

Issue

Topic

Model Transformation Framework + Amalthea to Inchron model transformation

Description

1 Setting up of development environment:

1.1 Code checkout

Source code of APP4MC Model Transformation framework along with examples is available on eclipse repository: <https://git.eclipse.org/r/app4mc/org.eclipse.app4mc.tools.git>

→ Use the following command on console to checkout code:

```
Git clone https://git.eclipse.org/r/app4mc/org.eclipse.app4mc.tools.git
```

Branch: **master** is the development branch

→ Use the following command on console to change branch:

```
Git checkout master
```

1.2 Selection of IDE

- It is recommended to use Eclipse APP4MC product for development, as it already contains Core Eclipse, EMF, Xtend 2, Google Guice plugins.

Latest version of APP4MC can be downloaded from:

<https://www.eclipse.org/app4mc/downloads/>

- In a case user wants to use “*custom Eclipse IDE*” or “*expect to develop against different target version of Amalthea model*”, it is required to use the target file available at the below location.

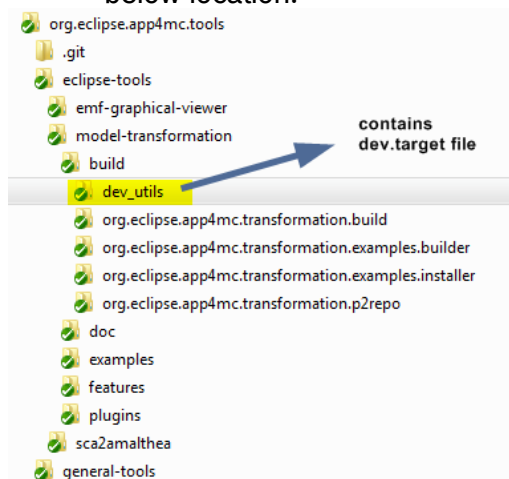


Figure 1 Folder structure of APP4MC tools repo content

Regarding usage of target file and setting up the target for development, have a look at the following [chapter](#)

1.3 Importing sources into APP4MC IDE

Follow the steps shown in the screenshot and import the sources into the workspace.

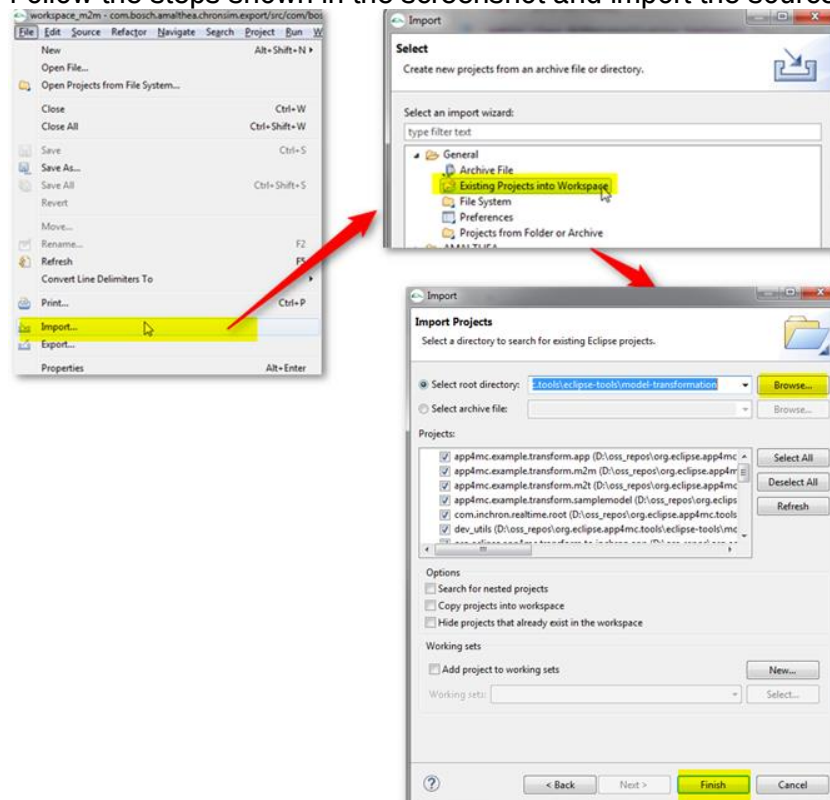


Figure 2 Steps for importing projects into eclipse workspace

1.4 Applying working sets on imported projects

Eclipse Working sets are used to group various projects into custom containers. This is helpful for the developers to organize the projects based on the type.

For organizing projects into working sets, psf file is used.

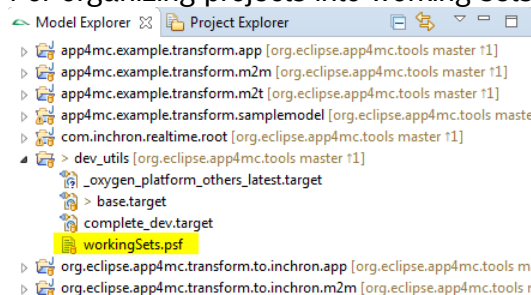


Figure 3 location of working sets psf file

Follow the steps shown in screenshot to use working sets

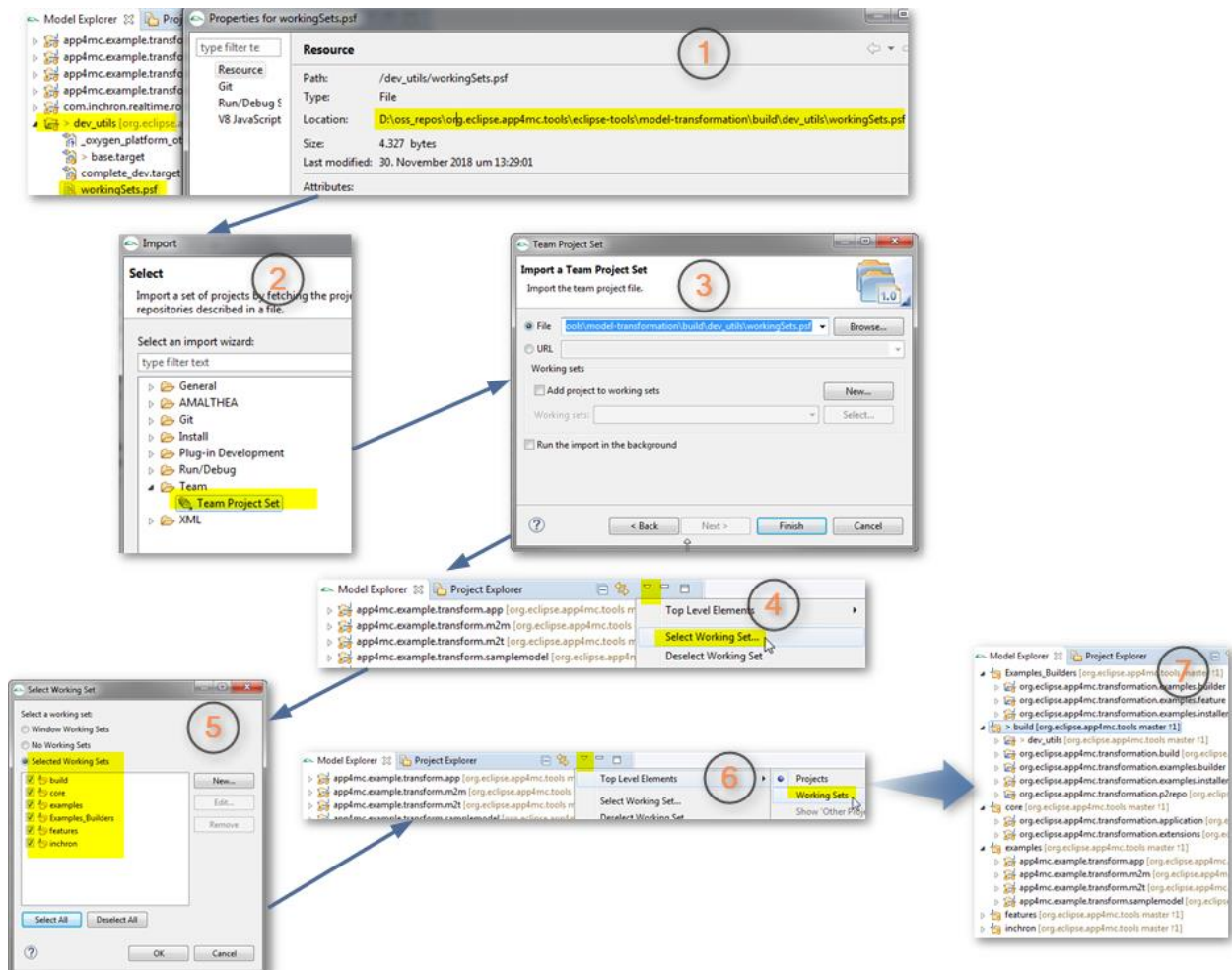


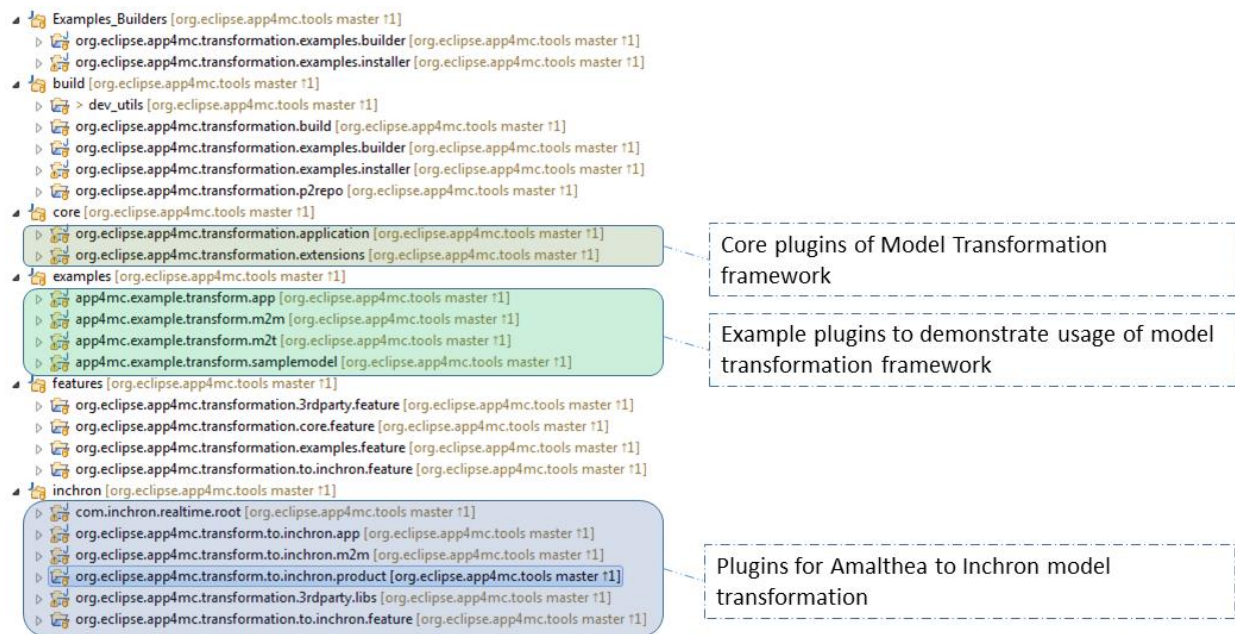
Figure 4 Steps for enabling predefined working sets

2 Use cases included in the current sources

- EMF model transformation framework (“model to model” and “model to text”)
 - o Example plugins for Amalthea to a “Simple Model” transformation (*both m2m and m2t*)
- Amalthea to Inchron model transformation (*productive use case: to convert the Amalthea model data to Inchron model*)

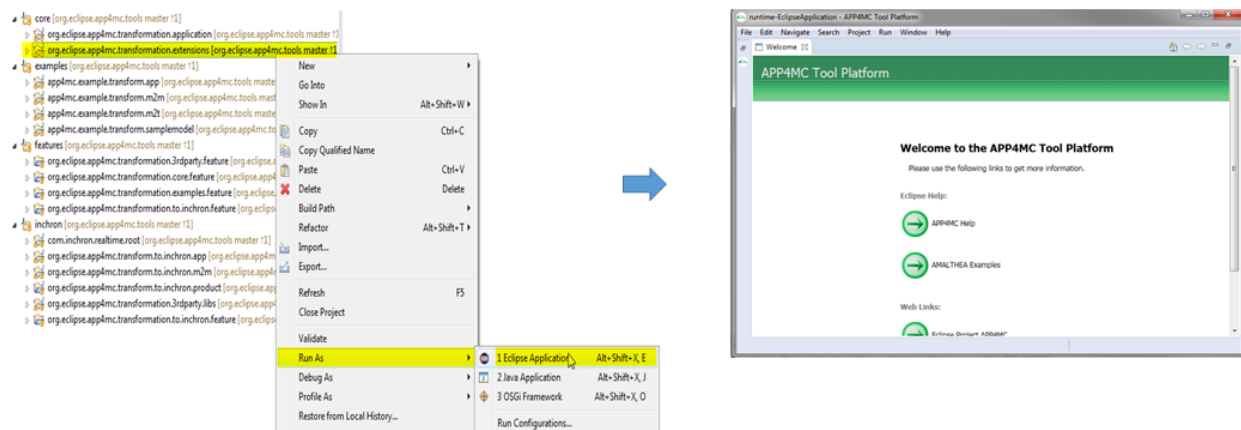
Note: EMF model transformation framework is independent of Amalthea model and it can be used for any model/text transformations

3 Source code structure



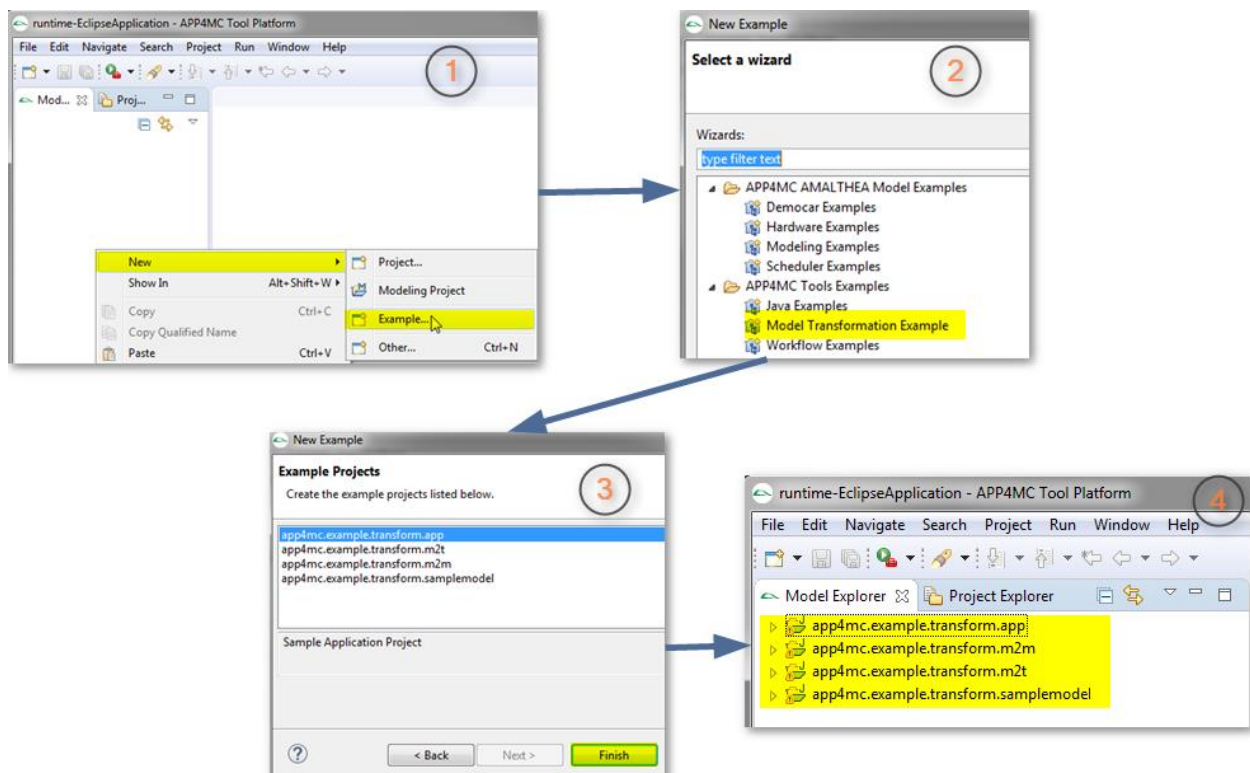
4 Execution of model transformation examples from development environment

4.1 Launch runtime Application



4.2 Create Model Transformation Examples in the runtime environment

Follow the steps as shown in the below screenshot in the runtime environment of eclipse



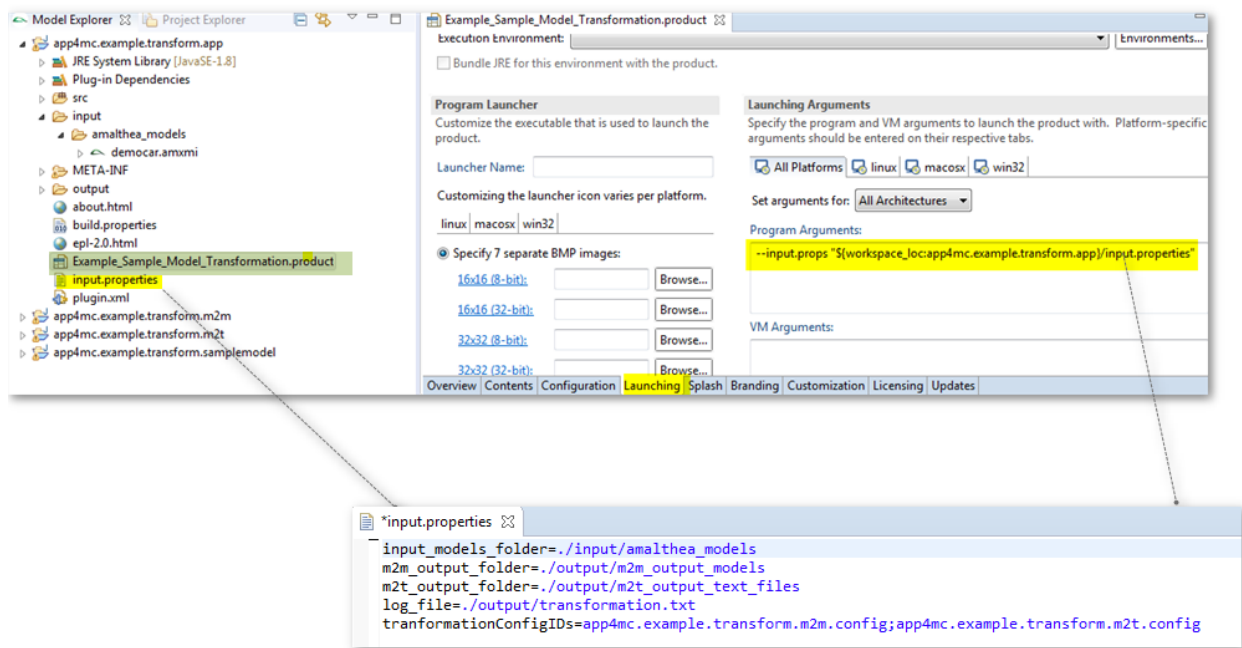
4.3 Input parameters for "Sample model transformation":

Input parameters for execution of transformation are specified in the *Example_Sample_Model_Transformation.product* file present in *app4mc.example.transform.app* plugin.

- ➔ As shown in the below screenshot, Launching tab of the product file contains location of the input properties file.

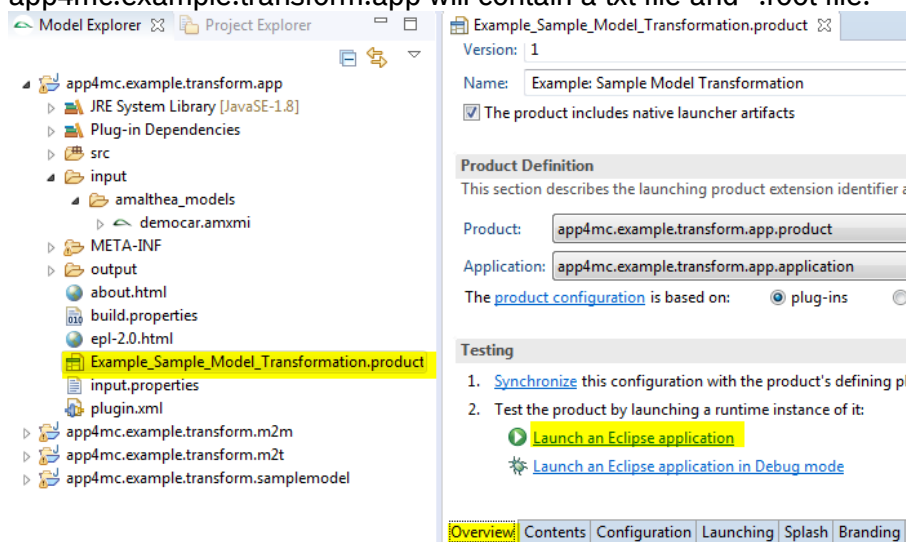
Input properties file should contain below properties in the form of key and value pairs

input_models_folder
m2m_output_folder
m2t_output_folder
log_file
transformationConfigIDs



4.4 Invocation of the model transformation

By clicking on Launch an Eclipse application in the product file (as shown in screenshot), Sample model transformation is invoked. On successful execution, output folder in plugin app4mc.example.transform.app will contain a txt file and *.root file.

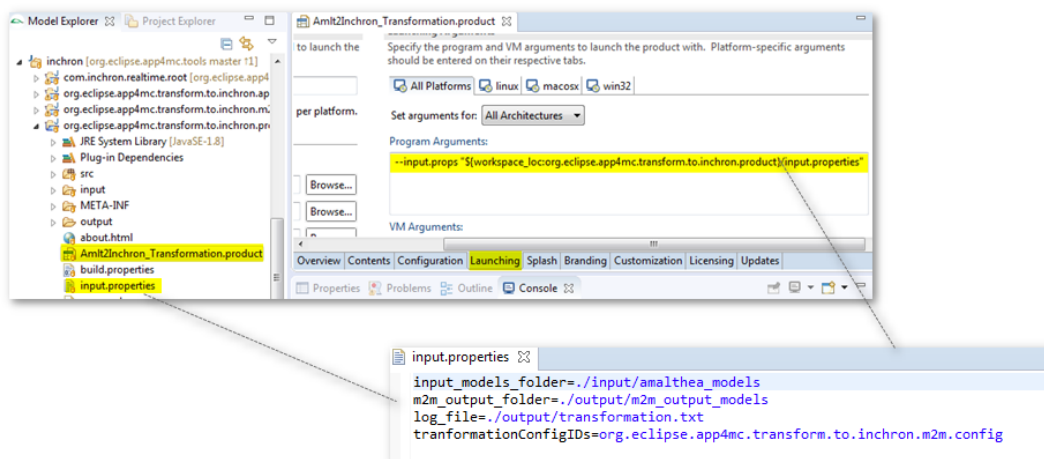


5 Execution of Amalthea to Inchron model transformation

As Amalthea to Inchron model transformation is an individual application which is making use of model transformation framework, it can be executed directly from development workspace without launching runtime application.

5.1 Input parameters to launch the application

Input parameters for execution of model transformation are supplied in file (Amlt2Inchron_Transformation.product) -> launching tab



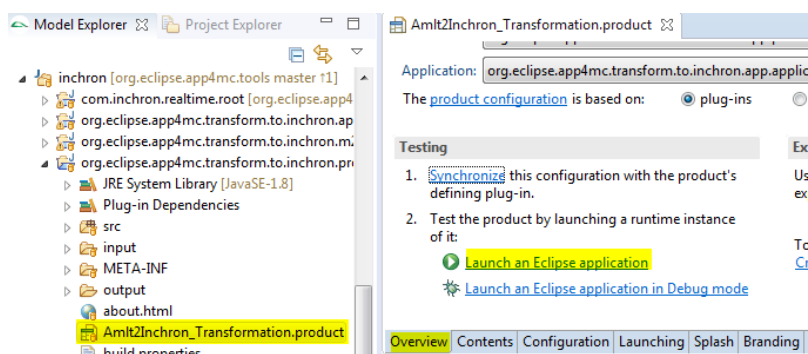
Input properties file should contain below properties in the form of key and value pairs

input_models_folder	Location of Amalthea model files
m2m_output_folder	Location to generate Inchcron model file
log_file	Location of log file of model transformation
transformationConfigIDs	Model transformation extension ID (<i>for Amalthea to Inchcron</i>)

Note: different location of the properties file can be specified in the product file. Also the contents of input.properties can be changed based on the need

5.2 Execution of model transformation

By clicking on Launch an Eclipse application in the product file (as shown in screenshot), Amalthea to Inchcron model transformation is invoked. On successful execution, output folder in plugin org.eclipse.app4mc.transform.to.inchcron.product will contain a txt file and *.root file.



6 Execution from exported product

TBD...

7 Setting runtime platform for model transformation

Target file is used to define the runtime platform. In case runtime platform is not set explicitly, all the plugins available in the dev environment will also be part of the runtime environment.

For Amalthea to Inchron model transformation, as the development is based on fixed Amalthea model version for each release.

- If Amalthea model version required and the APP4MC IDE which is used for development, are of different Amalthea versions -> then it is required to use the dev.target file and specify the required APP4MC release update site

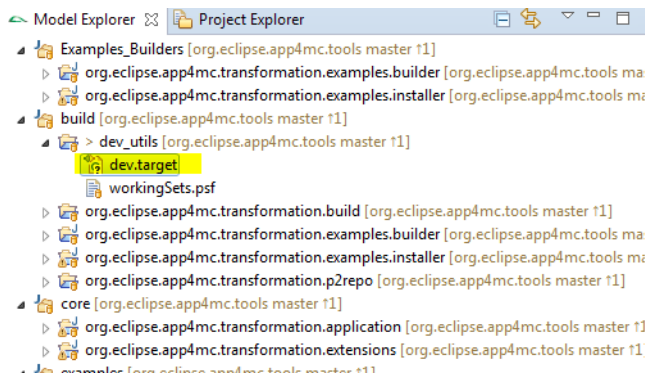


Figure 5 Target file location

Dev.target file present inside dev_utils project, contains the runtime target platform.

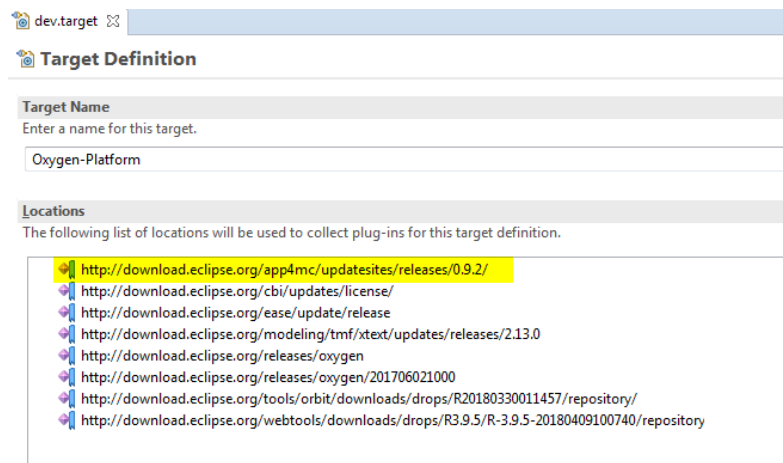
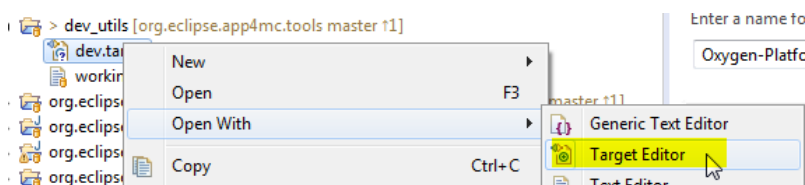


Figure 6 Target file contents

For setting the target platform, open the file with Target editor as shown below and wait till the platform is completely loaded. Once the loading is successful, ensure that there is no message in lower right corner of eclipse, and then press on the **“Set as Active Target Platform”**



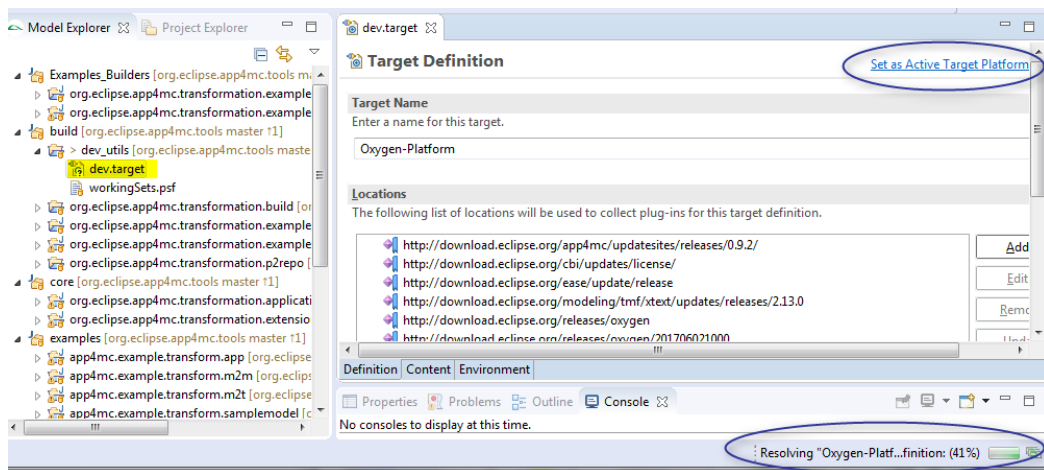


Figure 7 loading of target contents