Embedded Systems Innovation by TNO

POOSL Legacy Support User Manual

Tool version 3.1.0

22-10-2014

POOSL Legacy Support User Manual

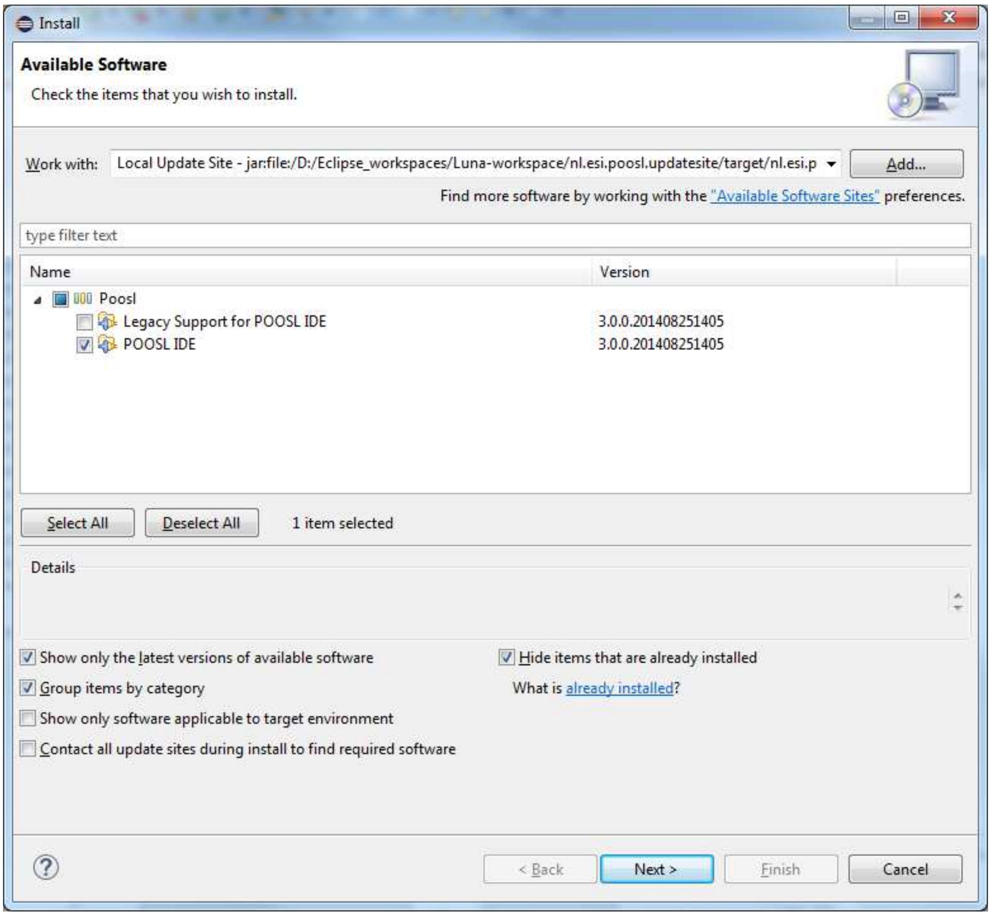
Introduction

This document explains the use and installation of the POOSL Legacy Support, which is an optional part of the POOSL IDE (http://poosl.esi.nl/). This manual describes all features that are needed to import and export POOSL models from and to the SHESim tool.

1 Installation

To install the Legacy Support functionality in the POOSL IDE, follow these steps:

1. Install the POOSL IDE as described in the manual that can be found at: http://poosl.esi.nl/downloads/manuals/UserManual.pdf
2. In Eclipse go to Help -> Install New Software…
3. In the drop-down menu next to "Work with…", select the POOSL repository that was created during the installation of the POOSL IDE.
4. Select the "Legacy Support for POOSL IDE" plug-in from the update site and uncheck the "Contact all update sites during installation to find required software" checkbox.



Available Software

Check the items that you wish to install.

Work with: Local Update Site - jar:file:/D:/Eclipse_workspaces/Luna-workspace/nl.esi.poosl.updatesite/target/nl.esi.p

Find more software by working with the "Available Software Sites" preferences.

type filter text

Name	Version
▪ Poosl	
□ Legacy Support for POOSL IDE	3.0.0.201408251405
☑ POOSL IDE	3.0.0.201408251405

1 item selected

Details

☑ Show only the latest versions of available software

☑ Group items by category

□ Show only software applicable to target environment

□ Contact all update sites during install to find required software

☑ Hide items that are already installed

What is already installed?

5. Click "Next" twice and accept the license agreement, click "Finish" to start the installation. Because the software is not signed by Eclipse it will give a security warning. Click OK on the security warning to continue installation.
6. When asked, restart Eclipse.

2 Export models to SHESim

To export a .poosl model from POOSL IDE to SHESim:

1. In Eclipse, right click on the .poosl file to be exported.
2. In the context menu, click on “Generate Legacy SHESim XML”.
3. In the simulator subfolder, a file is produced that ends with “_shesim.xml”. If this file is not visible, click on the project, and press F5 (Refresh).

The produced XML file can be opened in SHESim. Note: There are some differences in the set of reserved keywords for POOSL IDE and SHESim, which may cause complications.

3 Import legacy models from SHESim

To import a POOSL model from SHESim to POOSL IDE, we need to distinguish two different file formats that have been in use:

- _TL.top file (usually in combination with a _Cl.psl, a _Dt.psl and a _Pr.psl file)
- .xml file

3.1 Import a (legacy) _TL.top model to the new POOSL editor

First transform the _TL.top model into an XML model using SHESim2. SHESim2 can be obtained from: http://www.es.ele.tue.nl/~mgeilen/shesim/

For example, use the version labeled “Stand-alone MS-Windows Non-Commercial version of SHESim v.2”. In SHESim2, perform the following steps:

1. File -> Import Model from Old Format
2. File -> Save Model

Then the model is converted to .xml format. Afterwards proceed in Eclipse as described below for a (legacy) XML model.

3.2 Import a (legacy) .xml model to the new POOSL editor

1. Right click on the project where you want to import the model
2. In the context menu, click on Import…
3. In the list go to POOSL -> Import Legacy XML as Poosl
4. Click on Next
5. Click on Browse… to select the XML file to be imported
6. In the “new file name” box, type an appropriate filename
7. Click Finish

Then the model is converted to .poosl format. The produced .poosl file can be opened in POOSL IDE. Note: There are some differences in the set of reserved keywords for POOSL IDE and SHESim, which may cause complications.

3.3 Solving common problems in legacy models

When you open your imported .poosl model, there are some warnings and errors that are very common.

In the first place, imported .poosl models typically contain all basic data classes, whereas in POOSL IDE these are already implicitly imported. This causes a conflict that is reported as an error “There is another data class with the same name … in BasicClasses.poosl”. The solution is simple: just remove these basic data classes from the imported .poosl model, either manually or using the quick-fix.

POOSL IDE uses version 3.0 of the POOSL basic classes. The documentation of these basic classes can be found at: http://poosl.esi.nl/downloads/manuals/BasicClasses.pdf . The differences with previous versions of the basic classes is described at: http://poosl.esi.nl/downloads/manuals/BasicClasses-Conversion.pdf . In particular, the basic data class Number does not exists anymore, which may lead to the error "Couldn't resolve reference to DataClass 'Number'". This error could typically occur in old basic classes, which should be removed from the file (if present).

In addition, there may be many warnings from the type checker related to numbers. The type Number does not exists anymore and hence Integers and Reals need to be distinguished more explicitly. The value 4 is of type Integer, whereas the value 4.0 is of type Real. Conversions between Integer and Real must be done explicitly, using the methods asInteger and asReal.

4 Reserved Keywords in POOSL IDE

The following keywords cannot be used as the identifier of a class, instance, port or message:

- abort
- and
- channels
- class
- cluster
- currentTime
- data
- delay
- do
- else
- extends
- false
- fi
- if
- init
- instances
- interrupt
- les
- messages
- methods
- native
- new
- nil
- od
- or
- par
- ports
- process
- rap
- return
- sel
- self
- skip
- then
- true
- system
- variables
- while
- with

The following keywords cannot be used as the identifier of a method, parameter or variable:

- abort
- and
- cluster
- currentTime
- data
- delay
- do
- else
- false
- fi
- if
- interrupt
- les
- native
- new
- nil
- od
- or
- par
- process
- rap
- return
- sel
- self
- skip
- then
- true
- system
- while
- with