Grant Agreement 260057

Model-based Analysis & Engineering of Novel Architectures for Dependable Electric Vehicles

Report type	**Deliverable D5.3.1**
Report name	**EAXML Demonstration Platform on Artop/Sphinx.**

Dissemination level	**PU**
Status	**Intermediate**
Version number	**V 1.0**
Date of preparation	**2011-11-28**

Authors

Editor	E-mail
Pascal Gula	pascal.gula@pulse-ar.com

Authors	E-mail
Pascal Gula	pascal.gula@pulse-ar.com
Ali Akar	ali.akar@pulse-ar.com

The Consortium

Volvo Technology Corporation (S) Centro Ricerche Fiat (I)

Continental Automotive (D) Delphi/Mecel (S) 4S Group (I)

MetaCase (Fi) Pulse-AR (Fr) Systemite (SE) CEA LIST (F)

Kungliga Tekniska Högskolan (S) Technische Universität Berlin (D) University of Hull (GB)

Revision chart and history log

Version	Date	Reason
0.9	2011-09-01	Initial version
1.0	2011-11-28	Intermediate version

Table of contents

1 Introduction

1.1 Purpose of document

This document constitutes one part of Deliverable D5.3.1 on Tool adaptations for EAST-ADL. It describes the adaptation of an Eclipse framework, Artop/Sphinx to enable EAST-ADL modeling.

The purpose to have the chosen tools, in MAENAD is to make the EAST-ADL more useable to OEM's and subcontractors in business projects. The purpose is also to do the tools interoperable with the EAST-ADL model using an exchange format, so that analysis performed in the context of the Modeling Analysis Workbench (see WT5.2) will also be possible with the existing tools being adapted in the context of WT5.3

1.2 Introduction to contents of the document

This intermediate release consists of the preliminary development of a demonstrator based on Eclipse/Sphinx framework to be able to model and serialize EAST-ADL model with a common exchange format, therefore enabling tool interoperability.

The document provide a step by step installation process and highlight the first usage of the demonstrator to be able to create/modify an EA model that will be serialized in an EA-XML format.

1.3 Abbreviations

EA	EAST-ADL
TP	Target Platform

1.4 Software Requirement

Name	Version	Note
Java	1.6	
TortoiseSVN	1.6.16	
Eclipse	3.6	
Subversive (Eclipse plug-in)	0.7	
EMF	2.6.1	
EMF Compare	1.1.0	
EMF Query	1.4.0	
EMF Transaction	1.4.0	
EMF Validation	1.4.0	
GEF	3.6.2	
GMF Runtime	1.4.2	
MDT-UML2	3.1.2	
MDT-OCL	3.0.2	

M2M-QVT	3.0.1	
Orbit	R20110523182458	
EMFT-MWE	2.0.1	
EMFT-MWE2	2.0.1	
EMFT-MWE2-Lang	2.0.1	
M2T-Xpand	1.1.0	
Sphinx	0.7.0	

2 Setup Environment

2.1 Install Java

The latest version of Java at this time is Java SDK 1.6 update 27, you can download it via link below:

http://www.oracle.com/technetwork/java/javase/downloads/index.html

The process to install it is easy on Windows platform: you will have **jdk-6u27-windows-i586.exe** after downloading and run it to install Java 1.6. If you already have Java 1.5 or 1.6 version on your computer, this step can be skipped.

2.2 Install Eclipse

You need to install Eclipse as the actual Java IDE. Of course, this can be skipped if you already have recent Eclipse version installed (recommend to have Eclipse 3.6 version because all others plug-ins below are Eclipse 3.6 compatible).

You can download Eclipse 3.6 via link below:

http://www.eclipse.org/downloads/ (prefer to use Eclipse Classic version)

The same to install Java, you'll have **eclipse-SDK-3.6-win32.zip** after downloading and just extract it to have Eclipse Classic 3.6.

Further information on downloading and installing Eclipse can be found in the Eclipse FAQ and in the Eclipse Readme. A general introduction to Eclipse as a Java IDE can be found in this article.

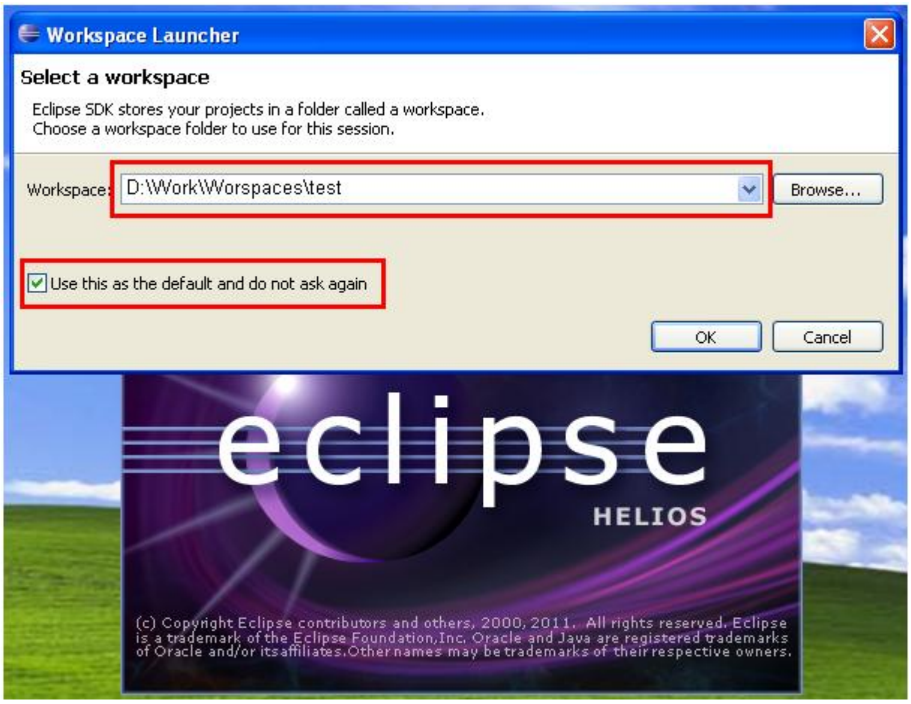
Note: Please pay attention when you define the path to extract eclipse, it should be a short path under windows, for ex: C:\Work\Eclipses

2.3 Install Subversive

The next step is to install a Subversion client plug-in into your Eclipse installation, since such a plug-in does not come with Eclipse out-of-the-box. Subversion is used as the version control system for all source code and other development related resources within the Eclipse community.

Currently two different Subversion client implementations exist for Eclipse: Subversive and Subclipse. It is hard to recommend one of the implementations since both have been around for a while. The authors impression is that Subversive is more feature rich but not as stable as Subclipse. As Manaed project support Subversive only, we'll explore the way to install it below.

Run Eclipse (for ex., my case is **C:\Work\Eclipse\eclipse.exe**), the first time Eclipse run will ask to define workspace's location, please find a short relative path, such as **D:\Work\Workspaces\test1**



Workspace Launcher

Select a workspace

Eclipse SDK stores your projects in a folder called a workspace.
Choose a workspace folder to use for this session.

Workspace: D:\Work\Worspaces\test

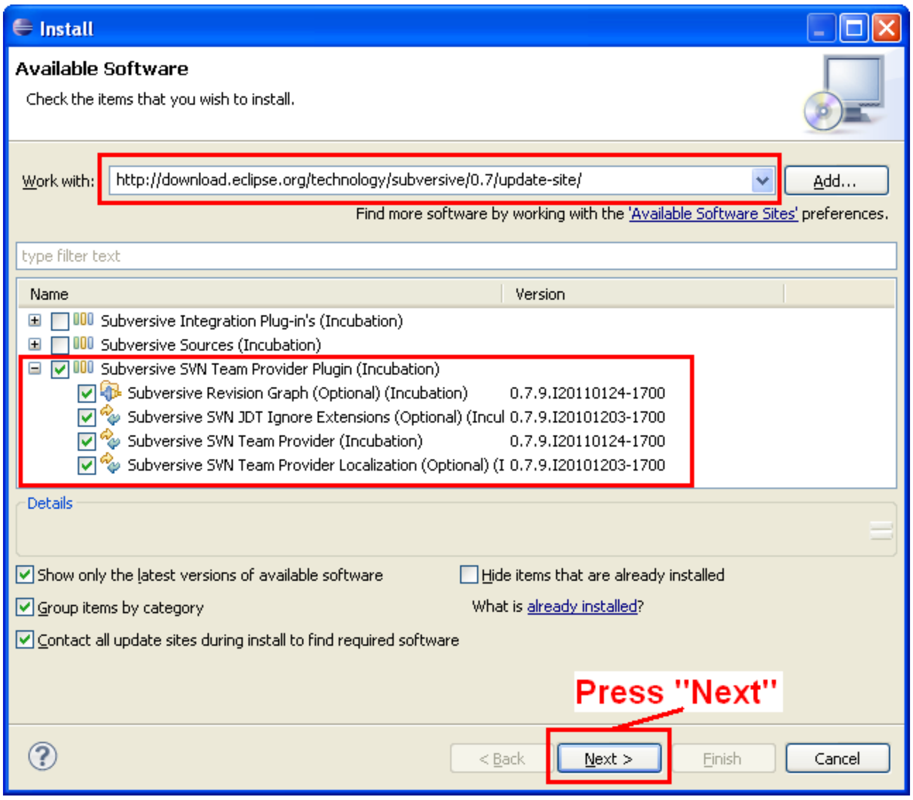
☑ Use this as the default and do not ask again

Browse...
OK
Cancel
eclipse
HELIOS
(c) Copyright Eclipse contributors and others, 2000, 2011. All rights reserved. Eclipse is a trademark of the Eclipse Foundation,Inc. Oracle and Java are registered trademarks of Oracle and/or its affiliates.Other names may be trademarks of their respective owners.

Figure1: Define Eclipse workspace's location

When Eclipse open. Run Help → Install New Software …. Add the update site below and install feature group named "***Subversive SVN Team Provider Plugin (Incubation)***"

http://download.eclipse.org/technology/subversive/0.7/update-site/



Install

Available Software

Check the items that you wish to install.

Work with: http://download.eclipse.org/technology/subversive/0.7/update-site/

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

type filter text

Name	Version
⊞ □ Subversive Integration Plug-in's (Incubation)	
⊞ □ Subversive Sources (Incubation)	
⊟ ☑ Subversive SVN Team Provider Plugin (Incubation)	
☑ Subversive Revision Graph (Optional) (Incubation)	0.7.9.I20110124-1700
☑ Subversive SVN JDT Ignore Extensions (Optional) (Incul	0.7.9.I20101203-1700
☑ Subversive SVN Team Provider (Incubation)	0.7.9.I20110124-1700
☑ Subversive SVN Team Provider Localization (Optional) (I	0.7.9.I20101203-1700

Details

☑ Show only the latest versions of available software

□ Hide items that are already installed

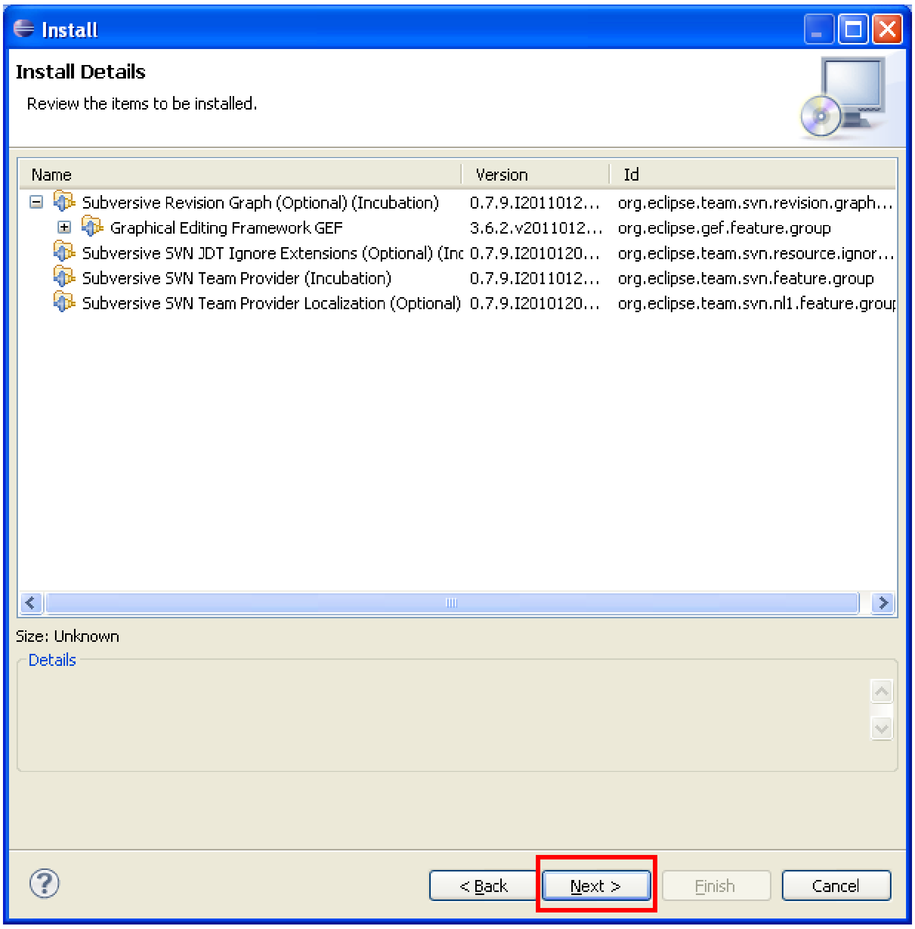
☑ Group items by category

What is already installed?

☑ Contact all update sites during install to find required software

Press "Next"

Figure 2: Install Eclipse Subversive (1)



Install Details

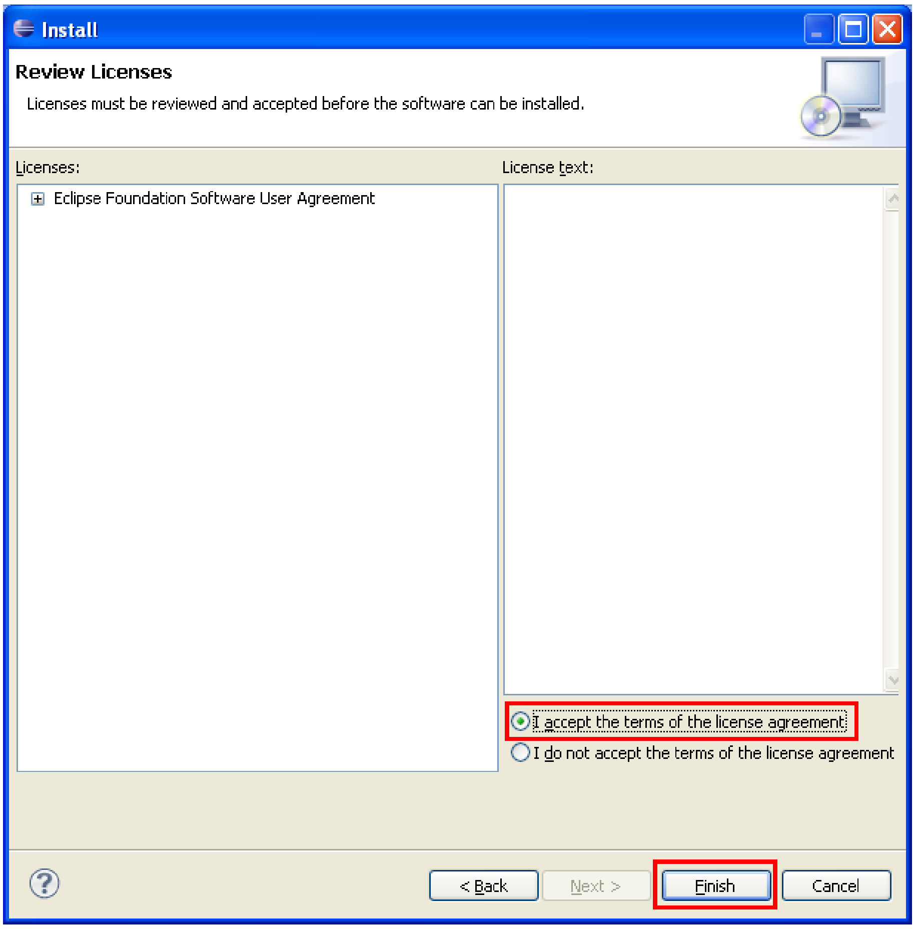
Review the items to be installed.

Name	Version	Id
Subversive Revision Graph (Optional) (Incubation)	0.7.9.I2011012...	org.eclipse.team.svn.revision.graph...
Graphical Editing Framework GEF	3.6.2.v2011012...	org.eclipse.gef.feature.group
Subversive SVN JDT Ignore Extensions (Optional) (Inc	0.7.9.I2010120...	org.eclipse.team.svn.resource.ignor...
Subversive SVN Team Provider (Incubation)	0.7.9.I2011012...	org.eclipse.team.svn.feature.group
Subversive SVN Team Provider Localization (Optional)	0.7.9.I2010120...	org.eclipse.team.svn.nl1.feature.grou

Size: Unknown

Details

Figure 3: Install Eclipse Subversive (2)



Review Licenses

Licenses must be reviewed and accepted before the software can be installed.

Licenses:

⊞ Eclipse Foundation Software User Agreement

License text:

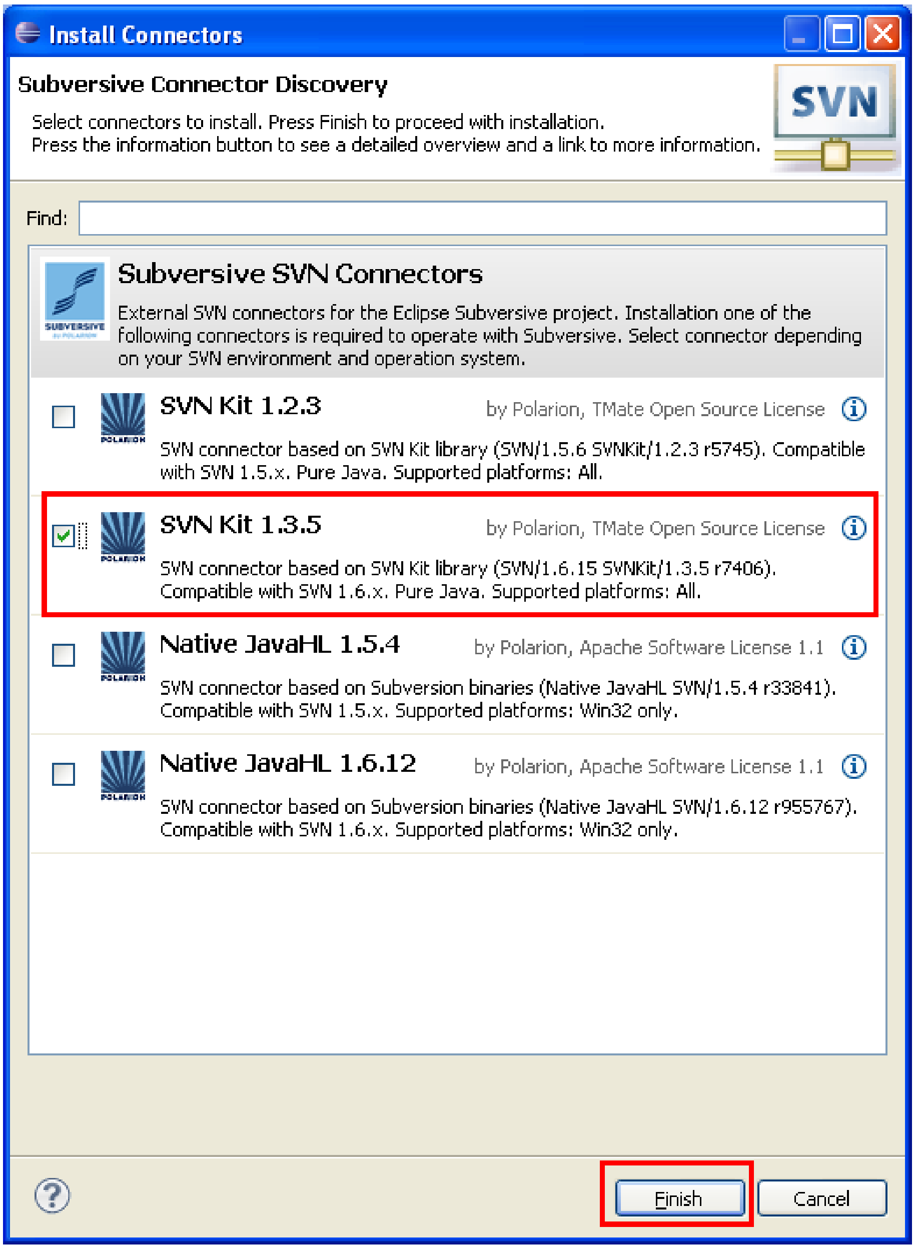
● I accept the terms of the license agreement

○ I do not accept the terms of the license agreement

Figure 4: Install Eclipse Subversive (3)

After Subversive is installed successfully, please choose **Restart Now** to restart Eclipse. Then execute the following actions:

Open Window → Open Perspective → Other …, select SVN Repository Exploring, and Eclipse will ask to install SVN Connector. You can select “SVN Kit 1.3.5” or “Native JavaHL 1.6.12”



Subversive Connector Discovery

Select connectors to install. Press Finish to proceed with installation.
Press the information button to see a detailed overview and a link to more information.

Find:

Subversive SVN Connectors

External SVN connectors for the Eclipse Subversive project. Installation one of the following connectors is required to operate with Subversive. Select connector depending on your SVN environment and operation system.

□ **SVN Kit 1.2.3** by Polarion, TMate Open Source License

SVN connector based on SVN Kit library (SVN/1.5.6 SVNKit/1.2.3 r5745). Compatible with SVN 1.5.x. Pure Java. Supported platforms: All.

☑ **SVN Kit 1.3.5** by Polarion, TMate Open Source License

SVN connector based on SVN Kit library (SVN/1.6.15 SVNKit/1.3.5 r7406). Compatible with SVN 1.6.x. Pure Java. Supported platforms: All.

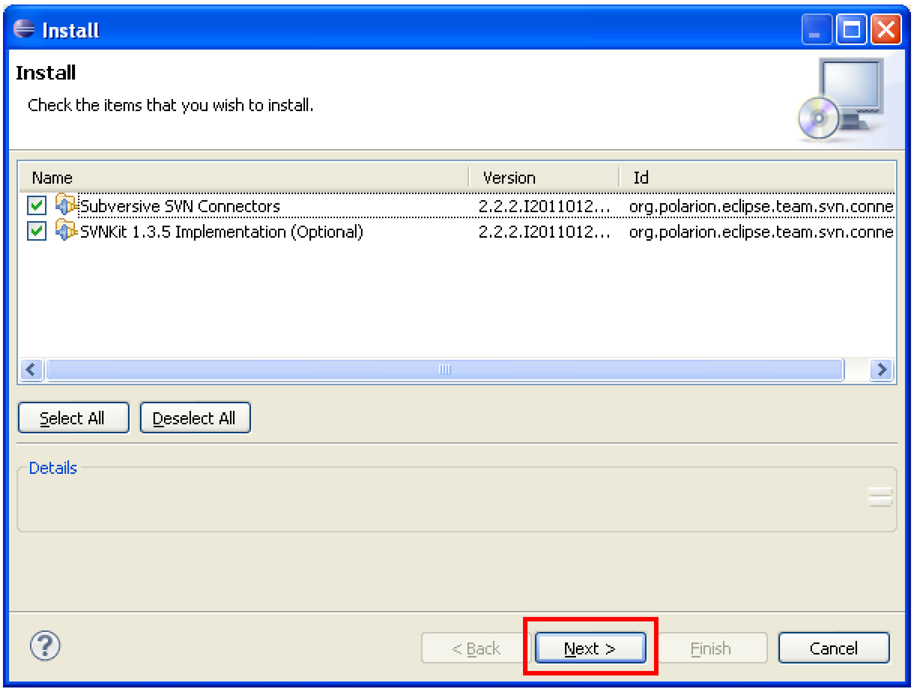
□ **Native JavaHL 1.5.4** by Polarion, Apache Software License 1.1

SVN connector based on Subversion binaries (Native JavaHL SVN/1.5.4 r33841). Compatible with SVN 1.5.x. Supported platforms: Win32 only.

□ **Native JavaHL 1.6.12** by Polarion, Apache Software License 1.1

SVN connector based on Subversion binaries (Native JavaHL SVN/1.6.12 r955767). Compatible with SVN 1.6.x. Supported platforms: Win32 only.

Figure 5: Install Subversive Connector (1)



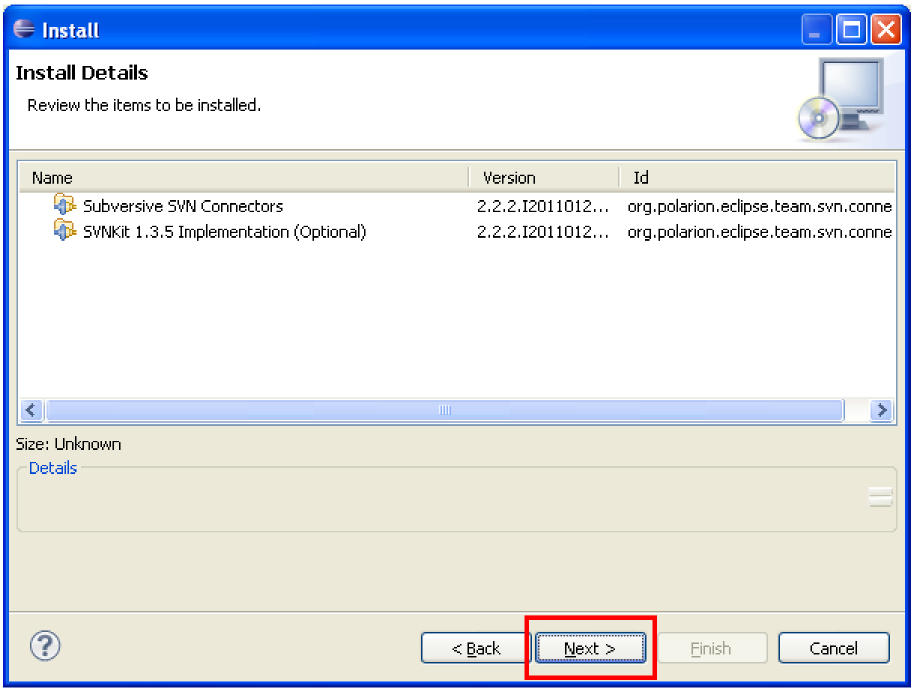
Install

Install

Check the items that you wish to install.

Name	Version	Id
☑ Subversive SVN Connectors	2.2.2.I2011012...	org.polarion.eclipse.team.svn.conne
☑ SVNKit 1.3.5 Implementation (Optional)	2.2.2.I2011012...	org.polarion.eclipse.team.svn.conne

Figure 6: Install Subversive Connector (2)



Install

Install Details

Review the items to be installed.

Name	Version	Id
Subversive SVN Connectors	2.2.2.I2011012...	org.polarion.eclipse.team.svn.conne
SVNKit 1.3.5 Implementation (Optional)	2.2.2.I2011012...	org.polarion.eclipse.team.svn.conne

Size: Unknown

Figure 7: Install Subversive Connector (3)

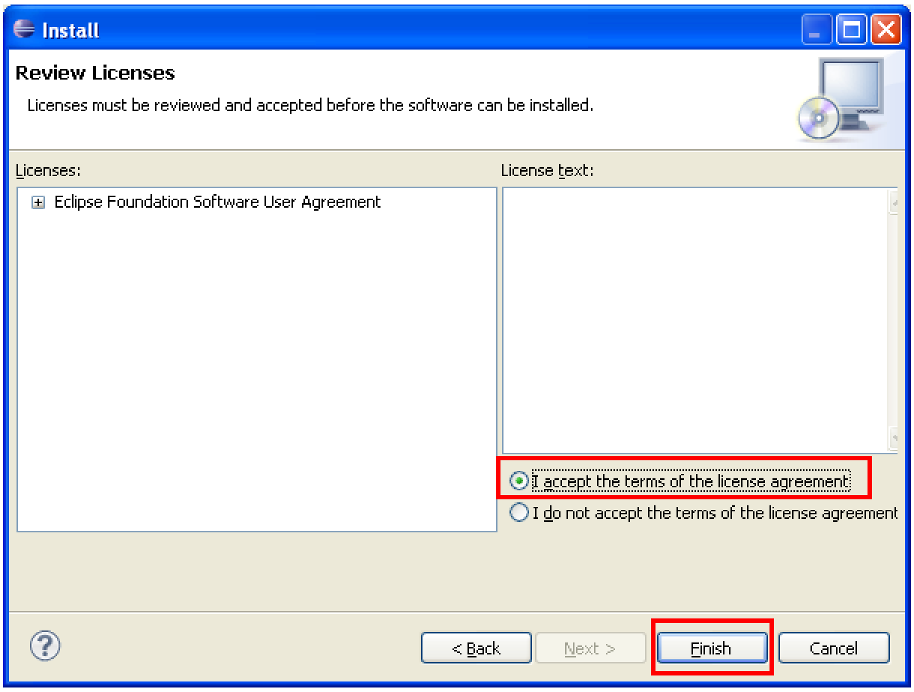

Install
Review Licenses
Licenses must be reviewed and accepted before the software can be installed.
Licenses:
⊞ Eclipse Foundation Software User Agreement
License text:
● I accept the terms of the license agreement
○ I do not accept the terms of the license agreement
?
< Back
Next >
Finish
Cancel

Figure 8: Install Subversive Connector (4)

Once Subversive Connector is installed successfully, Eclipse need to be restarted again.

2.3.1 Setup Target Platform

The development of our EAXML Demonstrator is built basically on Sphinx. Sphinx requires a specific target platform which is a set of libraries (Eclipse plug-ins) and our plug-ins are compiled against it. It is highly recommended to set an explicit target platform because otherwise the plug-ins are compiled against the development environment itself in a self-hosted way which can have unwanted side-effects.

We provide a formally defined target platform. The target platform itself in the current version is available on the Maenad repository located at

https://www-maenad.cea.fr/svn/WP5/eaxml-demonstrator/com.pulsear.targetdefs

To configure the target platform in Eclipse also a target definition file is needed. The corresponding target definition file for the above target platform can be obtained here.

Currently the target platform is based on Eclipse version 3.6, and mainly contains the following extras: (see 1.4 Software Requirement)

- EMF (including XSD, Validation, Transaction and Compare)
- GMF
- GEF
- ...

2.3.2 Add SVN Repository

Maenad projects use SVN repository. In order to check out and manage source code, the following repository needs to be added as 'SVN Repository' in development environment.

- Open *SVN Repository Exploring* perspective;
- In *SVN Repositories* view, select "New > Repository Location" context item;
- The "New Repository Location" wizard dialog opens;
- Fill fields with the following elements:
    - URL: https://www-maenad.cea.fr/svn/edesigner
    - User: your SVN user name; Password: your SVN password;
    - Select 'Save authentification' to avoid setting password at each action;
- Click on 'Finish' button to complete location creation.

2.3.3 Check Out Target Folder

Navigate the already created svn repository in 2.4.1 and go to WP5/eaxml-demonstrator/targets and select target 3.6; right click and then export it to the local file system.

Instead of exporting the target from whithin Eclipse, you can install a svn client for windows here: http://tortoisesvn.net/downloads.html

(Or use the embedded svn client on MAC Os via a terminal)

2.3.4 Define Windows Environment Variable

As the target definition file contains Windows environment variable **EA_TARGETS**, we should define it first.

EA_TARGETS points to directory on local hard disk where target is exported from svn, for ex.: **C:\Work\targets** (it does not point directly to target 3.6 directory)

Note: If you Eclipse is open, please restart it, so that it can recognize the newly set environment variable.

2.3.5 Configure Target Platform in Eclipse

Open the **target-3.6.target** file located in the **com.pulsear.targetdefs,** then in the editor, click on the "**Set as Target Platform**" in the top right corner of the editor.

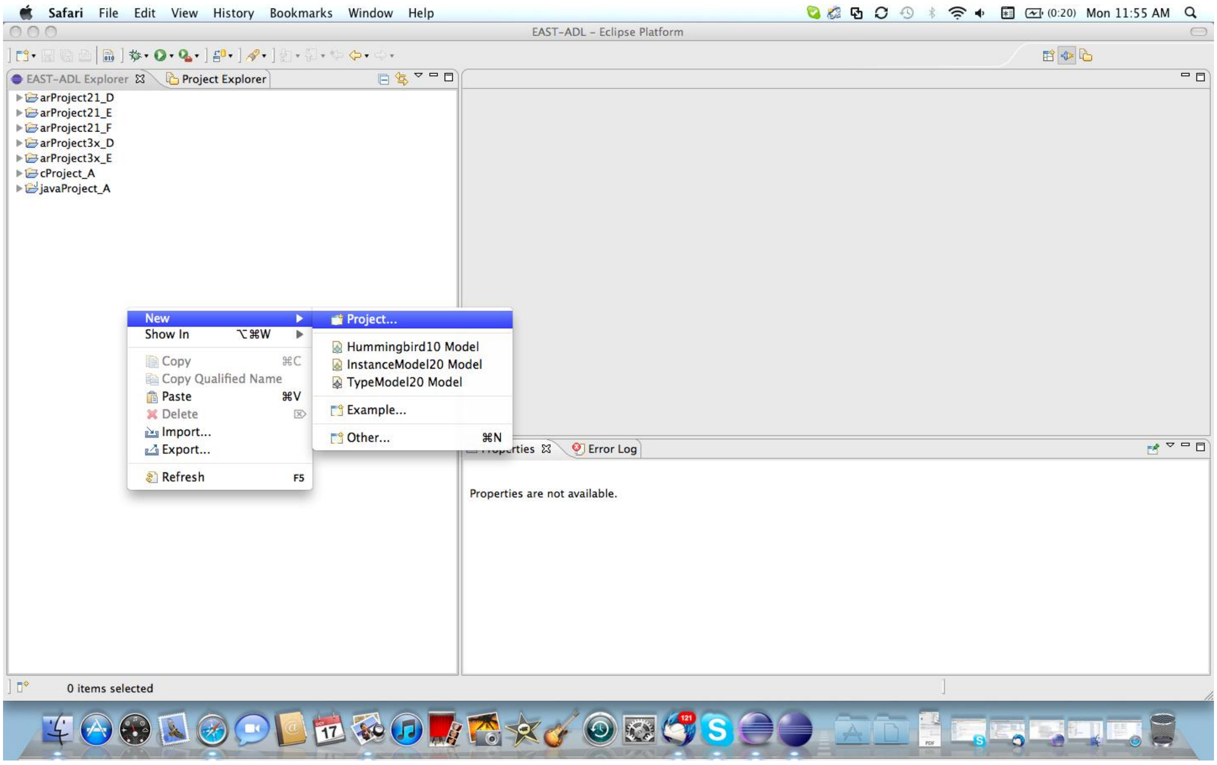
2.4 Check Out EastAdlOnSphinx plugins

Now and after achieving all previous steps successfully, our Eclipse Workspace is ready to receive our EAXML demonstrator plug-ins. Go to the SVN Repository Exploring perspective; navigate the created repository in WP5/EAXML-demonstrator, select all the plug-ins and check out them.

2.5 Launch EAXML Demonstrator

Create an Eclipse launch application to run the EAXML demonstrator.

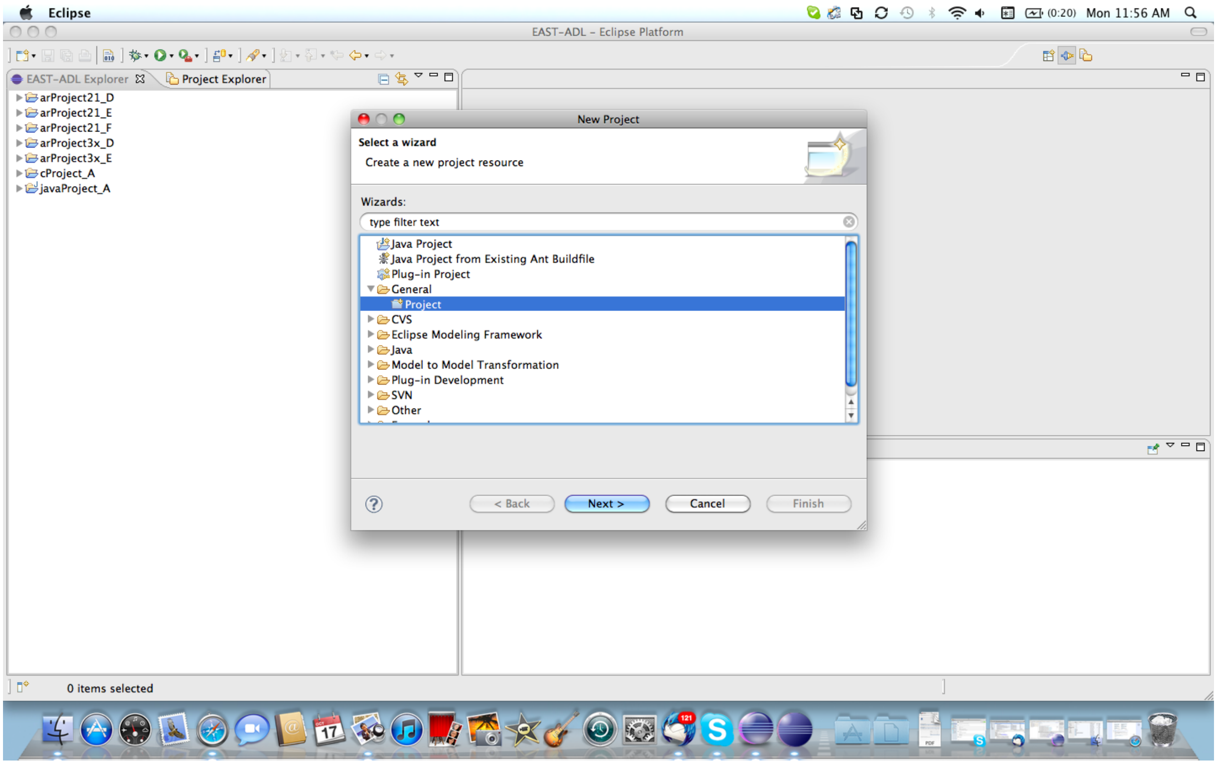
The current demonstrator does not have a specific wizard related to EAST-ADL, so that a generic project has to be first created, and then the EAST-ADL nature has to be manually added. See the 4 next figures below.



Properties are not available.

0 items selected

Figure 9: Create new project



New Project

Select a wizard

Create a new project resource

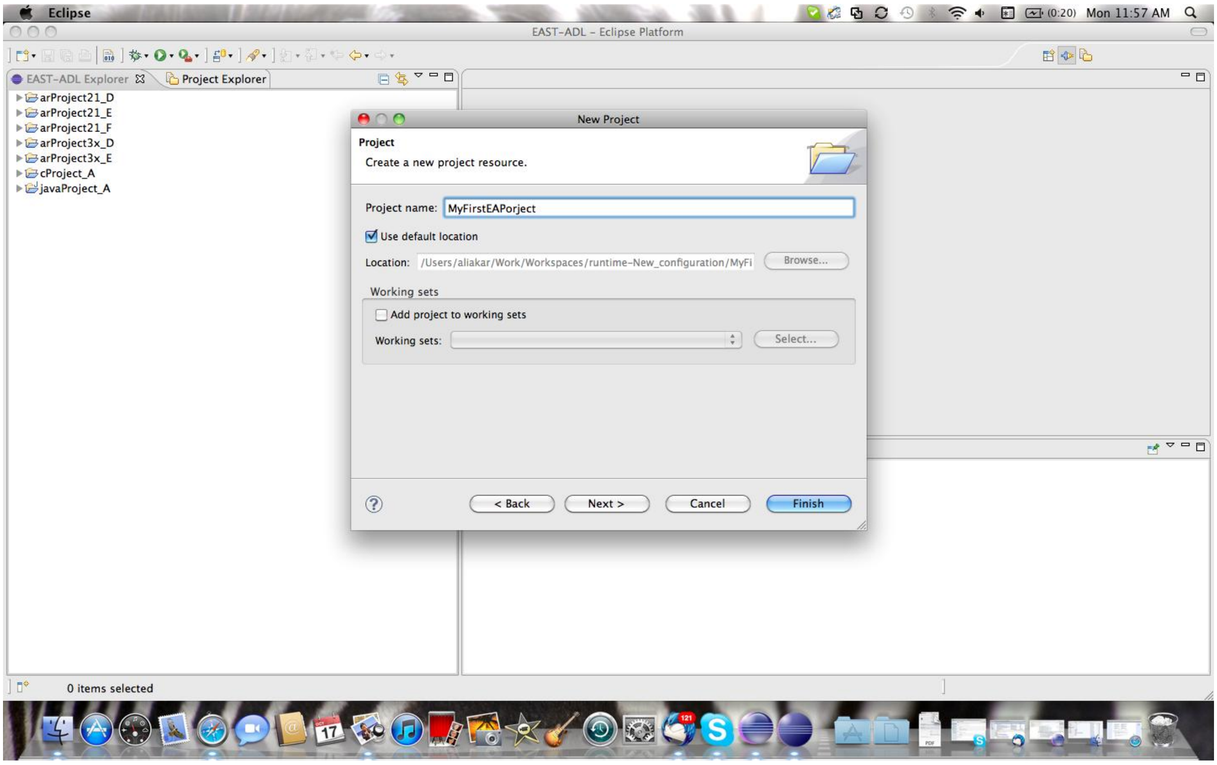
Wizards:

type filter text

- Java Project
- Java Project from Existing Ant Buildfile
- Plug-in Project
- General
  - Project
- CVS
- Eclipse Modeling Framework
- Java
- Model to Model Transformation
- Plug-in Development
- SVN
- Other

0 items selected

Figure 10: Create general project



New Project

Project

Create a new project resource.

Project name: MyFirstEAPorject

☑ Use default location

Location: /Users/aliakar/Work/Workspaces/runtime-New_configuration/MyFi

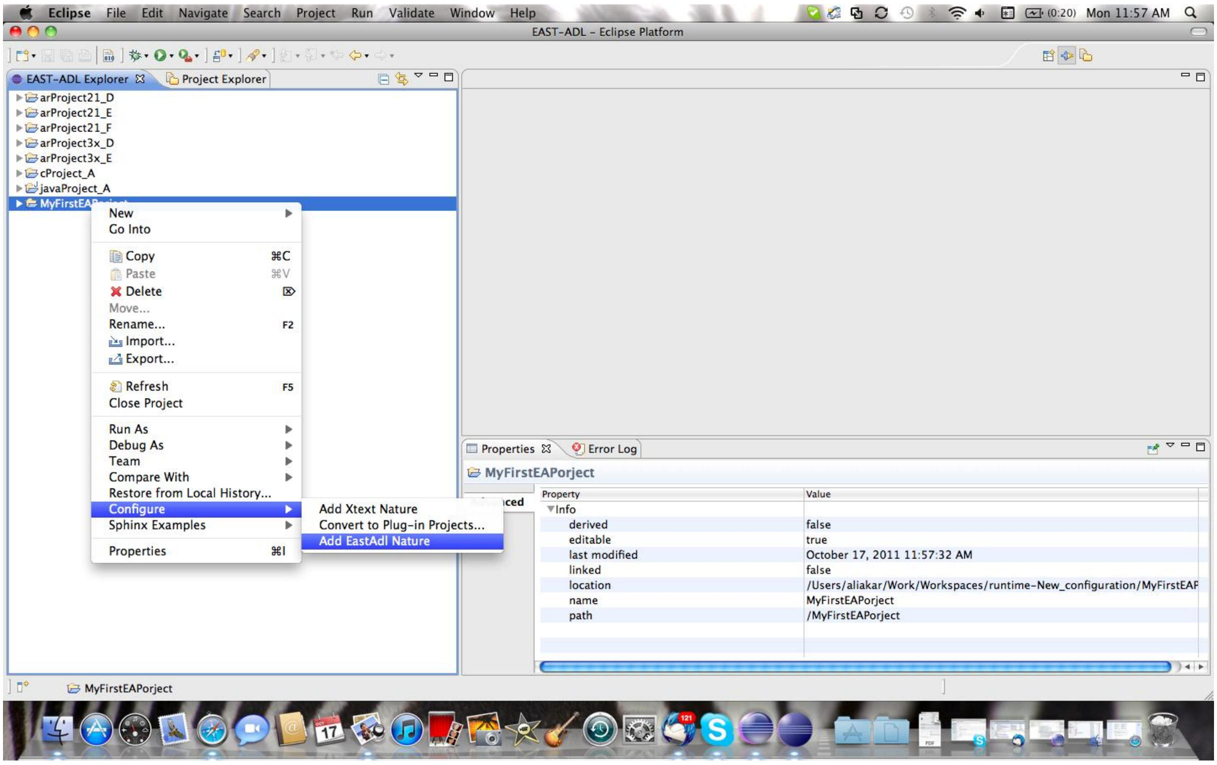
Working sets

□ Add project to working sets

Working sets:

0 items selected

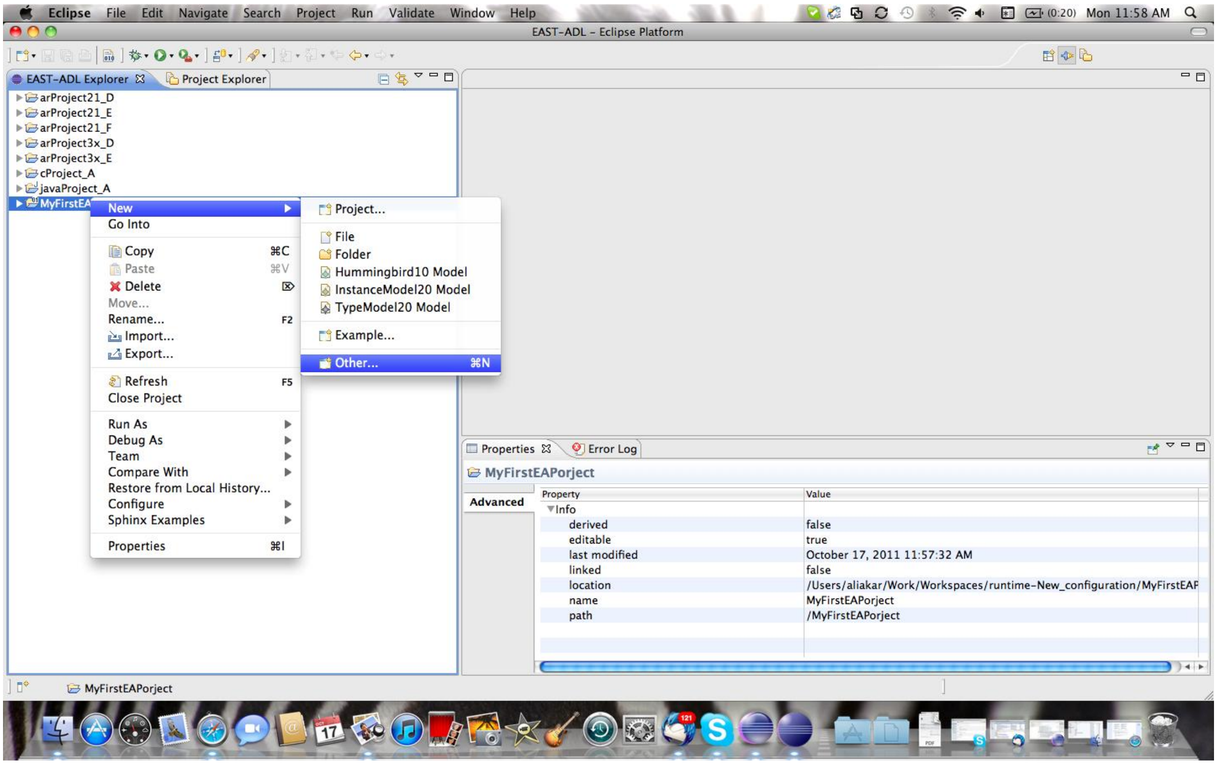
Figure 11: New project wizard



Property	Value
▼Info	
derived	false
editable	true
last modified	October 17, 2011 11:57:32 AM
linked	false
location	/Users/aliakar/Work/Workspaces/runtime-New_configuration/MyFirstEAF
name	MyFirstEAPorject
path	/MyFirstEAPorject

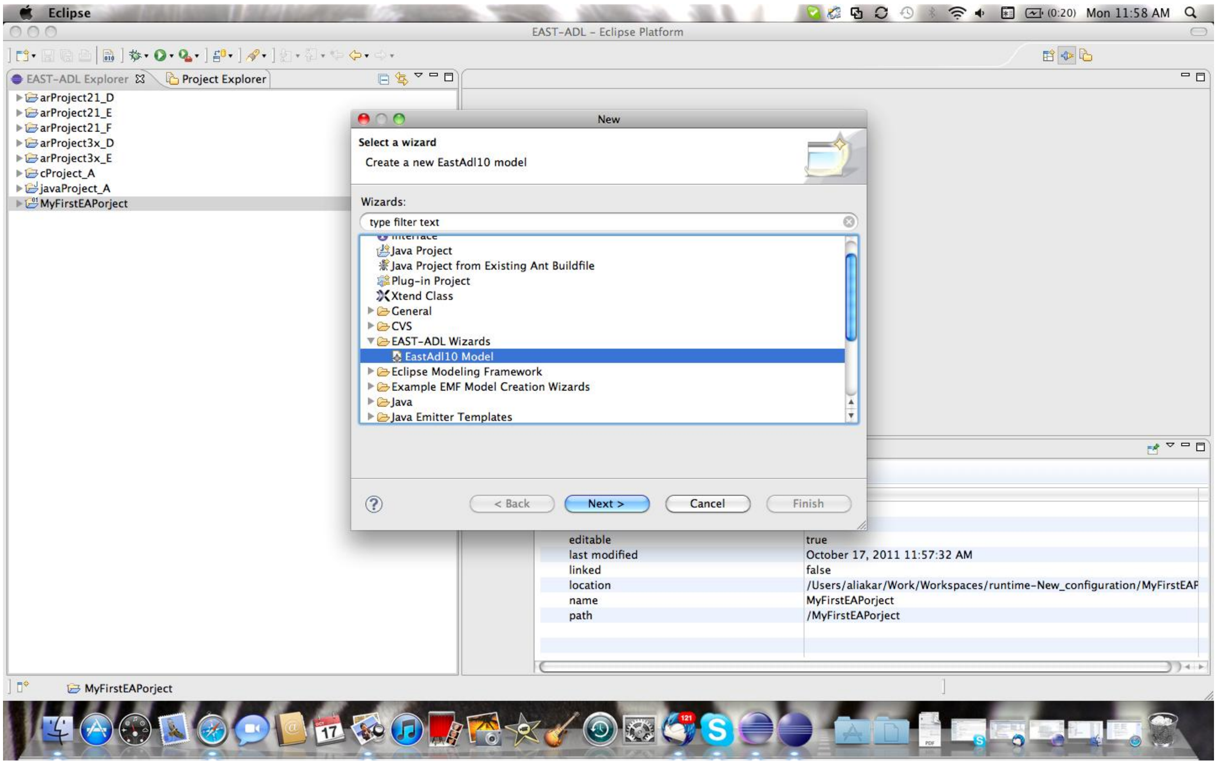
Figure 12: Add EAST-ADL Nature

The second step is the creation of a EAST-ADL model which is summarized in the following figures. This time a specific wizard is available to this end.



Property	Value
▼Info	
derived	false
editable	true
last modified	October 17, 2011 11:57:32 AM
linked	false
location	/Users/aliakar/Work/Workspaces/runtime-New_configuration/MyFirstEAF
name	MyFirstEAPorject
path	/MyFirstEAPorject

Figure 13: Create EAST-ADL Model

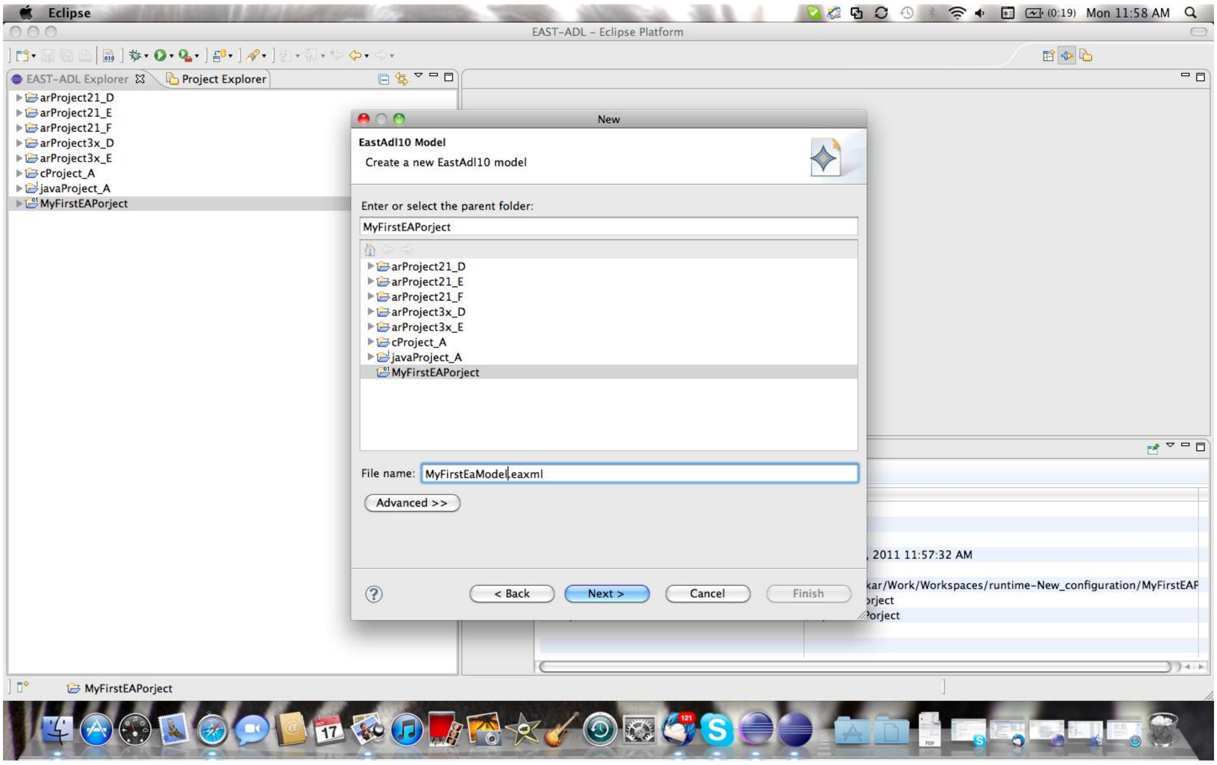


Select a wizard

Create a new EastAdl10 model

editable	true
last modified	October 17, 2011 11:57:32 AM
linked	false
location	/Users/aliakar/Work/Workspaces/runtime-New_configuration/MyFirstEAF
name	MyFirstEAPorject
path	/MyFirstEAPorject

Figure 14: Select EAST-ADL Wizard



EastAdl10 Model

Create a new EastAdl10 model

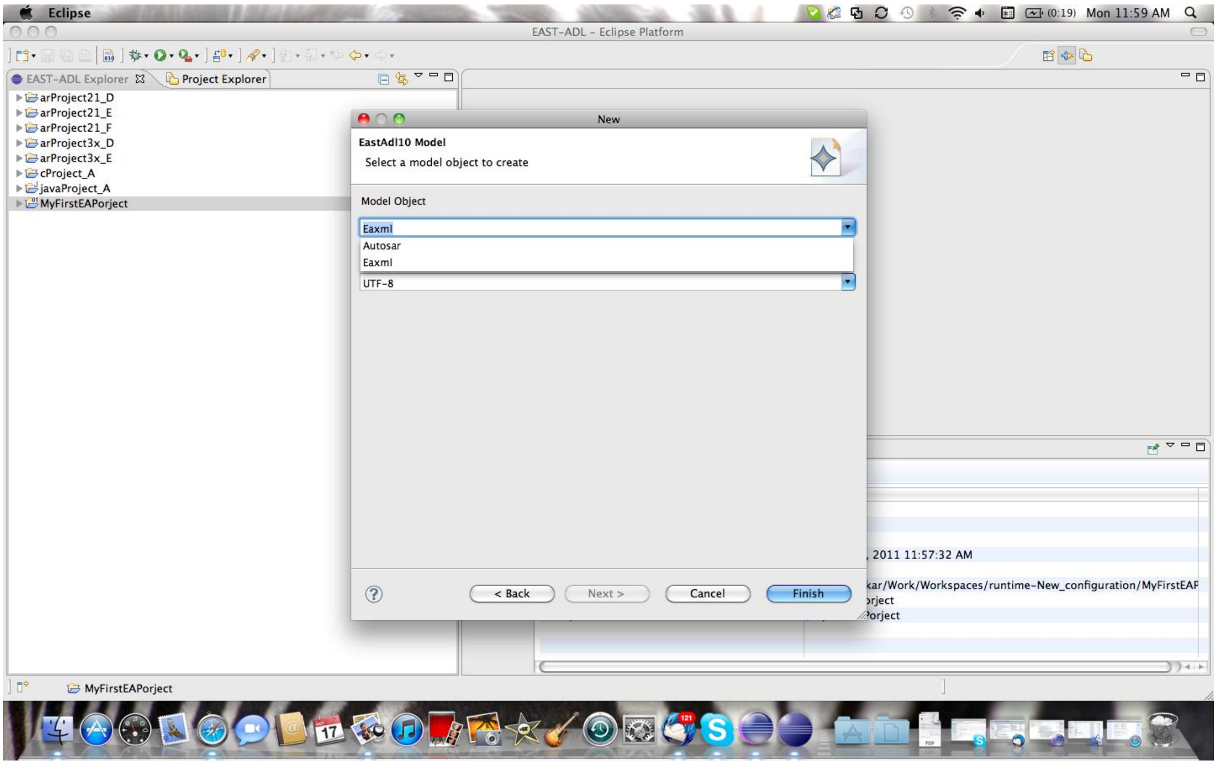
Enter or select the parent folder:

MyFirstEAPorject

File name: MyFirstEaModel.eaxml

Figure 15: Enter file name

Due to some minor issues on the XML schema used, you have to select the root object named Eaxml, in the last window of the creation wizard.



New

EastAdl10 Model

Select a model object to create

Model Object: Eaxml

UTF-8

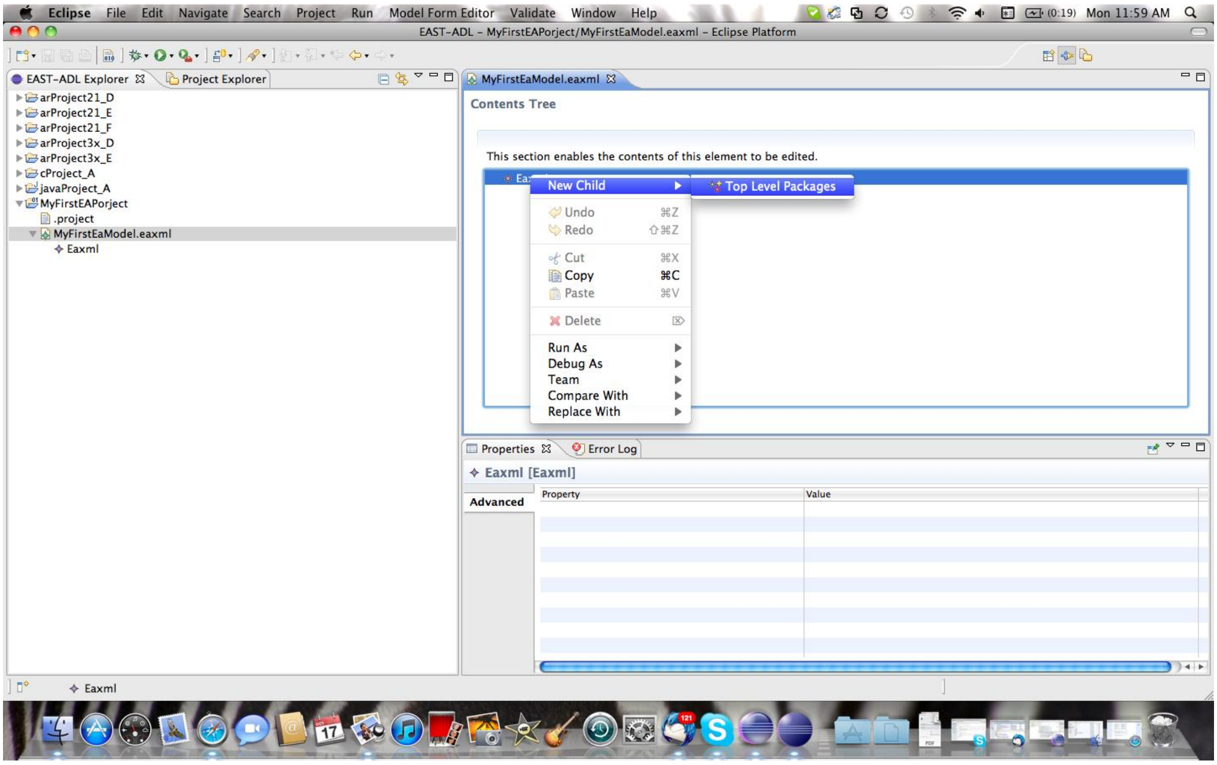
Figure 16: Select Root model

From now on, you could create whatever model element by right clicking on the root model, and then further by adding child to any existing element.

On the navigation side, you will see a tree representing the hierarchy of you model including all designed element (no filter are made up to now).

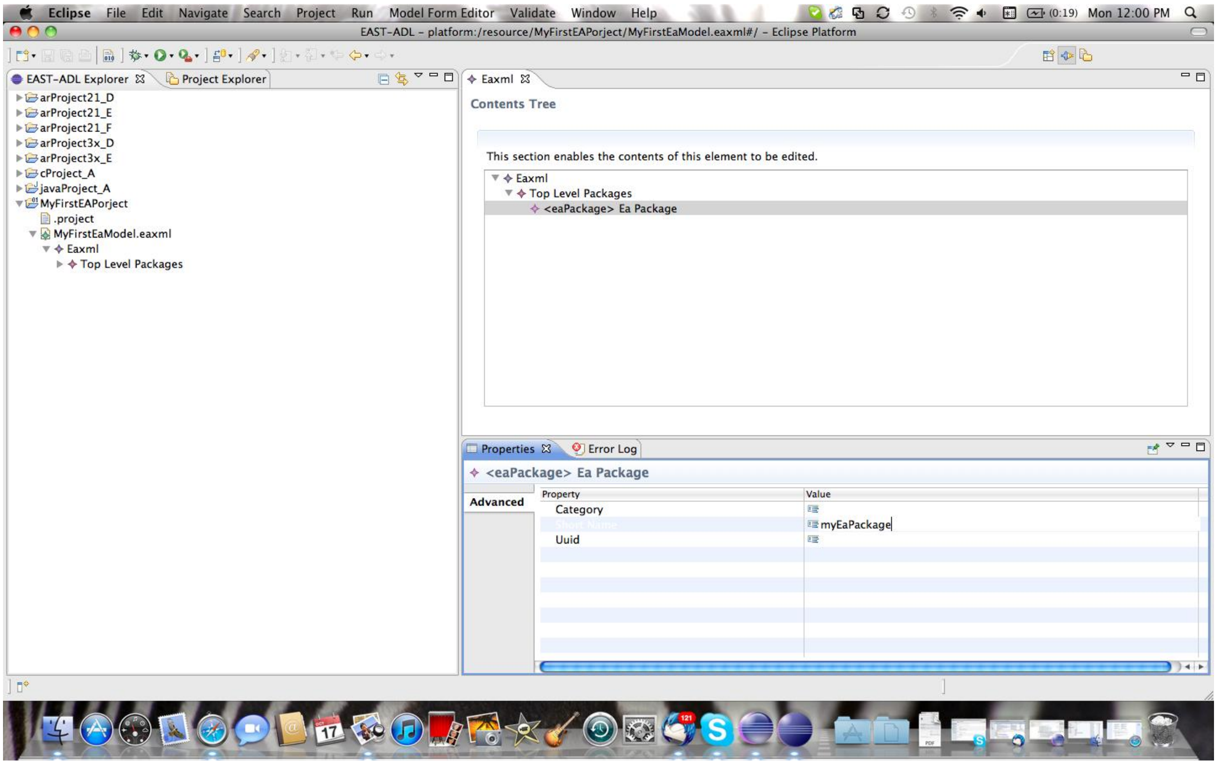
On the editor side, the generic view is also the same tree representation.

On the property tab, you could find all relevant properties for a given East-ADL object.



This section enables the contents of this element to be edited.

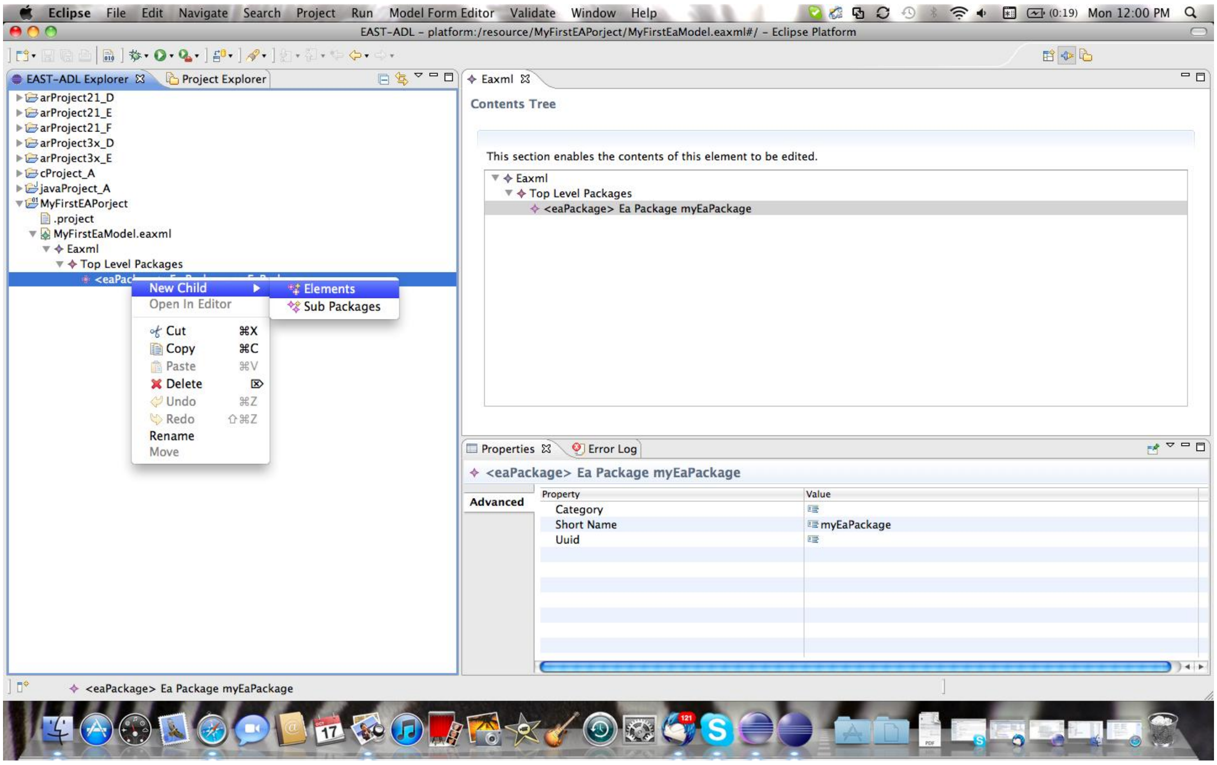
Figure 17: EAST-ADLEditor



This section enables the contents of this element to be edited.

Property	Value
Category	
Short Name	myEaPackage
Uuid	

Figure 18: Edit EAST-ADLModel



Contents Tree

This section enables the contents of this element to be edited.

- Eaxml
  - Top Level Packages
    - <eaPackage> Ea Package myEaPackage

<eaPackage> Ea Package myEaPackage

Property	Value
Category	
Short Name	myEaPackage
Uuid	

Figure 19: Create New Child

3 Summary and Conclusions

The whole EAST-ADL model is implemented in this demonstrator. Since it is based on a version of a Schema which still contains several minor issues, the corresponding modeling representation might suffer from some inconsistencies.

The next step will be to resolve those issues and integrate several enhancements like wizard, filters on the tree to have a more confortable editing environment.

The second point consist in the investigation to be able to integrate Papyrus modeling capabilities through the EAST-ADL profile with automatic transformation.