Contract number: ITEA2 – 10039

Safe Automotive soFtware architEcture (SAFE)

ITEA Roadmap application domains:

Major: Services, Systems & Software Creation

Minor: Society

ITEA Roadmap technology categories:

Major: Systems Engineering & Software Engineering

Minor 1: Engineering Process Support

WT4.1
Deliverable D4.1.1.d
SAFE Tool Platform User Guide

Due date of deliverable: 30.11.2014

Actual submission date: 27.11.2014

Start date of the project: 01/07/2011 **Duration:** 42 months

Project coordinator name: Stefan Voget

Organization name of lead contractor for this deliverable:

Editor: Yue Ma, Stefan Voget

Contributors: -

Revision chart and history log

Version	Date	Reason
0.1	11.04.2013	Draft
1.0	29.05.2013	Editorial Changes
1.1	26.03.2014	MMGen Tool Changes
2.0	27.11.2014	Update due to update of meta-model and new target-definition

1 Table of contents

2 Introduction

The scope of SAFE WT 4.1 "Meta-model implementation" is the implementation of the SAFE meta-model in WT 3.5. The requirements for SAFE WT 4.1 "Meta-model implementation" are described in [1]. The traceability between these requirements and the higher-level SAFE requirements is captured in the respective requirements traceability document.

This report summarizes the contributions to the deliverable D4.1.d and gives a manual for the usage of SAFE Tool Platform. The goal of the deliverable is to create an initial version of an Ecore for system, hardware and software in Eclipse.

It consists of two independent parts.

- A set of Eclipse based plug-ins created by itemis France. This set is an implementation of SAFE target tool platform.
- A Meta-Model Gen tool (MMGen) developed by itemis France. This includes the generation of Ecore metamodel, EMF code implementation and XSD schema.

3 SAFE Tool Platform

This chapter describes the contribution from itemis France to WT4.1.

3.1 License

The code of the platform from itemis France is made available under the terms of the ITEA2 SAFE project consortium agreement.

3.2 Installation guide

3.2.1 Prerequisites

1. Windows Vista, 7
2. Eclipse Kepler SR2
3. If you expect to install SAFE Tool Platform by checking out source code or by update site, you should also install Sphinx, EATOP (EAST-ADL Tool Platform) and ARTOP (AUTOSAR Tool Platform).

   You can:

   - either simply apply the SAFE target definition, which you can find in: https://safe.offis.de/svn/svndav/34_WP4_Technology_Platform/10_WT4_1_Meta_model_implementation/30_Development_artifacts/SAFE_Tool_Platform/org.safe.targetdefs/kepler/kepler.target
   - Or use update site for installing each of the following:

   - Nebula : http://download.eclipse.org/technology/nebula/snapshot
   - Sphinx: http://download.eclipse.org/sphinx/updates/interim
   - ARTOP: https://www.artop.org/containers/artop-sdk-update-site-4.2/
   - EATOP: http://download.eclipse.org/eatop/updates/interim

3.2.2 Installation

You can install SAFE Tool Platform with one of the following solutions. Please make sure that you have Nebula, Sphinx, ARTOP and EATOP installed in your Eclipse, if you choose solution 1 or solution 2.

1. Source code

   If you have already an Eclipse Kepler installed, you could check out the source and import all the *org.safe.** plug-ins under *SAFE_Tool_Patform* directory into your workspace. The URL of source repository for SAFE is:

   https://safe.offis.de/svn/svndav/34_WP4_Technology_Platform/10_WT4_1_Meta_model_implementation/30_Development_artifacts/SAFE_Tool_Platform

2. SAFE Tool Platform update site

   http://safe.offis.de/updates/safe/platform/updates/interim/

3. SAFE Technology Demonstrator Download

The SAFE Technology Demonstrator is a collection of examples that demonstrate the features and capabilities of the underlying platform. It is a standalone application that can be used to conveniently explore SAFE. It does not require an existing Eclipse installation.

http://safe.offis.de/updates/safe/demonstrator/downloads/

3.3 Main features

1. SAFE project/file creation support

   It provides services for creating SAFE project and file in the workspace.

2. SAFE perspective

   The SAFE perspective contains a SAFE explorer view and a SAFE form editor. They are created operating on shared model instances in Sphinx-based modeling tool applications.

3. SAFE reference mechanism

   This feature allows SAFE models could make reference to AUTOSAR and EAST-ADL items.

3.4 User Guide

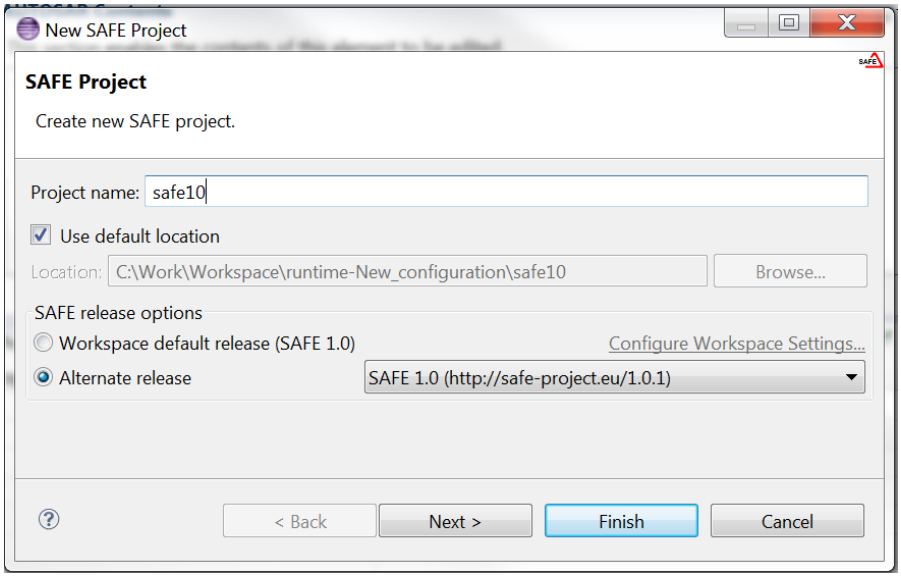
For more details guide tour, you could reference to the SAFE cheat sheet, "*Help -> Cheat sheets… -> SAFE Tool Development -> Guided Tour of the SAFE Technology Demonstrator*".

3.4.1 Open SAFE perspective

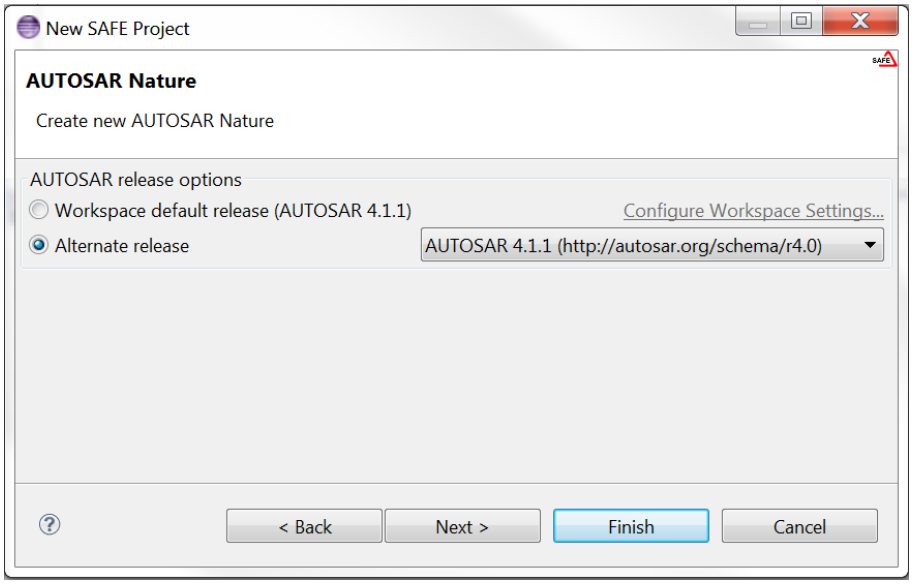
In the main menu, select "*Window > Open Perspective > Other",* and then select "*SAFE".*

3.4.2 Create a SAFE project

1. In the main menu select "*File > New > SAFE Project"*.
2. A dialog box invites you to name the project. You can set the SAFE release used for the project by selecting "*Alternate Release*" and by choosing a release in the list or decide to use the "*Workspace default release*". Click "*Next"*.



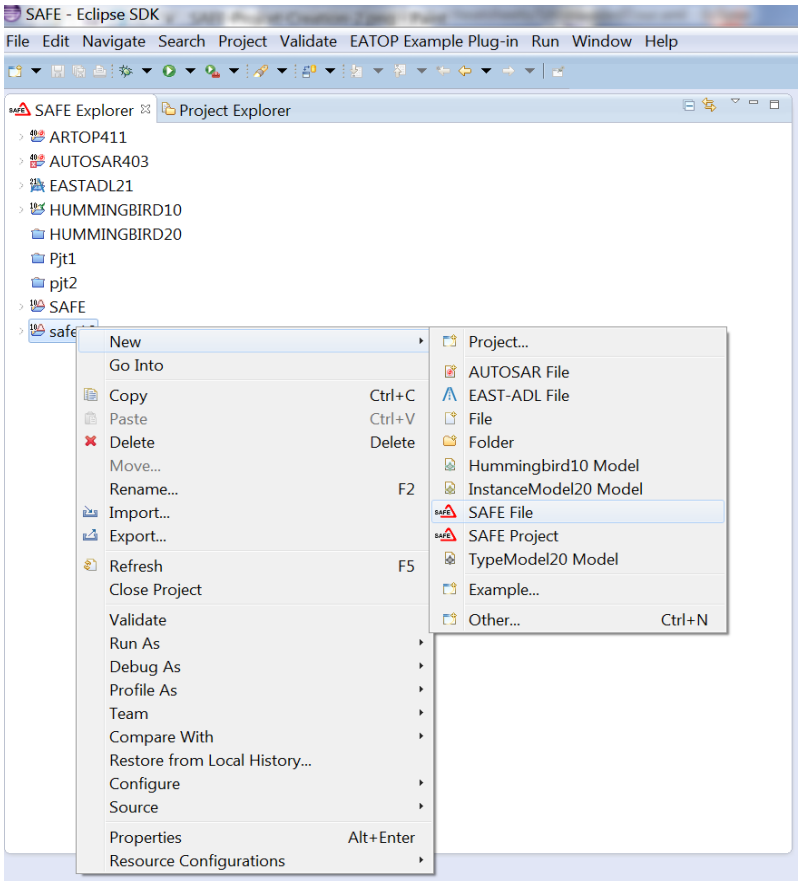
3. A second page is opened to select the "*AUTOSAR Nature*". You can set the AUTOSAR release by selecting "*Alternate Release*" and by choosing a release in the list or decide to use the "*Workspace default release*". Click "*Next*".



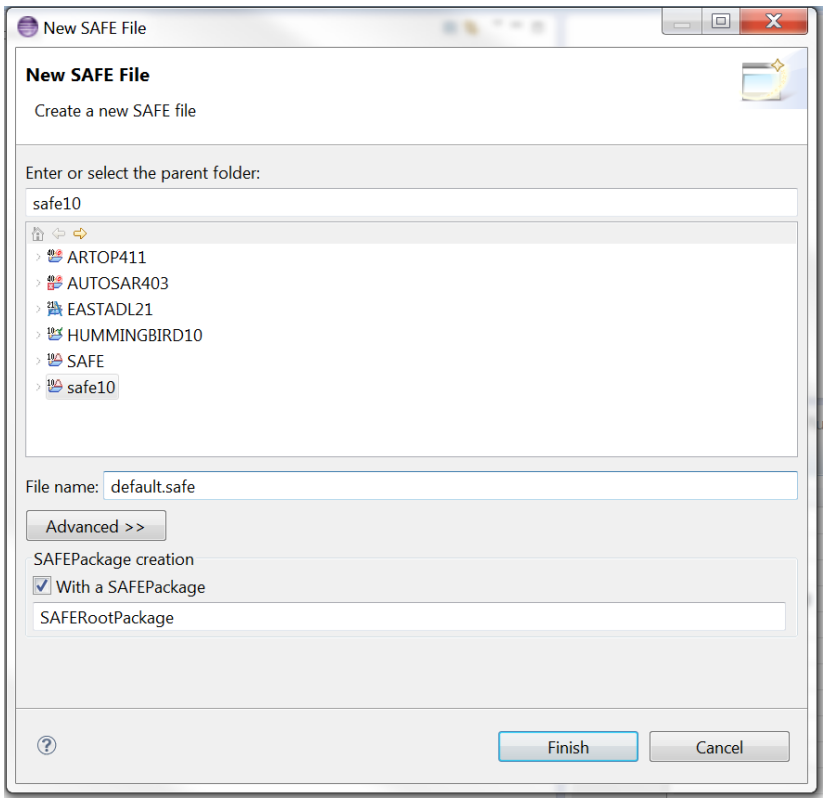
4. A third page is opened to select the "*EAST-ADL Nature*". Set the EAST-ADL release by selecting "*Alternate Release*" and by choosing a release in the list or decide to use the "*Workspace default release*".
5. Now you have set projects properties, and you can click on the "*Finish*" button to perform creation.

3.4.3 Create a SAFE file

1. Right click on a created SAFE project, select "*New >SAFE file*".



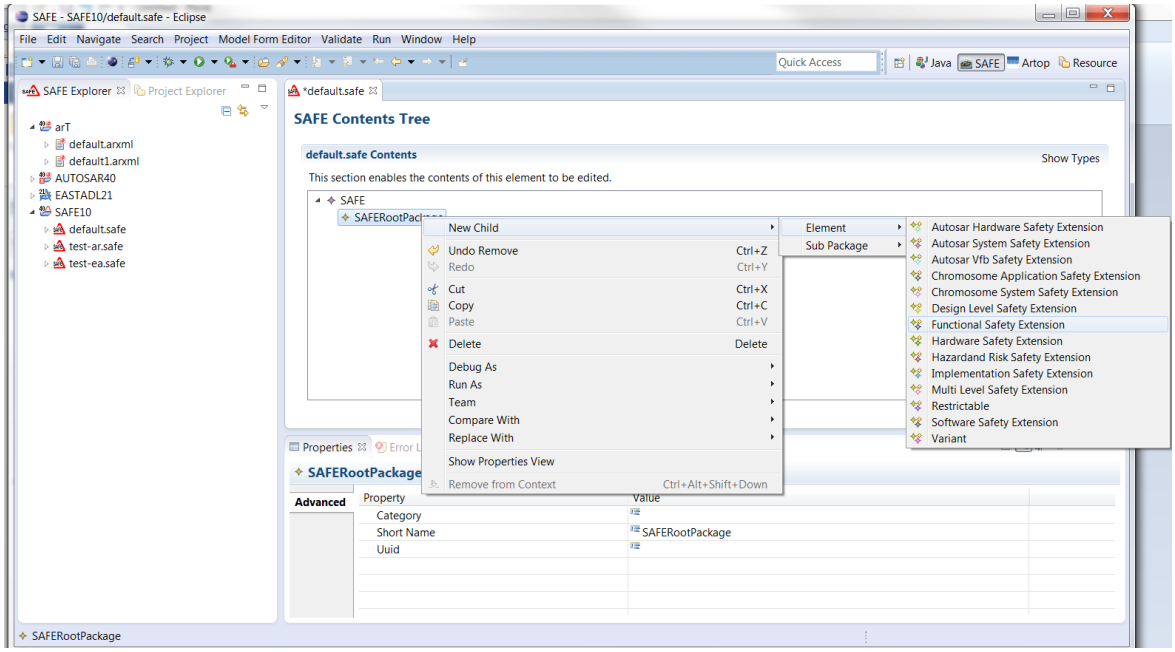
2. A wizard is opened. You can name the file or choose to keep the default name "*default.safe*". A "*SAFEPackage*" root can be automatically added into the file by selecting "*With a SAFEpackage*" in the "*SAFEPpackage Creation*" section.



3. Then click on the "Finish" button to perform creation.

3.4.4 Create a SAFE object

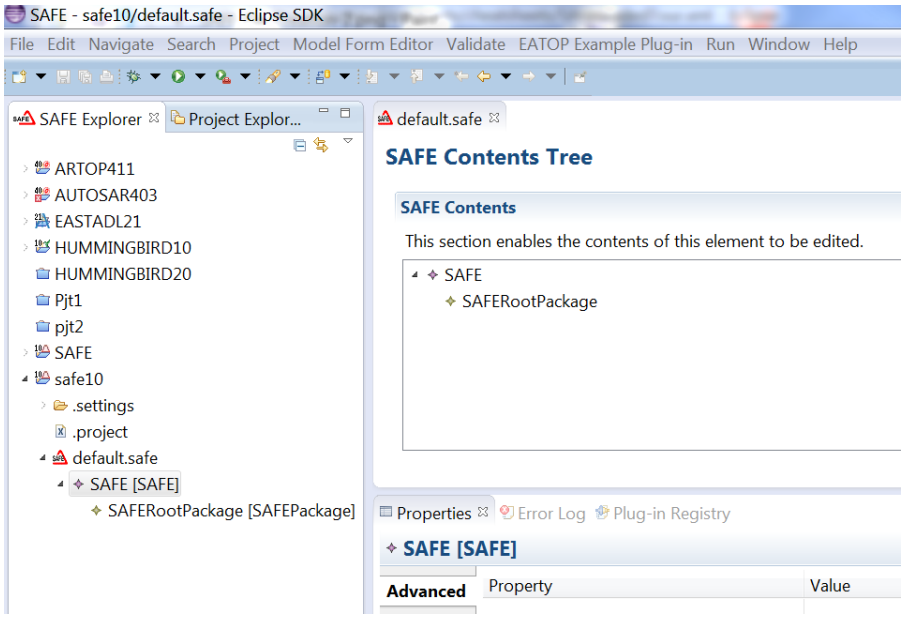
1. Open the "*SAFE explorer view*". Expand project composition and elements below SAFE file. Double click on SAFE element below SAFE file. The project editor view is now opened.



2. Expand elements, right click on an element, e.g., *SAFEPackage*, a menu appears, select "*New Child > Element > Functional Safety Extension*" and left click on it. You should now see a new Functional Safety Extension element below the selected package.

3.4.5 Explorer SAFE project

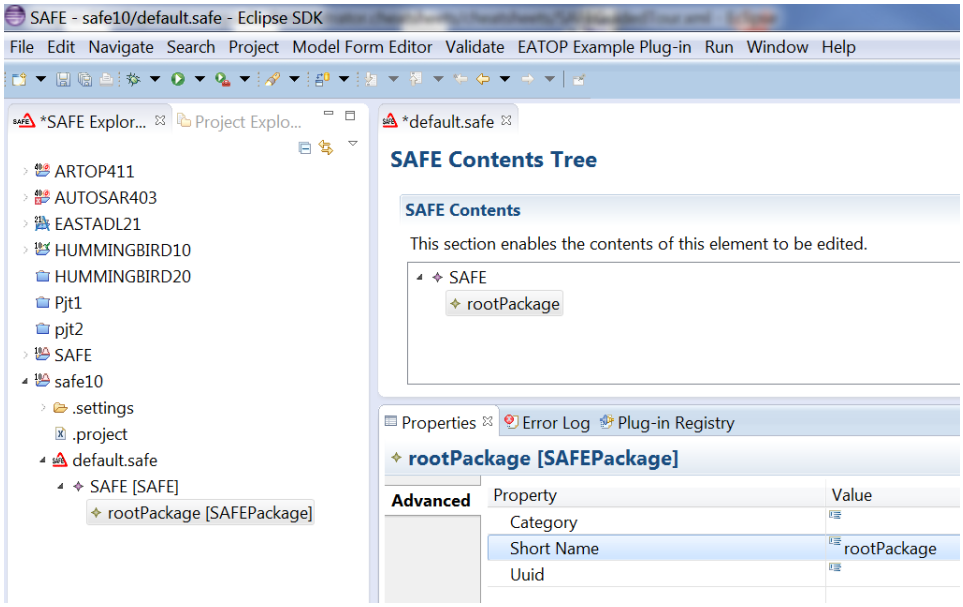
1. To open the editor in the main view, you have to select the SAFE element below the SAFE file in the "*SAFE explorer view*" and double click on it.



2. The editor is opened and you can see a tree representation of the elements it contains. You can explore project composition by expanding elements below the root object.

3.4.6 Edit SAFE object properties

1. You can modify any of the project elements by using the editor and the properties tab displayed below the view. Open the editor.
2. Expand project elements and select one element. You can see the properties of the element in a tab below the editor view.
3. Change the value of the field "*Short name*" in its properties. Once your modification is made you can observe that the name of the SAFE object changed in the editor view.

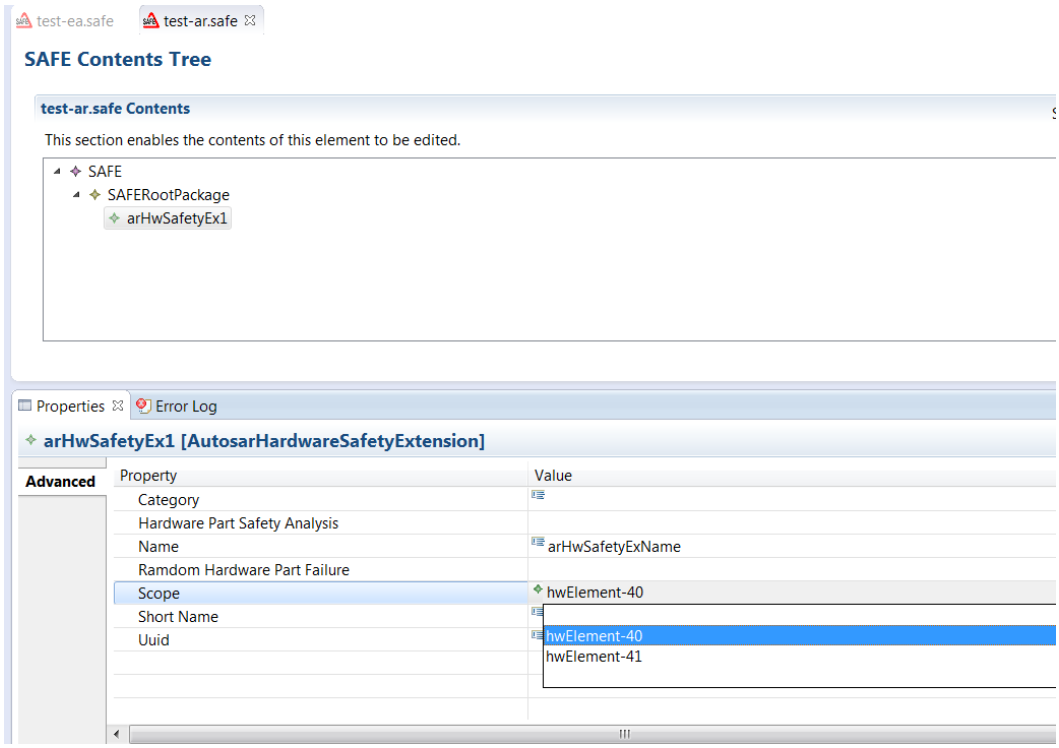


4. Each time you modify or add elements in the project a star appears in the top label of the editor view, which means that modification has been made but not saved. To save modifications simply click on the disk item in the top menu toolbar or use "*ctrl+s*" shortcut.

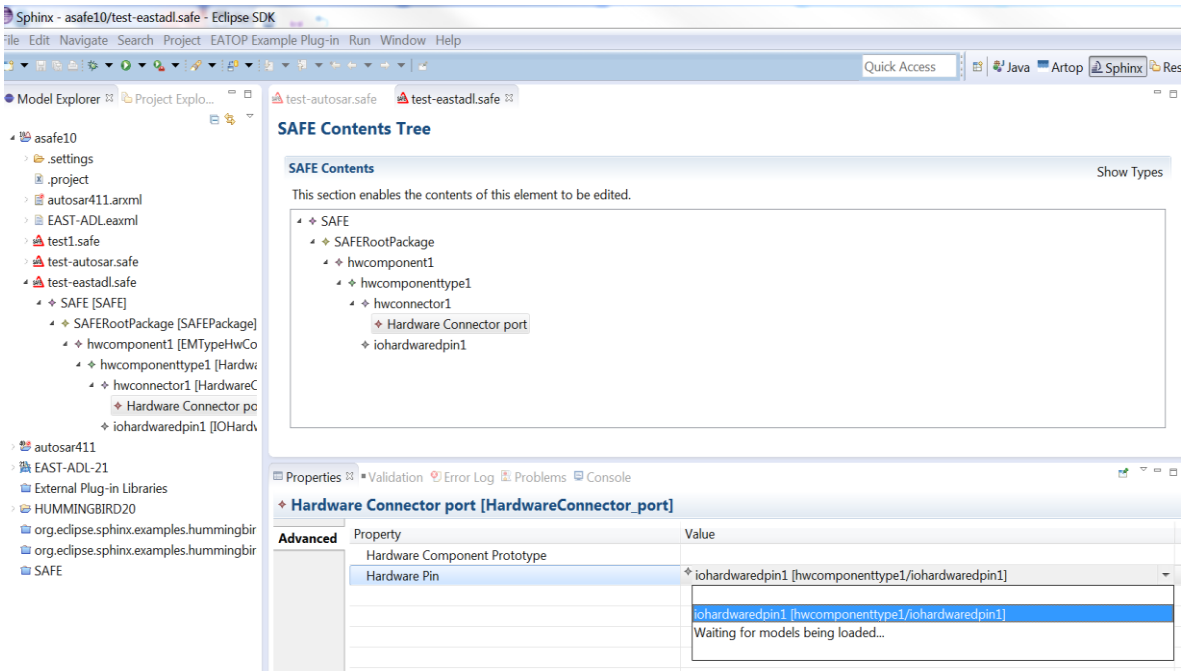
3.4.7 Reference to AUTOSAR and EAST-ADL

With ARTOP and EATOP installed along SAFE platform, you can use reference mechanism making references to EAST-ADL and AUTOSAR elements. Once serialized, SAFE file only contain a reference to AUTOSAR/EAST-ADL elements, and AUTOSAR/EAST-ADL data are located in standard AUTOSAR/EAST-ADL files.

1. Create a SAFE file.
2. Add AUTOSAR and EAST-ADL referable elements exactly the same way that you would do for other SAFE elements. Click on the attribute filed to get the available referenced models:
    1) If the available referenced models are already loaded, then you can directly choose the model that you would like to reference.



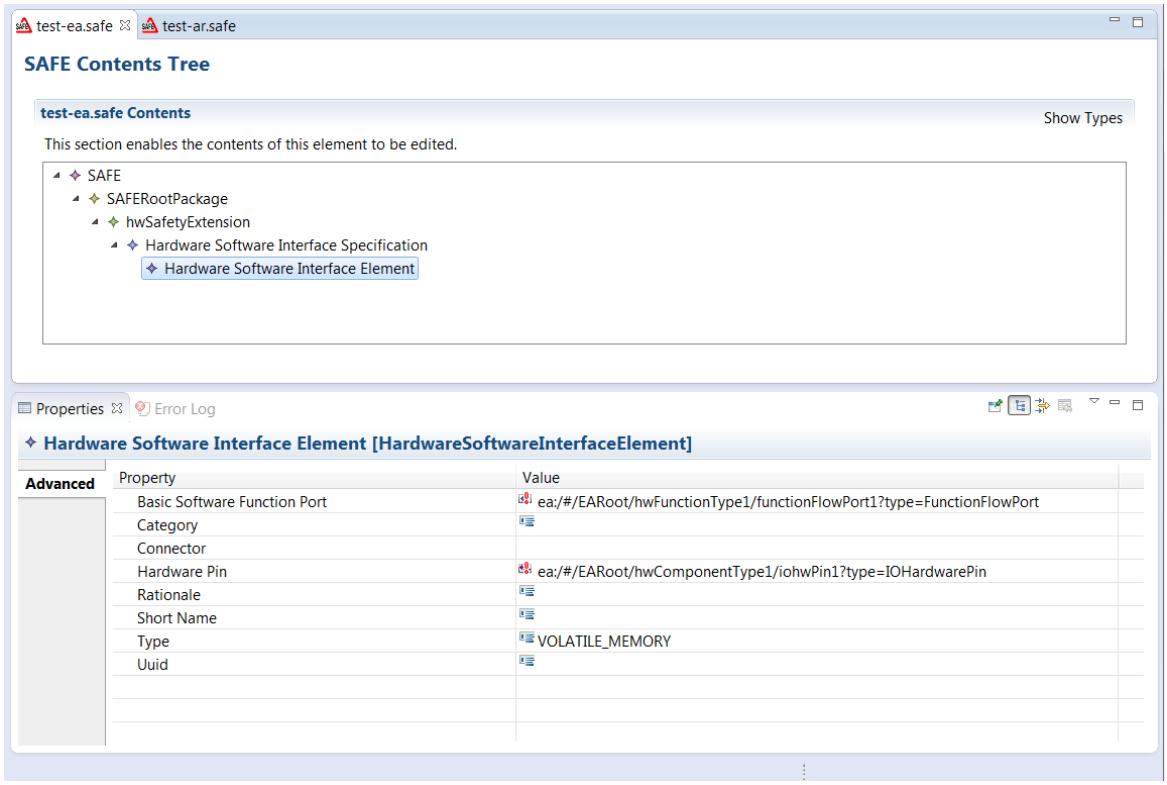
2) If the available referenced models are not yet loaded, then a message object is shown in the list: "*Waiting for models being loaded...*" The models are loaded asynchronously in background. It may take several minutes if the loaded models are huge. You can continue with other stuffs, and come back to refresh and get the reachable models some time later.



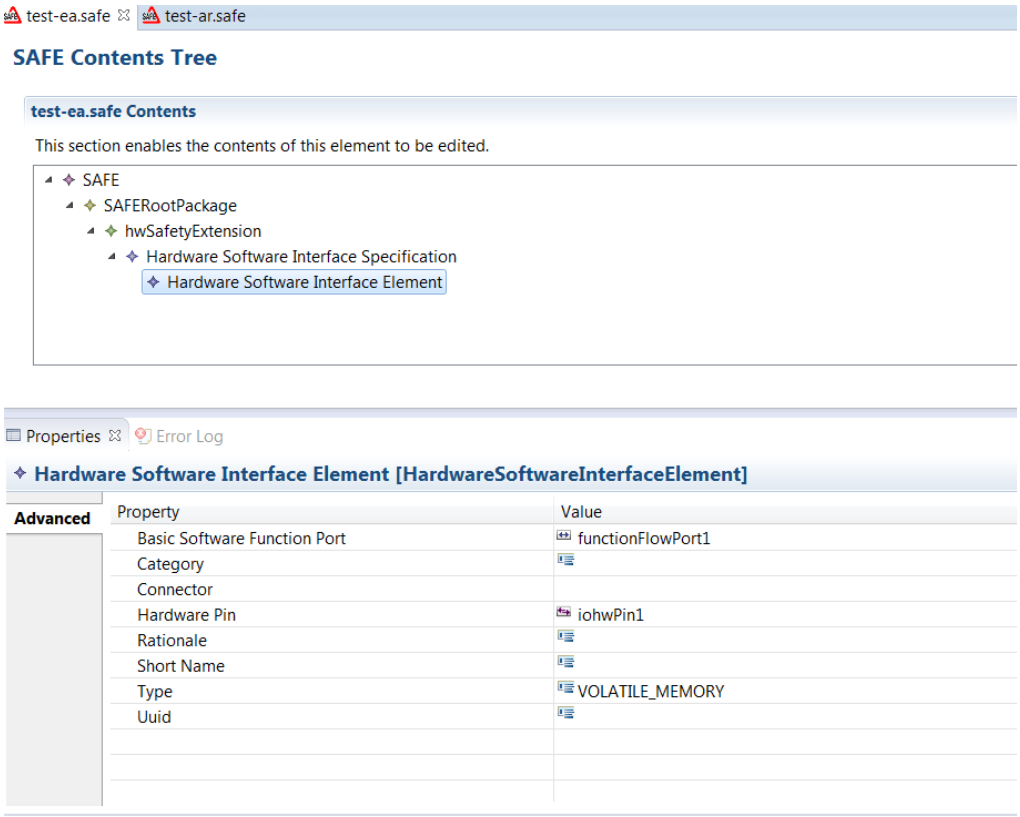
3. When opening a SAFE file that references to EAST-ADL or AUTOSAR elements, the reference proxy is resolved. The referenced EAST-ADL/AUTOSAR elements are located in a separated EAST-ADL or AUTOSAR file.

   If the associated referenced file has already been loaded, the reference proxy will be resolved immediately.

Otherwise, the file will be loaded asynchronously. For example, the EAST-ADL element "*iohwPin1*" is declared in an EAST-ADL file other than the current SAFE file, the reference proxy is not resolved immediately.



You have to refresh and get the reference resolved. For example, you can click on "*Hardware Software Interface Specification*", re-click on "*Hardware Software Interface Element",* then you will find that the "*iohwPin1*" is resolved.



4 SAFE MM Generator Tool

This SAFE MMGen tool is contributed by itemis France. It is refactored and enhanced to be adapted for the SAFE meta-model by itemis France. With the adaptations done in the WT3.5 meta-model, the tool generates an Ecore for the SAFE meta-model, as well as an XSD schema.

4.1 License

The code provided by itemis France is made available under the Eclipse public license EPL. The code provided by Continental Automotive which is part of this deliverable is made available under the terms of the Eclipse public license EPL.

4.2 Abbreviations

- Artop See AUTOSAR Tool Platform Website www.artop.org
- EA Enterprise Architect
- EAST-ADL See EAST-ADL association at www.east-adl.info
- SAFE See SAFE project Website www.safe-project.eu

4.3 Installation guide

4.3.1 Prerequisites

1. Windows XP, Vista, 7 (Linux or Mac is not supported)
2. Enterprise Architect 9.1 or greater
3. Eclipse 32Bit
4. Install the following software in your Eclipse:
   Nebula: http://download.eclipse.org/technology/nebula/snapshot

   Sphinx: http://download.eclipse.org/sphinx/updates/interim

   Xtext and Xtend 2.5.3:
   http://download.eclipse.org/modeling/tmf/xtext/updates/composite/releases

4.3.2 Source checkout

1. Clone the git://git.eclipse.org/gitroot/eatop/org.eclipse.eatop.git/metamodelgen EATOP MMGen Tool repository to your computer.
2. Add Enterprise Architect API in EATOP MMGen Tool:

   Copy *eaapi.jar* from your installation, e.g., *C:\Program Files (x86)\Sparx Systems\EA\Java API*, to:

   org.eclipse.eatop.eaadapter/lib

   org.eclipse.eatop.metamodelgen/lib

Copy *SSJavaCOM.dll* to:

org.eclipse.eatop.eaadapter

org.eclipse.eatop.eaadapter/lib

org.eclipse.eatop.metamodelgen

org.eclipse.eatop.metamodelgen/lib

3. Check out the plugins *org.safe.** from the SAFE MMGen *tools* repository into your workspace. The URL of source repository for SAFE MMGen tool is: https://safe.offis.de/svn/svndav/34_WP4_Technology_Platform/10_WT4_1_Meta_model_implementation/30_Development_artifacts/tools
4. Apply the target definition *4.3.target* in *org.safe.metamodelgen.tool.targetdefs*.

4.4 User guide

4.4.1 Step 1: Start MMGen

Execute "*Run Eclipse Application*". This operation opens a new Eclipse instance, including the MMGen tool plug-ins from your workspace. The further work is now done in the second Eclipse instance.

4.4.2 Step 2: Create a project for SAFE metamodel

In the second Eclipse instance:

1. Create a new Project, "*File->New->Other->Java Project*". Choose a name, e.g., "*org.safe.metamodel10*", and press "*finish*"
2. Create a new folder "*model*" within this project. Copy the SAFE metamodel Enterprise Architect EAP file into the "*model*" file. Rename the file to be "*safe10.eap*".

4.4.3 Step 3: Create SAFE Ecore model

1. Import the following projects from the SAFE *MMGen tools* repository or *MMGen.zip* into the MMGen's runtime workspace:

   org.eatop.stub

   org.artop.stub

2. Right click on the "*safe.eap*" file, select "*SAFE – Generate SAFE Ecore MM*", then select one of the following:

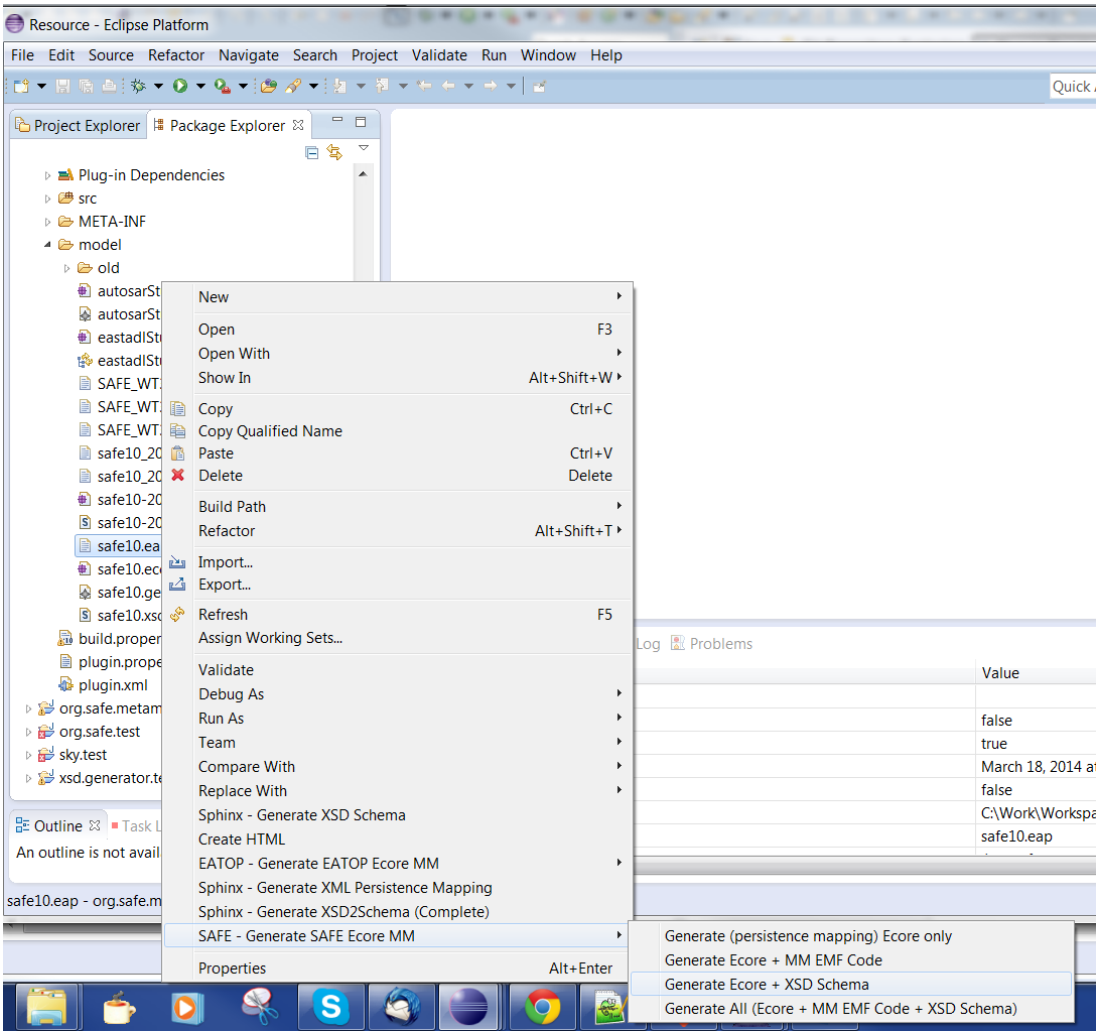
- "*Generate (persistence mapping only) Ecore only*": to generate persistence mapping Ecore only
- "*Generate Ecore + MM EMF Code*": to generate persistence mapping Ecore and the metamodel code implementation
- "*Generate Ecore + XSD Schema*": to generate persistence mapping Ecore and the XSD shcema

- "*Generate All (Ecore + MM EMF Code + XSD Schema)*": to generate persistence mapping Ecore, metamodel code implementation and the XSD schema.

  Note: Please make sure that the following resources should be removed before you start to generate the metamodel:

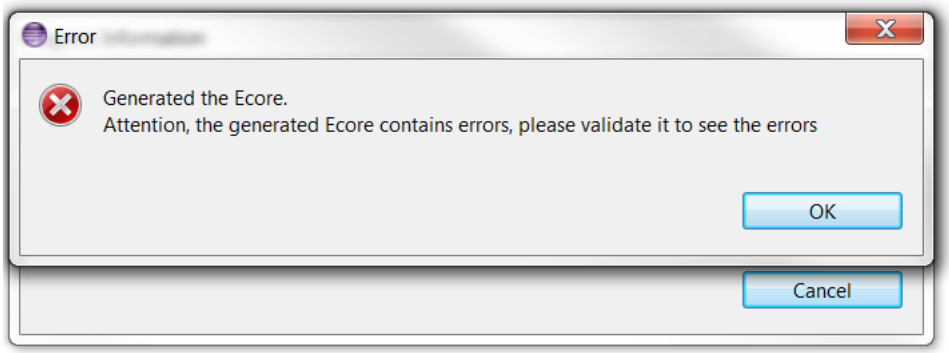
- *org.safe.metamodel10*: /src-gen, /META-INF, /model/safe.ecore, /model/safe10.genmodel, /model/safe10.xsd, /model/autosarStub.ecore, /model/autosarStub.genmodel, /model/eastadlStub.ecore, , /model/eastadlStub.genmodel, build.properties, plugin.properties, plugin.xml

- *org.safe.metamodel10.edit* plugin

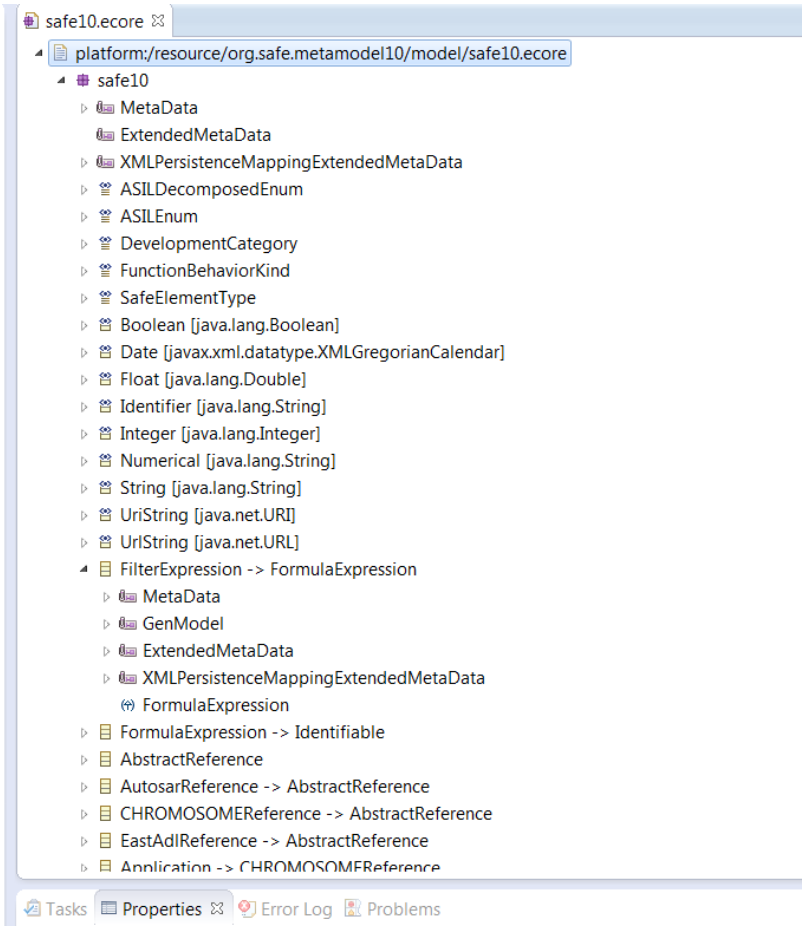


Wait until the Ecore metamodel generation has been finished.

Note: at the end of the generation, a validation of the Ecore is performed. Some validation failure error is pumped up. It does not affect the generated metamodel, the reason is caused by some default values not setting. Just click "*OK*".



The generated SAFE Ecore:



The generated SAFE XSD schema:


safe10.xsd
<xsd:schema xmlns:Q1="http://east-adl.info/2.1.12" xmlns:Q2="http://autosar.org/schema/r4.0" xmlns:SAFE="http://safe-prc
<xsd:import namespace="http://autosar.org/schema/r4.0" schemaLocation="platform:/plugin/org.artop.aal.autosar411/model
<xsd:import namespace="http://east-adl.info/2.1.12" schemaLocation="platform:/plugin/org.eatop.eel.eastadl2112/model/e
<xsd:element name="SAFE" type="SAFE:SAFE"/>
<xsd:complexType name="FILTER-EXPRESSION">
<xsd:sequence>
<xsd:element maxOccurs="1" minOccurs="1" name="SHORT-NAME" type="SAFE:IDENTIFIER"/>
<xsd:element maxOccurs="1" minOccurs="1" name="FORMULA" type="xsd:string"/>
</xsd:sequence>
<xsd:attribute name="CATEGORY" type="SAFE:IDENTIFIER" use="optional"/>
<xsd:attribute name="UUID" type="xsd:string" use="optional"/>
</xsd:complexType>
<xsd:complexType name="FORMULA-EXPRESSION">
<xsd:sequence>
<xsd:element maxOccurs="1" minOccurs="1" name="SHORT-NAME" type="SAFE:IDENTIFIER"/>
<xsd:element maxOccurs="1" minOccurs="1" name="FORMULA" type="xsd:string"/>
</xsd:sequence>
<xsd:attribute name="CATEGORY" type="SAFE:IDENTIFIER" use="optional"/>
<xsd:attribute name="UUID" type="xsd:string" use="optional"/>
</xsd:complexType>
<xsd:complexType name="ABSTRACT-REFERENCE">
Design
Source
Tasks
Properties
Error Log
Problems

5 References

[1] SAFE Requirements
https://safe.offis.de/svn/svndav/40_Deliverables/SAFE_D2.1.a/SAFE_D2.1.a.pdf

[2] SAFE Risk List
https://safe.offis.de/svn/svndav/10_Project_Management/SAFE_Plus-Minus-Risks.xlsx

[3] SAFE_D2.1.a-ISO-Part_2.pdf (Management of functional safety)

[4] SAFE_D2.1.a-ISO-Part_3.pdf (Concept Phase)

[5] SAFE_D2.1.a-ISO-Part_4.pdf (Product development at the system level)

[6] SAFE_D2.1.a-ISO-Part_5.pdf (Product development at the hardware level)

[7] SAFE_D2.1.a-ISO-Part_6.pdf (Product development at the software level)

[8] SAFE_D2.1.a-ISO-Part_7.pdf (Production and operation)

[9] SAFE_D2.1.a-ISO-Part_8.pdf (Supporting Processes)

[10] SAFE_D2.1.a-ISO-Part_9.pdf (Automotive Safety Integrity Level (ASIL)-oriented safety-oriented analysis

[11] ISO/FDIS 26262 parts 2-9: 2011.