

Building Web Services as a Plugin

Additional SOAP web services can be added to ClearSpace. These web services can be deployed by using the ClearSpace plugin mechanism. The ClearSpace web services are built using the XFire web services toolkit. Additional Documentation on XFire can be found at <http://xfire.codehaus.org/>.

Note: By default, web services are disabled in ClearSpace. You can enable them in the admin console. In the console, go to **System > Settings > Web Services**, then click **Enable** for the style you want. Be sure that the **User Access** and **Force SSL** settings are what you want also.

Building a Web Service Class

Building a web services class in jive is very simple. First all of make a new plugin, please review the [Plugin Development Guide](#) for the details.

A web service for the most part is a simple plain old java object. A couple of annotations must be set on the class as should in the example below:

```
package com.jivesoftware.clearspace.plugin.example;

import javax.jws.WebService;
import javax.jws.WebMethod;
import javax.jws.soap.SOAPBinding;

/**
 * Simple service that echoes a string back telling who said it.
 */
@WebService(serviceName = "EchoService", targetNamespace =
"http://jivesoftware.com/webservices")
@SOAPBinding(style = SOAPBinding.Style.DOCUMENT,
    use = SOAPBinding.Use.LITERAL,
    parameterStyle = SOAPBinding.ParameterStyle.WRAPPED)
public class EchoService {

    @WebMethod
    public String echo(String something) throws UserNotFoundException {
        return something;
    }
}
```

Required Annotations:

javax.jws.WebService: Contains the name of the service, the namespace should be <http://jivesoftware.com/webservices>.

javax.jws.soap.SOAPBinding: Use exact annotation as described above. All services in jive software must be Document Literal with wrapped parameters.

javax.jws.WebMethod: Must be defined on every method that should be exposed as an operation.

If you webservice needs access to the `com.jivesoftware.community.JiveContext` (Core API entry point) or a particular a particular users authentication information (`com.jivesoftware.base.AuthFactory`) you can acquire this by using the static utility method exposed on `com.jivesoftware.community.webservices.server.WSUtil`.

For all custom objects that must be exposed, the class must be annotated with the `com.jivesoftware.community.webservices.annotations.WSBean` annotation. This annotation tells XML-RPC support that it is safe to serialize/deserialize the object. Service deployment will fail if this annotation is not included.

Deploying a Web Service Inside a Plugin

In your `plugin.xml` file add `webservice` element for each of your web service classes:

```
<plugin xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="http://www.jivesoftware.com/schemas/clearspace/1_1/plugin.xsd">
  <name>example</name>
  <description>Simple example of a plugin using webwork.</description>
  <author>Jive Software</author>
  <version>1.0.0</version>
  <minServerVersion>1.1.0</minServerVersion>

  <webservice class="com.jivesoftware.clearspace.plugin.example.EchoService" />
</plugin>
```

Finally deploy the plugin in your Clearspace application and verify that the plugin installed by visiting the url `http://<clearspace server path>/rpc/soap`.