MAENAD

Grant Agreement 260057

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

Report type — Deliverable D5.3.1

Report name — EATOP: An EAST-ADL Tool Platform for Eclipse

Dissemination level	**PU**
Status	**Intermediate**
Version number	**V1.0**
Date of preparation	**2012-08-31**

Authors

Editor	E-mail
Stefan Voget	Stefan.voget@continental-corporation.com

Authors	E-mail

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
0.1	2012-06-17	Initial version
0.2	2012-07-09	EATOP project description added
1.0	2012-08-31	First Release

Table of contents

1 Introduction

This document constitutes one part of Deliverable D5.3.1 on Tool adaptations for EAST-ADL. It introduces the Eclipse initiative “EATOP” which is an abbreviation for “EAST-ADL Tool Platform”.

EATOP is an Eclipse-based implementation of the EAST-ADL meta-model. EAST-ADL is a domain specific language established in the automotive industry.

The general outcome will be an extensible platform, which provides model editors and analysis plug-ins, which will support the EAST-ADL language for functional-, system-, software- and hardware-description. For the integration of already existing tools, which work on proprietary meta-models, model to model transformations will be provided.

With the work of EATOP the usage of EAST-ADL should be simplified and the number of already existing intermediate formats should be reduced.

2 Background

2.1 EAST-ADL

EAST-ADL is a domain specific language to model functional-, system-, software-, and hardware-architecture in the automotive domain. EAST-ADL has been created by the ITEA (www.itea2.org) funded project EAST/EEA. Further development has been done in two funded projects ATESST and ATESST2. Since 2010 a European funded project MAENAD (www.maenad.eu) maintains and extends the language with respect to electrified vehicles and safety development lifecycle modeling. For long-term maintenance and dissemination the EAST-ADL association (www.east-adl.info) has been founded. It maintains the meta-model definition and makes the latest version available to the public.

Until now, the EAST-ADL has been used in several tools:

- In the Eclipse MDT project Papyrus a UML2 profile of EAST-ADL has been provided
- TopCased initiative and the project OPEES – both are established in the airbus industry – are using EAST-ADL
- The commercial tool Volcano Vehicle Systems Architect (VSA) from Mentor Graphics provides editors for EAST-ADL modeling
- The commercial tool MetaEdit+ from MetaCase provides a meta-model implementation of EAST-ADL and generic editors of this meta-model
- The commercial tool System Weaver from Systemite provides editors for the EAST-ADL modeling
- Further tools exist from Arcticus, Symtavision, Rapita, RealTime@Work, Chronos
- …

EATOP shall support the work of the EAST-ADL association by providing an Eclipse based tool platform implementation of the EAST-ADL standard. Until now there had been several initiatives to create Eclipse based implementations of EAST-ADL. Goal of EATOP is to reconcile such initiatives in Eclipse.

2.2 Relationship to AUTOSAR

AUTOSAR is a design standard in the automotive industry. It focuses on the software architecture for electronic control units (ECU) in road vehicles, its deployment to networked ECUs in vehicles, and the configuration of the basic software in such ECUs. The AUTOSAR Tool Platform User Group (ARTOP) develops common base functionality for creating modeling tools supporting the AUTOSAR standard. ARTOP essentially encompasses implementations of the different releases of the AUTOSAR meta-model plus a rich set of services and components for managing and processing AUTOSAR models.

The EAST-ADL meta-model has a close relationship to the AUTOSAR meta-model as both address the model based development part in a process. East-ADL covers the function- and system-architecture and AUTOSAR acts with the system configuration and software architecture.

Therefore, EATOP will have a close relationship to the ARTOP (www.artop.org) initiative.

Figure 1 shows the principle how ARTOP generates the EMF- / Ecore-based meta-model implementation. Similar steps will be realized in EATOP for the EAST-ADL meta-model.

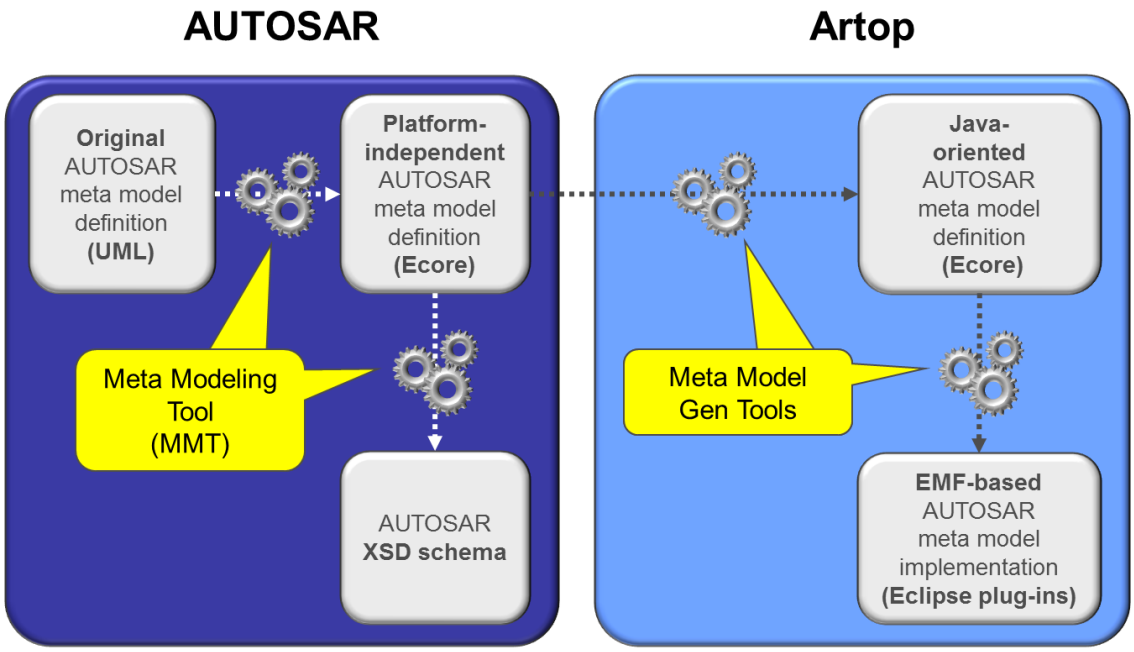

Figure 1: Principle relationship between meta-model and EMF-/Ecore-based implementation shown with the help of the example “AUTOSAR”.

3 Scope

EATOP is an infrastructure platform implementation of common base functionality for development tools that are used to design EAST-ADL compliant systems. This includes at least

- An EAST-ADL meta-model implementation supports several versions of EAST-ADL meta-model releases. These releases are published by the EAST-ADL association as an Enterprise Architect project and a XML schema. This input is taken to generate an Ecore-based meta-model implementation.
- Serialization is supported to enable file- and repository-based persistency of EAST-ADL models. Serializing and de-serializing EAST-ADL models to and from EAST-ADL XML files and databases.
- Refactoring contains a number of mechanisms to modify EAST-ADL models in a safe way.
- Workspace Management supports managing of EAST-ADL models, which are spread over more than one XML file.
- Model to model transformation between the different abstraction layers within the EAST-ADL meta-model.
- Further utilities simplify the handling of EAST-ADL models.
- To enable a seamless workflow in a development process, interfacing and model exchange with
    - Requirements Engineering (via ReqIF),
    - Software modeling (e.g. via AUTOSAR) and
    - HW modeling (e.g. via IPXACT) is enabled.
- Specific platform developments enable safety analysis and timing modeling.
- Consolidation of bridges between EATOP and Papyrus and the synchronization of EAST-ADL EMF API with the profile implementation.
- Variability management will be included; supporting both, the definition of variant-rich EAST-ADL models, as well as derivation of fully/partly configured instances of these models.
- Interoperability with domain independent abstractions of EAST-ADL like the CESAR reference technology platform (www.cesarproject.eu) or the MBAT reference technology platform (www.mbat-artemis.eu) will be addressed.

4 Description

4.1.1 Initial Components

EATOP will initially provide a set of EAST-ADL model editors. The components realizing these services already exist and will be contributed along with the creation of this project (see Code contributions).

- An EMF based meta-model implementation of EAST-ADL.
- A set of graphical design editors for the different abstraction layers of an EAST-ADL based model. These editors enable a developer to design an automotive system on vehicle level, through the functional analysis level down to the functional design level and hardware level. A package diagram is the root diagram and the entry point for each EAST-ADL model. On each abstraction level several editors are provided to support the type/prototype principle of the meta-model.

4.1.2 Future Components

As ARTOP is based on the Eclipse project Sphinx (www.eclipse.org/sphinx), EATOP will also be. Sphinx eases the creation of integrated modeling tool environments supporting individual or multiple modeling languages. Figure 2 presents the proposed architecture based on Sphinx.

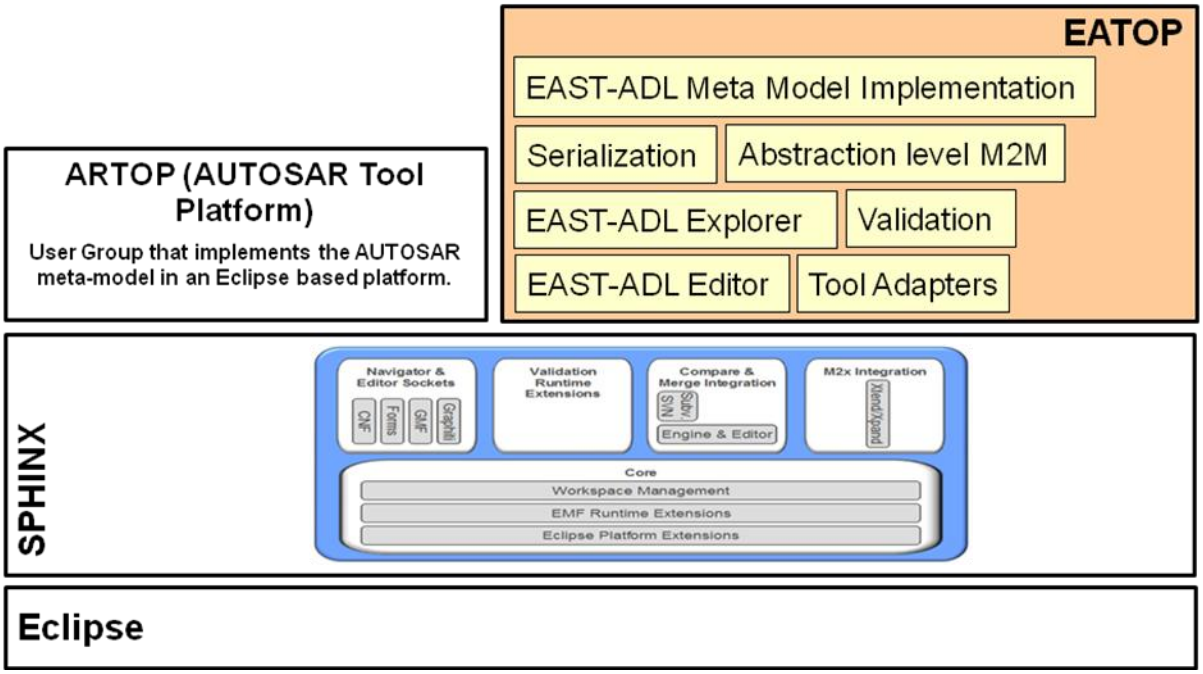

Figure 2: The architecture of EATOP resembles ARTOP and is based on Sphinx and Eclipse.

- The **EAST-ADL meta-model is implemented** using the Java programming language and the Eclipse Modeling Framework (EMF). The implementation supports several releases of the EAST-ADL meta-model. The goal is to provide meta-model implementations in EATOP close to the point in time when the EAST-ADL association releases a new meta-model version.
- The **Serialization component** provides file- and repository-based persistence for EAST-ADL models, serializing and de-serializing EAST-ADL models to and from EAST-ADL XML-files and databases.

- The **abstraction level M2M** component realizes a set of model to model converters that enable the generation of a skeleton on a more concrete meta-model level based on a model on a more abstract meta-model level.
- The **EAST-ADL Explorer** organizes the model elements and provides an adapted view of the generic Eclipse explorer.
- Based on the Sphinx validation runtime extensions a generic **validation** engine that minimizes the effort to create validation constraints for EAST-ADL models will be provided. A set of generic constraints will be included.
- The initial contributed **EAST-ADL editors** will be adapted to the architecture and enhanced in their feature set.
- Specific **tool adapters** enable the integration of the EATOP components into existing commercial or non-commercial tools. These adapters either enable import/export of models between the tools and EATOP or enable the integration using the Eclipse plug-in concept.

One core part of the work in EATOP is the translation of the EAST-ADL meta-model definition, which is provided in the form of a XML model authorized with Enterprise Architect, into an EMF-/Ecore-based implementation. The following picture summarizes the most important relationships.

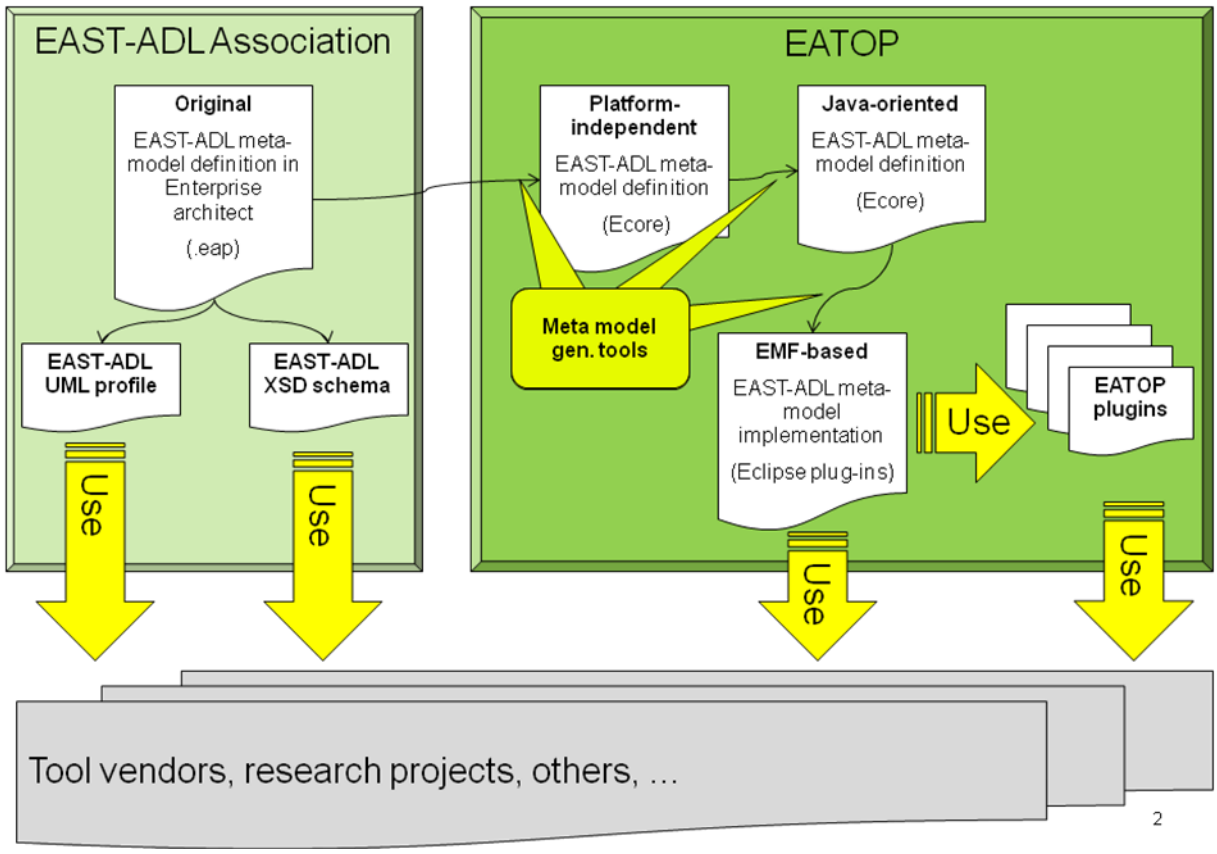

Figure 3: A tool chain for providing an EAST-ADL meta-model implementation ranging from language definition, via language development tools to user modeling tools. Intermediate artifacts are also indicated to show the language information needed to implement tools.

Figure 3 shows the different steps. These are described below.

1. Enterprise Architect is used to model the EAST-ADL meta-model in UML. This meta-model contains all elements of the language, including their attributes and relationships. Diagrams

are used to represent these elements. The resulting meta-model is in the Enterprise Architect project format (EAP). Enterprise Architect can also export to XMI format.

2. The EAST-ADL association already provides a schema definition and a UML2 profile which can directly be used. E.g. the Eclipse project Papyrus uses the UM2 profile.
3. EATOP generates an API to access and manipulate a model. This API can be used for building all kinds of applications on top of EAST-ADL. Examples are form-based and graphical EAST-ADL editors. EATOP will provide a set of such editors. These can be used as starting point for full-featured EAST-ADL editors in commercial tools or custom editors developed by the EAST-ADL tool users.

5 Relationship to other Eclipse Projects

- EATOP will be built on top of the Eclipse Platform and EMF.
- EATOP will also use complementary components of the Eclipse Modeling Project (EMP). EMF Transaction, EMF Validation and GMF are needed immediately. Others like Graphiti, M2T, M2M, EMF search, etc. are likely to be required later on.
- The architecture will be set up on top of Sphinx mainly using the workspace management, the validation runtime extensions, compare & merge integration, and the EMF runtime and Eclipse platform extensions. If reusable components for using in ARTOP as well as in EATOP will be identified, such components will be provided to Sphinx and used in EATOP.
- There are already several usages and demonstrator models of the EAST-ADL UML2 profile based solution in Papyrus. Services, examples and ideas from Papyrus will be included. Integration of the editors from Papyrus is planned. At least, model transformations will be provided.
- An interface to the Requirements Modeling Framework (RMF) - which is based on MDT – is intended to enable the requirements traceability through the modeling process steps.