TWOEAGLES
Short User Guide

Pre-requisites to install TWOEAGLES are:

- Indigo Eclipse environment (https://www.eclipse.org/downloads/packages/eclipse-modeling-tools/indigosr2)
- A TwoTowers tool (http://www.sti.uniurb.it/bernardo/twotowers/)
- TwoEagles plugins (https://svn.codespot.com/a/eclipselabs.org/two-eagles/TwoEaglesPlugins.zip)
- External plugins (https://svn.codespot.com/a/eclipselabs.org/two-eagles/ExternalPlugins.zip)

TWOEAGLES is an Eclipse-based tool. In order to extend TwoTowers, a graphical user interface has been developed as an Eclipse plugin (i.e., the TT_GUI component), so to allow the original tool to be embedded in Eclipse. TT_GUI tailors the Eclipse environment to offer all the original TwoTowers functionalities to users, along with the extensions introduced in TWOEAGLES. Hence, Eclipse becomes the software platform that hosts all the tools interacting within the TWOEAGLES environment. This wrapping of TwoTowers has allowed its implementation to be kept basically unchanged, whereas its interfaces can be invoked, as they are, through the TT_GUI component. Note that TwoTowers still can make use of the NuSMV (http://nusmv.irst.itc.it/) external model checker (not embedded in Eclipse), as in its original release.

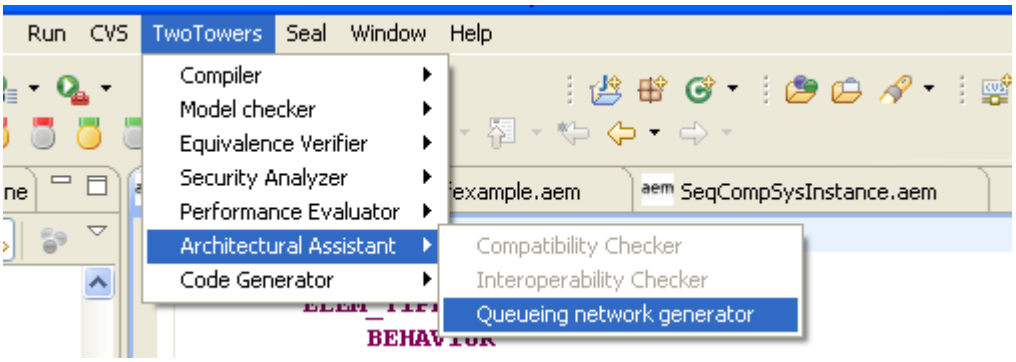
The ÆMILIA_To_QN component performs the (Java-based) transformation of an ÆMILIA specification into a queueing network represented in PMIF (http://dmi.uib.es/~cllado/pmif/), and it has also been implemented as an Eclipse plugin.

In order to install TWOEAGLES all the provided plugins must be copied in the Eclipse plugins folder. Thereafter, through the *Window -> Preferences -> TwoEagles Preferences* menu voice, the whole path of the TTKernel.exe file of TwoTowers must be specified.

A typical scenario of usage can be as follows.

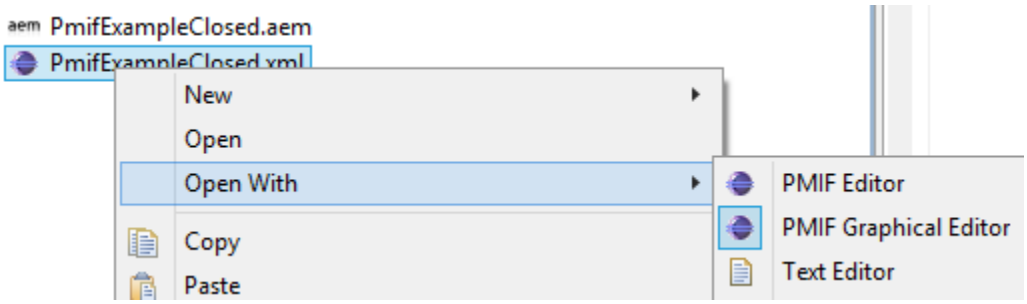
TWOEAGLES starts and TT_GUI, within Eclipse, is ready to accept user commands. The user opens the ÆMILIA editor, which is part of TT_GUI, and enters an ÆMILIA description (i.e. a .aem file). Then the user can run any (functional or performance) analysis technique provided by the original TwoTowers release, including the NuSMV model checker. In addition the user may decide to invoke, as follows, a model transformation that generates a queueing network from the ÆMILIA description.

It is possible to invoke a transformation towards a queueing network through the *TwoTowers -> Architectural Assistant -> Queueing network generator* menu voice, as illustrated in the screenshot here below.



If the ÆMILIA specification is compliant with all the requirements of the underlying methodology, then a PMIF-based file (with the same name as the original file and a .xml extension) will appear in the same directory.

The output queueing network is represented in PMIF and can be rendered in two ways: (i) in a textual XML format through the standard Eclipse XML editor, (ii) in a graphical format through the QN_Editor component. The latter is able to import and export queueing networks in PMIF format and to graphically represent them within Eclipse.



Finally, the user can invoke the QN_Solver, which is a web service able to invoke different existing solvers. The solution results are represented in the standard Eclipse console.