Eclipse plugin Designer for Joram 5

Last modified 19/05/2010 by ScalAgent D.T. release 5.3.1

Contents

Figures and tables

Figures index

Tables index

1. Introduction

This document describes installation and usage of the Joram architecture designer.

2. Installation

2.1. Requirements

The designer plugin needs an Eclipse 3.5 platform with additional plugins:

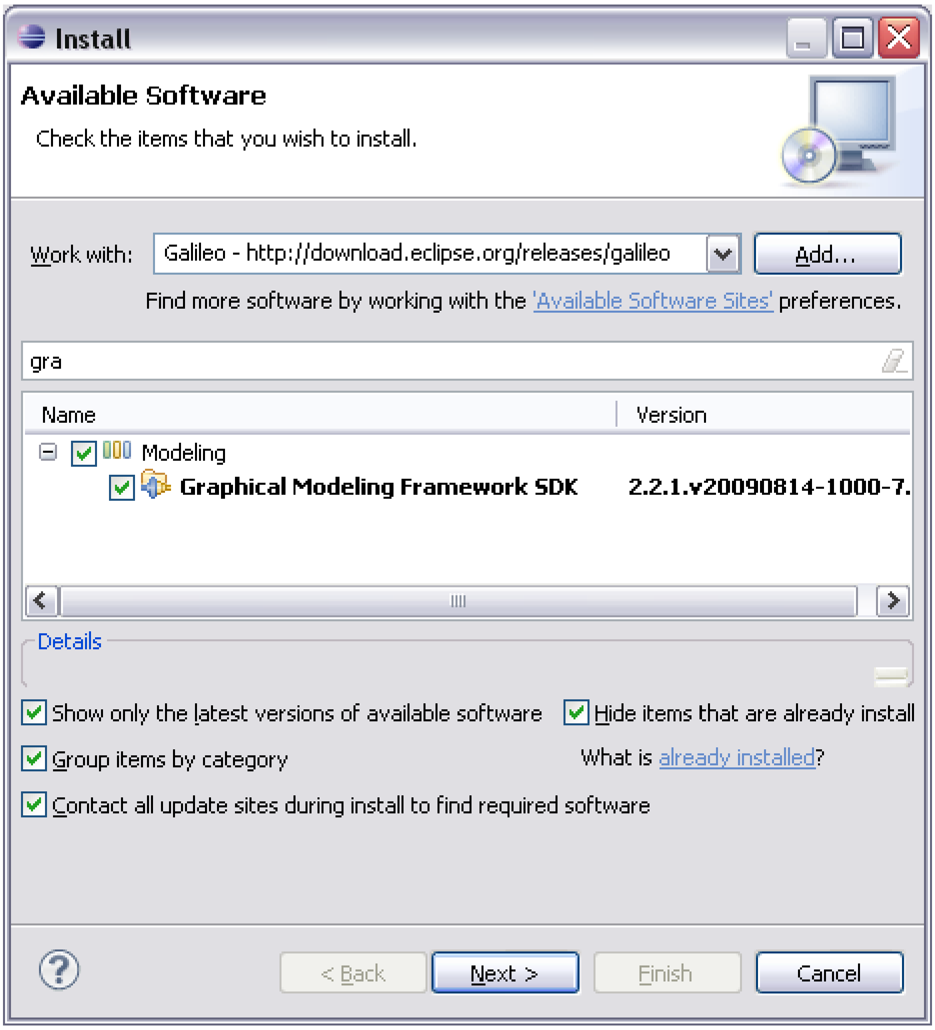
- EMF 2.5.0 (http://www.eclipse.org/modeling/emf/)
- OCL 1.3.0 (http://www.eclipse.org/modeling/mdt/?project=ocl#ocl)
- EMF Validation Framework 1.3.1 (http://www.eclipse.org/modeling/emf/?project=validation)
- GMF 2.2.1 (http://www.eclipse.org/downloads/)

2.2. Installation

2.2.1. Eclipse with modeling tools

Prior to install the plugin you have to obtain an Eclipse with modeling tools, the simplest way if you don’t have Eclipse installed consists to download eclipse modeling tools from «http://www.eclipse.org/downloads/».

If you already have eclipse installed, which will produce a lighter eclipse, run eclipse and install GMF using "Help > Install New Software..." menu (see figure below).



Available Software

Check the items that you wish to install.

Work with: Galileo - http://download.eclipse.org/releases/galileo

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

gra

Name	Version
⊟ ☑ Modeling	
☑ Graphical Modeling Framework SDK	2.2.1.v20090814-1000-7.

Details

☑ Show only the latest versions of available software ☑ Hide items that are already install

☑ Group items by category What is already installed?

☑ Contact all update sites during install to find required software

Figure 1 - Installing GMF

Then download Eclipse « Validation Framework >> Runtime » and « Build dependencies >> OCL » from http://www.eclipse.org/modeling/emf/downloads/?project=validation.

Extract them in your eclipse folder.

2.2.2. Installation of the Joram designer plugin

Download the 5 jar files from http://joram.ow2.org/download/index.html:

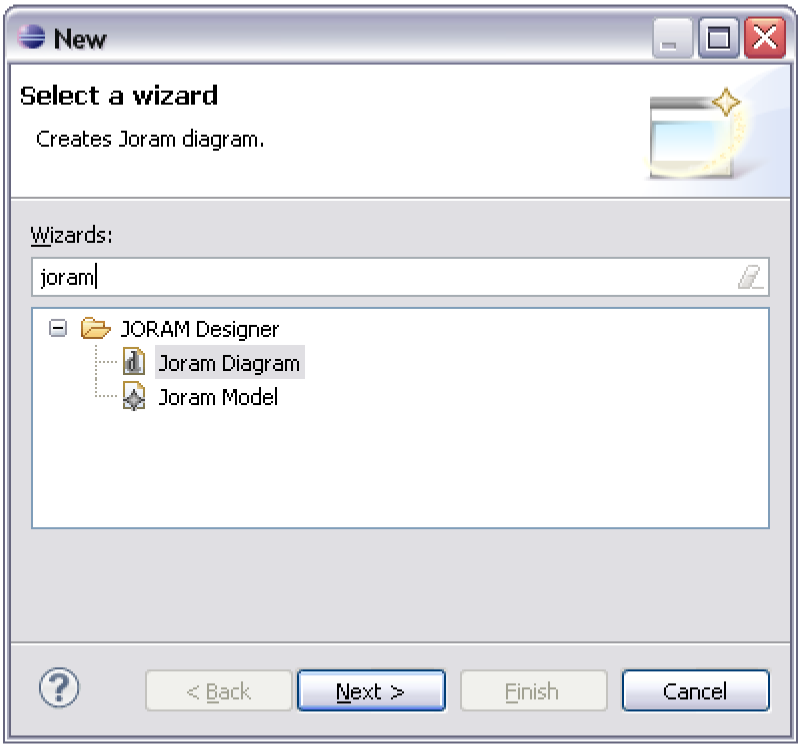
- Follow the « Click here to start the download process link ».
- Search the « JORAM_DESIGNER » package at the end of the page.

Copy the 5 jar files in Eclipse `dropins` folder, then restart Eclipse.

3. Getting started

3.1. Creation of a new configuration

Select « File > New > Other... » menu, then select « JORAM Designer > Joram Diagram » and click the « Next » button.



Select a wizard

Creates Joram diagram.

Wizards: joram

Figure 2: JORAM Designer wizard

Fix the name of the configuration file then click the « Finish » button to create an empty configuration. Using tools in the right palette you can now define « Joram servers », connect them through « Domain «, and locate them on « Host ».

A tutorial is available from http://joram.ow2.org/doc/tutorials.html to guide you to a full configuration as the one below:

- 3 Joram servers interconnected through a PoolNetwork domain named D0.
- 2 servers are located on host « tiga », the last on « aroe ».
- Each servers defined a TCP entry point (TCPProxyService), the server 0 defined a JNDI service.
- Each server defined a user named « anonymous » and 2 destinations are defined on server 0.

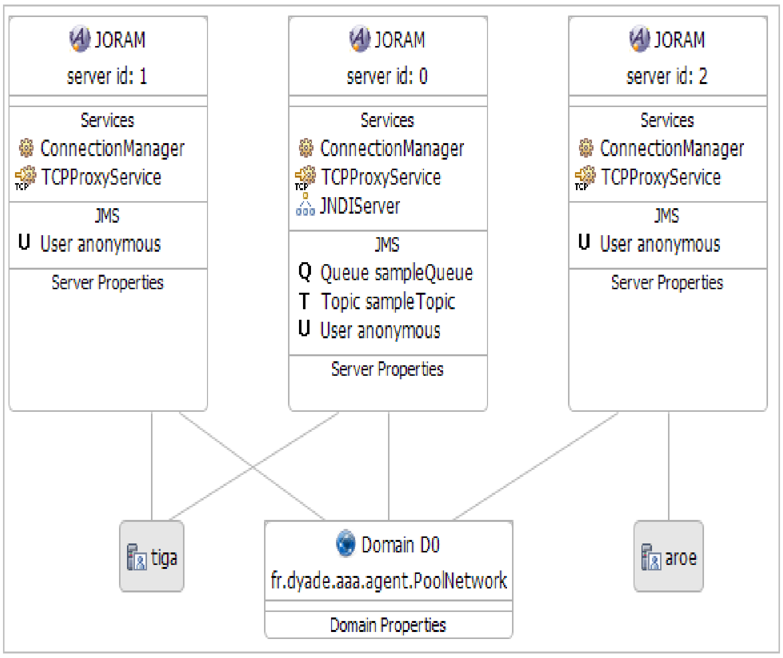
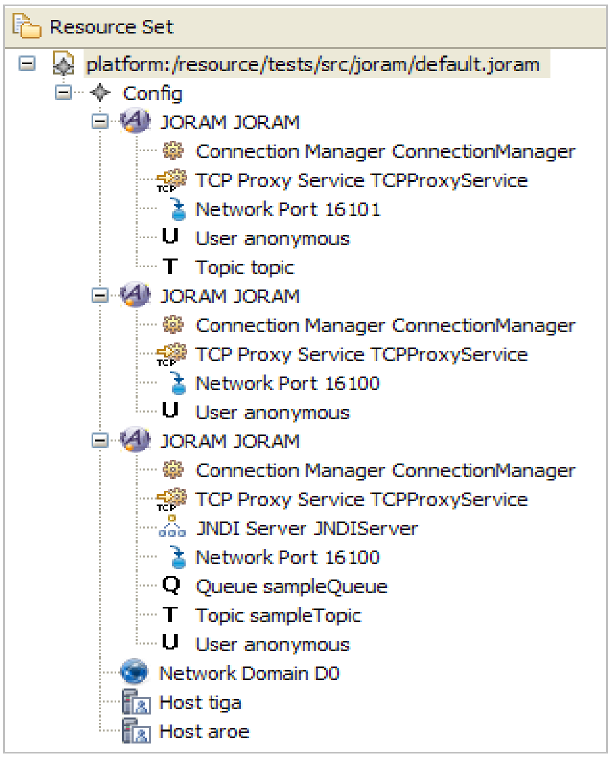


Figure 3: Sample Joram configuration

This configuration can be viewed as a tree opening the corresponding « .joram » file. The figure below shows the previous configuration as a tree.

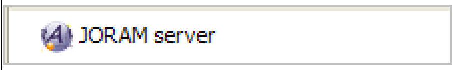
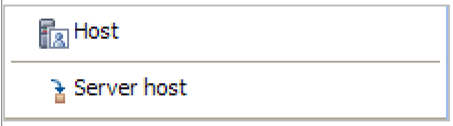
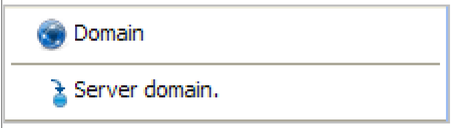


```
Resource Set
platform:/resource/tests/src/joram/default.joram
  Config
    JORAM JORAM
      Connection Manager ConnectionManager
      TCP Proxy Service TCPProxyService
      Network Port 16101
      User anonymous
      Topic topic
    JORAM JORAM
      Connection Manager ConnectionManager
      TCP Proxy Service TCPProxyService
      Network Port 16100
      User anonymous
    JORAM JORAM
      Connection Manager ConnectionManager
      TCP Proxy Service TCPProxyService
      JNDI Server JNDIServer
      Network Port 16100
      Queue sampleQueue
      Topic sampleTopic
      User anonymous
    Network Domain D0
    Host tiga
    Host aroe
```

Figure 4: Configuration tee view

3.2. Configuration elements

This chapter describes the tools in Palette at the right side of the window.

 JORAM server	Allow to create a new Joram server with default configuration: • server id. 0, • a ConnectionMAnager and a TCPProxyService defined, • a JNDIServer, • an user named « anonymous » and 2 destinations, a queue named « sampleQueue » and a topic named « sampleTopic ». Additional elements can be added using configuration elements below.
 Host Server host	« Host » allow to create a new physical host. « Server host » allow to link a Joram server to a physical host.
 Domain Server domain.	« Domain » allow the creation of a new communication domain, default name is « D0 » and protocol is « PoolNetwork ». « Server domain » allow to link a Joram server to a communication domain.
T Topic	Allow the creation of JMS Topic: • default implementation class is used, • internal name is « topic », by default the destination is not registered in JNDI, • nobody is authorized to send or receive messages. These values can be modified using properties (see next chapter).
Q Queue	Allow the creation of JMS Queue: • default implementation class is used, • internal name is « queue », by default the destination is not registered in JNDI, • nobody is authorized to send or receive messages. These values can be modified using properties (see next chapter).
U User	Allow the creation of an user with default name and password set to « anonymous ».

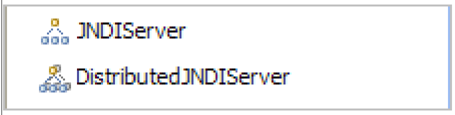
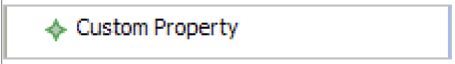
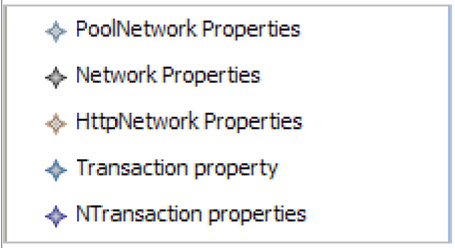
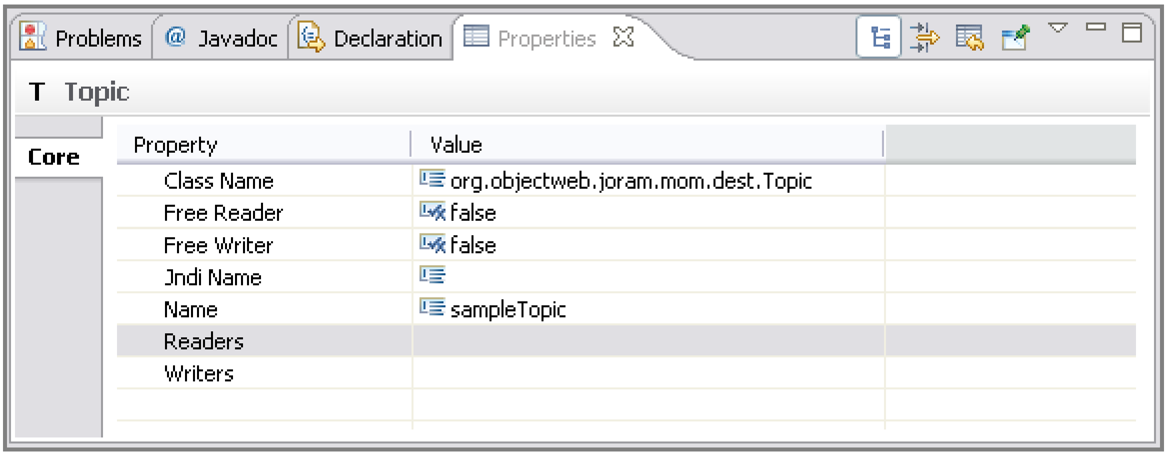
 JNDIServer DistributedJNDIServer	Allow the definition of sample or distributed JNDI service.
Global Properties Set	Allow to define a global set of properties which can be applied to all servers and communication domains.
 Custom Property	Allow to define a custom property with a <name, value> pair.
 PoolNetwork Properties Network Properties HttpNetwork Properties Transaction property NTransaction properties	Allow to add predefined property sets for specific components: • Transaction, NTransaction, • Network, PoolNetwork, HttpNetwork. These sets of properties can be added either at server, domain or global level.

Tableau 1: Elements from right palette

3.3. Properties setting

Many elements need to be configured. The properties of a selected elements can be changed using eclipse properties view. For example for a « Topic » element:



Property	Value
Class Name	org.objectweb.joram.mom.dest.Topic
Free Reader	false
Free Writer	false
Jndi Name	
Name	sampleTopic
Readers	
Writers	

Figure 5: Topics propertties

3.4. Configuration generation

Each time the file is saved your configuration is verified and fault are reported using ⊗ icon, an associated tooltip describes the error: « server id must be unique » for example.

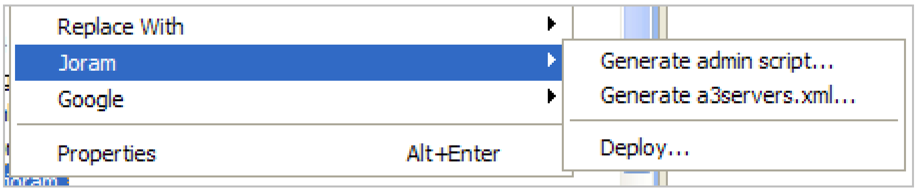
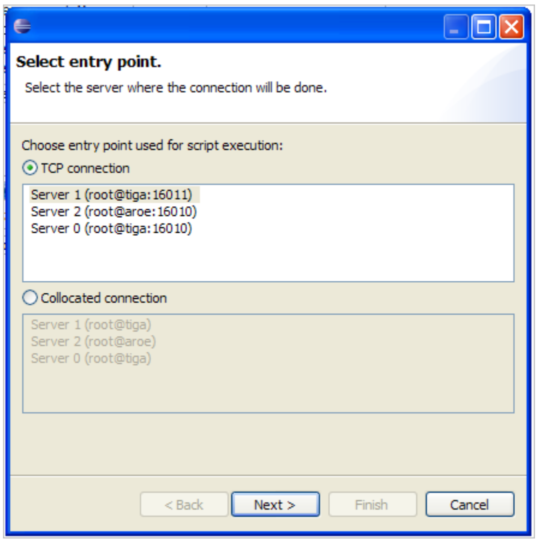
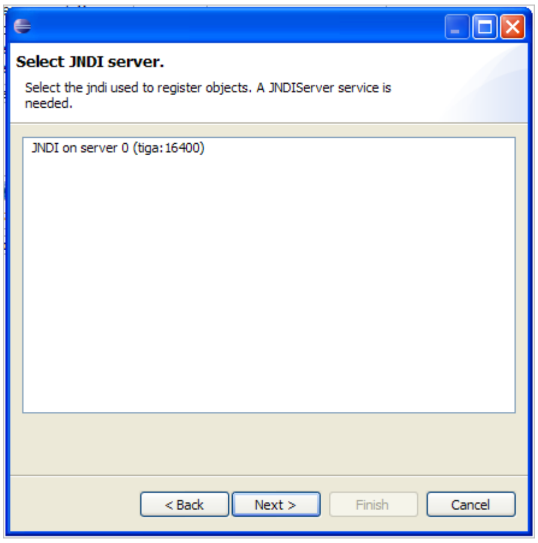


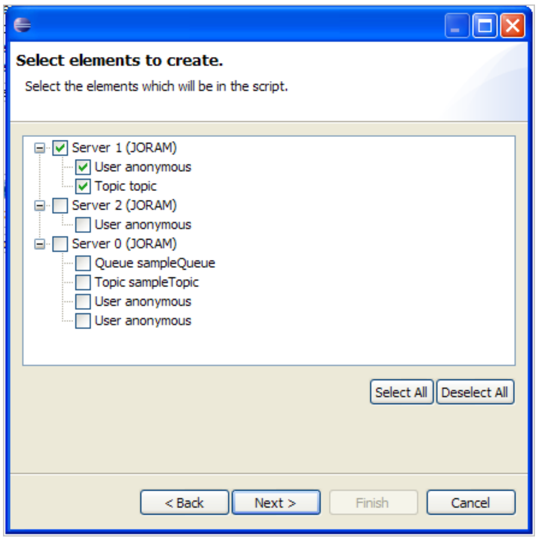
Figure 6: Generation menu

To export your configuration in standard Joram configuration file, right click on your « .joram » file then select the « Joram » sub-menu:

- The « Generate a3servers.xml... » menu allows to produce the corresponding a3servers.xml configuration file.
- The « Generate admin script » menu executes a wizard allowing the generation of XML administration scripts.
- The « Deploy... » menu generates and executes a DeployWare script.

The generation of XML administration script comprises 4 steps:

 **Select entry point.** Select the server where the connection will be done. Choose entry point used for script execution: ● TCP connection Server 1 (root@tiga:16011) Server 2 (root@aroe:16010) Server 0 (root@tiga:16010) ○ Collocated connection Server 1 (root@tiga) Server 2 (root@aroe) Server 0 (root@tiga) 	The first step allows to define the administration entry point, either through TCP or local connection.
 **Select JNDI server.** Select the jndi used to register objects. A JNDIServer service is needed. JNDI on server 0 (tiga:16400) 	The second step allows to choose the JNDI server used to register created destinations.



Select elements to create.

Select the elements which will be in the script.

- ☑ Server 1 (JORAM)
  - ☑ User anonymous
  - ☑ Topic topic
- □ Server 2 (JORAM)
  - □ User anonymous
- □ Server 0 (JORAM)
  - □ Queue sampleQueue
  - □ Topic sampleTopic
  - □ User anonymous
  - □ User anonymous

The third step allows to choose elements to create through this script. For example, all elements can be created from a single script, or elements can be created through different scripts associated to each server.

The last step allows to choose the file containing the created XML administration script.

Tableau 2: Script generation wizard