

# ATESST

ATESST2 newsletter #3 2010

---



[Click here to download the tool CVM, a framework for compositional variability management aligned with the EAST-ADL2.](#)



[Click here to download the UML Modelling tool Papyrus and the EAST-ADL2 profile](#)

**Dear Colleague,**

in this ATESST2 newsletter, we will present how EAST-ADL2 supports modelling of requirements and verification & validation information. We will also discuss how requirements in an EAST-ADL2 model can be exchanged by means of the RIF standard (Requirements Interchange Format).

Further information about the ATESST2 project and the EAST-ADL2 language can be found at <http://www.atesst.org/>.

IMPORTANT: Please find an intermediate version of the EAST-ADL2 specification [here](#).

## In This Issue

- [Final Open Workshop of the ATESST2 Project, 21 June 2010, Frankfurt](#)
  - [Webinar: EAST-ADL2 Requirements Management Concept Presentation](#)
  - [EAST-ADL2 Requirements Management—Introduction](#)
  - [Basic Requirements Relations](#)
  - [Requirements and AUTOSAR](#)
  - [Requirements, V&V Modelling, and Exchange](#)
  - [Exchanging Requirements Specifications](#)
- 

## Final Open Workshop of the ATESST2 Project, 21 June 2010, Frankfurt

We will present the major results of the project and have a tools and demonstrator session. Participation to the workshop is free of charge and open to everyone interested. Please confirm your participation by email [atesst-coordinator@vtec.se](mailto:atesst-coordinator@vtec.se)

### Agenda

08:30 Welcome coffee

09:00 Seminar session

12:00 Lunch

13:00 Break-out sessions: Small presentations and demonstrations on selected topics:

17:00 End of Day

### Venue

Mercure, FRANKFURT ESCHBORN OST

Helfmann-Park 6

D - 65760 Eschborn

Tel +49(0)6196-901-0

Fax +49(0)6196-901-900

[h0491@accor.com](mailto:h0491@accor.com)



[www.mercure-frankfurteschborn-ost.com](http://www.mercure-frankfurteschborn-ost.com)

Topics in the seminar session include EAST-ADL2 Overview, relation to AUTOSAR, methodology, tooling, modelling concepts for variability, safety, requirements and V&V, cooperative systems and environment.

Modelling example and in-depth presentations will concern requirements modelling, cooperative systems modelling, safety and variability.

Tooling examples will include Simulink exchange, fault-tree analysis, feature modelling and variability resolution, AUTOSAR synthesis, timing analysis and cost-dependability-timing optimization.

---

### Webinar: EAST-ADL2 Requirements Management Concept Presentation

[Click here to download a presentation with an overview of EAST-ADL2 requirements management](#)

You are invited to take part in a web-based overview presentation of the EAST-ADL2 requirements management. Please send an email to [owner-sig-adl@vtec.volvo.se](mailto:owner-sig-adl@vtec.volvo.se) and we will come back with a meeting details.

---

#### EAST-ADL2 Spotlight

*ATESST2 defines EAST-ADL2 as a domain-specific language using meta-modelling constructs such as classes, attributes, and relationships.*

*The project also implements a UML2 profile which is used in UML2 tools for user modelling.*

*The EAST-ADL2 definition also serves as the specification for implementation in domain-specific tools.*

**EAST-ADL2**

#### EAST-ADL2 Requirements Management—Introduction

In order to better support the development of automotive embedded systems, EAST-ADL2 does not only include means to create analysis and design models of the system to be developed (at varying abstraction levels), but also language means

- to specify required properties of the system (at varying degrees of abstraction);
- to trace requirements between system refinement and system decomposition levels;
- to require satisfaction of requirements for system components;
- to refine the specification of requirements by behavioural models; and
- to verify requirements by verification and validation activities.

EAST-ADL2 does not start from scratch but closely aligns its requirements concepts to SysML [[www.sysml.org](http://www.sysml.org)]. However, extensions and adjustments are made to these proposals based on the needs of the automotive application domain.

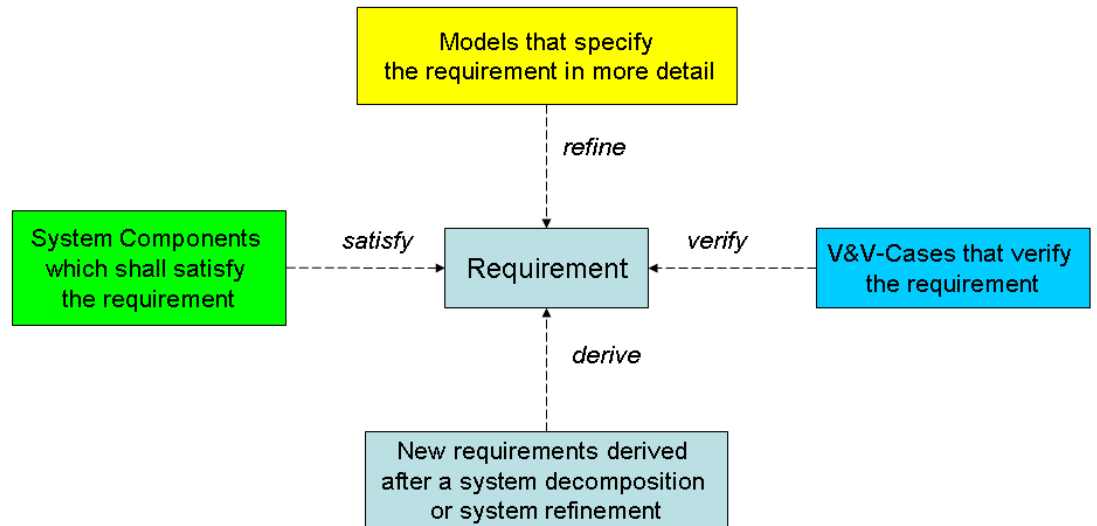
Previous newsletters have discussed the abstraction levels and their content, see [www.atesst.org](http://www.atesst.org) → [Newsletter](#).

---

## Basic Requirements Relations

Four requirement relations are used in EAST-ADL2 (Figure 1). Requirements may be used to textually specify required properties of the system to be developed. The textual specification of a requirement can be refined by relating models to requirements through the **refine** relation.

Please click on the figure to enlarge.



**Figure 1: The basic requirements relations “refine”, “verify”, “derive”, and “satisfy” (adopted from SysML).**

Requirements are refined into more detailed requirements after a system refinement or a system decomposition step. This concept is supported by the **derived** requirement relationship to allow a traceability from any requirement to its original. Such requirement hierarchy will thus span over the architectural decomposition of the system, but also across abstraction levels.

Requirements apply to the various system components which are introduced to satisfy them. This concept is modelled using the **satisfy** relation. Requirements’ traceability is completed by the verification and validation relationships. These define how the verification and validation activities, such as the testing activities fit criteria explicit for the verification and validation goals and how its associated verification and validation cases, such as test cases, verify the requirement. This is modelled using the **verify** relationship along with constructs that capture and relate V&V to the requirement and system components (Figure 2).

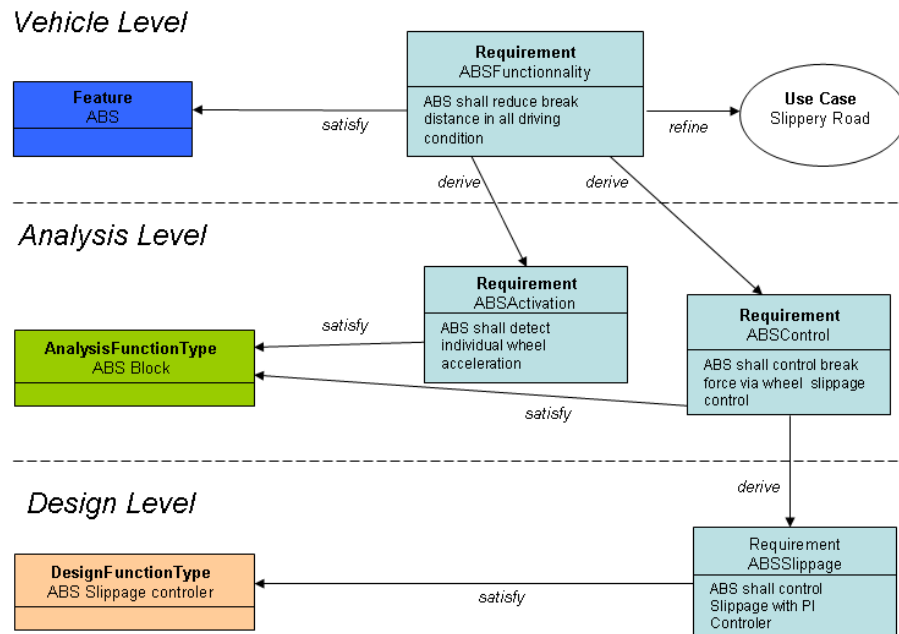


Figure 2: Requirements tracing and linking to system components.

## AUTOSAR Spotlight

*AUTOSAR defines a software architecture platform by standardization of its infra-structure and a communication layer suitable for distributed systems.*

*The standard also defines description means for the execution platform including control units, network topology, I/O, and middleware and application software components.*

*The **platform** and the description means make it possible to integrate software from different suppliers on the same hardware. Reuse is favoured and dependencies between application software and hardware are avoided.*



## Requirements, V&V Modelling, and Exchange

In order to support the development of dependable systems, the EAST-ADL2 offers detailed means to explicitly model central artefacts of verification and validation activities and to relate these artefacts to requirements.

This allows for explicitly and continuously planning, tracking, updating, and managing of important V&V activities and their impact on the system in parallel to the development of the system.

In EAST-ADL2 a V&V case will take different forms, depending on the kind of the V&V activity performed, e.g., safety analysis, specification, design, or implementation review, functional analysis by simulation, SIL-testing, HIL-testing, or vehicle testing. In general, it consists of a number of V&V procedures to be applied to the target object, which are recorded close to the modelling artefacts. Each V&V case may contribute to a dedicated aspect of the verification of some requirement and EAST-ADL2 allows documenting this relationship by means of the “verifies” relation (**Error! Reference source not found.**). These relations between model, requirements, and verification & validation information provides means to ensure all requirements are satisfied in the configuration that correspond to the produced vehicle. The context of function reuse will also benefit from these formal associations because consistent functional definition with requirements and V&V information may be defined.

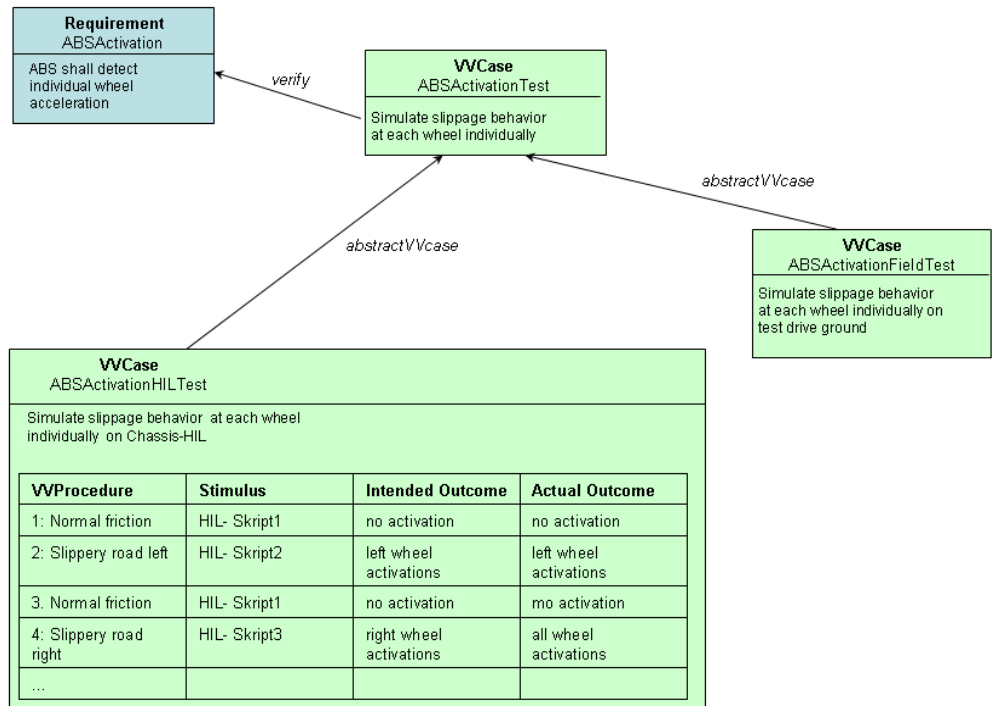


Figure 3. The figure illustrates the verification of a requirement.

## Exchanging Requirements Specifications

It is typical for the automotive domain that the engineering of embedded systems is done across several stakeholders, for example a vehicle manufacturer and tier-1 supplier may share the development effort of certain functionality. One typical use case is that a manufacturer sets up an initial requirements specification and shares certain parts of it with several suppliers to adjust the specification according to changes or comments made by the suppliers. That is why the exchange of requirements specifications is not only unidirectional but also a bidirectional flow of data (in a round-trip manner). During such round-trip exchanges, requirements and also the structuring of requirements may be exchanged between a manufacturer and a supplier and the receiver of such a changed specification (manufacturer or supplier) may adjust its current specification according to the changes made before.

To support the exchange of requirements specifications across several partners, the German automotive organization Herstellerinitiative Software (HIS) started a definition of a tool independent representation of such specifications [<http://www.automotive-his.de/rif>]. The outcome of this definition is the XML based Requirements Interchange Format (RIF) which is specified as an XML Schema (XSD) that is including all the concepts for representing requirements specifications including the structuring of requirements and relations between them. So, using RIF for exchanging requirements specifications, tools such as IBM Rational DOORS [[www.ibm.com/software/awdtools/doors](http://www.ibm.com/software/awdtools/doors)], Polarion Requirements [[www.polarion.com/products/requirements](http://www.polarion.com/products/requirements)], Borland Caliber RM [[www.borland.com/de/products/caliber/rm.html](http://www.borland.com/de/products/caliber/rm.html)] and also a typical EAST-ADL2 tool suite should support this exchange format. To keep track of certain requirement elements (remember the round trip manner mentioned above) RIF gives all its elements a globally unique identifier (UUID, [<http://tools.ietf.org/html/rfc4122>]).

The EAST-ADL2 tool suite supports three use cases for exchanging requirements specifications via RIF:

1. Instantiate a new EAST-ADL2 model based on a given RIF file. The new model will then contain an initial requirements specification.
2. Import a requirements specification of a given RIF file into an already existing EAST-ADL2 model. Here, requirements specification elements will be newly created, changed or deleted.
3. Export a requirements specification out of an already existing EAST-ADL2 model into a newly created RIF file.

## Exchanging Requirements Specifications

It is typical for the automotive domain that the engineering of embedded systems is done across several stakeholders, for example a vehicle manufacturer and tier-1 supplier may share the development effort of certain functionality. One typical use case is that a manufacturer sets up an initial requirements specification and shares certain parts of it with several suppliers to adjust the specification according to changes or comments made by the suppliers. That is why the exchange of requirements specifications is not only unidirectional but also a bidirectional flow of data (in a round-trip manner). During such round-trip exchanges, requirements and also the structuring of requirements may be exchanged between a manufacturer and a supplier and the receiver of such a changed specification (manufacturer or supplier) may adjust its current specification according to the changes made before.

To support the exchange of requirements specifications across several partners, the German automotive organization Herstellerinitiative Software (HIS) started a definition of a tool independent representation of such specifications [<http://www.automotive-his.de/rif>]. The outcome of this definition is the XML based Requirements Interchange Format (RIF) which is specified as an XML Schema (XSD) that is including all the concepts for representing requirements specifications including the structuring of requirements and relations between them. So, using RIF for exchanging requirements specifications, tools such as IBM Rational DOORS [[www.ibm.com/software/awdtools/doors](http://www.ibm.com/software/awdtools/doors)], Polarion Requirements [[www.polarion.com/products/requirements](http://www.polarion.com/products/requirements)], Borland Caliber RM [[www.borland.com/de/products/caliber/rm.html](http://www.borland.com/de/products/caliber/rm.html)] and also a typical EAST-ADL2 tool suite should support this exchange format. To keep track of certain requirement elements (remember the round trip manner mentioned above) RIF gives all its elements a globally unique identifier (UUID, [<http://tools.ietf.org/html/rfc4122>]).

The EAST-ADL2 tool suite supports three use cases for exchanging requirements specifications via RIF:

4. Instantiate a new EAST-ADL2 model based on a given RIF file. The new model will then contain an initial requirements specification.
5. Import a requirements specification of a given RIF file into an already existing EAST-ADL2 model. Here, requirements specification elements will be newly created, changed or deleted.
6. Export a requirements specification out of an already existing EAST-ADL2 model into a newly created RIF file.

