Castor

Introduction

Castor is a Castor is an open source data binding framework for Java. This simple-to-use technology is available from http://castor.exolab.org.

The Exercise

In this exercise, you will use Castor to generate a set of Java class files from an XML schema to enable you to use Java to easily manipulate the corresponding XML documents. Castor is easy to use and does not require any knowledge of other processing technologies such as SAX or DOM.

Installing Software

Your instructor will provide you with the *Castor 0.9.3.21* and *Xerces-J 2.0.2* software packages and give you instructions regarding how these should be installed in your system.

Both should be installed before you do this exercise.

You should also ensure that a Java Development Kit is installed on your system before undertaking this exercise.

Setting Up

Make a new directory for this exercise. Call this directory *Castor*: For example:

```
C:\> mkdir Castor
C:\> cd Castor
```

All the files that you subsequently create as you do this exercise should be contained in this directory.

Create an XML Schema

In this exercise, Castor will be 'driven' by a standard XML Schema.

Create the file *person.xsd* as follows:

```
<?xml version="1.0"?>
<schema xmlns="http://www.w3.org/2001/XMLSchema">
  <element name="person" type="PersonType"/>
  <complexType name="PersonType">
    <sequence>
      <element name="name" type="string"/>
      <element name="domicile" type="string"/>
      <element name="dob" type="date" minOccurs="0" maxOccurs="1"/>
    </sequence>
    <attribute name="valid" type="boolean"/>
  </complexType>
</schema>
```

Create a Simple Properties File

Much of what Castor does is parameterisable. When a member of the Castor toolset starts up, it looks for a properties file called castor.properties and initialises itself according to the contents of the file.

Create the file *castor.properties* and ensure that it contains the following:

```
org.exolab.castor.indent=true
```

Generate the Castor Mapping Classes

Open a command prompt window and execute the following command sequence(*notes: (1) substitute the actual locations on your system for* castor-0.9.3.21-xml.jar *and* xercesImpl.jar; *(2) there are only two command lines here, the second has been wrapped artificially*):

```
C:\Castor> set CLASSPATH=.;castor-0.9.3.21-xml.jar;xercesImpl.jar
C:\Castor> java org.exolab.castor.builder.SourceGenerator -i person.xsd -package
  person.mapping
```

Create a Simple Java Application

The simple application you will make here utilises the files created by Castor's SourceGenerator to create and persist a single Person object.

Create the file *CastorExample.java* as follows:

```
package person;

import person.mapping.*;
import java.io.*;

public class CastorExample
  {
  public static void main(String[] args)
    throws Exception
    {
    Person person = new Person();

    person.setName("Fred Flintstone");
    person.setDomicile("Bedrock");
    person.setDob(new org.exolab.castor.types.Date());
    person.setValid(true);

    FileWriter writer = new FileWriter("FredFlintstone.xml");
    person.marshal(writer);
    }
  }
```

Compilation and Execution

Very straightforward. Type the following commands into the *same* command prompt window you used earlier:

```
C:\Castor> javac -d . CastorExample.java
C:\Castor> java person.CastorExample
C:\Castor> type FredFlintstone.xml
```

Congratulations! You have successfully undertaken a simple Castor project.