

Remedy

White Paper

AR System 5.1

Instructions for Creating Web Service Clients

Written By: Venkat Dokiparthi

Copyright © 2002 Peregrine Remedy, Inc. or its subsidiaries. All rights reserved.

Information contained in this document is proprietary to Peregrine Remedy, Incorporated, and may be used or disclosed only with written permission from Peregrine Remedy, Inc. This white paper, or any part thereof, may not be reproduced without the prior written permission of Peregrine Remedy, Inc. This document refers to numerous products by their trade names. In most, if not all, cases these designations are claimed as Trademarks or Registered Trademarks by their respective companies.

Remedy, the Remedy Corporation logo and design, and Action Request System and AR System are registered trademarks of Peregrine Remedy, Inc. or its subsidiaries.

This document and the related software described in this manual are supplied under license or nondisclosure agreement and may be used or copied only in accordance with the terms of the agreement. The information in this document is subject to change without notice and does not represent a commitment on the part of Peregrine Remedy, Inc. Contact Remedy Customer Support to verify the date of the latest version of this document.

Creating Web Service Clients

You can create web service clients using either .Net Web Service studio or Apache Axis client library. You must first create a simple web service in AR System 5.1.

Creating a Simple Web Service

1. Create a regular form with core fields only and name the form `Simple`.
2. Right-click on `Simple` in the Forms window, and select Create Web Service.
3. Change the name of the web service to `SimpleWebService`.
4. Click the Permissions tab and select Add All to give permissions to all.
5. Save the web service.
6. Click the WSDL (Web Services Description Language) tab and change `<midtier_server>` to the location where the mid tier is installed, and click View. The WSDL for the web service should appear.

Creating a .Net Web Service Client

1. Install Microsoft .Net run-time and Visual Studio .Net 7.0.
2. Open Visual Studio .Net 7.0.
3. Choose File > New > Project.
4. Select Visual C# Projects from the set of available project types.
5. Select Console Application from Templates.
6. Enter `WSClient` for Name.
7. Enter the directory name (for example, `C:\temp`) in the location in which you choose to store the project files.
8. Click OK.
9. Choose View > Solution Explorer.
10. In Solution Explorer, right-click `WSClient` and select Add Web Reference.
11. In the Address bar, type the URL for WSDL (for example, http://<midtier_server>/arsys/WSDL/<arserver>/SimpleWebService).
12. Enter the username and password, and click Login (this is only needed if public access is not given to `SimpleWebService`).
13. Click the Add Reference button.
14. Choose Project > Show All Files.
15. Navigate to `SimpleWebService.cs` under Web References in the Solution Explorer. Right-click `SimpleWebService.cs` and select View Code.
`SimpleWebService.cs` is the proxy class generated for making requests to `SimpleWebService`.
16. Click the `Class1.cs` tab and start editing.

17. After the "using System" statement, enter the following statement on a separate line:

```
using WSClient.<midtier_server>;
```

18. Enter the following code in Main method (after the //TODO... statement):

```
// create a proxy object to make web service requests
SimpleWebService proxy = new SimpleWebService();

// set up authentication info
AuthenticationInfo authInfo = new AuthenticationInfo();
authInfo.userName = "Demo"; // give a valid AR user name
authInfo.password = ""; // give a valid password
proxy.AuthenticationInfoValue = authInfo;

// declare variables
String Assigned_To;
String Short_Description;
StatusType Status;
String Submitter;
String Request_ID;

// supply values for creating an entry using OpCreate
Assigned_To = "Frank";
Short_Description = "Testing web service";
Status = StatusType.New;
Submitter = "Joe";

// make web service request to create an entry
Request_ID = proxy.OpCreate(Assigned_To, Short_Description, Status, Submitter);
Console.WriteLine("Successfully created a request with id: " + Request_ID);

// declare additional variables for get operation
DateTime Create_Date;
String Last_Modified_By;
DateTime Modified_Date;
String Status_History;

// make web service request to get the entry that was created above
Assigned_To = proxy.OpGet(ref Request_ID, out Create_Date, out Last_Modified_By, out
Modified_Date,
    out Short_Description, out Status, out Status_History, out Submitter);

Console.WriteLine();
Console.WriteLine("Following values have been returned by OpGet");
Console.WriteLine();
Console.WriteLine("Request_ID : " + Request_ID);
Console.WriteLine("Create_Date : " + Create_Date);
Console.WriteLine("Last_Modified_By : " + Last_Modified_By);
Console.WriteLine("Modified_Date : " + Modified_Date);
Console.WriteLine("Short_Description : " + Short_Description);
Console.WriteLine("Status : " + Status);
Console.WriteLine("Status_History : " + Status_History);
Console.WriteLine("Submitter : " + Submitter);
Console.WriteLine("Assigned_To : " + Assigned_To);
```

19. Choose Build > Rebuild All.
20. Change directory to C:\temp\WSClient\bin\Debug.
21. Type WSClient.exe at the command prompt to run the client.

The following output appears:

```
Successfully created a request with id: 000000000000001
```

```
Following values have been returned by OpGet
```

```
Request_ID : 000000000000001
Create_Date : 7/17/2002 12:31:04 PM
Last_Modified_By : Demo
Modified_Date : 7/17/2002 12:31:04 PM
Short_Description : Testing web service
Status : New
Status_History : 2002-07-17T11:31:04-08:00 Demo

Submitter : Joe
Assigned_To : Frank
```

Creating an Apache Axis Web Service Client

1. Install JDK 1.3.1 or above. Make sure java and javac are available in the PATH.
2. Install Apache Axis 1.0 from <http://xml.apache.org/axis>.
3. Install Xerces 2 Java Parser 2.2.0 from <http://xml.apache.org/xerces2-j/index.html>.
4. Install JavaServer Pages 1.2 classfiles from <http://java.sun.com/products/jsp/download.html>.
5. Set CLASSPATH to include Axis and Xerces jar files as follows:

```
set AXIS_DIR=<Axis Install Dir> (For example, S:\products\xml-axis-10)
set AXIS_LIB_DIR=%AXIS_DIR%\lib
set XERCES_DIR=<Xerces Install Dir> (For example, S:\xerces-2_2_0)
set JSP_DIR=<JSP Install Dir> (For example, S:\products\jsp-1_2)
set CLASSPATH=.;%AXIS_LIB_DIR%\axis.jar;%AXIS_LIB_DIR%\saa.jar;%AXIS_LIB_DIR%\log4j-1.2.4.jar;%AXIS_LIB_DIR%\wsdl4j.jar;%AXIS_LIB_DIR%\commons-discovery.jar;%AXIS_LIB_DIR%\jaxrpc.jar;%AXIS_LIB_DIR%\commons-logging.jar;%XERCES_DIR%\xercesImpl.jar;%XERCES_DIR%\xmlParserAPIs.jar;%JSP_DIR%\servlet.jar
```

6. Create a directory for building an Axis web service client (for example, C:\temp\axisclient).
7. Change directory to C:\temp\axisclient.
8. At the command prompt, type the following:

```
java org.apache.axis.wsdl.WSDL2Java -W -p ARWSClient
http://<midtier_server>/arsys/WSDL/<arserver>/SimpleWebService
```

This generates proxy classes for writing the client program.

9. Create a directory for storing class files (for example, C:\temp\axisclient\classes).
10. Change directory to ARWSClient.

11. Create a new file called `TestClient.java` and add the following lines of code:

```
/**
 * TestClient.java
 *
 */

package ARWSClient;

import javax.xml.rpc.holders.StringHolder;
import javax.xml.rpc.holders.CalendarHolder;

public class TestClient {
    public static void main(String args[]) throws Exception
    {
        SimpleWebServicePortType binding;

        binding = new SimpleWebServiceServiceLocator().getSimpleWebServiceSoap();

        // declare variables
        CreateInputMap createInput = new CreateInputMap();

        // supply values for creating an entry using OpCreate
        createInput.setAssigned_To("Frank");
        createInput.setShort_Description("Testing Axis Client");
        createInput.setStatus(StatusType.New);
        createInput.setSubmitter("Joe");

        // make web service request to create an entry
        CreateOutputMap createOuput = binding.opCreate(createInput);
        System.out.println("Successfully created a request with id: " +
createOuput.getRequest_ID());

        // declare additional variables for get operation
        GetInputMap getInput = new GetInputMap();
        getInput.setRequest_ID(createOuput.getRequest_ID());

        ARWSClient.GetOutputMap getOutput;
        getOutput = binding.opGet(getInput);

        System.out.println();
        System.out.println("Following values have been returned by OpGet");
        System.out.println();
        System.out.println("Request_ID : " + getOutput.getRequest_ID());
        System.out.println("Assigned_To : " + getOutput.getAssigned_To());
        System.out.println("Create_Date : " + getOutput.getCreate_Date().getTime());
        System.out.println("Last_Modified_By : " + getOutput.getLast_Modified_By());
        System.out.println("Modified_Date : " + getOutput.getModified_Date().getTime());
        System.out.println("Short_Description : " + getOutput.getShort_Description());
        System.out.println("Status : " + getOutput.getStatus());
        System.out.println("Status_History : " + getOutput.getStatus_History());
        System.out.println("Submitter : " + getOutput.getSubmitter());
    }
}
```

12. Type the following command:

```
javac -d ..\classes *.java
```

13. Open the mid tier configuration tool, and enter an Anonymous User Name under WebService Settings. (For example, Demo. The password of the user must be blank.)

14. Change directory to ..\classes.

15. Type the following command to run the client:

```
java ARWSSClient.TestClient
```

16. The output should appear as follows:

```
Successfully created a request with id: 0000000000000015
```

```
Following values have been returned by OpGet
```

```
Request_ID : 0000000000000015
```

```
Assigned_To : Frank
```

```
Create_Date : Wed Jul 17 20:12:10 PDT 2002
```

```
Last_Modified_By : Demo
```

```
Modified_Date : Wed Jul 17 20:12:10 PDT 2002
```

```
Short_Description : Testing Axis Client
```

```
Status : New
```

```
Status_History : 2002-07-17T19:12:10-08:00 Demo
```

```
Submitter : Joe
```