

When you use JDeveloper for Artix work you will need to generate code from WSDL files. The code generated may be for a container or a client at any one time. However, regardless of the specific target type you need to use the same utility named `wsdl2java` that is provided by Artix for this task. JDeveloper provides a mechanism that allows you to do this with relative ease via the External Tools... option under the Tools menu. Once you select that option from the menu you get a list of currently defined external tools you can execute along the option to manage the list. The list of tools comes from the `tools.xml` file that is located in your `<install dir>\jdev\system\oracle.jdeveloper.10.1.3.40.66` directory when you are using JDeveloper version 10.1.3.2.0.4066. To work with Artix code generation you will need to define three external tools based on the `wsdl2java` utility. Depending on your Artix installation and existing contents of your computer's PATH, you may or may not be able to call the `wsdl2java` utility directly. For example, on my computer my PATH has some entries in it that breaks the out of the box Artix batch file which forced me to edit up a custom one. Whatever batch file you use, you should test that it can be called from the command line. Once that is done you will define three external tools:

- External tool to generate types and interfaces for a container. This is the tool you use when you first create generate code for a WSDL, or you alter the port or binding for a service. For that use the following arguments: `-d ${project.dir}\gen -merge -plugin -types -interface -verbose ${file.path}`
- External tool to generate just types for either a client or a container. This is the tool you use when you have already generated code for a WSDL and you need to update code to reflect a change in message structure. For that use the following arguments: `-d ${project.dir}\gen -types -all -verbose ${file.path}`
- External tool to generate a client interface. You use this call to generate code for a new or updated WSDL where Artix will call a web service. For that you use the following arguments: `-d ${project.dir}\gen -merge -all -verbose -client ${file.path}`

All of the arguments for the tools assume you want to generate code within the folder named "gen" for a given JDeveloper project directory. For each project where you generate code you need to add the "gen" directory to your project source content. You do this via the project properties dialog within the top level of the project content tree. Just add a reference to the "gen" directory and make sure the option to include content from subfolders is selected.

For those who might want to take a crack at directly editing the `tools.xml` file, the specific XML for the tools mentioned in that file are as follows:

```
<Item class="oracle.jdevimpl.externaltools.program.ExternalProgramTool">
  <arguments>-d ${project.dir}\gen -merge -plugin -types -interface -verbose
${file.path}</arguments>
  <availability>0</availability>
  <availabilityExpression/>
  <availableTypes class="java.util.ArrayList"/>
  <caption>Artix Plug-In ( types & interface)</caption>
  <executable>D:\Dev\ATT\my_wsdltojava.bat</executable>
  <iconURL protocol="ide.fileicon" path="D:/Dev/ATT/my_wsdltojava.bat" />
  <integrationMask>1</integrationMask>
  <reloadBuffers>>false</reloadBuffers>
  <runDirectory>D:\IONA\artix\4.1\bin</runDirectory>
```

```

        <scannerTag/>
        <toolTipText/>
    </Item>
    <Item class="oracle.jdevimpl.externaltools.program.ExternalProgramTool">
        <arguments>-d ${project.dir}\gen -types -all -verbose
${file.path}</arguments>
        <availability>0</availability>
        <availabilityExpression/>
        <availableTypes class="java.util.ArrayList"/>
        <caption>Artix wsdl2java - generate types</caption>
        <executable>D:\Dev\ATT\my_wsdltojava.bat</executable>
        <iconURL path="../../lib/jdev.jar" jar-
entry="oracle/jdevimpl/externaltools/program/programtool.gif"/>
        <integrationMask>1</integrationMask>
        <reloadBuffers>>false</reloadBuffers>
        <runDirectory>D:\IONA\artix\4.1\bin</runDirectory>
        <scannerTag/>
        <toolTipText/>
    </Item>
    <Item class="oracle.jdevimpl.externaltools.program.ExternalProgramTool">
        <arguments>-d ${project.dir}\gen -merge -all -verbose -client
${file.path}</arguments>
        <availability>0</availability>
        <availabilityExpression/>
        <availableTypes class="java.util.ArrayList"/>
        <caption>Artix wsdl2java - generate client</caption>
        <executable>D:\Dev\ATT\my_wsdltojava.bat</executable>
        <iconURL protocol="ide.fileicon" path="D:/Dev/ATT/my_wsdltojava.bat"/>
        <integrationMask>1</integrationMask>
        <reloadBuffers>>false</reloadBuffers>
        <runDirectory>D:\IONA\artix\4.1\bin</runDirectory>
        <scannerTag/>
        <toolTipText/>
    </Item>

```

If you try to use the XML shown you will likely need to edit the paths in the `<executable>` and `<runDirectory>` tag