

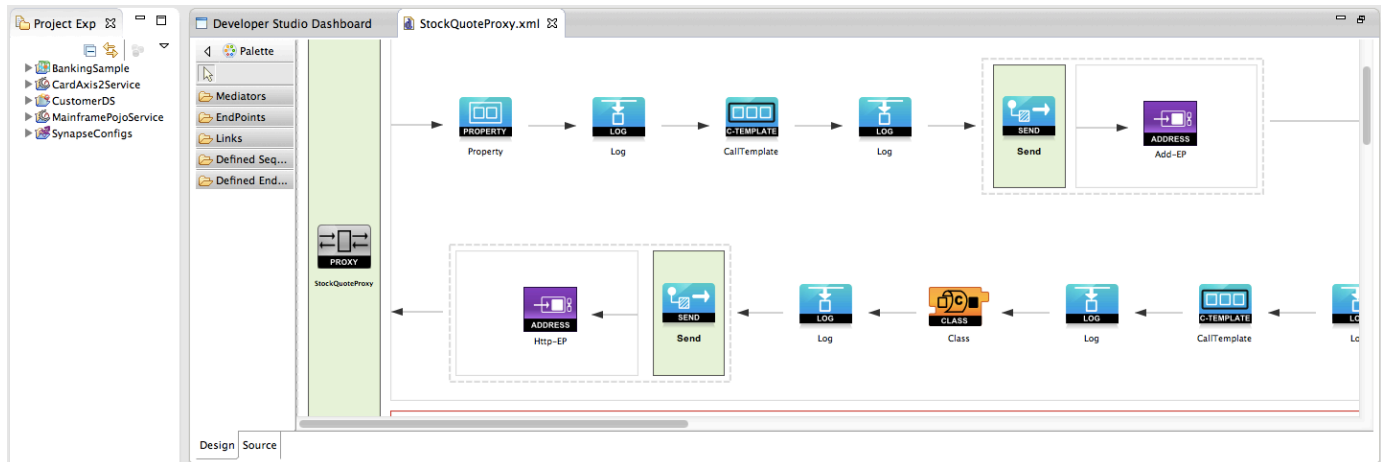
Packaging Artifacts Into Deployable Archives

This page describes how to package your artifacts into archives that you can deploy to [Carbon](#) or other servers. It contains the following sections:

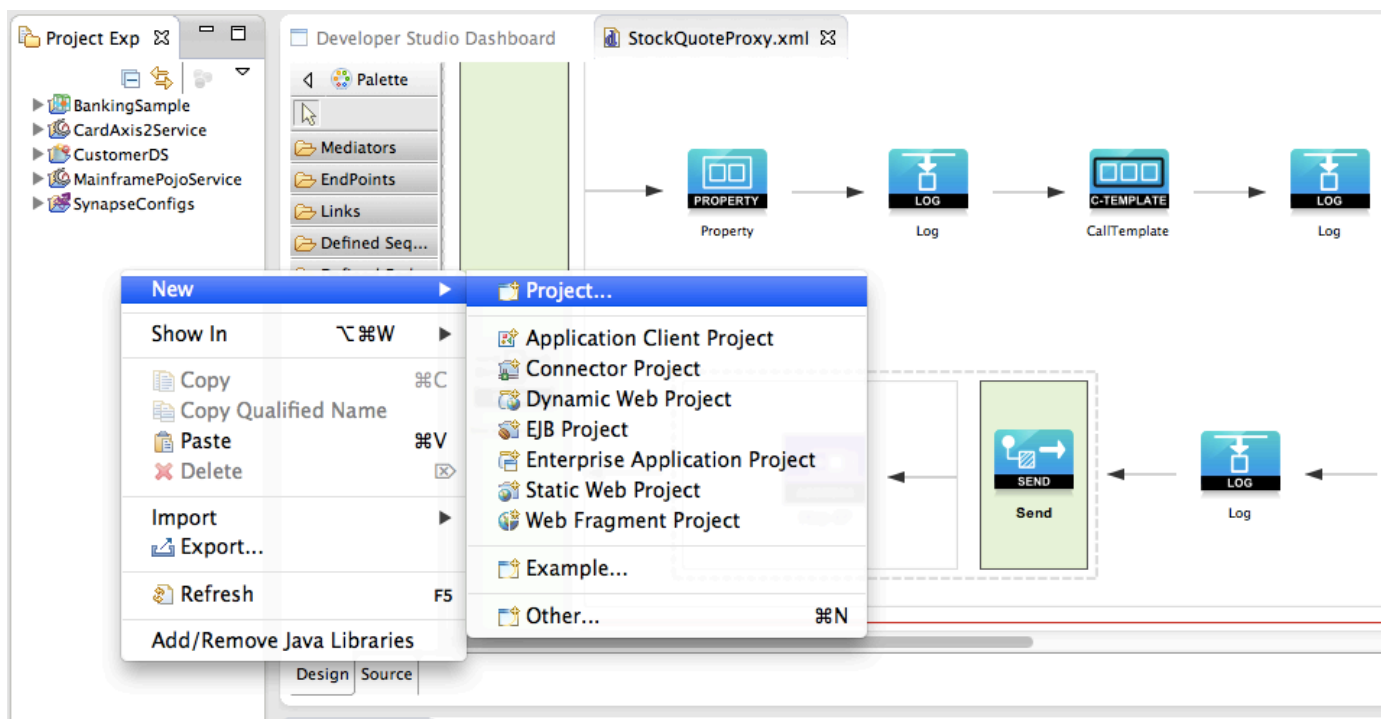
- [Grouping artifacts into a Composite Application Project](#)
- [Creating a Composite Application Archive \(CAR\) file](#)
- [Packaging individual artifacts](#)

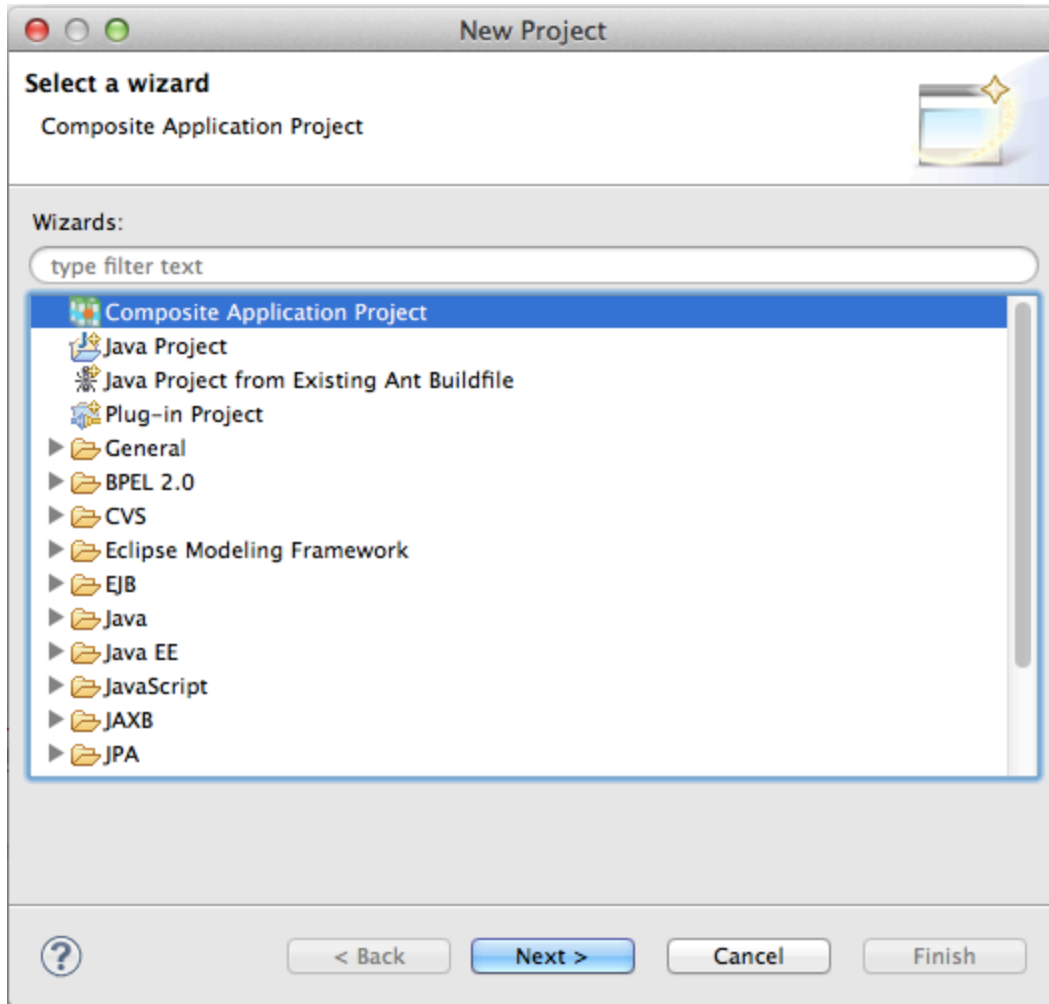
Grouping artifacts into a Composite Application Project

Suppose you have already created all the artifact projects that you need.

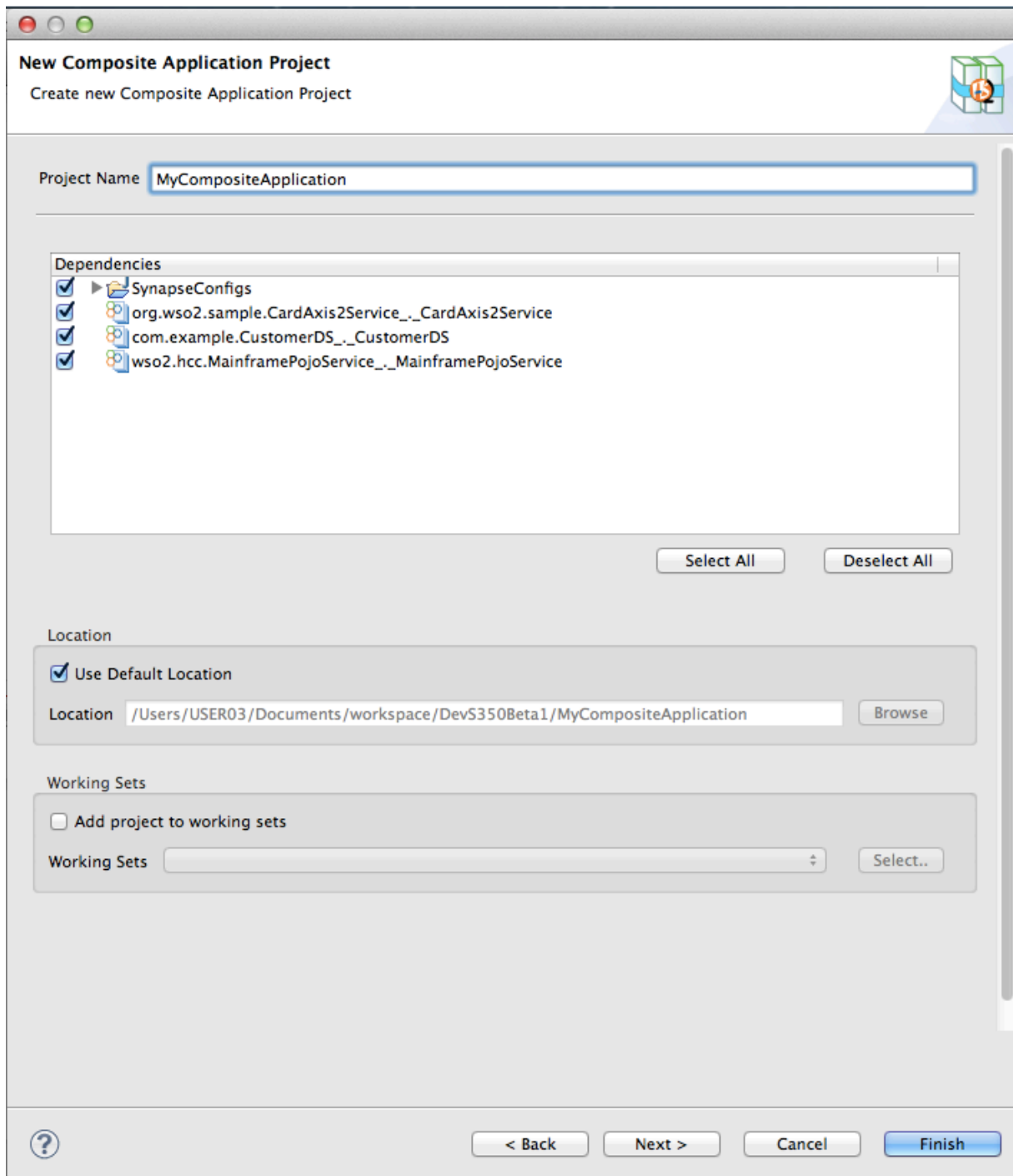


You can now create a Composite Application Project to group these artifacts. To create a Composite Application Project, right-click on the **Project Explorer**, choose **New -> Project** and select **Composite Application Project**.

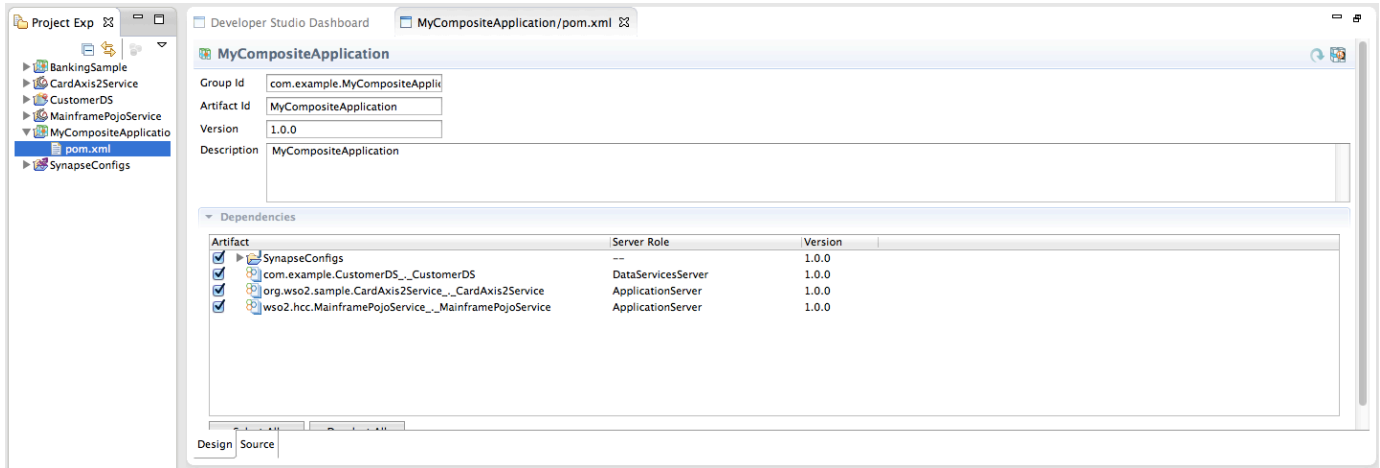




Give a name to the **Composite Application** project. As shown in the following image, it will display all the artifact projects that are available in the workspace. Select the projects that you need to group into your Composite Application Project.



Once you create the Composite Application Project, the **Composite Application Project POM Editor** appears.



Under **Dependencies**, you will see information for each of the projects you selected, such as Server Role and Version. From the editor, you can change those details.

Creating a Composite Application Archive (CAR) file

To deploy a C-App, you first create a Composite Application Archive (CAR) file of that C-App and then deploy the CAR file. This step is not required if you will deploy the C-App [using the Maven CAR Deploy plug-in](#), which generates the CAR file and deploys it in a single step.

You can create the CAR file in one of two ways:

- Right-click the C-App project and select **Export Composite Application Project** from the pop-up menu.
OR
- Open the `pom.xml` file in the Composite Application Project POM Editor and click the button in the upper-right corner for creating an archive.

You then specify the location for the CAR file and the artifacts you want to include in it. After the CAR file is created, you are ready to [deploy it to a Carbon server](#).

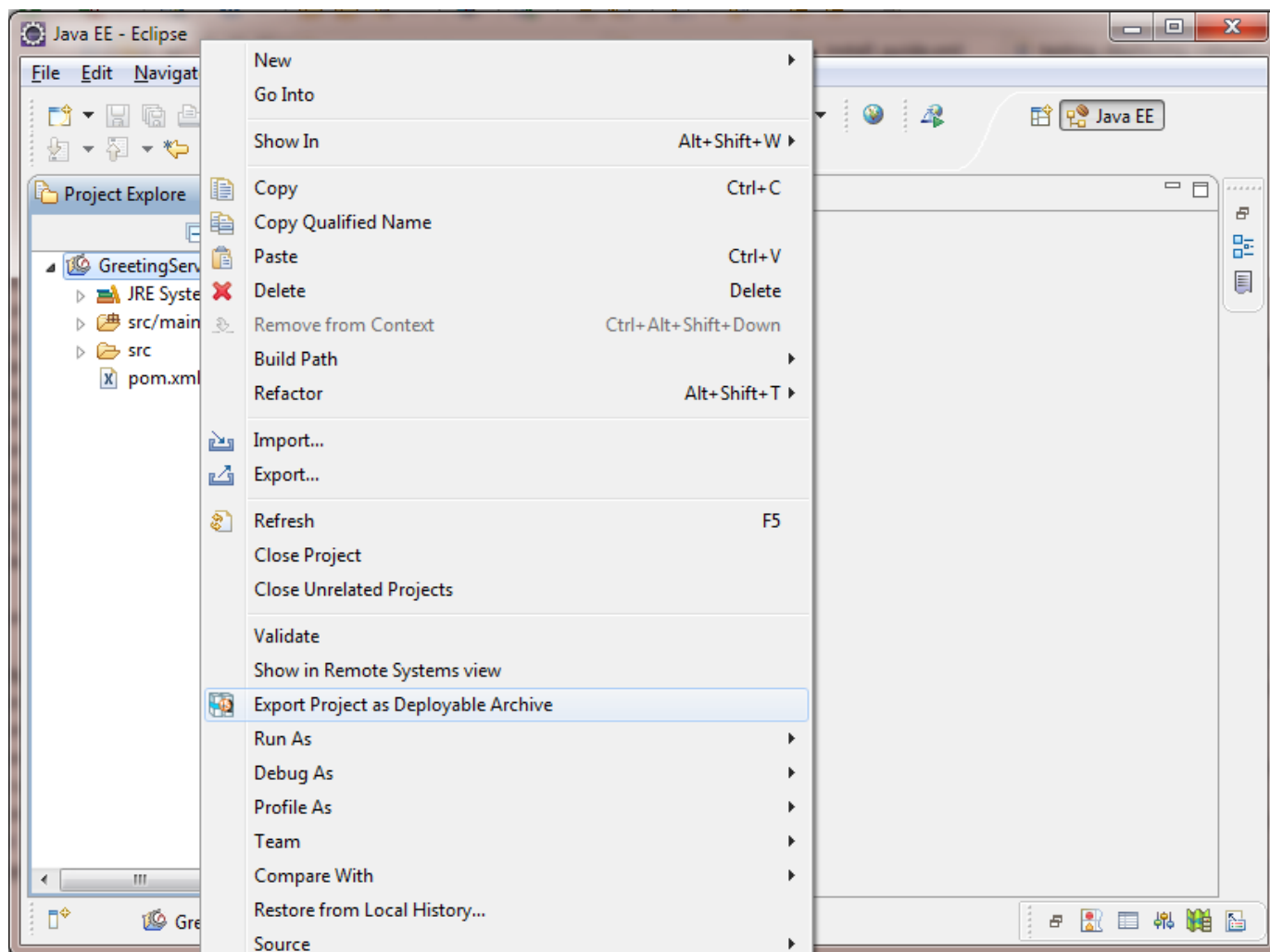
i When you try to export a CAR file that was created in 3.1.0, you may get the following exception in your Eclipse log:

```
org.eclipse.e4.core.di.InjectionException: java.lang.UnsupportedClassVersionError:
org.wso2/developerstudio/eclipse/esb/core/preferences/ui/PreferenceData : Unsupported major.minor
version 51.0
```

Upgrading the JRE version to 1.7.0_04 or later should solve the issue.

Packaging individual artifacts

You can create a separate deployable archive for each artifact. For example, suppose you created an Apache Axis2 Service. When you right-click the Axis2 Service Project, there is an option called **Export Project as Deployable Archive**. This will create the relevant deployable archive in the location you specified, such as an Axis2 Archive (.aar) file for an Apache Axis2 service.



Following are the deployable archives that will be generated for each artifact type.

Artifact Type	Project Type
Apache Axis2 Artifact	Axis2 Archive [aar]
Web Application	Web Application Archive [war]
JAX-WS Artifact	Java Archive [jar]
Endpoint Artifact	xml
Sequence Artifact	xml
Proxy Service Artifact	xml
Local Entry Artifact	xml
Synapse Configuration	xml
ESB Mediator	Java Archive [jar]
Registry Resource	Registry Resource with necessary meta-data
Registry Handler Artifact	OSGI Bundle [jar]
Registry Filter Artifact	OSGI Bundle [jar]

Data Service Artifact	Data Services Descriptor file [dbs]
Data Services Validator Artifact	OSGI Bundle [jar]
BPEL Workflow	BPEL Archive [zip]
Gadget	Dashboard Archive [dar]
Carbon UI Bundle	OSGI Bundle [jar]
Third Party Library Artifact	OSGI Bundle [jar]