

Code generation IDEA plug-in

Introduction

This document explains installation and usage of IDEA code generation plug-in.

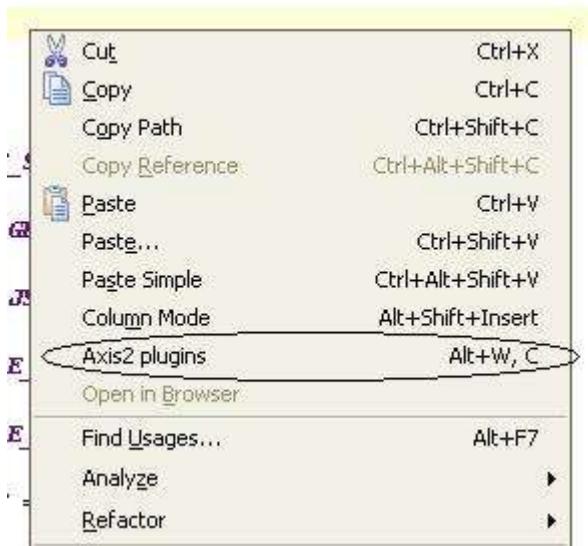
Installation

First download the plug-in [link](#) which is a zipped file, and extract that into plugins directory in IDEA installation directory or plugins directory in user directory (usr/idea/conf/plugins), if you have correctly extracted the file you would see a directory called Axis2 and inside there should be only one directory called lib and bunch of .jar files can be seen inside that.

Next step is to restart IDEA to changes to be taken place , when you restart IDEA you would see following icons in IDEA if the plug-in has been installed correctly.



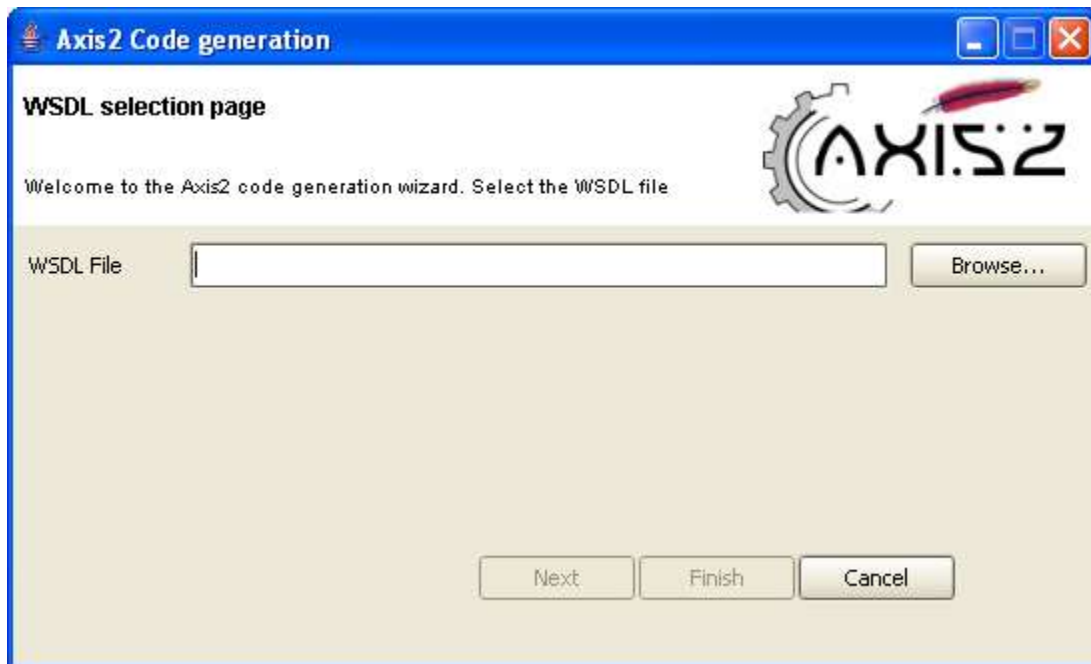
As well as if you right-click on editor page you would see a link for the same plug-in form that too.



When you click one of them a window will appear by asking to select one from two options.

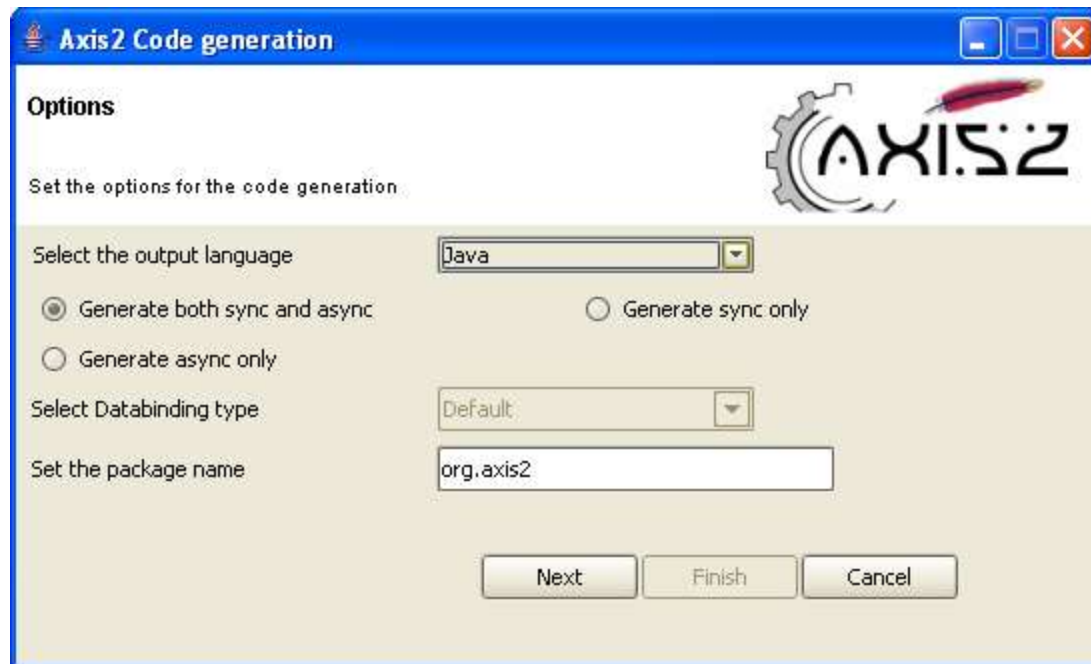


Click on WSDL2Java codegenaration and click OK to generate code from a WSDL file. Then the following window will appear.



To move on to the next page the WSDL file location must be given. The browse button can be used to easily browse for a file rather than typing the whole path.

Once the WSDL file is selected, codegen options are to be selected. By far this is the most important page in this wizard, which determines the characteristics of the code being generated. Novices need not worry about these options since the most common options are defaulted, But advanced users will find it very easy to "turn the knobs" using these options.

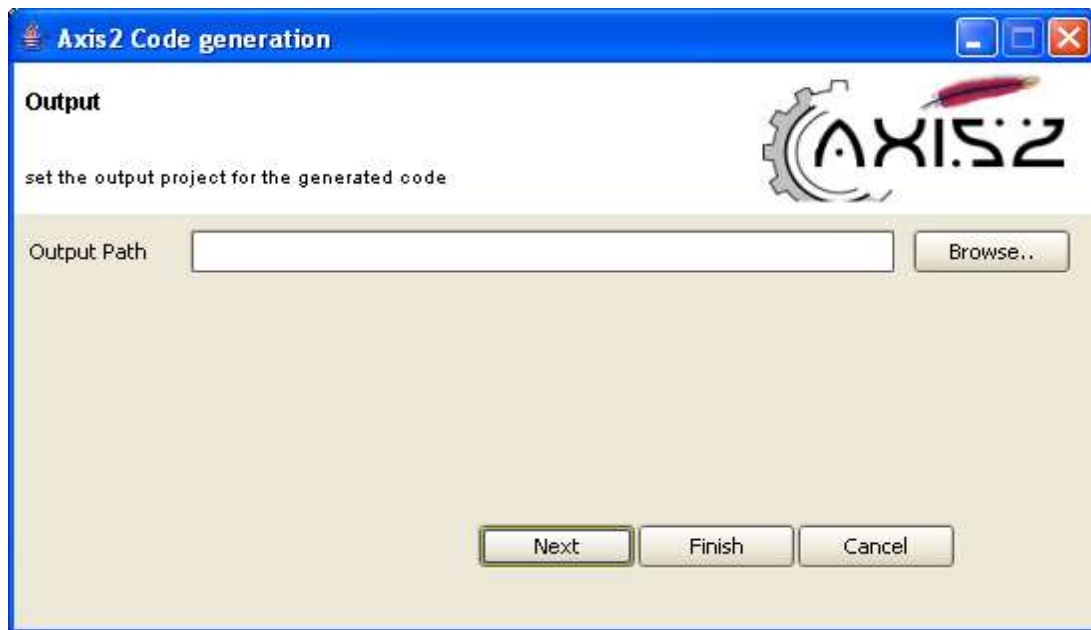


Output language can be Java, C# or C++ but we not fully tested C# and C++ codegeneration there for it is better to select Java as output language.

Since Axis2 support both synchronous and asynchronous client programming model , the tool has provide a way to select the options.

Data binding type , the default data binding type is pure OM, though the tool is capable of providing XML beans due to class loading issues in XML beans current implementation only generate code with OM.

Package name : The package name of the generated code , you can set the package name that you want.



To select the output location or the location that the to be generated , you do not need to type the output location you can brows and select the location.



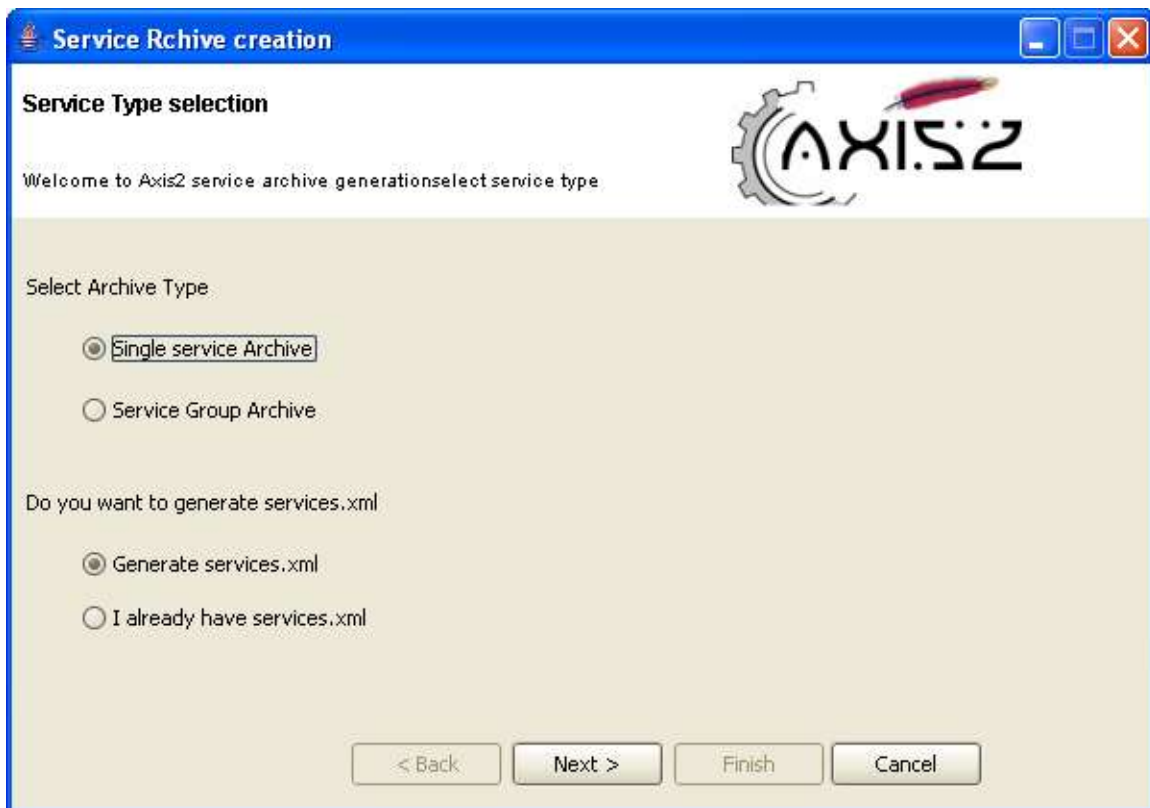
If everything has gone well you will see output like above.

IDEA service archive generation plug-in

First step is to click on Axis2 IDEA plug-in icon and select service archive creation on that.



Then a wizard will appear by asking to select archive type



Select archive type : In Axis2 user can deploy single service or service group so you can select the service type you want to create.

If you already have services.xml you can skip some of the steps in the wizard so if you have services.xml with you click on “I already have services.xml” and click on next , if you do not have the tool will create services.xml for you.

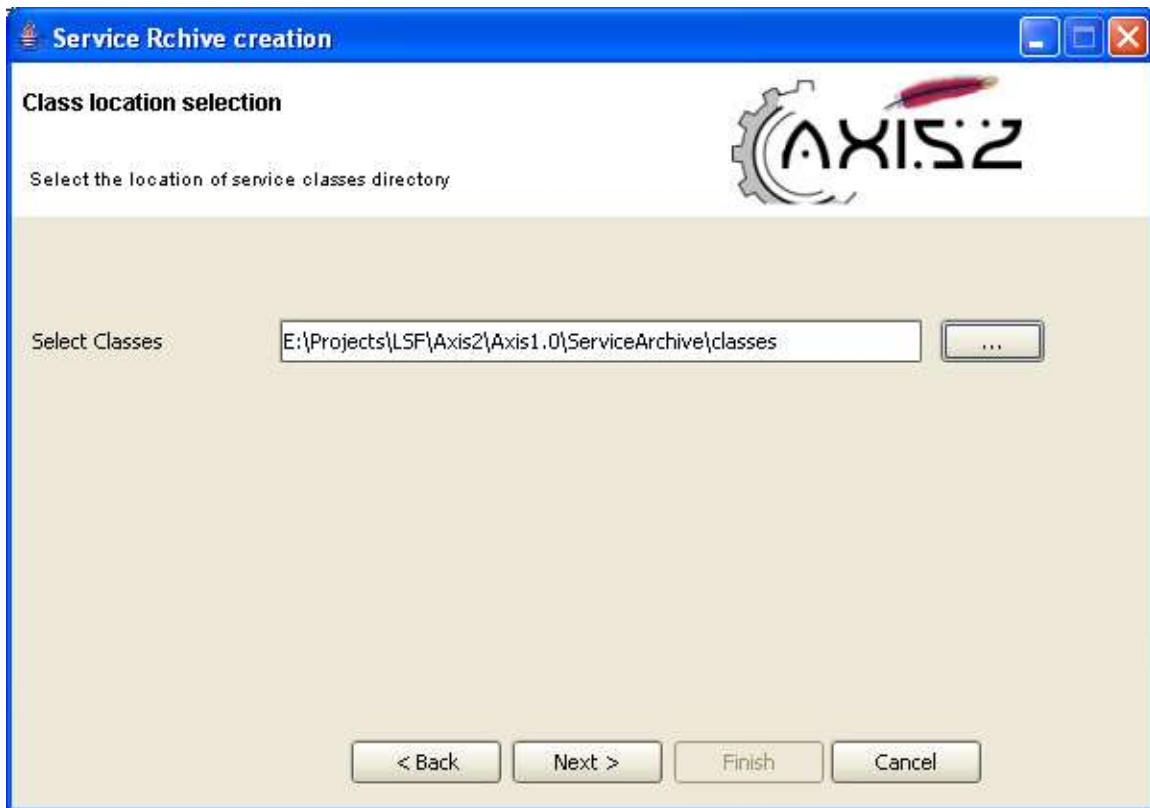
Depending on the option you select there can be three sub wizards,

1. Sub wizard 1(create single service and its services.xml)
2. Sub wizard 2 (create service group and its services.xml)
3. Sub wizard 3 (create service/service group using give services.xml)

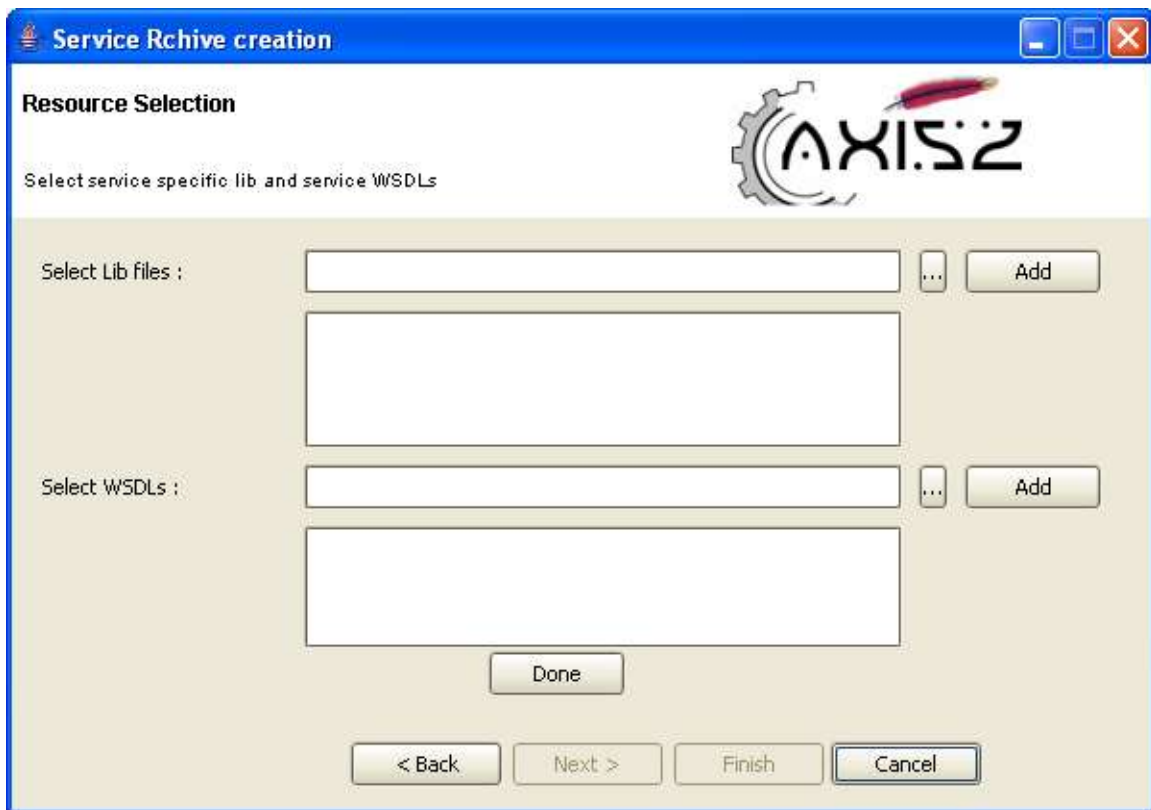
Option 1 and option 2 follow the same set of steps except some looping mechanism in the middle of the wizard.

Sub wizard 1 & Sub wizard 3

From the following page you have to select the location of classes directory , where you compile output location , you do not need to type it, you can easily browse and select

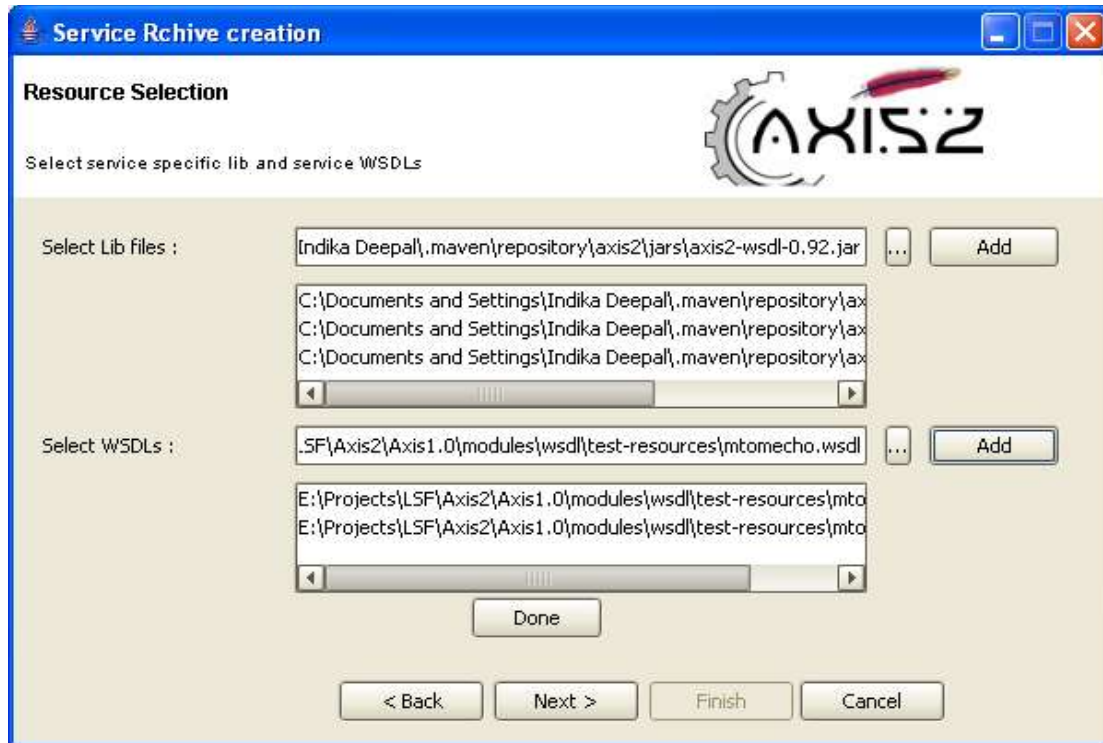


When you click on next button , the page is to select service specific external libraries and service WSDL file(if you want to add multiple WSDL files you can do that too , in the case of service group you can add any number of WSDL files depending on the services in service group)



Adding libraries first click on small button to browse library file and then click on Add then the selected file will be appear in the list box.

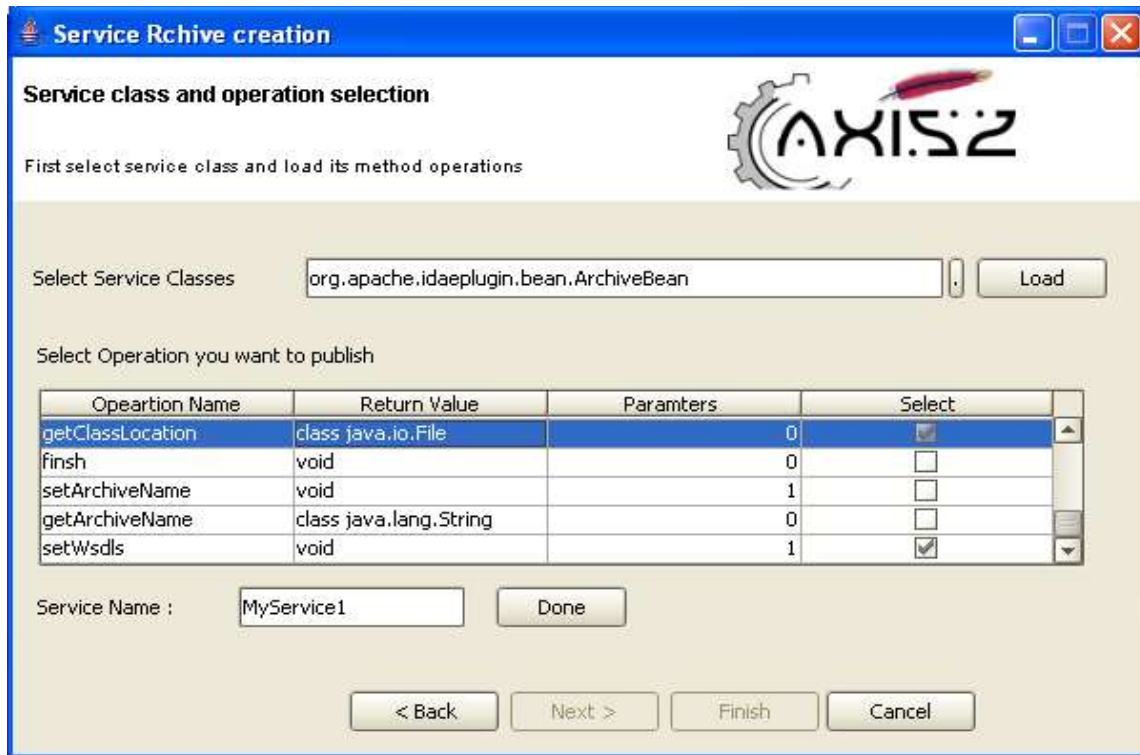
To add WSDLs first click on small button to select WSDL file and then click Add button to add the file to the list.



Next page is to select service implementation class (in the case of service group same page will be looped to select multiple service implementation classes)

Step 1:

Select a service implementation class by browsing and the press Load button to load all the public methods in that class and you can select the methods that you want to publish.



Service Rchive creation

Service class and operation selection

First select service class and load its method operations

Select Service Classes:

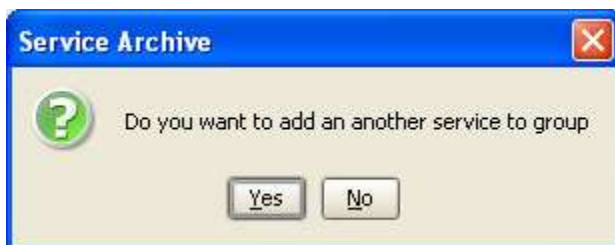
Select Operation you want to publish

Opeartion Name	Return Value	Paramters	Select
getClassLocation	class java.io.File	0	<input checked="" type="checkbox"/>
finsh	void	0	<input type="checkbox"/>
setArchiveName	void	1	<input type="checkbox"/>
getArchiveName	class java.lang.String	0	<input type="checkbox"/>
setWsdls	void	1	<input checked="" type="checkbox"/>


Service Name :

In service name textbox you can type the name of the service you want , but remember service name should be across the system.

When you have complete on this particular service press Done (in the first place if you have selected service group when you press the button then the following dialog will appear to add some more service to service group)



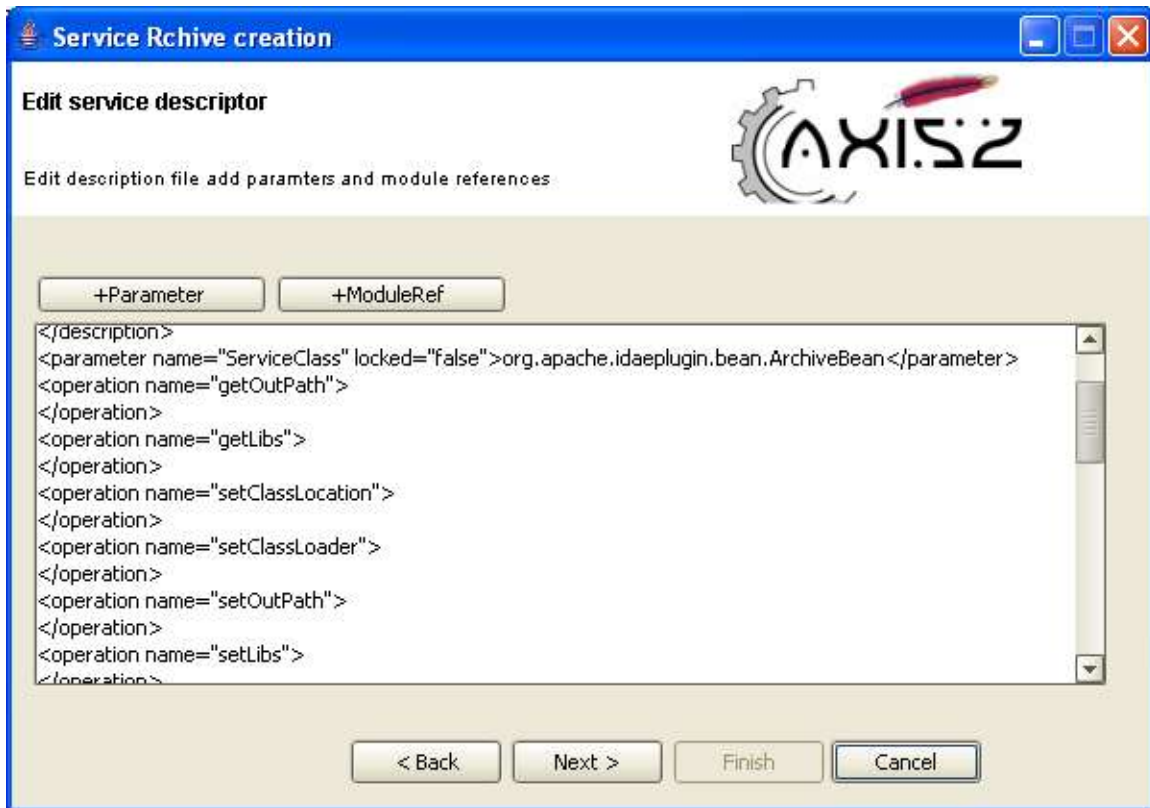
Service Archive

 Do you want to add an another service to group

If you press yes , you have to follow the same procedure to add some other service to service group.

If you press No you can , Next button will be enabled and you can go to next page.

Note from this onward the steps are similar to all the sub wizards.



This page displays the services.xml file (either you given one in the case of “I already have services.xml” or the system generated one for you.)

This page can editable and provide a way to add parameters and module references to any level.

Note : When you click on one of those button remember corresponding texts will be added to the current mouse position. Therefor click on the location you want to add the parameter and then press the button . If you click on the parameter button a window will appear asking to give parameter name and parameter value.

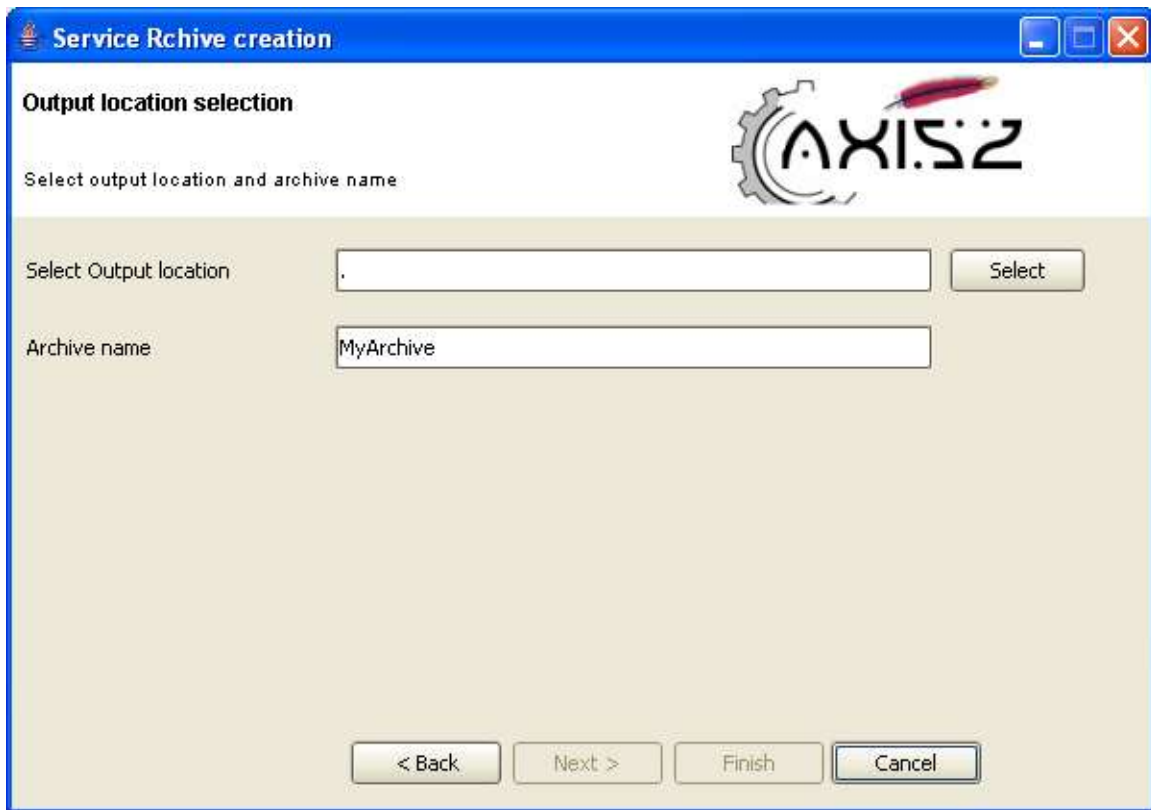


Or you can manually add parameters to any where in the document as you wish.

Adding module references can be done by clicking on +ModuleRef button in the window you have to type the name of the module to be engaged.



When you complete press the Next button to go to final stage.



Next step is to select output location , the location output archive file should be created.

In the Archive Name just type the name of the archive file you want to put and that will finally become the service group name.

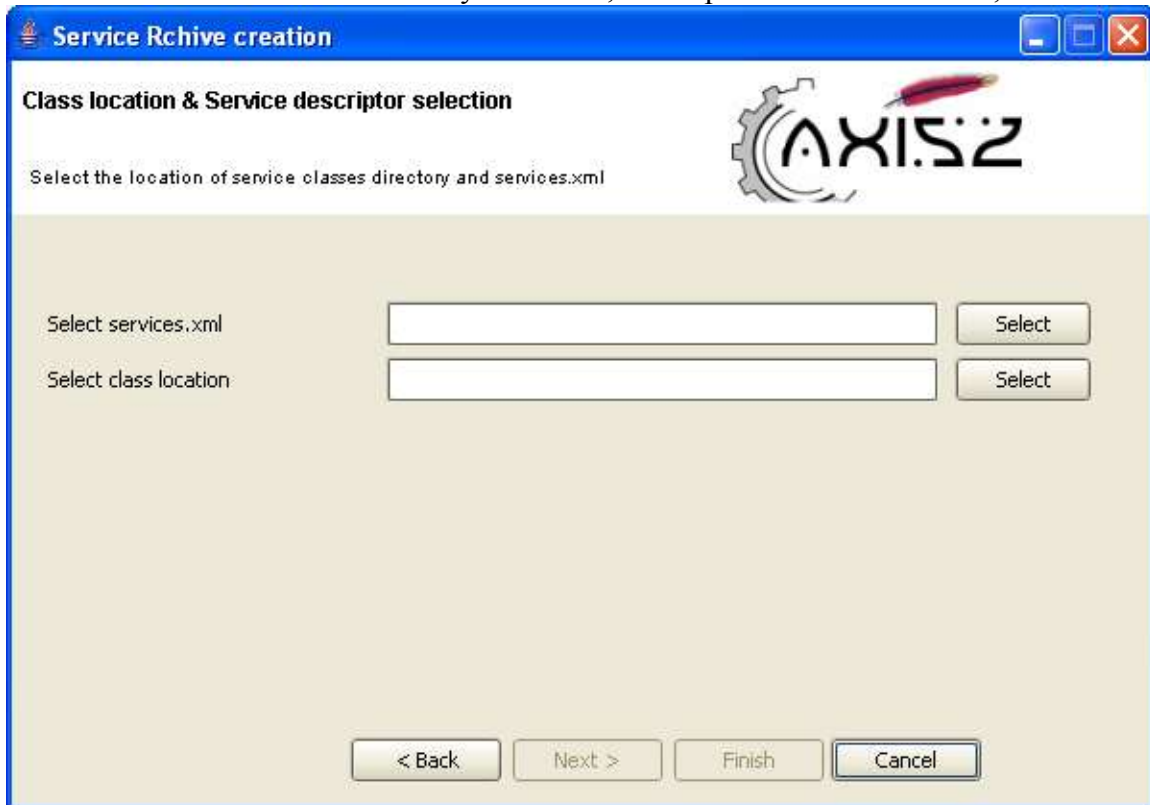
Note : Do not include file extension name when you write archive name we will create that for you.

When you are done click the Finish button , if everything has gone well you will see following message.



Sub wizard 2

In the case of services.xml is already available , the steps look like as follows,



This page is to select both location of services.xml and the location of classes directory, you can press select button and browse the file system to find required document and location.

If we click on Next button then the procedure is same as above, in next page provide a way to edit services.xml as mentioned earlier.