}"/>
    <delete dir="${dist}"/>
</target>
</project>
```

Homework #1 – Jakarta Ant

- How to run:

  - ant
    - Using "build.xml" file in the current directory

  - ant -buildfile test.xml

  - More option ...

Homework #1 – Jakarta Ant

- Tasks:
  - Archive Tasks: Jar, Unjar, Zip, Unzip, War, Unwar ...
  - Compile Tasks: Javac, Rmic, …
  - Documentation Tasks: Javadoc, …
  - Execution Tasks: Java, Exec, Ant, …
  - File Tasks: Copy, Delete, Mkdir, Move, …
  - Mail Tasks: Mail, …

- More:
  Audit/Coverage Tasks, Deployment Tasks, EJB Tasks, Java2 Extensions, Tasks, Logging Tasks, Miscellaneous Tasks, .NET Tasks, Pre-process Tasks, Property Tasks, Remote Tasks, SCM Tasks, Testing Tasks, Visual Age for Java Tasks

---

Homework #1 - XJC

- Edit AnnotatedSchema.xml (http://www.imit.kth.se/courses/2G1523/Homeworks/AnnotatedSchema.xml) that it will fit to your CV (You can delete non-relevant elements, add new elements or/and modify the description).

- Compile the edited XML schema into Java classes using xjc.

Homework #1 - AnnotatedSchema.xml

- Contact Info Definition:

```
<xs:element name="PersonName" type="PersonNameType"/>
    <xs:element name="GivenName" type="NameType"/>
    <xs:element name="MiddleName" type="NameType"
    minOccurs="0"/>
    <xs:element name="FamilyName" type="NameType"/>

<xs:element name="ContactMethod" type="ContactMethodType"/>
    <xs:element name="Phone" type="PhoneNumberType"
    maxOccurs="3"/>
    <xs:element name="Fax" type="PhoneNumberType"
    minOccurs="0"/>
    <xs:element name="Pager" type="PhoneNumberType"
    minOccurs="0"/>
```

Homework #1 - AnnotatedSchema.xml

- Objective Definition:

```
<xs:enumeration value="CRL-Business Transformation
Solution"/>
<xs:enumeration value="CRL-Supply Chain and Logistics
Management Research"/>
<xs:enumeration value="CRL-Management Science and
Operations Research"/>
<xs:enumeration value="CRL-User Paradigm Research"/>
<xs:enumeration value="CRL-Distributed Computing"/>
<xs:enumeration value="CRL-Data Management"/>
```

3

Homework #1 - AnnotatedSchema.xml

- Education History:

  `<xs:element name="SchoolName" type="NameType"/>`

  `<xs:element name="Degree" type="DegreeType"/>`
  `        <xs:enumeration value="Doctor"/>`
  `        <xs:enumeration value="Master"/>`
  `        <xs:enumeration value="Bachelor"/>`

  `<xs:element name="Major" type="MajorType"/>`
  `        <xs:enumeration value="Computer Science"/>`
  `        <xs:enumeration value="Electronic Engineering"/>`
  `        <xs:enumeration value="Math"/>`
  `        <xs:enumeration value="Information"/>`
  `        <xs:enumeration value="Physics"/>`

25-01-2007   2G1523 Homeworks   Slide 19 / 26

Homework #1 - JAXB

- Construct your own resume data using JAXB. For doing this create a Java content tree from scratch and marshal it to XML data file, for example, Resume.xml. Please refer to create-marshal example for guidance.

- Print your generated Resume.xml file.

25-01-2007   2G1523 Homeworks   Slide 20 / 26

Homework #1 - create-marshal

```java
public class Main {
    public static void main( String[] args ) {
        try {
            // create a JAXBContext
            JAXBContext jc = JAXBContext.newInstance( "primer.po" );
            // create an ObjectFactory instance.
            ObjectFactory objFactory = new ObjectFactory();
            // create an empty PurchaseOrder
            PurchaseOrder po = objFactory.createPurchaseOrder();
            // set the required orderDate attribute
            po.setOrderDate( Calendar.getInstance() );
            // create shipTo USAddress object
            USAddress shipTo = createUSAddress( objFactory,
                    "Alice Smith",
                    "123 Maple Street",
                    "Cambridge",
                    "MA",
                    "12345" );
            // set the required shipTo address
            po.setShipTo( shipTo );
            // create billTo USAddress object
            USAddress billTo = createUSAddress( objFactory,
                    "Robert Smith",
                    "8 Oak Avenue",
                    "Cambridge",
                    "MA",
                    "12345" );
            // set the required billTo address
            po.setBillTo( billTo );
```

25-01-2007   2G1523 Homeworks   Slide 21 / 26

Homework #1 - create-marshal

```java
            // create an empty Items object
            Items items = objFactory.createItems();
            // get a reference to the ItemType list
            List itemList = items.getItem();
            // start adding ItemType objects into it
            itemList.add( createItemType( objFactory,
                    "Nosferatu - Special Edition (1929)", new BigInteger( "5" ), new BigDecimal( "19.99" ), null, null,
"242-NO" ) );
            itemList.add( createItemType( objFactory,
                    "The Mummy (1959)", new BigInteger( "3" ), new BigDecimal( "19.98" ), null, null, "242-MU" ) );
            itemList.add( createItemType( objFactory,
                    "Godzilla and Mothra: Battle for Earth/Godzilla vs. King Ghidora",
                    new BigInteger( "3" ), new BigDecimal( "27.95" ), null, null, "242-GZ" ) );
            // set the required Items list
            po.setItems( items );
            // create a Marshaller and marshal to System.out
            Marshaller m = jc.createMarshaller();
            m.setProperty( Marshaller.JAXB_FORMATTED_OUTPUT, Boolean.TRUE );
            m.marshal( po, System.out );

        } catch( JAXBException je ) {
            je.printStackTrace();
        }
    }
}
```

25-01-2007   2G1523 Homeworks   Slide 22 / 26

Homework #1 - create-marshal

```java
public static USAddress createUSAddress( ObjectFactory objFactory,
        String name, String street, String city, String state, String zip )
    throws JAXBException {

    USAddress address = objFactory.createUSAddress();

    address.setName( name );
    address.setStreet( street );
    address.setCity( city );
    address.setState( state );
    address.setZip( new BigDecimal( zip ) );

    return address;
}
```

25-01-2007   2G1523 Homeworks   Slide 23 / 26

Homework #1 - create-marshal

```java
public static Items.ItemType createItemType( ObjectFactory objFactory,
                String productName,  BigInteger quantity, BigDecimal price,
                String comment, Calendar shipDate, String partNum )
    throws JAXBException {

    Items.ItemType itemType = objFactory.createItemsItemType();

    itemType.setProductName( productName );
    itemType.setQuantity( quantity );
    itemType.setUSPrice( price );
    itemType.setComment( comment );
    itemType.setShipDate( shipDate );
    itemType.setPartNum( partNum );

    return itemType;
}
```

25-01-2007   2G1523 Homeworks   Slide 24 / 26

Homework #1 - Report

- Report:
  - A short description of your activities for installation of the above-mentioned tools (what problems did you face)
  - Edited XML Schema for your resume (with comments on what was modified compare to initial schema)
  - The tree of Java classes generated after compiling your XML Schema
  - Source code of Java programs that creates Java content tree for your resume and marshals it to the Resume.xml file.
  - Printout of your generated Resume.xml file

Homework #1 - Delivery

- Delivery:
  Send your report by e-mail to both misha@imit.kth.se and hrmizani@imit.kth.se.

- Deadline: 2007-02-04

- See course web for more information.

  GOOD LUCK!