CHESS Platform

Installation Guide

Release:	1.10
Date:	14 November 2018
Author:	Stefano Puri

TABLE OF CONTENTS

List of Figures

Acronyms

API	Application Programming Interface
ATL	ATL Transformation Language
BDD	Block Definition Diagrams
BVR	Base Variability Resolution
CDO	Connected Data Objects
CDT	Eclipse C/C++ Development Tooling
CHESS	Composition with Guarantees for High-integrity Embedded Software Components Assembly
EEF	Extended Editing Framework
ELK	Eclipse Layout Kernel
EPF	Eclipse Process Framework
EST	Eclipse Standard Tools
GEF	Eclipse Graphical Editing Framework
GMF	Graphical Modelling Framework
GUI	Graphical User Interface
IBD	Internal Block Diagram
LTL	Linear Temporal Logic
MARTE	Modelling and Analysis of Real Time and Embedded systems
OCRA	Othello Contracts Refinement Analysis
QVT	Query/View/Transformation
SDK	Software Development Kit
SQL	Structured Query Language
SSL	Secure Sockets Layer
SysML	System Modelling Language
UI	User Interface
URI	Uniform Resource Identifier
V&V	Verification & Validation
XMI	XML Metadata Interchange
XML	eXtended Markup Language
XSD	XML Schema Definition

1. Introduction

1.1 Scope

The objective of this document is to describe the steps needed to setup a CHESS modelling environment.

1.2 Pre-requisites

The CHESS Modelling Tools runs based on Papyrus, thus it requires, as basic environment:

- **Java Runtime Environment (JRE) 1.8** or greather;
- **Eclipse IDE 4.6 (Neon) - Eclipse Modeling Package**;
- **Papyrus 2.0 (Neon)**.

Java Development Kit (JDK) 8 is only required to compile and run the PolarSys CHESS from the source code.

Additional components and dependencies are described in the section 2 “CHESS installation”.

1.3 Naming Conventions

1.3.1 CHESS

The CHESS code uses the following naming convention for its plugins:

org.polarsys.CHESS.*[module].[plugin-name]*

where *[module]* can be one of the following modules:

- ***contracts***: support for contract-based design and analysis
- ***fla***: support for failure logic analysis
- ***sba***: support for state-based analysis

APIs with visibility limited to the owning plugin should be stored in a package named as “xxx.internal.yyy” (e.g. *org.polarsys.chess.service.**internal**.utils*); an internal package should not be exported to an external client (i.e. it should not appear in the *runtime* tab of the plugin.xml editor).

2. CHESS Installation

2.1 Pre-requisite: Java Runtime Environment (JRE) 1.8 and Java Development Kit (JDK) 8

Both JRE and JDK are available from Oracle Java SE web site: https://www.oracle.com/technetwork/java/javase/downloads/index.html. Follow the installation instructions available on that site. Supported platforms are Microsoft Windows, Linux, macOS and Oracle Solaris.

2.2 Pre-requisite: Eclipse Neon IDE

Eclipse Modelling Tools package, Neon 3 version, is available on Eclipse Foundation site at: https://www.eclipse.org/downloads/packages/release/neon/3. Follow the installation instructions available on that site (https://wiki.eclipse.org/Eclipse/Installation).

2.3 Pre-requisite: Papyrus Neon

Papyrus is available in on Papyrus download page on Eclipse Foundation site at: https://www.eclipse.org/papyrus/download.html. Follow the instructions available on that page.

The following steps are suggested:

1. Use the discovery interface from the menu "**Help->Install Modelling components**" and select **Papyrus**. As an alternative, use the Eclipse Neon update site (http://download.eclipse.org/releases/neon) from "Help->Install new software" wizard, by having the "Contact all update sites during install to find required software" feature enabled; Papyrus can be found under the Modelling category (see Figure 1).

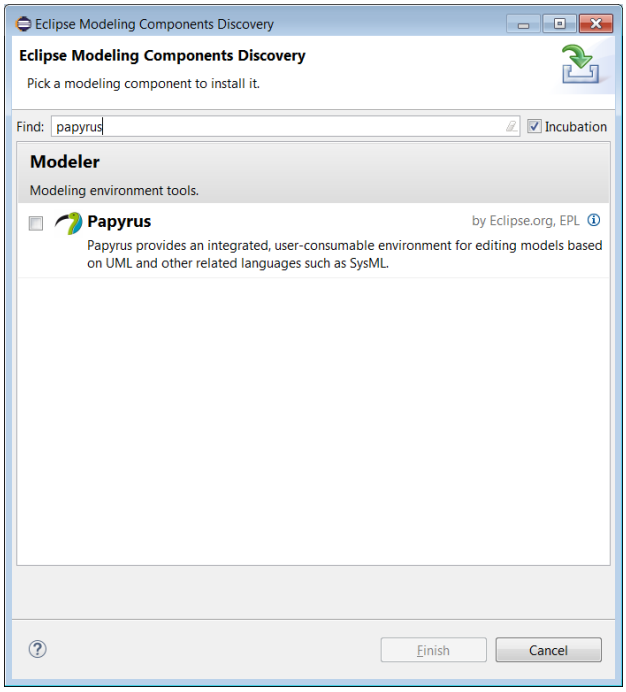

Figure 1. Installation of the Papyrus features

2. Restart Eclipse.

3. Install the **CDO Model Repository** from the menu option "**Help->Install Papyrus Additional Components**" (see Figure 2).

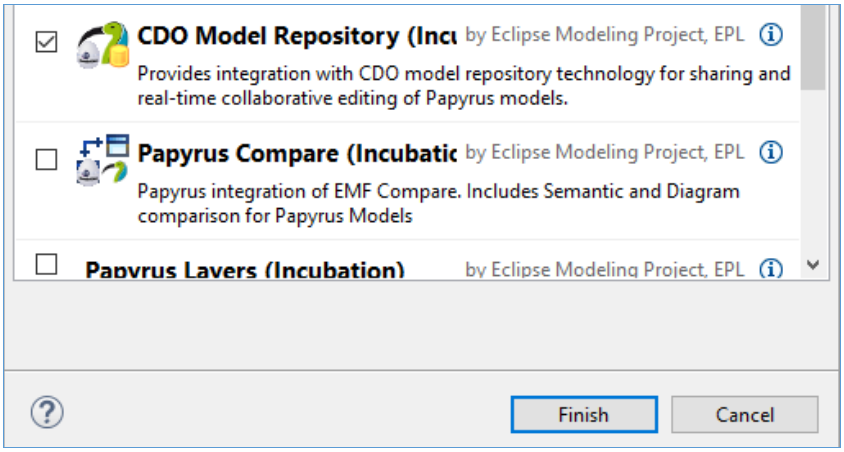

Figure 2. Installation of the CDO Model Repository

4. Install **SysML 1.1** from the following update site (from "**Eclipse Help->Install new software**"): http://download.eclipse.org/modeling/mdt/papyrus/updates/releases/neon

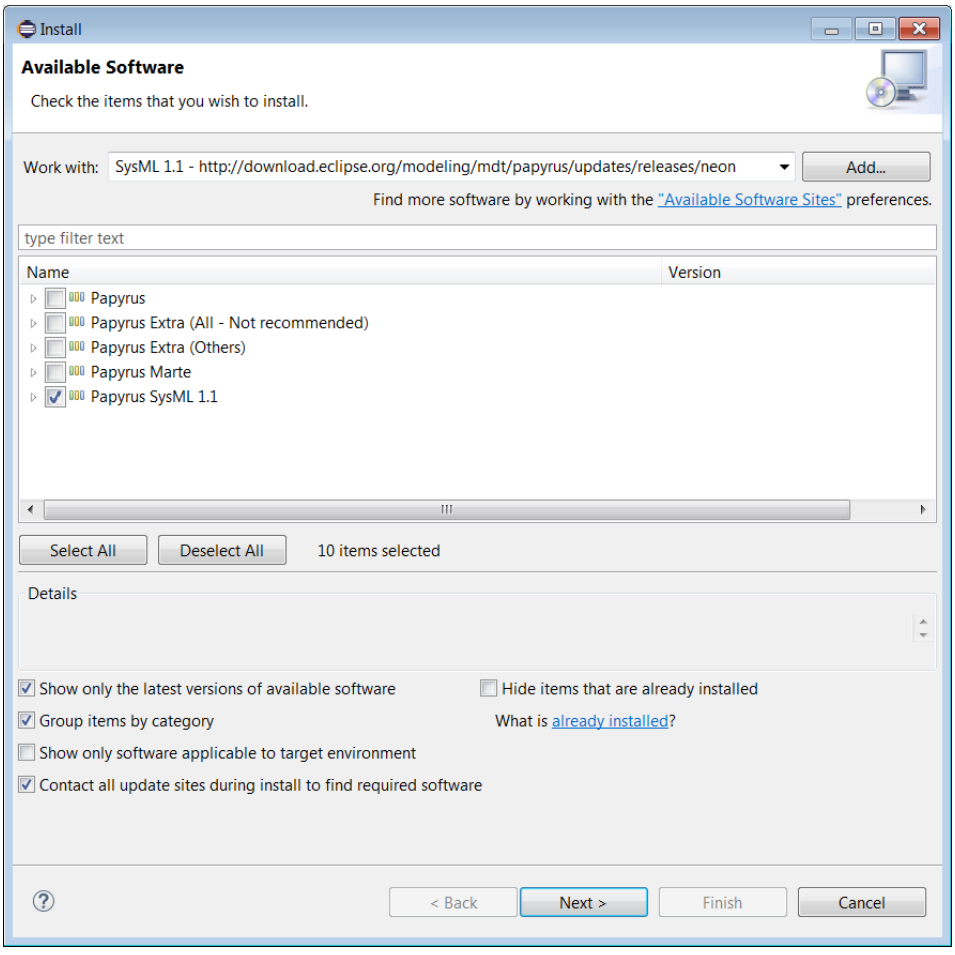

Figure 3. Installation of SysML 1.1

5. Install **MARTE** archive update site from: https://hudson.eclipse.org/papyrus/view/Marte/job/papyrus-marte-neon/lastSuccessfulBuild/artifact/releng/org.eclipse.papyrus.marte.p2/target/repository/*zip*/repository.zip
6. Install the MARTE features as shown in Figure 4 (required to fix issues with CDO).

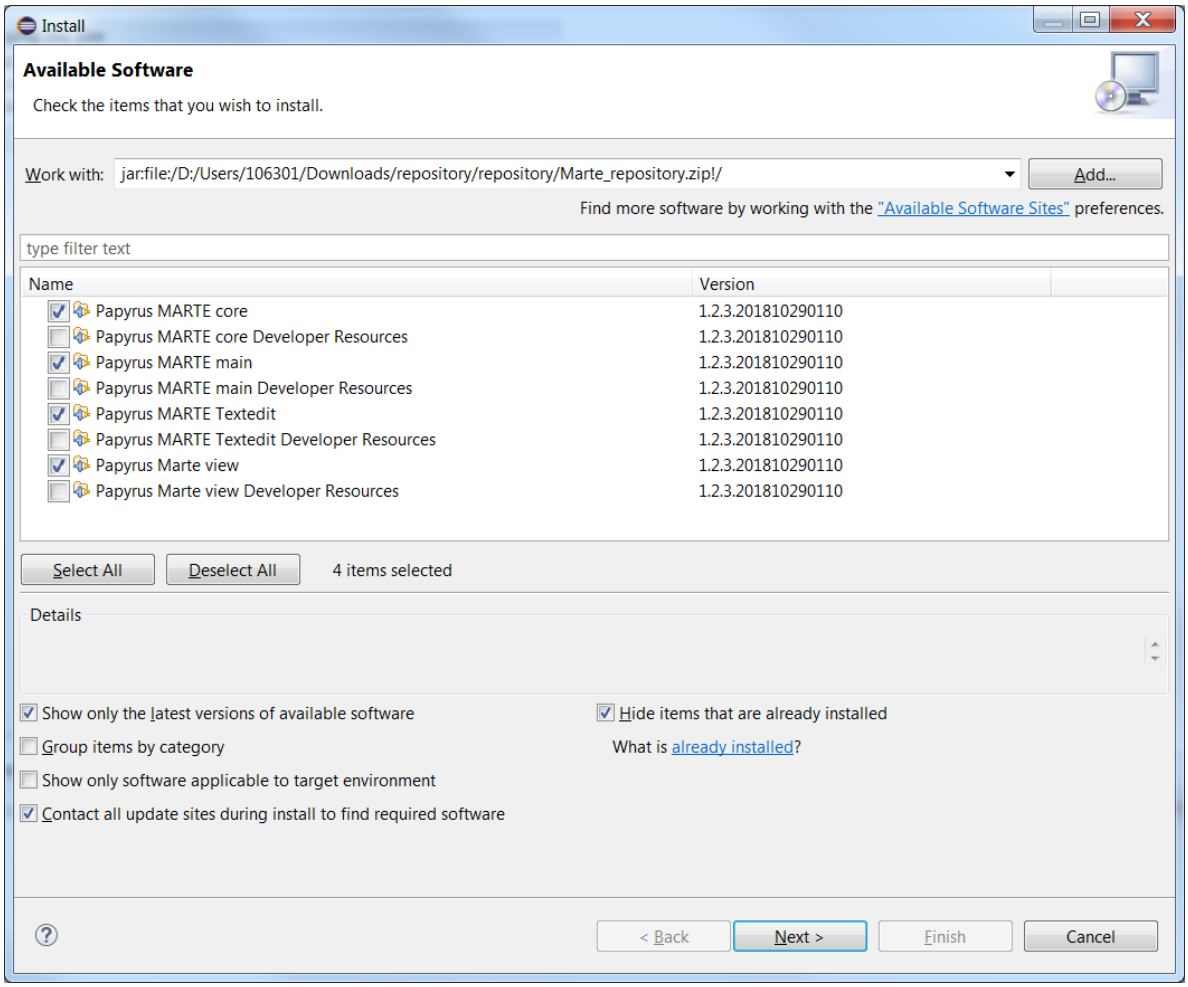

Figure 4. Installation of MARTE

2.4 CHESS dependencies

The following plugins are required to install the CHESS tool.

2.4.1 XText SDK 2.12

Install **XText SDK 2.12**. It can be installed from the following Eclipse update site http://download.eclipse.org/modeling/tmf/xtext/updates/composite/releases/. To see the version 2.12 remember to uncheck the box "Show only the latest versions of available software". Note that this will require to remove the EMF Parsley plugins.

2.4.2 Operational QVT, Acceleo, ATL features

Install **Operational QVT, Acceleo, ATL features** from the menu "**Help->Install Modelling components**".

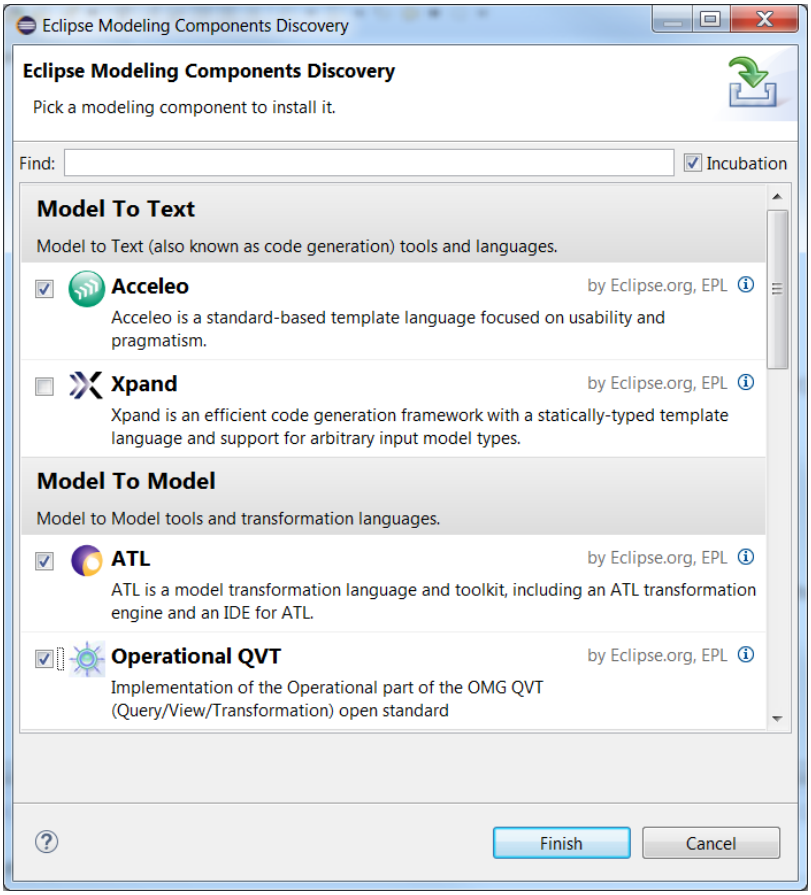

Figure 5. Installation of Operational QVT, Acceleo and ATL features (I)

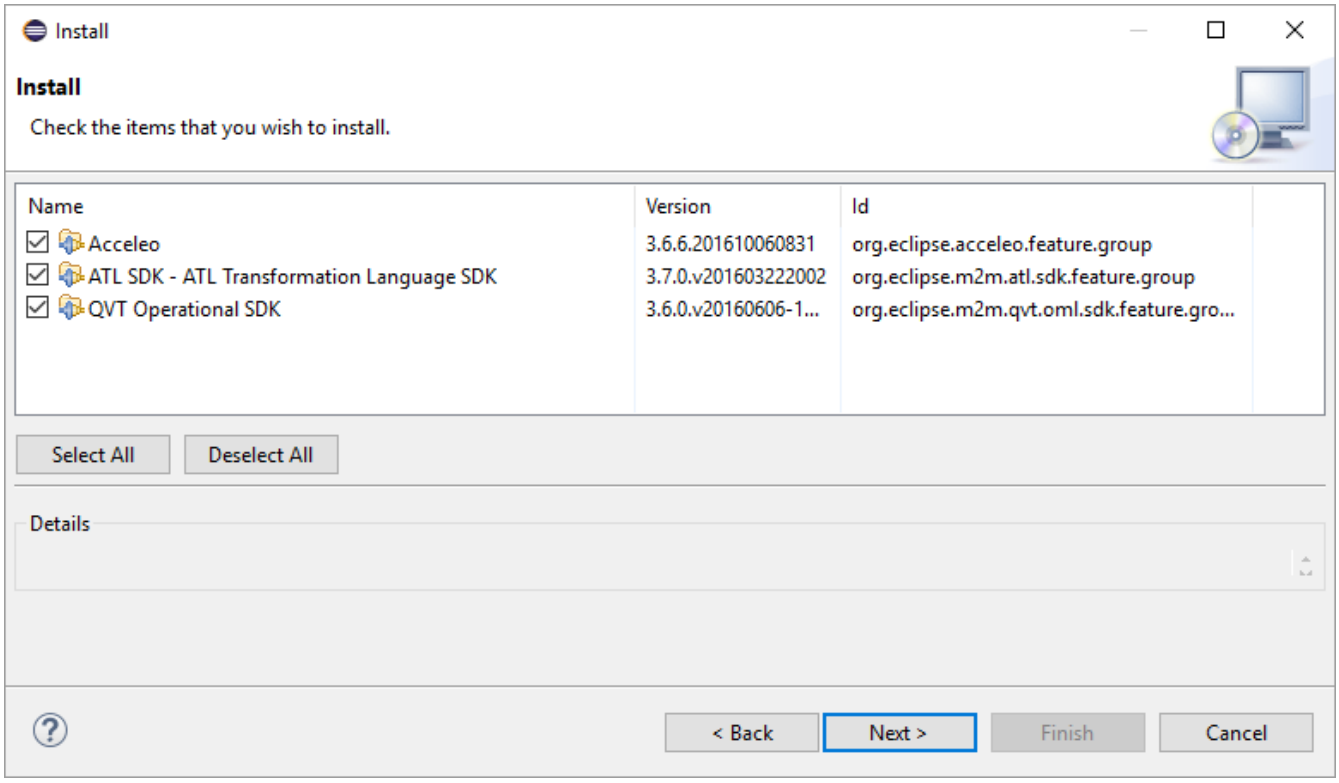

Figure 6. Installation of Operational QVT, Acceleo and ATL features (II)

2.4.3 Sirius

The plugins[1] that provide the graphical representation of fault trees, require Sirius. To install **Sirius** go to the following update site (from "**Eclipse Help -> Install new software**"):

http://download.eclipse.org/sirius/updates/releases/4.0.0/neon (select the Sirius Category)

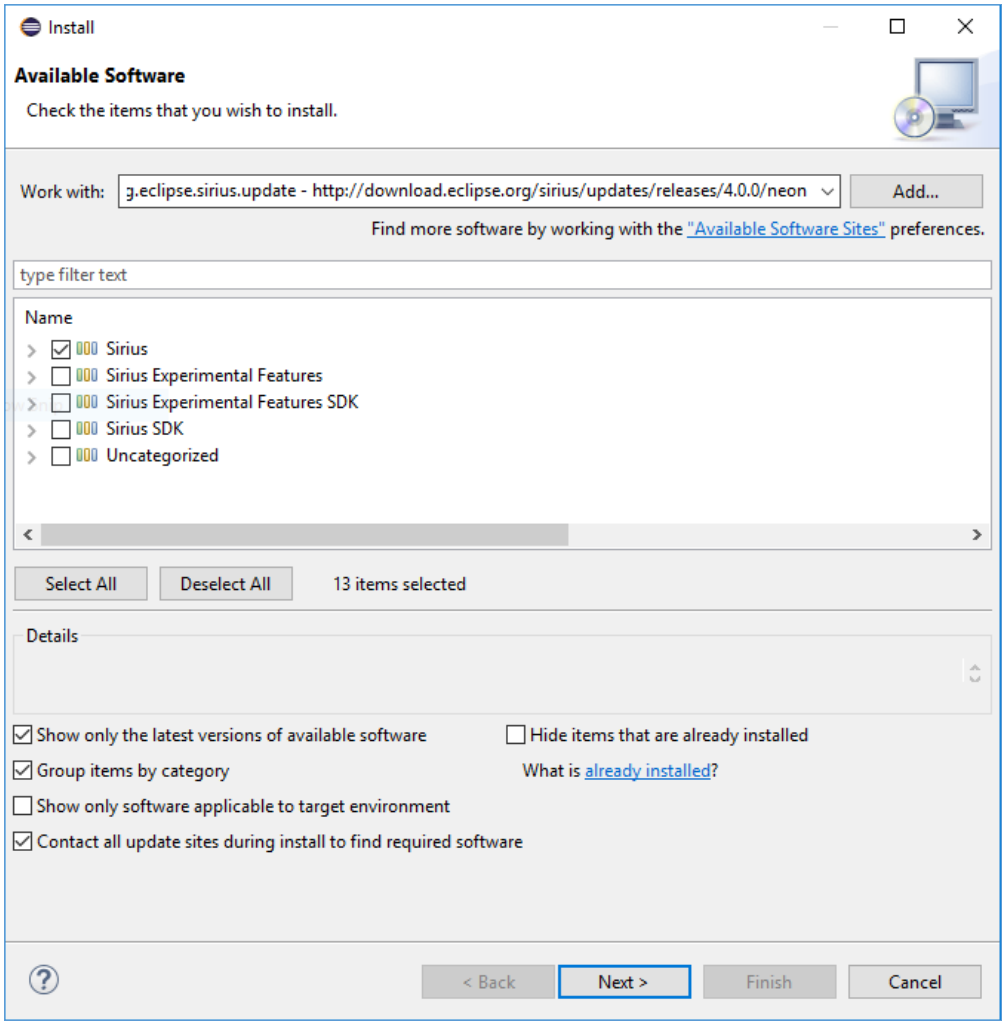

Figure 7. Installation of Sirius

[1] eu.fbk.eclipse.standardtools.faultTreeViewer, eu.fbk.eclipse.standardtools.faultTreeViewer.emfta, eu.fbk.eclipse.standardtools.faultTreeViewer.emfta.design

2.4.4 ELK

The plugins that create the BDD and IBD diagrams, require ELK (Eclipse Layout Kernel). To install **ELK** (required version >= 0.4.0) go to the following update site (from "Eclipse Help->Install new software"): http://download.eclipse.org/elk/updates/releases/0.4.0 and install the items as in Figure 8.

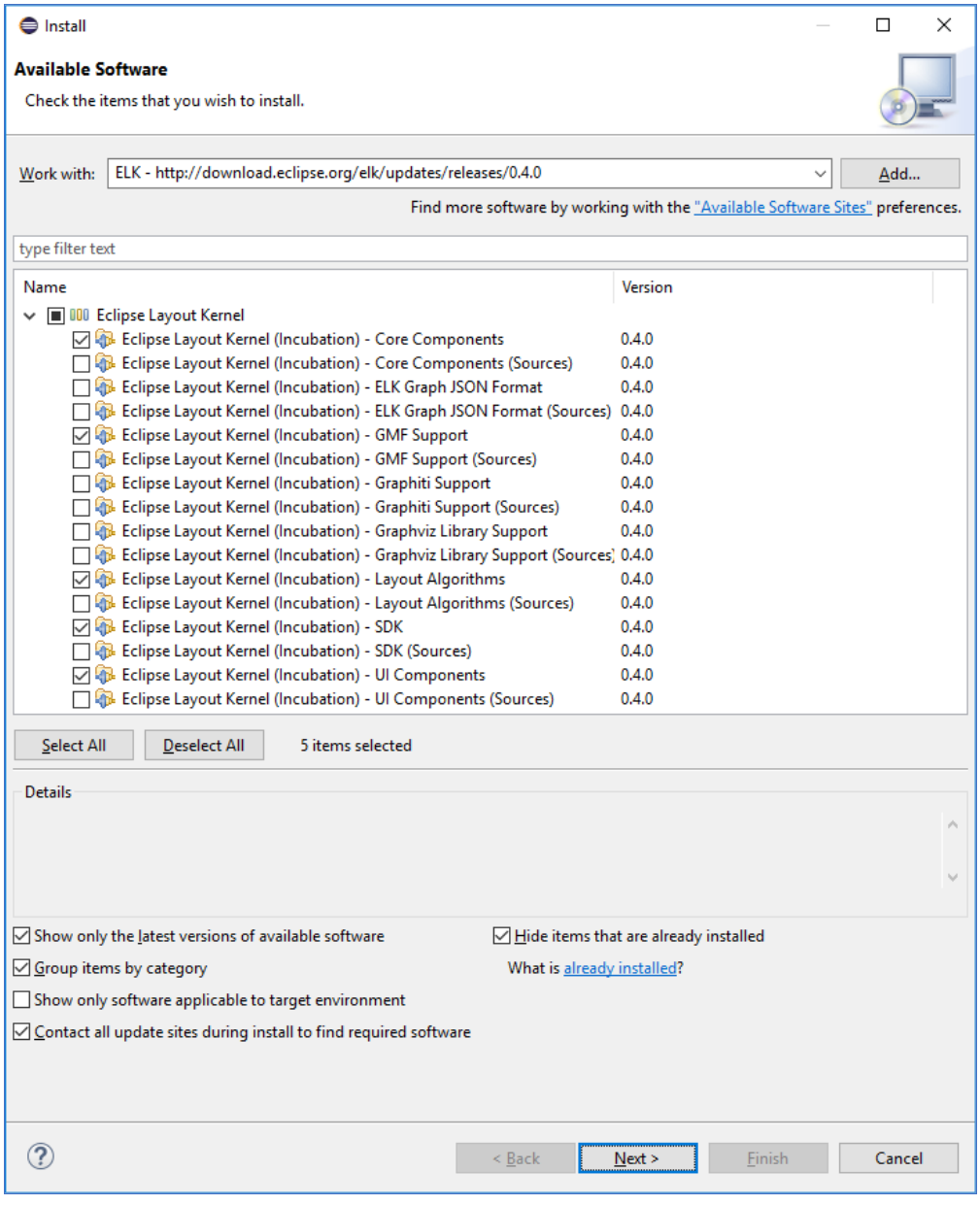

Figure 8. Installation of ELK

2.4.5 FBK

CHESS plugins depend on a set of Eclipse plugins provided as external libraries but not implemented in the project. They can be installed from the following Eclipse update site http://es-static.fbk.eu/tools/devel_sde. The available plugins are:

- eu.fbk.tools.adapter.*: plugins provided by FBK that enable the interaction with V&VTools such as OCRA, nuXmv and XSap.
- eu.fbk.tools.editor.*: plugins provided by FBK that enrich a text area with content assist for an LTL grammar. Do not install the "Xtext Redistributable" plugin because it is needed only to final users and not to developers.

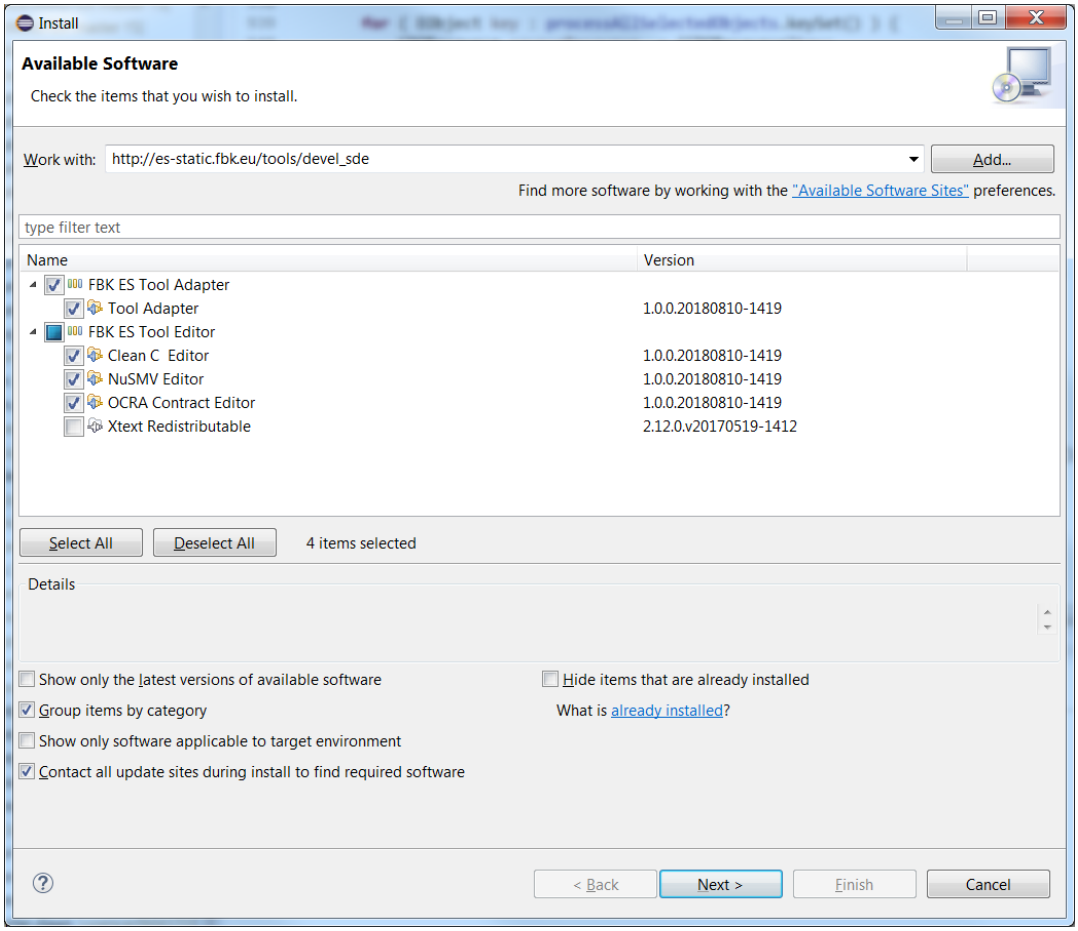

Figure 9. Installation of FBK tools

2.4.6 EST

Install **EST plugins** from http://es-static.fbk.eu/tools/devel_est, unchecking the "Group by category option" (see section 2.5 for more information about EST plugins).

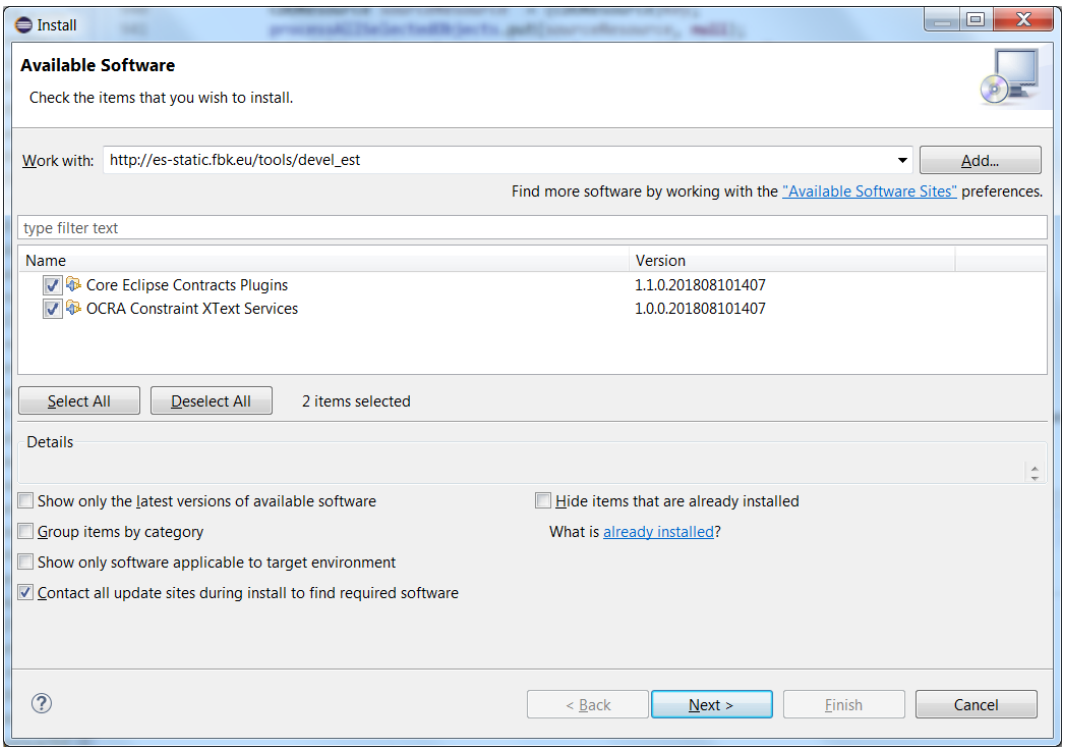

Figure 10. Installation of EST plugins

2.4.7 GEF 5.0

Download the **GEF 5.0** update site archive from here: https://www.eclipse.org/downloads/download.php?file=/tools/gef/downloads/drops//5.0.0/R201706131249/GEF-Update-5.0.0.zip and install the features showed in Figure 11.

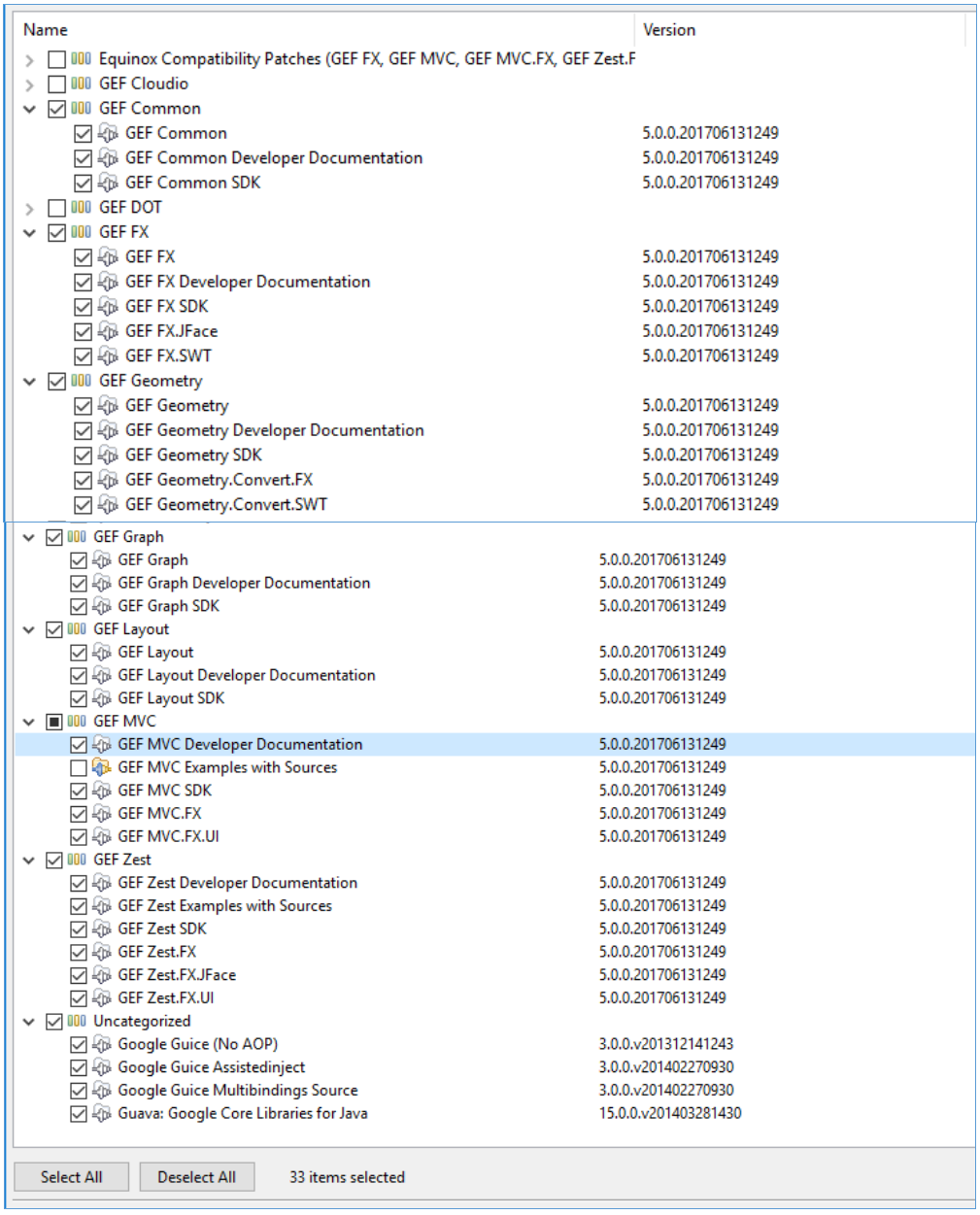

Figure 11. Installation of GEF 5.0 features

2.4.8 VIATRA

Install the **VIATRA** framework from http://download.eclipse.org/viatra/updates/release/1.7.2, in particular the features showed in Figure 12.

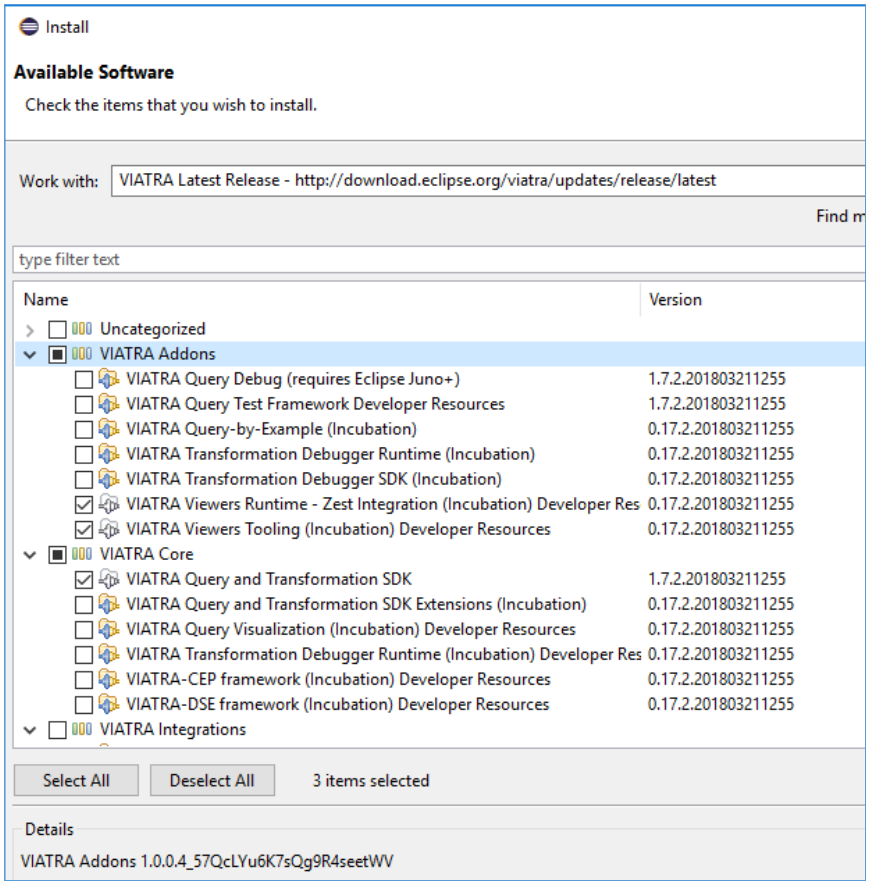

Figure 12. Installation of VIATRA features

2.5 Getting and Installing the CHESS Plugins

The CHESS plugins, and related libraries, can be installed in the given Eclipse environment through update sites, or their source version can be imported in the current workspace; the latter option must be preferred in case of CHESS plugins development. Otherwise, to avoid having too much plugins in the workspace, the first option can be adopted.

2.5.1.1 Install the CHESS plugins via update site

To install CHESS plugins on the current Eclipse environment:

1. Download the CHESS plugins from https://github.com/megamart2/tool-chess and install them from the downloaded zip file (unchecking the “Group by category option”).

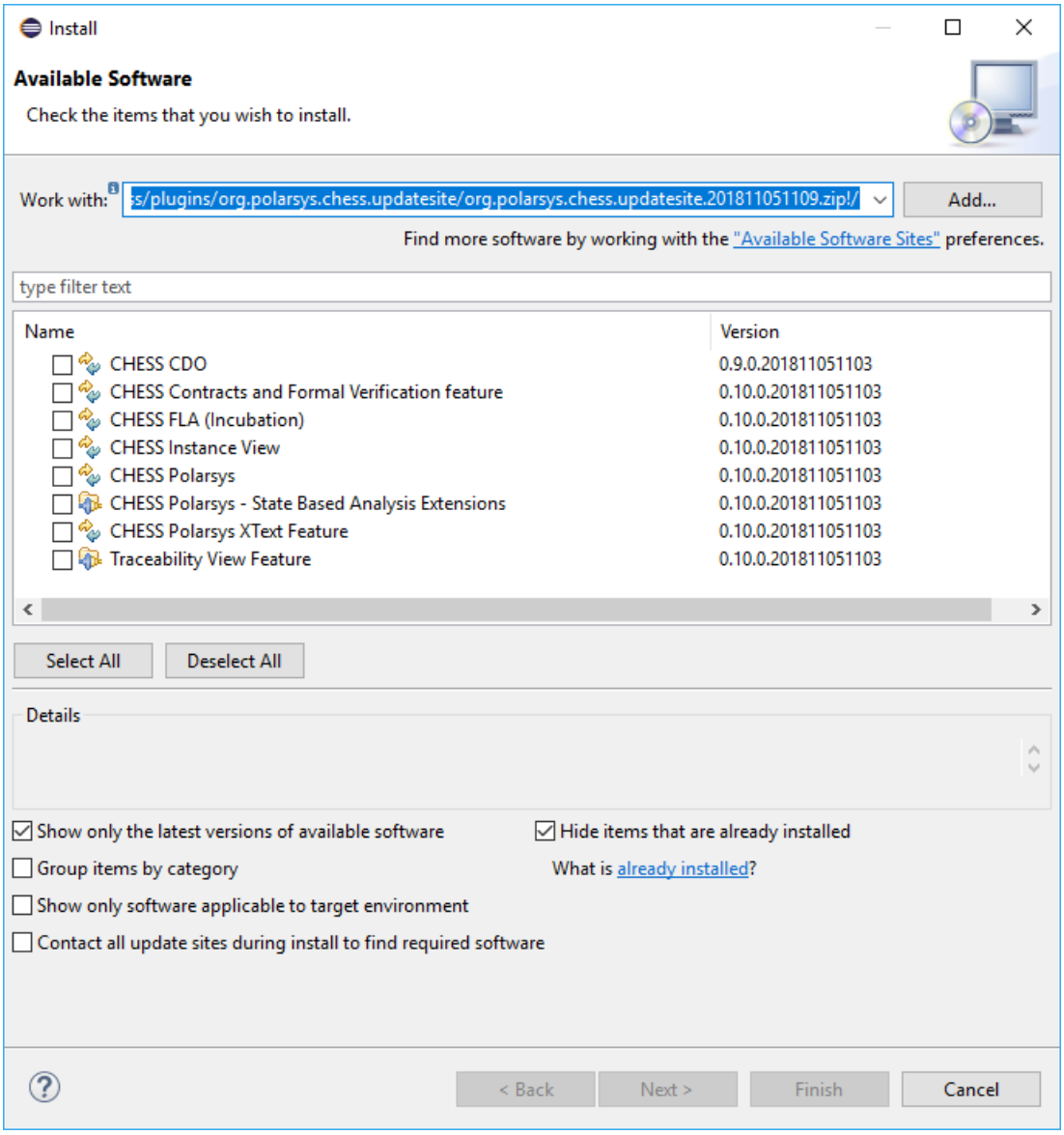

Figure 13. Installation of the CHESS plugins

The CHESS plugins are currently available at the Polarsys Git server:
https://git.polarsys.org/c/chess/chess.git

3. Installation of optional tools

3.1 BVR Installation

BVR is a tool for variability management; BVR can be installed and used together with CHESS to manage variability at architecture level.
The BVR plugins should work on Linux as well as on Windows. Make sure to have **Java 8** installed**.** The support for Neon 3 is taken into consideration.

3.1.1 Getting the BVR Plugins

The BVR tool bundle, i.e., a set of plug-ins for Eclipse that implements and supports the BVR language can be cloned from the particular GitHub link: https://github.com/SINTEF-9012/bvr.git. The source code of the BVR Tool is imported into the workspace for development environment setup.

The BVR update site https://bvr-tool.sintef.cloud/update/site.xml is also built from the BVR sources. To avoid having many source code plugins, the installation is done from the update site, as shown in Figure 14.

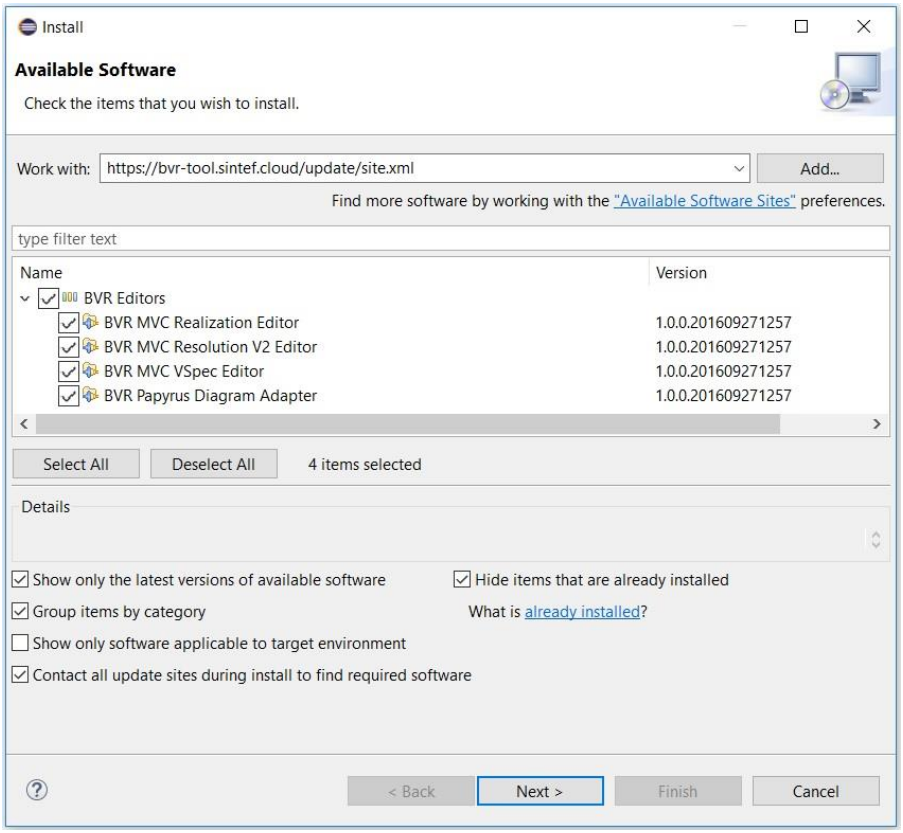

Figure 14. Installation of BVR Tool from the update site

3.2 Capra Installation

Capra is a tool for traceability, it is not strictly required to run and use CHESS. Capra can be installed and used to managed traceability between CHESS artefacts (e.g. model files, analysis artefacts) or model entities and any other resource inside and outside Eclipse.

3.2.1 Required Features

1. Install **Mylyn** from http://download.eclipse.org/mylyn/releases/latest/.

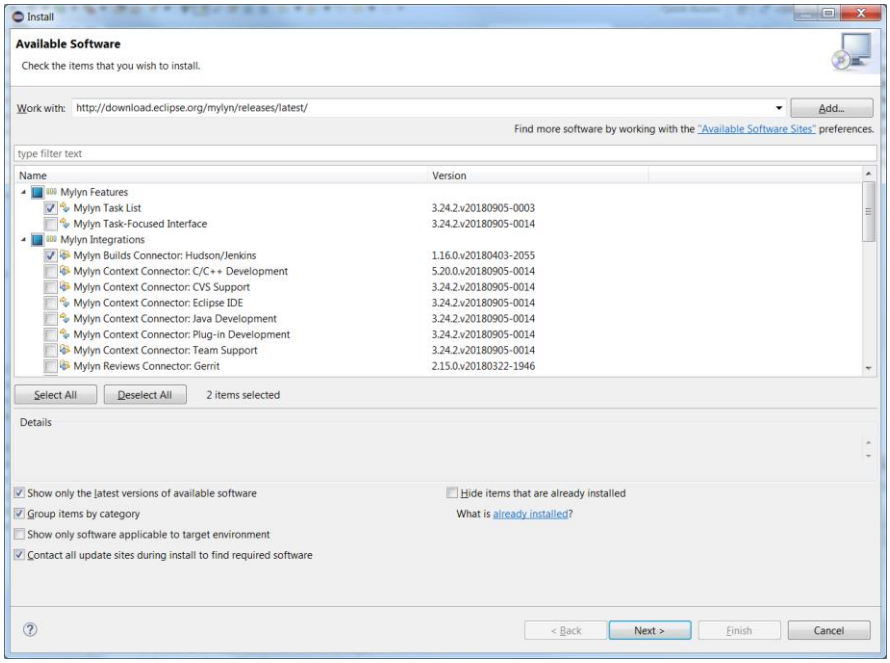

Figure 15. Installation of Mylyn

2. Install **CDT for Eclipse Neon**: http://download.eclipse.org/tools/cdt/releases/9.2

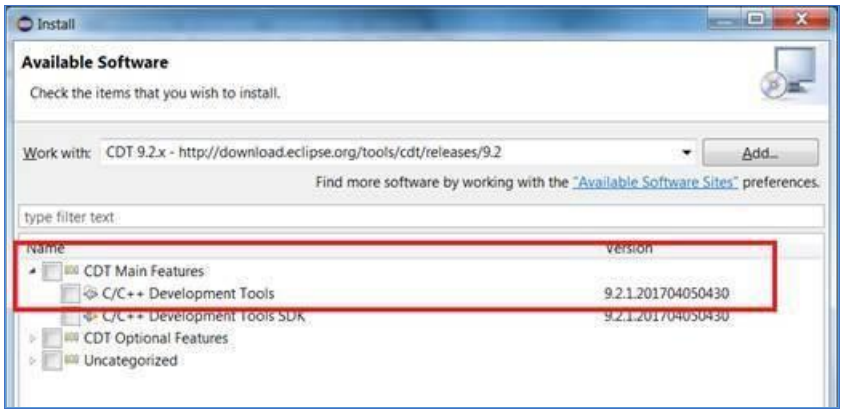

Figure 16. Installation of CDO for Eclipse Neon

3. Move the plugin *com.google.guava_21.0.0.v20170206-1425.jar* from the "plugins" folder to the "dropins" folder.

3.2.2 Getting the Capra Plugin

To install the Capra plugin, follow the next steps:

1. Clone the Git Repository available at https://github.com/jmauersberger/TraceabilityManagement
2. Import all the available plugins.