

Base Language - Feature #2208

create a java application running web services for testing purposes

12/10/2013 04:45 PM - Constantin Asofiei

Status:	Closed	Start date:	
Priority:	Normal	Due date:	
Assignee:	Evgeny Kiselev	% Done:	0%
Category:		Estimated time:	0.00 hour
Target version:	Cleanup and Stablization for Server Features	vendor_id:	GCD
billable:	No		
Description			
Related issues:			
Related to Base Language - Feature #1645: implement web services support		Closed	02/20/2013 08/09/2013
Related to Base Language - Feature #2232: finish testing the wrapped mode usi...		Closed	

History

#1 - 12/10/2013 04:55 PM - Constantin Asofiei

Build a java application running web services covering all features of a SOAP message (soap header, parameter types, fault, etc) and of a WSDL document defining the services (custom types, binding, port type, etc).

The code for this application should be in a non-P2J project, under the testcases/uast/web_services/ folder.

#2 - 12/11/2013 07:39 PM - Evgeny Kiselev

- Status changed from New to WIP

#3 - 12/12/2013 01:31 AM - Constantin Asofiei

First, please create some simple SOAP services: exposed over both HTTP and HTTPS, with and without parameters; I need this as soon as possible, to be able to move deeper into testing. There are a few freely available SOAP services out there, but I don't want to rely on them.

#4 - 12/16/2013 03:57 AM - Constantin Asofiei

Some more notes about what should be covered:

- for the various CONNECT statement options, I need more than one WSDL file, as we need to cover:
 - zero, one or more bindings
 - zero, one or more port types
 - zero, one or more services
 - port types associated with more than one binding
 - services associated with zero, one or more port/binding pairs
 - http and https support
 - user/password access to the WSDL document and to the SOAP endpoint.As only the -WSDL option is mandatory, I need to make sure the target service/port/binding is determined properly.
- for the invocation of service operations, considering that the parameters can be passed in both wrapped and unwrapped mode (see the dvwsv.pdf document), there should be operations with:
 - only output messages (one or more)
 - only input messages (one or more)
 - one input and one output
 - one input, more than one output
 - more than one input, one output
- at first, is enough to use int and char data types for the parameters (combinations of these two). Subsequently, I need:
 - parameters with all supported data types. See the Defining and passing Web service method parameters chapter in the dvwsv.pdf document for what XML Schema data types are supported.
 - complex type parameters, custom data types (see the GetSupplierBySupplyType defined by the <http://www.webservices.net/medicareSupplier.asmx?WSDL> file as an example).

4. support for the request/response SOAP headers. As I understand, this is custom XML data passed or received from the service.
5. support for SOAP Fault responses.
6. lower priority for now: invoke web services using TABLE and TABLE-HANDLE parameters.

To conclude:

- make sure the network port on which the web services are running can be configured (maybe passed as an argument).
- use axis2 or what other web service library you find useful. tomcat as a container I think is acceptable.
- the server-side of the web service must not be complicated. In some cases, is enough to return the input parameters or, if input parameters are not passed, to return something like "this is nth invocation of this operation".
- build this in phases, from the simple to complex ones. Start with writing WSDL documents configured in various ways, as the operations (and the web service server) can be implemented later. Use the browsdoc tool to validate the WSDL document.

#5 - 01/03/2014 07:10 AM - Constantin Asofiei

Evgeny, please make it a priority to add basic services which work with SOAP Faults and SOAP Headers (response and request).

#6 - 01/06/2014 01:20 PM - Constantin Asofiei

Some more notes: 4GL seems to alter the received SOAP Envelope by adding i.e. additional XMLNS attributes at the root node for a SOAP Fault Detail. For this reason, is best for the application to log, in distinct XML files:

- the full received SOAP Envelope
- the full sent SOAP Envelope

This will allow us to find what additional prepare is done to the envelope and/or to the nodes exposed to 4GL: SOAP Fault, SOAP Request/Response Header, Request/Response Messages.

#7 - 01/24/2014 09:10 AM - Greg Shah

- Target version changed from Milestone 7 to Milestone 11

#8 - 02/10/2014 08:17 PM - Evgeny Kiselev

I have problem with SSL. It's not possible to do it in the internal Axis2 web server(SimpleHttpServer).

Here is a link, which explain how it can works in tomcat or other server like jetty.

<http://axis.apache.org/axis2/java/core/docs/servlet-transport.html>

But it's not about common Axis2 Simple Server. This article about additional settings in Axis2 for servlets in tomcat/jetty/or other app server.

But I've find interesting axis2 jira issue.

<https://issues.apache.org/jira/browse/AXIS2-5180>

In the comment there is solution how to do it, but it's very old solution.

I'm trying to run it in embedded Jetty.

#9 - 02/11/2014 02:12 PM - Constantin Asofiei

OK, I think I understand your problem: you are using embedded AXIS2 web server, which does not support HTTPS. And you need to put it in a container like Jetty/Tomcat, to be able to accept HTTPS requests, right?

#10 - 02/11/2014 05:48 PM - Evgeny Kiselev

Yes, you are right.

#11 - 02/12/2014 01:26 AM - Constantin Asofiei

Evgeny Kiselev wrote:

Yes, you are right.

If you think you can solve this in 2-4 hours, then go ahead and solve it. Else, document all your findings related to SSL support here and will postpone the SSL configuration for a later time.

#12 - 02/13/2014 07:23 PM - Evgeny Kiselev

Where I can upload update ?

I've ~20mb zip archive (with axis2 libs and configs).

#13 - 02/14/2014 02:16 AM - Constantin Asofiei

Evgeny Kiselev wrote:

Where I can upload update ?

I've ~20mb zip archive (with axis2 libs and configs).

Commit it to testcases/uast/web_services/server/ folder. You might want to upload the archive to devsrv01 and commit it from there - depending on your connection, committing directly from your machine might take a while. It will be faster if you checkout the testcases project on devsrv01 and commit from there.

#14 - 02/19/2014 11:59 PM - Evgeny Kiselev

I've committed intermediate version of test server to the testcases/uast/web_services/server/.

To run this from IDE need to run class com.p2j.soap.StartServer. Without parameters it runs on 8080 port.

There is 3 parameters:

- port
- axis2 repository location, by default it is root dir of project testcases/uast/web_services/server/
- axis2 configuration, by default it is root dir of project testcases/uast/web_services/server/

There are 3 example web services.

#simple_service directory - accept JavaBean object.

#test_service2 directory - simple service which can accept and return String and primitives objects.

#MediCareSupplier directory - this is example of <http://www.webservices.net/medicareSupplier.asmx?WSDL>. It does not have implementation of particular methods and was generated by wsdl2java utility from Axis2 package.

In each service directory there is exists ant's build.xml script. which generate service jar file and put it into Axis2 repository.
Also there is possible to debug each service from IDE.

Jetty implementation with SSL support is coming soon.

#15 - 06/24/2014 11:42 AM - Greg Shah

- *Status changed from WIP to Closed*

Marius has done a code review and he notes that the code is acceptable and that this task can be closed.

Please do update this task history with a note about the status of SSL support, so that any future development that is needed can be easily known by reading here.

#16 - 11/16/2016 12:07 PM - Greg Shah

- *Target version changed from Milestone 11 to Cleanup and Stabilization for Server Features*