How to Create a Spectra CX Build Configuration Plugin

Creating a Plug-in Project

1. **New > Project > Plug-in Development > Plug-in Project**
   - Fill in project name e.g. **com.prismtech.spectracx.target.eorb15-lynxos4-gcc-ppc**
   - Un-check **Create a Java project**
   - Click **Next**
2. For consistency, ensure the **Plug-in ID** field matches the project name. Set
   - **Plug-in Version → <version>.qualifier (i.e. 3.1.2.qualifier),**
   - **Name → eorb15-lynxos4-gcc-ppc Plugin**
   - **Provider → PrismTech**
   - Click **Finish**
3. Select the **Dependencies** tab in the new editor that was opened
   - Click **Add Required Plug-ins**
   - Select **com.ibm.xtools.uml.msl** and click **OK**
4. Now switch to the **Extensions** tab and click **Add**
   - Select **com.ibm.xtools.uml.msl.UMLLibraries** and click **Finish**
   - Fill in the **name**. This is what the user sees when selecting the model from a list in CX
     e.g. **eorb15-lynxos4-gcc-ppc**
   - Fill in the **path**. This is a plugin relative path to the model file
     e.g. **models/eorb15-lynxos4-gcc-ppc.library.uml**

At this point you can start creating your Build Configuration in your new UML Library from scratch or you can copy and modify an existing one, or migrate an old style one.

Migrating an old style Build Configuration

First, you need to get hold of the old model that contained all the old style Build Configuration objects.

1. Install subversive and set up access to the the Zeligsoft SVN repository:
   URL - https://access.zeligsoft.com/svn/Zeligsoft/domains/branches/CX_GA__3_1_0/com.zeligsoft.domain.sca
   User - prismtech_svn_user
   Password - chuspE3p
2. Checkout revision 4559 of the **com.zeligsoft.domain.sca/plugins/com.zeligsoft.domain.sca** plugin project. This revision of this project contains the **models/SCABuild.library.uml** model that contains all the old style Build Configuration objects.
3. Now, in the new plugin project you created for your new Build Configuration, you need to create a new SCA Model. You will copy/build your new Build Configuration objects in this model. You could construct your Build Configuration objects directly in a **uml** file. However the editor/property sheet is difficult to use. Instead, an easier way is to do your modeling in a Rational **emx** file and then export the required **uml** from it.
   - Create a new folder called **models** in your new plugin project.
   - Select the **models** folder and right click **New > Other > Spectra CX > SCA Model**
   - Set the **Model Name** to match the Build Configuration name e.g. **eorb15-lynxos4-gcc-ppc**
   - Set the **Destination folder** to the **models** folder you just created.
   - Click **Finish**
4. Expand **com.prismtech.spectracx.target.eorb15-lynxos4-gcc-ppc/Models/eorb15-lynxos4-gcc-ppc**
   - Delete the **Main**, **modelConfiguration, (CF_IDL), (IDLPrimitives), (SCALibrary) and (UMLPrimitiveTypes)** (all) objects. We don't need them.

5. Right click on the **eorb15-lynxos4-gcc-ppc** model and select **Import Model Library**.
    - Check the **File** option, and browse for the **models/SCABuild.library.uml** file in the
      com.zeligsoft.domain.sca project.
    - Click **OK** and the **SCA_BuildEnvironments** model will be imported into your model.
    - Select & Cut the relevant pair of **SCABuildConfiguration** & **SCAToolchain** objects from
      the imported SCA_BuildEnvironments model.
    - Paste the two objects into your model and save it
    - Delete the **SCA_BuildEnvironments** model import from your model.
6. Select the Model in the Spectra CX Modeling Perspective, open the **Properties view**,
    Stereotypes tab and click **Apply Stereotypes**. Set the Stereotype to modelLibrary
7. Select each object and edit their slot values, using the **Properties View**, as appropriate.
    In particular, set the **target_specific_dir** slot on the **SCAToolchain** instance, to point at the
    directory that contains the code snippet for the build configuration.
    - For example, create a **src** directory in the root of your project
    - Select **eorb15-lynxos4-gcc-ppc_tools and set** the **target_specific_dir** to
      platform:/plugin/com.prismtech.spectracx.target.eorb15-lynxos4-gcc-ppc/src/.
8. Copy Operating Environment specific cpp and h files to the
    com.prismtech.spectracx.target.eorb15-lynxos4-gcc-ppc/src directory
9. When you are happy with all the slot values in your **SCABuildConfiguration** and
    SCAToolchain objects, you can export them to a **uml** file.
    - Select the **com.prismtech.spectracx.target.eorb15-lynxos4-gcc-ppc/Models/eorb15-
      lynxos4-gcc-ppc** project
    - Right click and select **Export > UML 2.1 Model**.
    - Select the **models** folder within your project as the destination directory, and click **Finish**.
10. **Rename the resulting uml file** to match the **path** value that you entered in the UML Library
    extension point above in "Creating a Plug-in Project", step 4. (typically, UML library files have
    a *.library.uml extension).
11. To complete the **uml** file, there is one last step to fix it. In it's current form, it references the
    SCABuildConfiguration and **SCAToolchain** classes in the **SCABuild.library.uml** file in
    your workspace, using a **platform:/resource/....** style URI.
    - To fix this, open the **uml** file in a text editor and substitiute all occurrences of
      "platform:/resource/com.zeligsoft.domain.sca/models/" with "pathmap://SCA_LIBRARIES/".
12. Select elements to include in plugin
    - Open the **MANIFEST.MF**
    - Select the **Build tab**
    - Select **models/eorb15-lynxos4-gcc-ppc.library.uml**
    - Select **src directory**
13. Right click on the project and Export as **Deployable plug-ins and fragments**
14. **Copy** the generated Plug-in (jar file) to the **CX/dropins** folder

Using the Build Configuration in a Model
1. Create a SCA model
2. **Import Model Library**. Select your new build configuration and it should appear in the model.