Writing an SMS Service Client

Sending an SMS Text Message Programmatically

Developed by: transentia

Unit 1, 40 Skew St.
Sherwood
Queensland 4075
Australia

Mobile: 0416 078 813
Phone/FAX: (07) 3278 5890
Email: contact@transentia.com.au
WWW: www.transentia.com.au

ACN: 091-142-880
ABN: 680-911-428-80

Exercise

4

Writing an SMS Service Client

Sending an SMS Text Message Programmatically

Introduction

Oracle, via it's Oracle Technology Network, has published a 'live' WebService that allows client applications to send an SMS text message anywhere in the world.

Simple Java Client

In this session you will use Oracle's JDeveloper 9i IDE to create and execute a simple client that makes use of Oracle's SMS WebService.

In this session you will need:

- A 'live' web connection
- Oracle JDeveloper 9.0.3

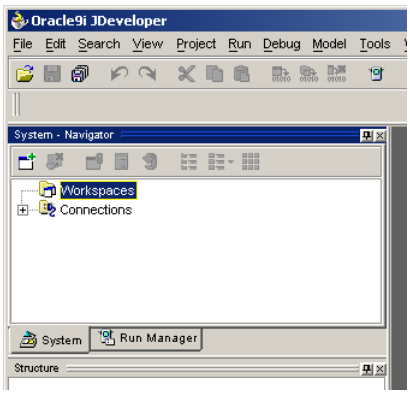
  JDeveloper 9i will be provided to you and should be installed prior to starting the exercise below (installation is easy: simply unzip the supplied file into C:\).
- A personal login on Oracle's "Oracle Technology Network." You will need to obtain this (it is generally very useful) from http://otn.oracle.com/membership.

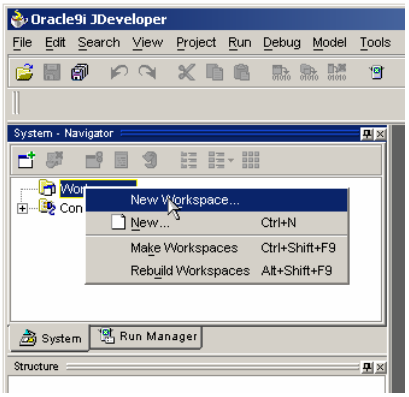
Create the Client

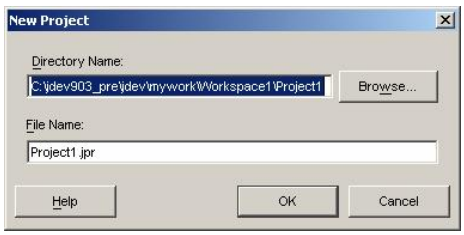
The process is simple, but requires a number of steps.

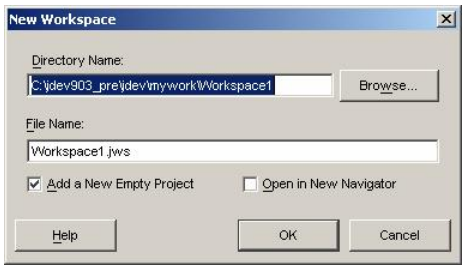
Create a JDeveloper 9i Workspace and Project

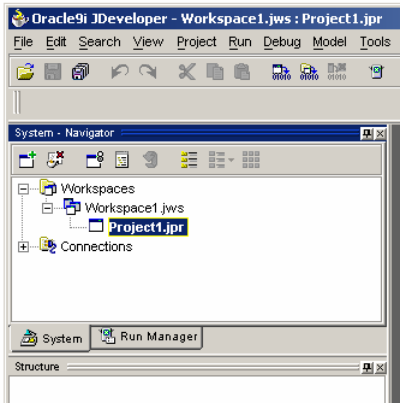
All activities in JDeveloper take place with respect to Workspaces and Projects. These may be created by the following process (illustrated by the following sequence of screenshots; it is acceptable to accept all the default values that are presented to you for this exercise):


Oracle9i JDeveloper
File Edit Search View Project Run Debug Model Tools
System - Navigator
Workspaces
Connections
System
Run Manager
Structure


Oracle9i JDeveloper
File Edit Search View Project Run Debug Model Tools
System - Navigator
New Workspace...
New... Ctrl+N
Make Workspaces Ctrl+Shift+F9
Rebuild Workspaces Alt+Shift+F9
System
Run Manager
Structure

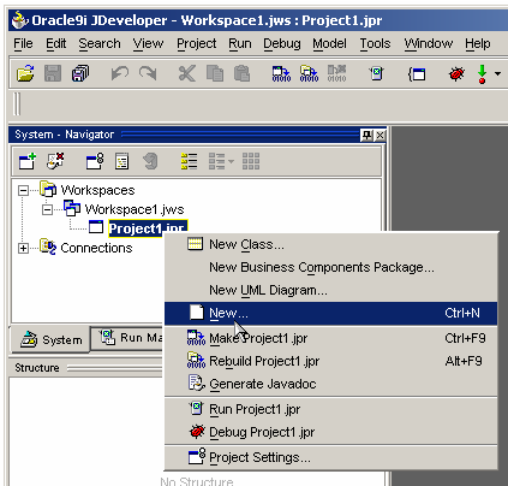

New Project
Directory Name:
C:\jdev903_pre\jdev\mywork\Workspace1\Project1
Browse...
File Name:
Project1.jpr
Help
OK
Cancel

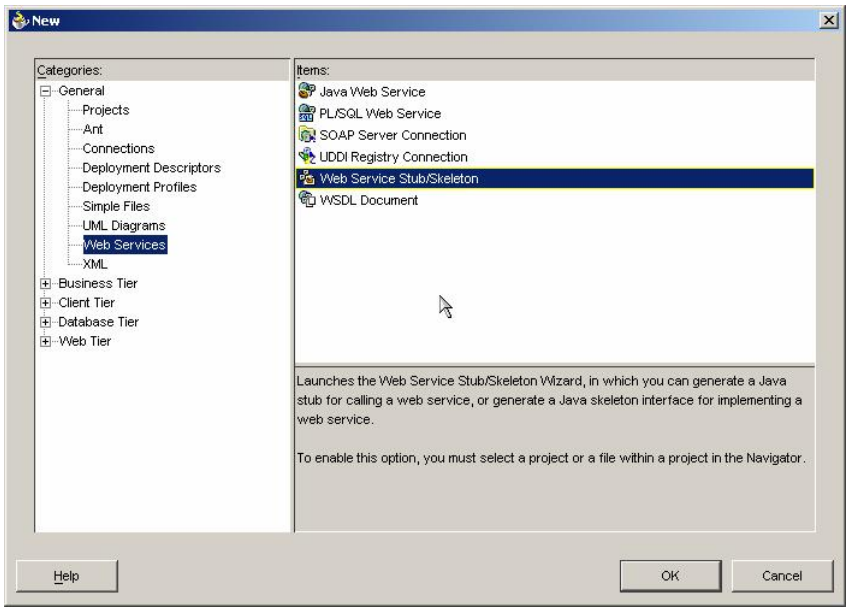

New Workspace
Directory Name:
C:\jdev903_pre\jdev\mywork\Workspace1
Browse...
File Name:
Workspace1.jws
Add a New Empty Project
Open in New Navigator
Help
OK
Cancel



Generate and Edit the WebService Client

Creating a simple client is largely a 'wizard' driven process. Follow the steps established by the WebServices wizard, as illustrated in the sequence of screenshots below:

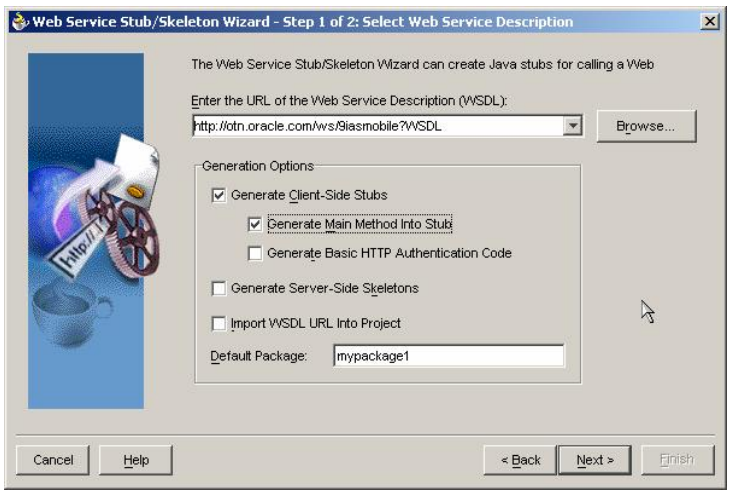


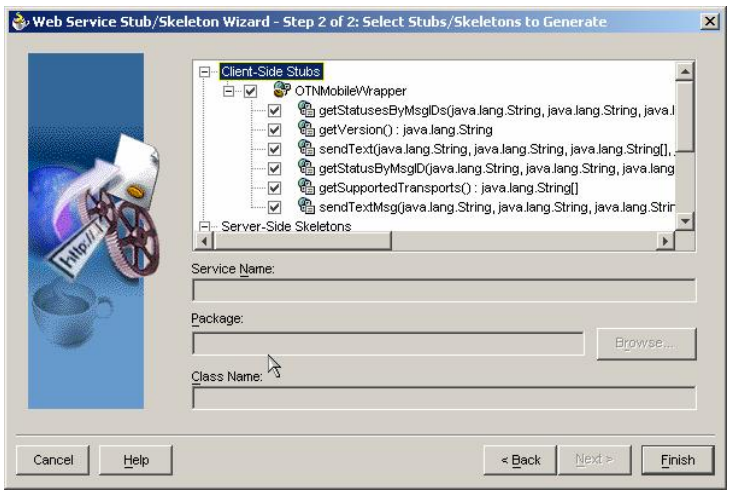


© Transentia Pty. Ltd., 10/11/2002 11:09 AM

In the following step it is important to specify the correct URL designating the location of the WSDL file which defines the SMS service's properties. The correct URL is **http://otn.oracle.com/ws/9iasmobile?WSDL**.

Also at this step, ensure that you generate a main method in the client.





Once the wizard finishes, it will create a framework for a client application in the file **OTNMobileWrapperStub.java**.

You should take a look at this file to see the code that is generated.

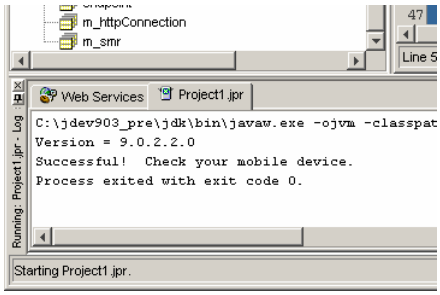
Open the generated code and edit the main method to become as follows (*note: replace* OTNusername*,* OTNpassword*,* myemail *and* myphone *with your own details*):

```
public static void main(String[] args)
  {
  try
    {
    OTNMobileWrapperStub stub = new OTNMobileWrapperStub();
    System.out.println("Version = " + stub.getVersion());

    String msgIDs[] =
```

```
      stub.sendTextMsg ("OTNusername",
                        "OTNpassword",
                         new String [] { "Email:myemail" },
                         new String [] { "SMS:myphone" },
                        "SMS From a Web Service!" );

    System.out.println("Successful! Check your mobile device.");
    }
  catch(Exception ex)
    {
    ex.printStackTrace();
    }
  }
```

Execution Time!

Press Ctrl-Shift-L to ensure that the log panel is showing at the base of the JDeveloper window and then press F11 to execute the client. You should see the result, similar to what is shown below:

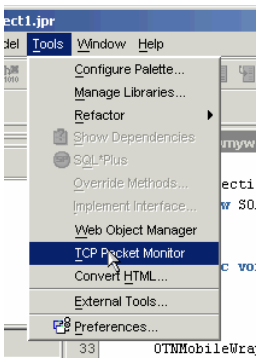


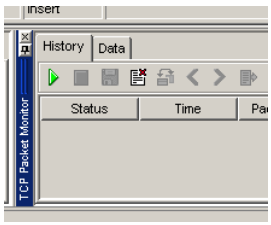
A short time later, the destination device you designated in the client application should inform you of the arrival of a new SMS message.

View the Underlying SOAP Interactions

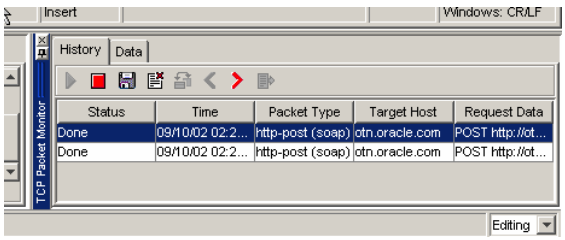
JDeveloper makes it possible to view the interactions between client and WebService.

Select Tools -> TCP Packet Monitor

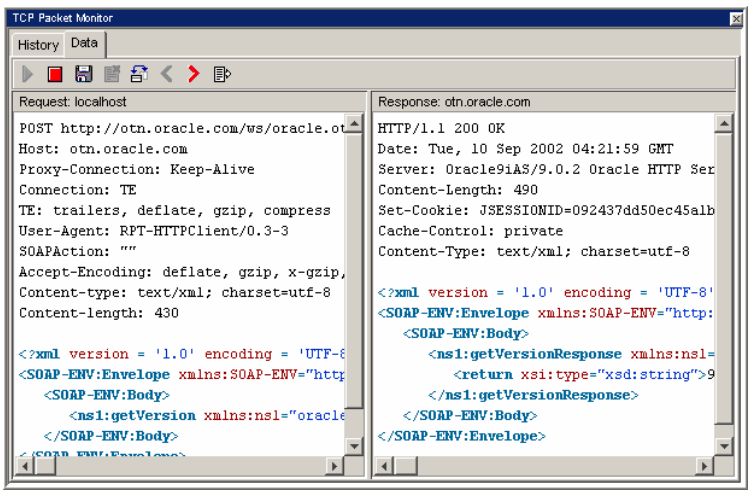




Click on the green arrow to start monitoring and then press F11 to execute the client a second time. As the interactions progress, you should see entries made in the panel, as shown:



If you double click an entry, you will see the request-response packets, similar to the illustration shown below (you may want to Ctrl-drag the window to undock it and resize it first):



Wrapup

Oracle has published a number of other useful 'live' WebServices. You can find out more about them at http://otn.oracle.com/tech/webservices/htdocs/live/content.html.

© Transentia Pty. Ltd., 10/11/2002 11:09 AM