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.b
SAFE Tool Platform User Guide

Due date of deliverable: 30.05.2013

Actual submission date: 30.05.2013

Start date of the project: 01/07/2011 **Duration:** 36 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 Table of contents

2 Introduction

The scope of SAFE WT 4.1 "Meta-model implementation" is the implementation of the SAFE meta-model. 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 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.
- An Ecore generated by a transformation initially created by Continental Automotive and enhanced by itemis France. The Ecore can run in an Eclipse platform.

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 XP, Vista, 7
2. Eclipse Juno 64 bit
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 simply apply the target definition: org.safe.targetdefs/4.2.target, which you can find in *SAFE_Tool_Platform.zip*.

   Or either you can use update site for installing:

   - Sphinx: http://download.eclipse.org/sphinx/updates/interim
   - ARTOP: https://www.artop.org/artifacts/nightly/artop-4.0/artop-Update/
   - EATOP: https://safe.offis.de/updates/eatop/updates/

   Note: For the current state, the EAST-ADL update site is not yet available.

   You could download the update site zip file from Eclipse lab: https://code.google.com/a/eclipselabs.org/p/eclipse-auto-iwg/downloads/detail?name=eatop-Update-1.0.0.v20130529-1615.zip&can=1&q=

   Extract to your local directory, e.g., C:\Work\temp, then apply the SAFE target definition to set as target (make sure to use your local directory instead of the original one in the target definition) or install by Eclipse new software installation.

3.2.2 Installation

Install SAFE Tool Platform with one of the following solutions. Please make sure that you have Sphinx, ARTOP and EAST-ADL installed in your Eclipse, if you choose solution 1 or solution 2.

1. Source code

   If you have already an Eclipse Juno 64 bit installed, you could extract *SAFE_Tool_Platform.zip* and import all the *org.safe.** plug-ins into your workspace.

2. (For SAFE project members only) SAFE Tool Platform update site

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

   For the current state, some problem of certification exists. The update site is not yet available. You can download the update site zip file *safe-Update-1.0.0.v20130529-1635.zip* from *Associated-Files* folder, then extract and install from local.

3. (For SAFE project members only) 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.

https://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

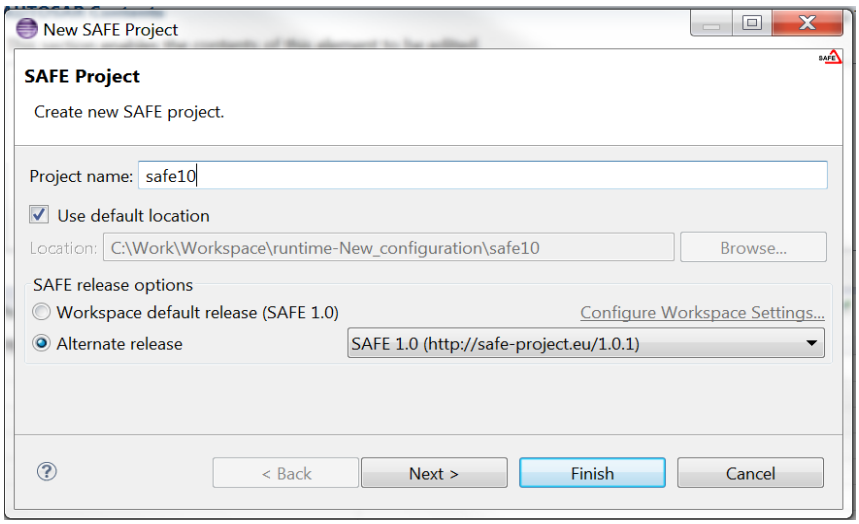
For more details guide tour, you could reference to the SAFE cheat sheet, "*Help -> Cheat sheets… -> SAFE Tool Development -> Guided Tour of the SAFE Technology Demonstrator*".

3.4.1 Open SAFE perspective

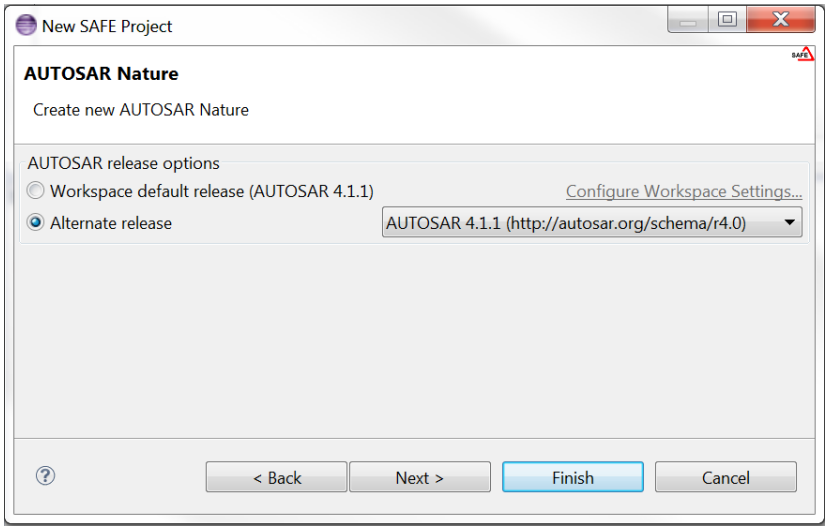
In the main menu, select "*Window* > *Open Perspective* > *Other",* and then select "*SAFE".*

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"*.



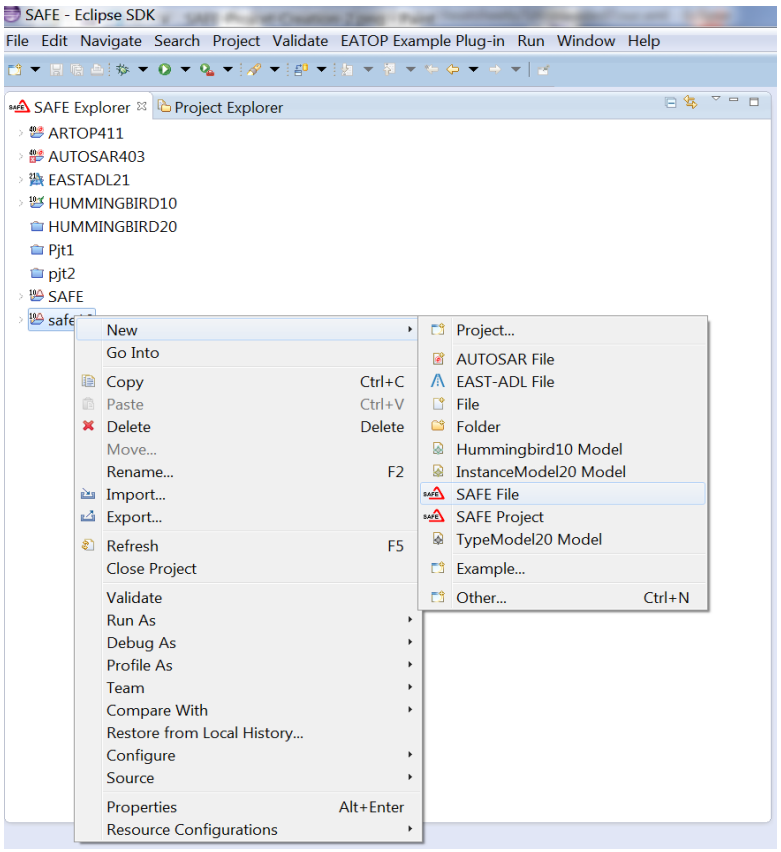
3. A second page is opened to select the "*AUTOSAR Nature*". You can set the AUTOSAR release by selecting "*Alternate Release*" and by choosing a release in the list or decide to use the "*Workspace default release*". Click "*Next*".



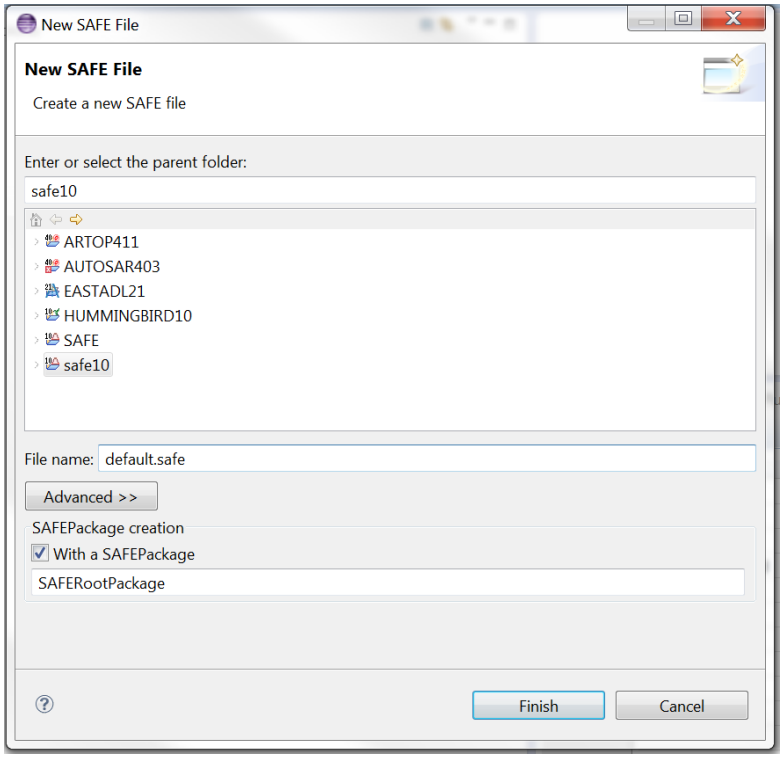
4. A third page is opened to select the "*EAST-ADL Nature*". Set the EAST-ADL release by selecting "*Alternate Release*" and by choosing a release in the list or decide to use the "*Workspace default release*".
5. Now you have set projects properties, and you can click on the "*Finish*" button to perform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

3.4.3 Create a SAFE file

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



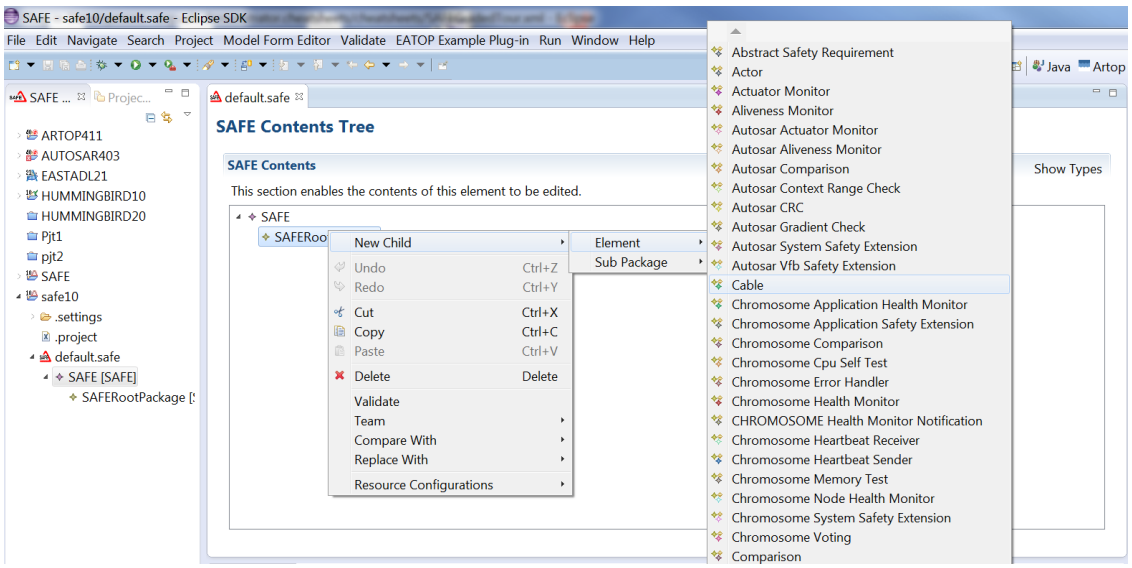
2. A wizard is opened. You can name the file or choose to keep the default name "*default.safe*". A "*SAFEPackage*" root can be automatically added into the file by selecting "*With a SAFEpackage*" in the "*SAFEPpackage Creation*" section.



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

3.4.4 Create a SAFE object

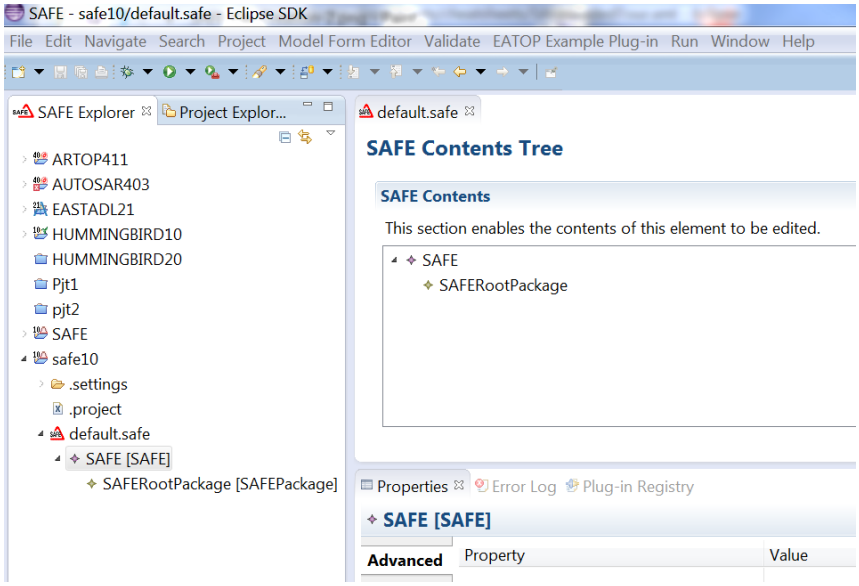
1. Open the "SAFE explorer view". Expand project composition and elements below SAFE file. Double click on SAFE element below SAFE file. The project editor view is now opened.



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

3.4.5 Explorer SAFE project

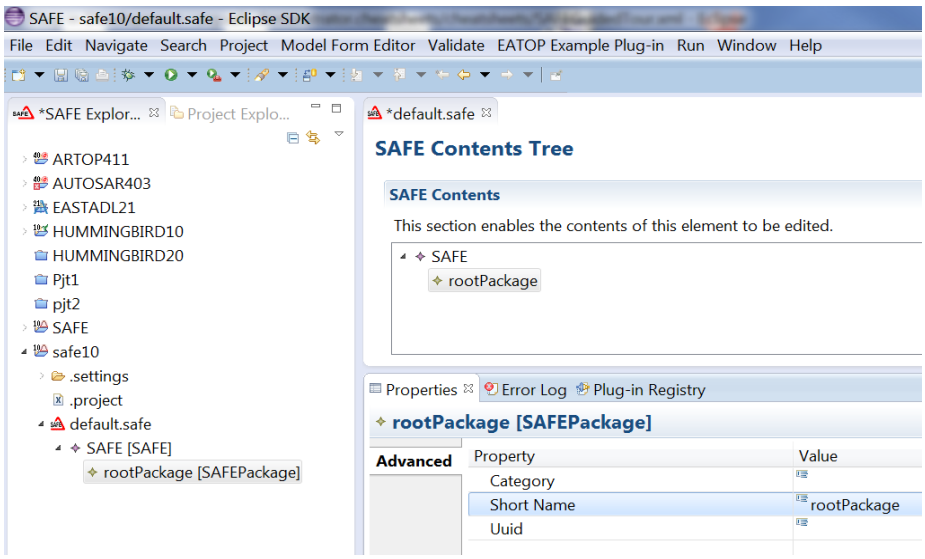
1. To open the editor in the main view, you have to select the SAFE element below the SAFE file in the “SAFE explorer view” and double click on it.



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.

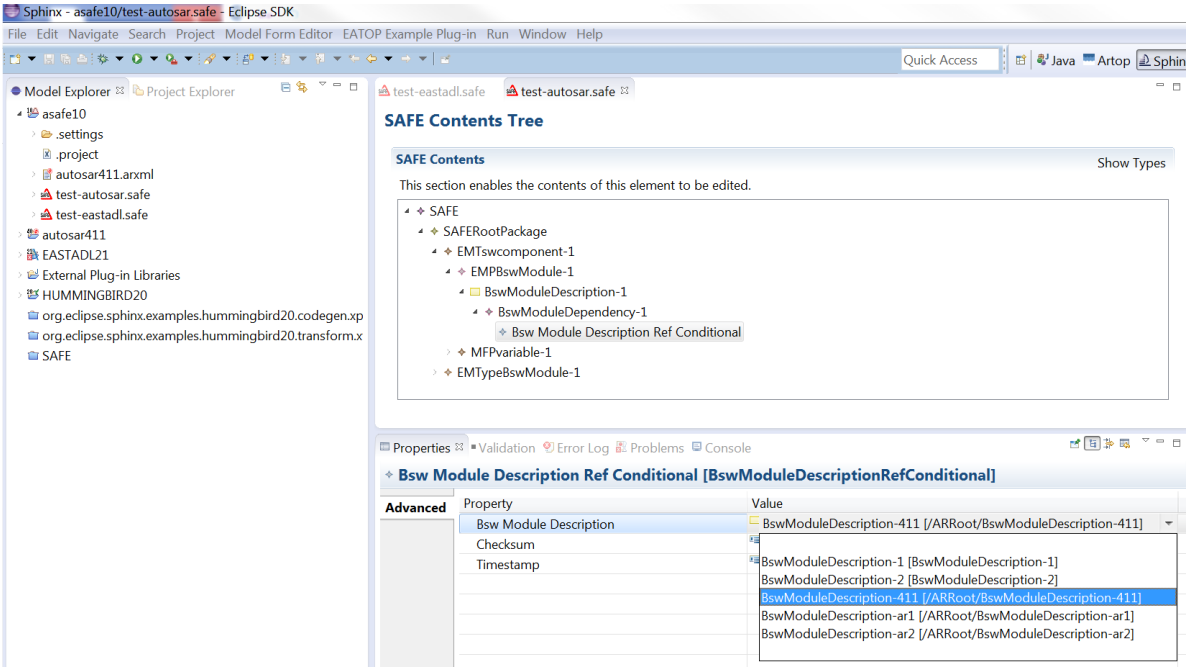


4. Each time you modify or add elements in the project a star appears in the top label of the editor view, which means that modification has been made but not saved. To save modifications simply click on the disk item in the top menu toolbar or use "*ctrl+s*" shortcut.

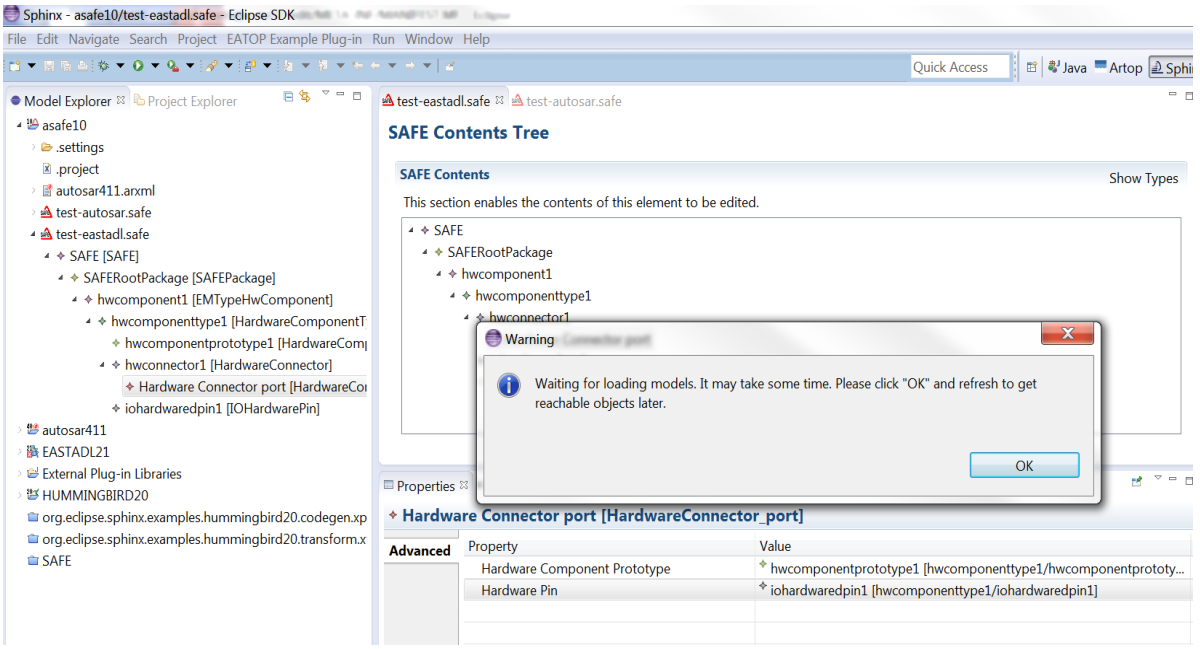
3.4.7 Reference to AUTOSAR and EAST-ADL

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.

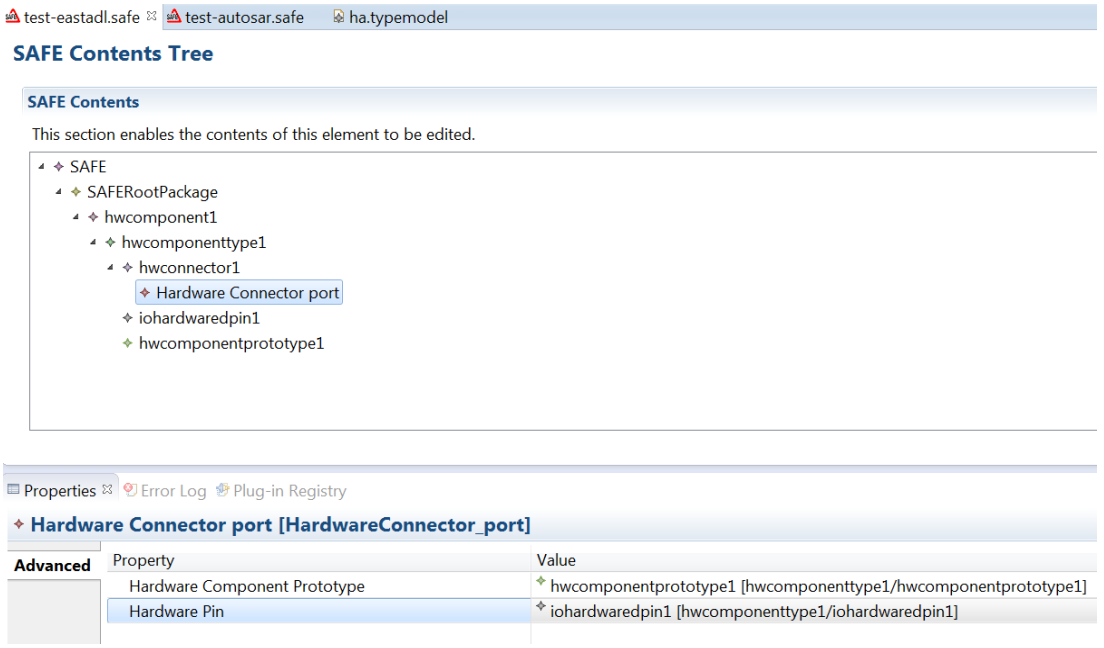


    2) If the available referenced models are not yet loaded, then a warning is pop up: "Waiting for loading models. It may take some time. Please click "OK" and refresh to get reachable objects later." The models are loaded asynchronously in background. Press "OK". It may take several minutes if the loaded 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:

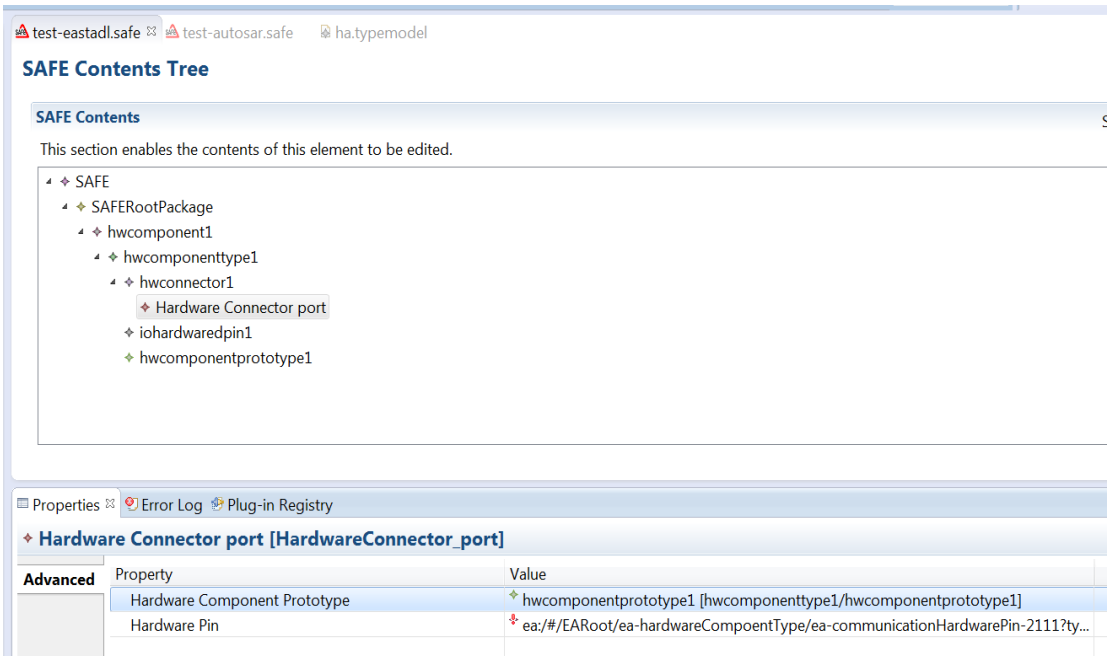
    1) If the referenced EAST-ADL/AUTOSAR elements are created in the same SAFE file, the reference proxy will be resolved exactly the same way that the SAFE elements would be resolved. The resolved object is displayed in the attribute field immediately. For example, the referenced Hardware Pin “iohardwaredpin1” is declared in the same SAFE file, the referenced object is resolved once you click on “Hardware Connector port”.



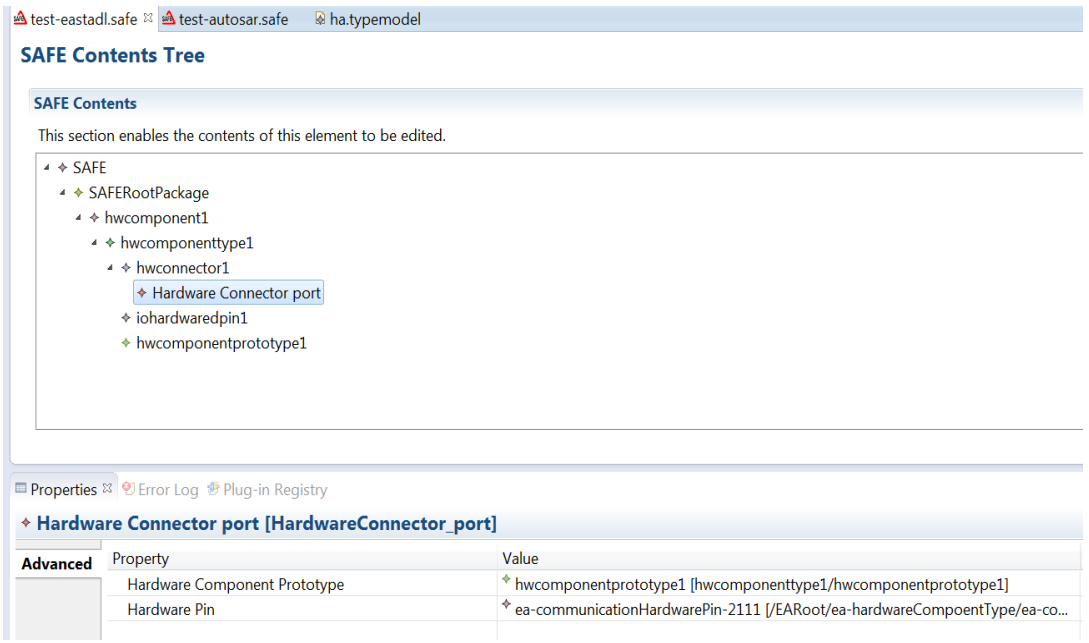
    2) If the referenced EAST-ADL/AUTOSAR elements are in a separated EAST-ADL or AUTOSAR file:

        If the associated referenced file has already been loaded, the reference proxy will be resolved immediately as above. Otherwise, the file will be loaded asynchronously. For example, the EAST-ADL element “ea-

communicationHardwarePin-2111" 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 "hwconnector1", re-click on "Hardware Connector port", then you will find that the "ea-communicationHardwarePin-2111" is resolved.



3.5 Known issues

There are still some issues to be resolved as listed in the known issue list: Associated-Files/SAFE_Tool_Platform_Known_Issur_List.doc

4 EAAdapter

This contribution is based on the EAAdapter for EAST-ADL created by Continental Automotive. It is enhanced and adjusted to be adapted for the meta-models from the SAFE project by itemis France. With the adaptations done in the WT3.5 meta-model, the transformation creates also an Ecore for the SAFE meta-model.

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

4.3.2 GIT checkout

1. Clone the https://code.google.com/a/eclipselabs.org/p/eclipse-auto-iwg.eatop/ EATOP repository to your computer.
2. Open Eclipse, "*File -> Import -> Existing Projects into Workspace",* select the EAAdapter directory and import all *eaadapter*.* plug-ins.

4.3.3 Extract zip file

1. Extract *EAAdapter.zip* to your computer.
2. Open Eclipse, "*File -> Import -> Existing Projects into Workspace",* select the EAAdapter directory and import all *eaadapter*.* plug-ins

4.4 User guide

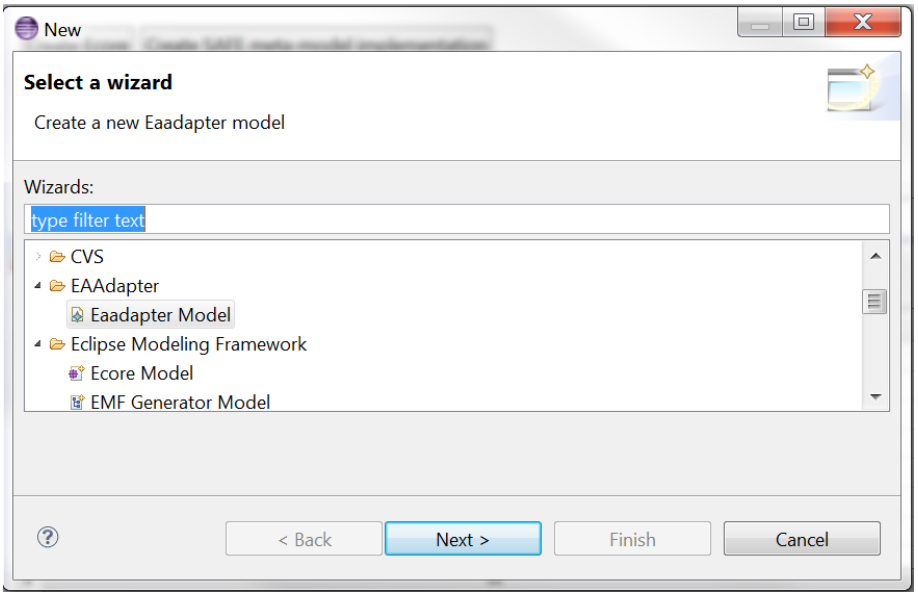
4.4.1 Step 1: Start EAAdapter

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

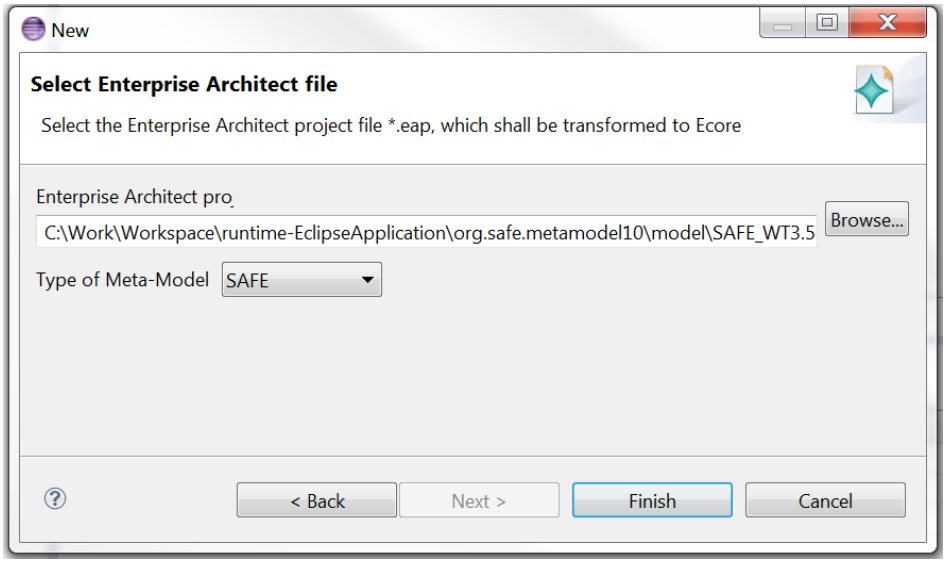
4.4.2 Step 2: Create a new EAAdapter file

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
3. Create a EAAdapter file in this "*model*" file, "*File->New->Other->EAAdapter->Eaadapter Model*". Select a file name, e.g., safe.eaadapter



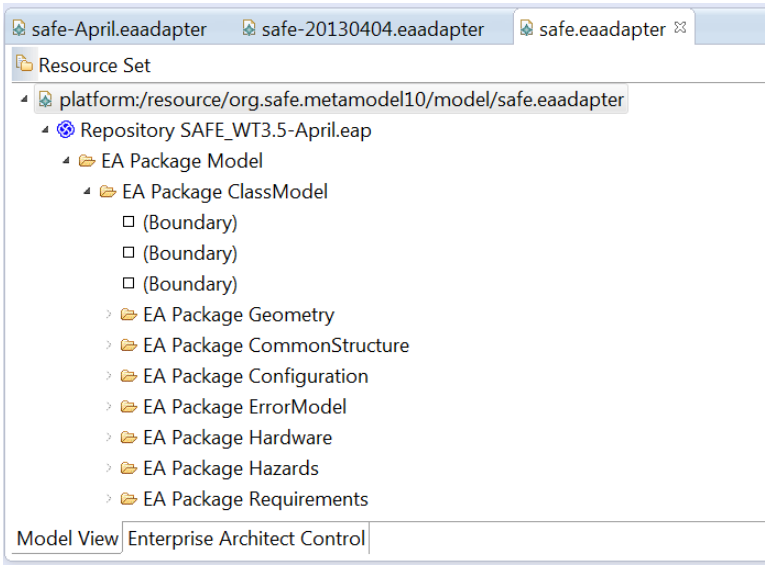
4. Select your "*Enterprise Architect file*" and choose the "*Type of Meta-Model*" of "*SAFE*".



5. Press "*finish*" to save the model.

4.4.3 Step 3: Load model

1. Double click the .eaadapter file you created. Switch to "*Enterprise Architect Control*" view, and select "*Load complete model*". This may take a few minutes, depending on the model you load.
2. Switch to "*Model View*". You can browse through the entire model.



3. Select the root element of the model you have loaded, and select "*Validate*". The "*Problems*" view of Eclipse shows you errors and warning associated to the model

4.4.4 Step 4: Create SAFE Ecore model

1. Import the following projects from the EATOP repository into the EAAdapter's runtime workspace:

   Org.eatop.stub

   Org.artop.stub

   Org.eatop.eel.common

   Org.eatop.eel.geastadl

   Org.eatop.eel.serialization

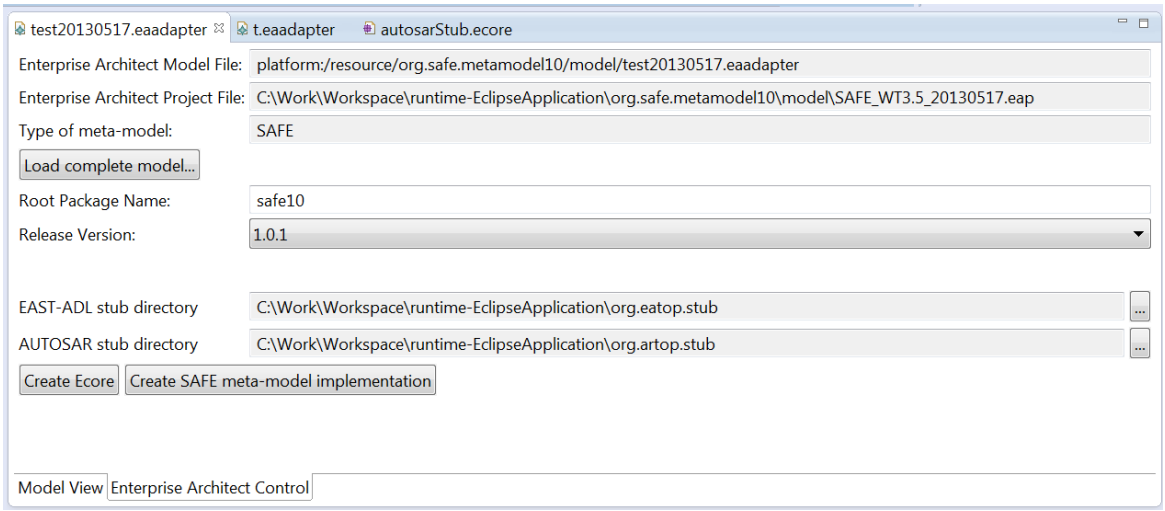
   Org.eatop.eel.eastadl21

   Org.eatop.eel.eastadl21.edit

   Org.eatop.eel.eastadl2111

   Org.eatop.eel.eastadl2111.edit

2. Switch back to the "*Enterprise Architect Control*" page. Set "*Root Package Name*" to "*safe10*". Select "*Release Version*" "*1.0.1*".
3. Set "EAST-ADL stub directory": select the org.eatop.stub project you imported earlier.



4. Set “AUTOSAR stub directory”: select the org.artop.stub project you imported earlier.
5. Select “*Create Ecore”* and specify the folder and filename.
6. Press “*save”* to start the transformation process.
7. When the Ecore has been generated, open it and select the root element, right click “*Validate”*. If no errors are shown, you're done!

4.4.5 Step 5: Create Meta-Model implementation for SAFE

1. Switch to “*Enterprise Architect Control”*, and set “*Root package Name”* to *“safe10”*. Select “Release Version” “1.0.1”.
2. Set “EAST-ADL stub directory” and “AUTOSAR stub directory” with the org.eatop.stub and org.artop.stub projects you imported earlier respectively.
3. Press “*Create SAFE meta-model implementation”* and specify the folder and filename. The generation process may take some minutes. When the ecore and implementation have been generated, if ecore is validated, and no errors are shown in the generated model and edit plugins, you’re done!

4.5 Known issues

There are still some issues to be resolved as listed in the known issue list: Associated-Files/SAFE_Tool_Platform_Known_Issur_List.doc

5 Conclusions and Discussion

To be added.

6 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.