

“Advancing Traffic Efficiency and Safety
through Software Technology”

ATESST2 Tools and Metamodeling

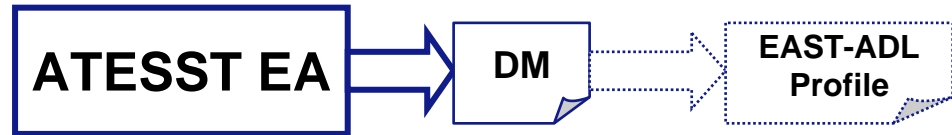
ATESST2 Concept presentation 2010 Q2



Contents

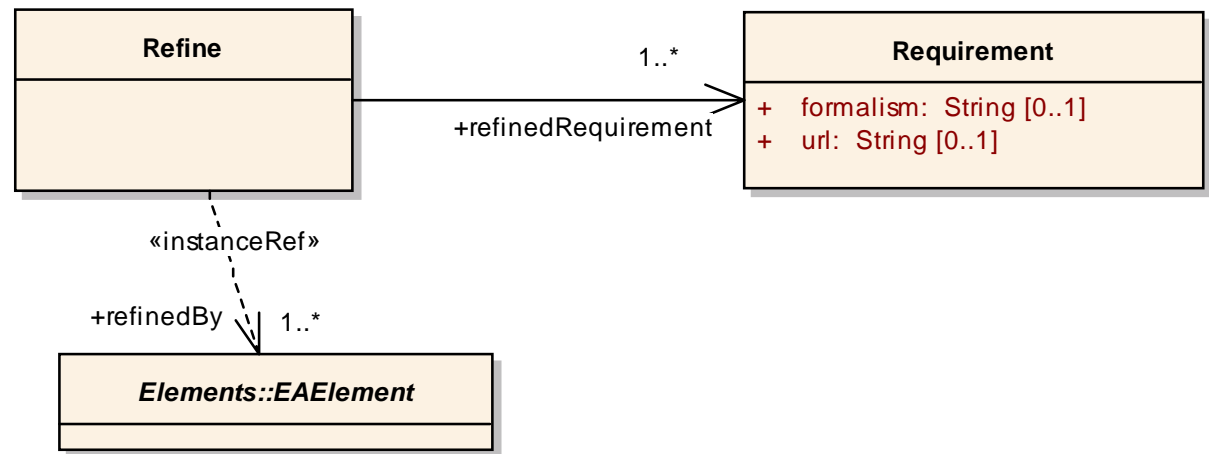
- **Domain Model for EAST-ADL**
- **Relations between Domain Model, Profile and UML2**
- **UML2 Profile for EAST-ADL**
- **EAST-ADL Profile relations**
- **The ATESST2 Tool Papyrus, with plugins**
- **Applying the profile**
- **Other ways to use the Domain Model**

Domain Model



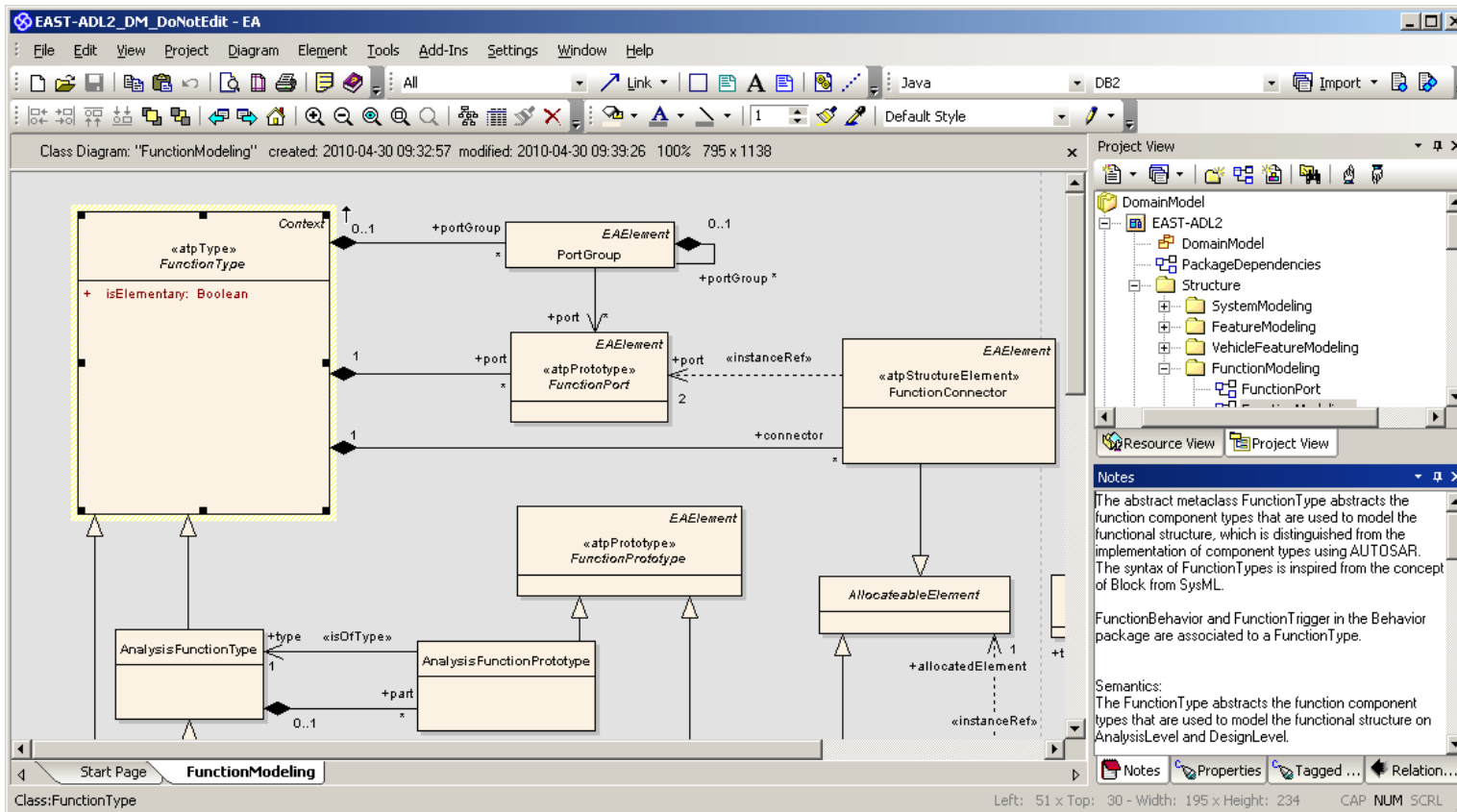
Enterprise Architect UML tool with class diagrams

- Description
- Semantics
- Attributes
- Associations
- Constraints
- Notation

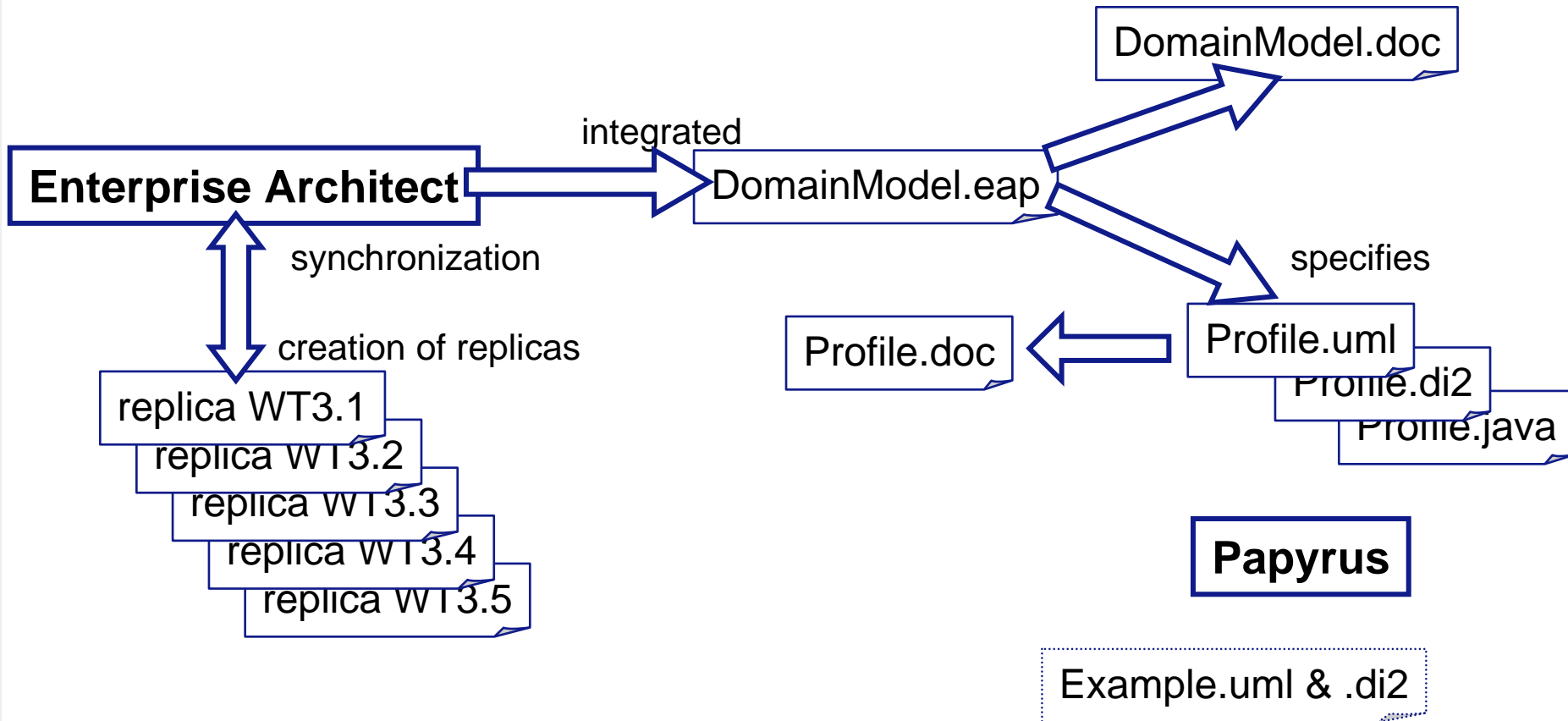


Enterprise Architect

EAST-ADL Domain Model



ATESST2 WT4.1 EAST-ADL Synchronization



Domain Model vs. Metamodel vs. Profile

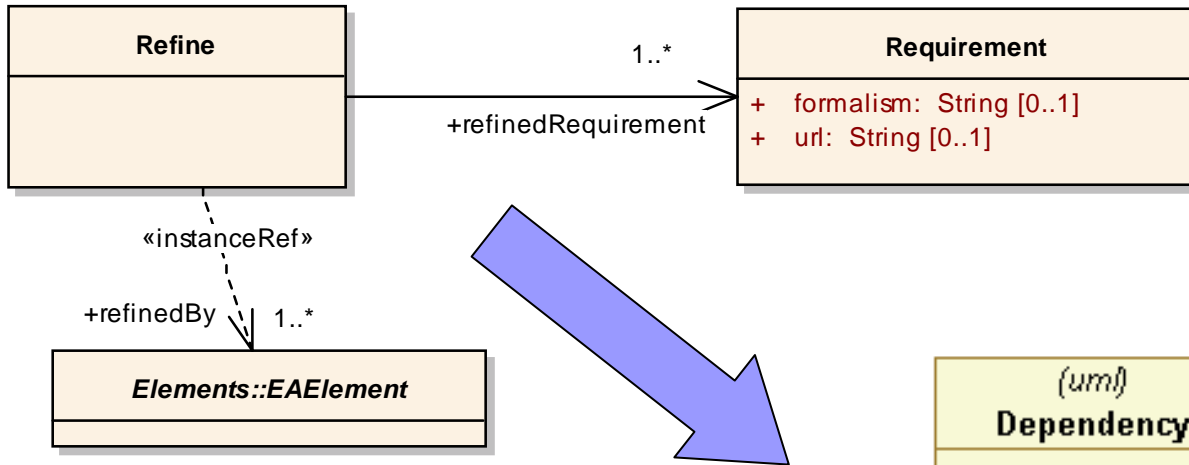
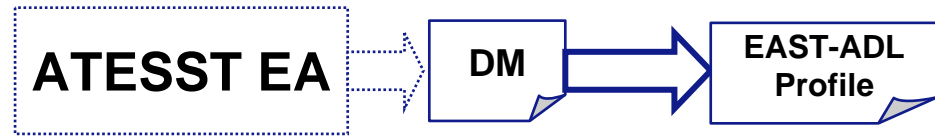
An AUTOSAR aligned Metamodel is already a Domain Model adapted for a specific purpose (XML exchange format). For example the details on the instanceRef may be excluded from the Domain view.

However the EAST-ADL Domain Model already has a lot of details regarding prototypes etc. In ATESST1 we gave hints to Profile definition in the EA Domain Model.

The Profile is also for a specific purpose (UML tool) and relies on the available UML elements.

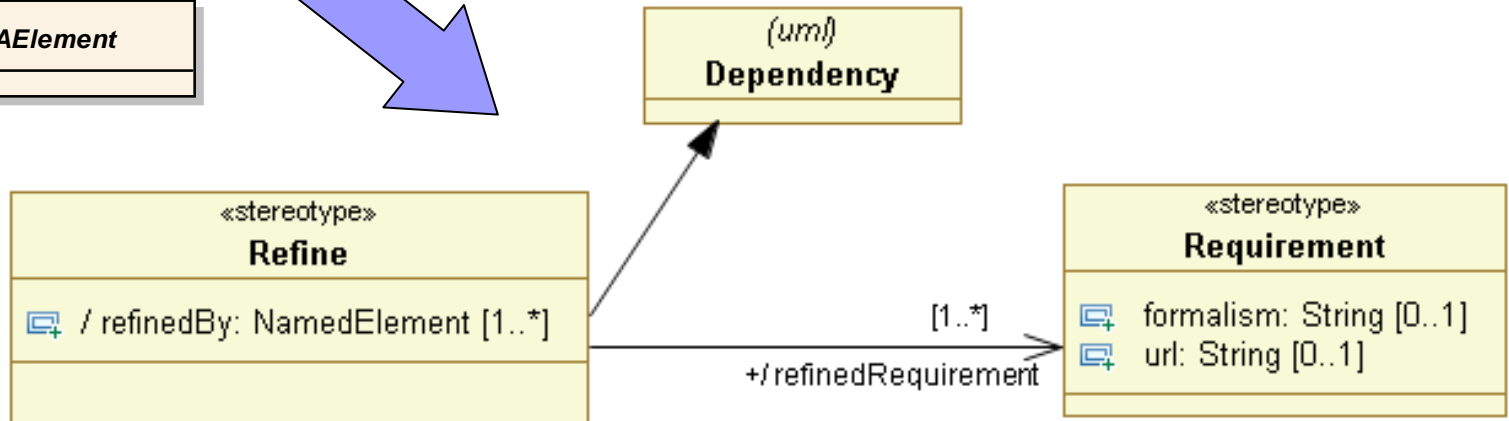
UML2 Profile

Domain Model



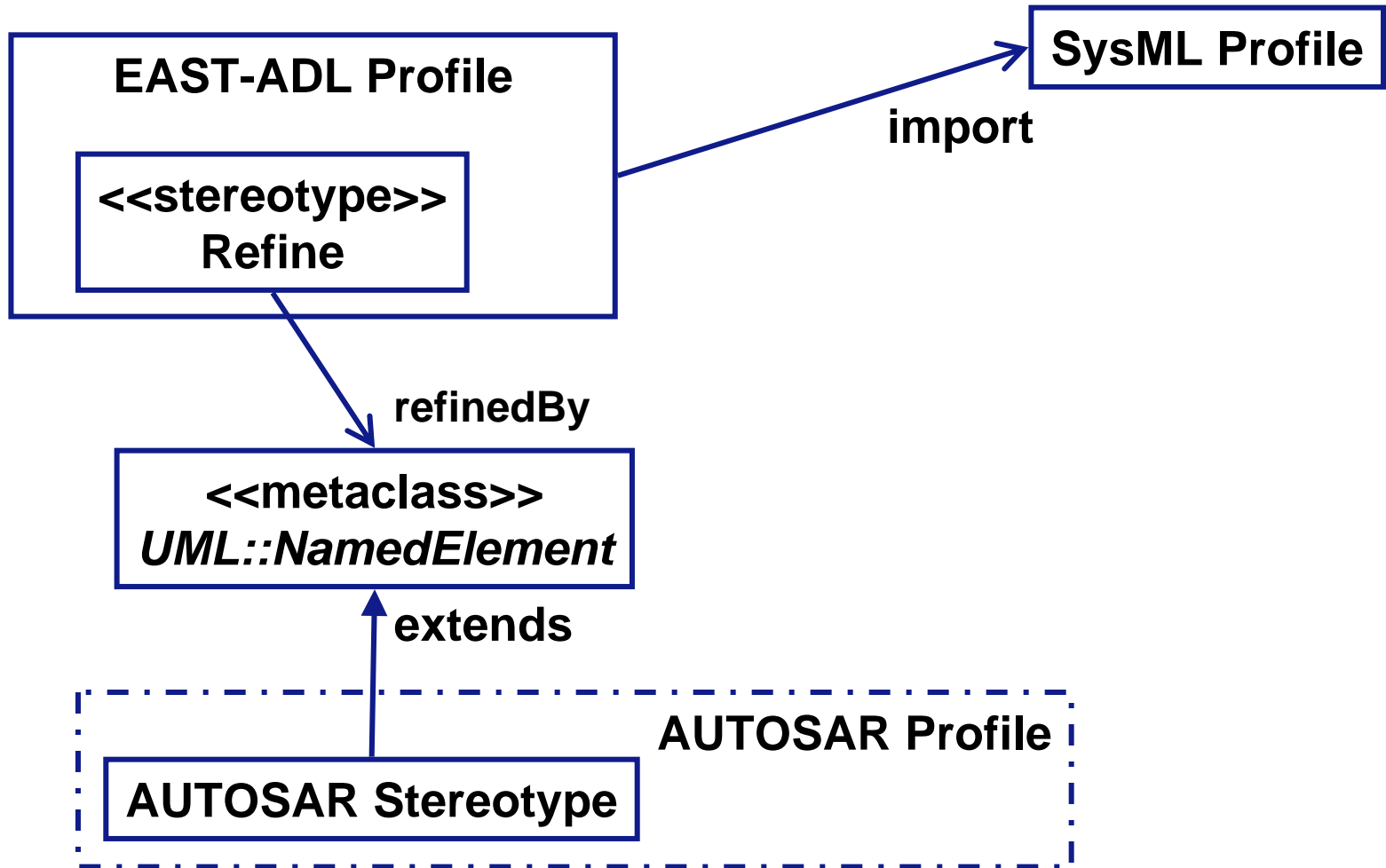
Choose UML2 metaclass
Define properties
Define constraints

Profile



Derived and ReadOnly properties,
are handled by static profile, assisting in User Modeling

Profile relations



Papyrus www.papyrusuml.org

- **Eclipse UML2 compliance**
- **Full respect of the UML2 standard as defined by the OMG**
- **Full respect of the DI2 (OMG Diagram Interchange) standard**
- **Extendable architecture of Papyrus that allows users to add new diagrams, new code generators, etc.**
- **Profile development support facilities for UML2 profiles**
- **Nested profiles support.** Papyrus enables the definition of a hierarchical profile - with internal subprofiles - and the use of them - i.e. correct "define" operation on such profiles as compared to the restricted Eclipse/UML2 "define" operation which does not operate recursively.
- **Available as UML Eclipse plug-in and standalone (Rich Client Platform)**

Papyrus for profile definition and user modeling

The screenshot displays the Papyrus IDE interface for defining a profile and user modeling. The main workspace shows a UML diagram with the following elements:

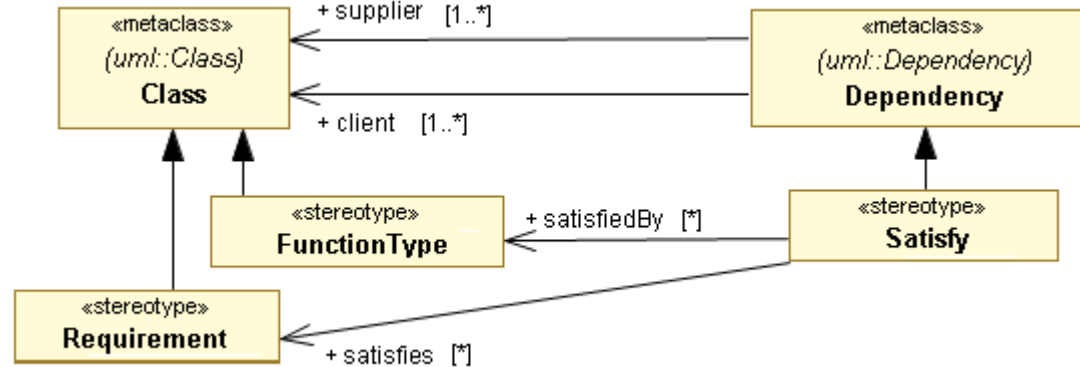
- «VariationPointConfiguration» Engine**: A green box representing a variation point configuration with properties: `isOptional = false`, `motivation = "What type of Engine: 4V or 6V?"`, and `kind = exclude`.
- «VariationPointConfiguration» Wheel**: An orange box representing a variation point configuration with properties: `isOptional = false`, `motivation = "What type of Wheel: large or normal?"`, and `kind = exclude`.
- «aDLFunctionType» Engine6V** and **«aDLFunctionType» Engine4V**: Two yellow boxes representing ADL function types, connected to the Engine variation point configuration via dashed arrows labeled `«variantLink»`.
- «aDLFunctionType» NormalWheel** and **«aDLFunctionType» LargeWheel**: Two yellow boxes representing ADL function types, connected to the Wheel variation point configuration via dashed arrows labeled `«variantLink»`.

The interface includes several panels:

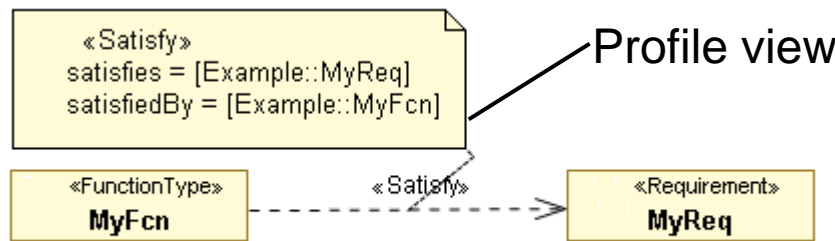
- Navigator**: Shows the project structure for `VariabilityExample`.
- Outline**: Lists the elements in the diagram, including `Engine6V`, `Engine4V`, `NormalWheel`, `LargeWheel`, `Engine`, `Wheel`, `Car`, `OldVariationManager`, and `Applied Profiles (17)`.
- Properties**: Shows the properties for the selected `VariabilityExample` package, including `Name: VariabilityExample` and `Visibility: public`.
- Palette**: Provides a list of UML elements and relationships for modeling, such as `Select`, `Marquee`, `Dependency`, `Generalization`, `Association`, `Link`, `Uml Element`, `VFM`, `Feature`, `VFMFeature`, `VV`, `FunctionMode...`, `ADLFunctionType`, `ADLFunctionPro...`, `ADLConnector...`, `FunctionalDevice`, `EnvironmentFun...`, `DesignDataType`, `ADLPortGroup`, `ADLInFlowPort`, and `Requirements`.

Profile view vs. UML2 view

M2



M1



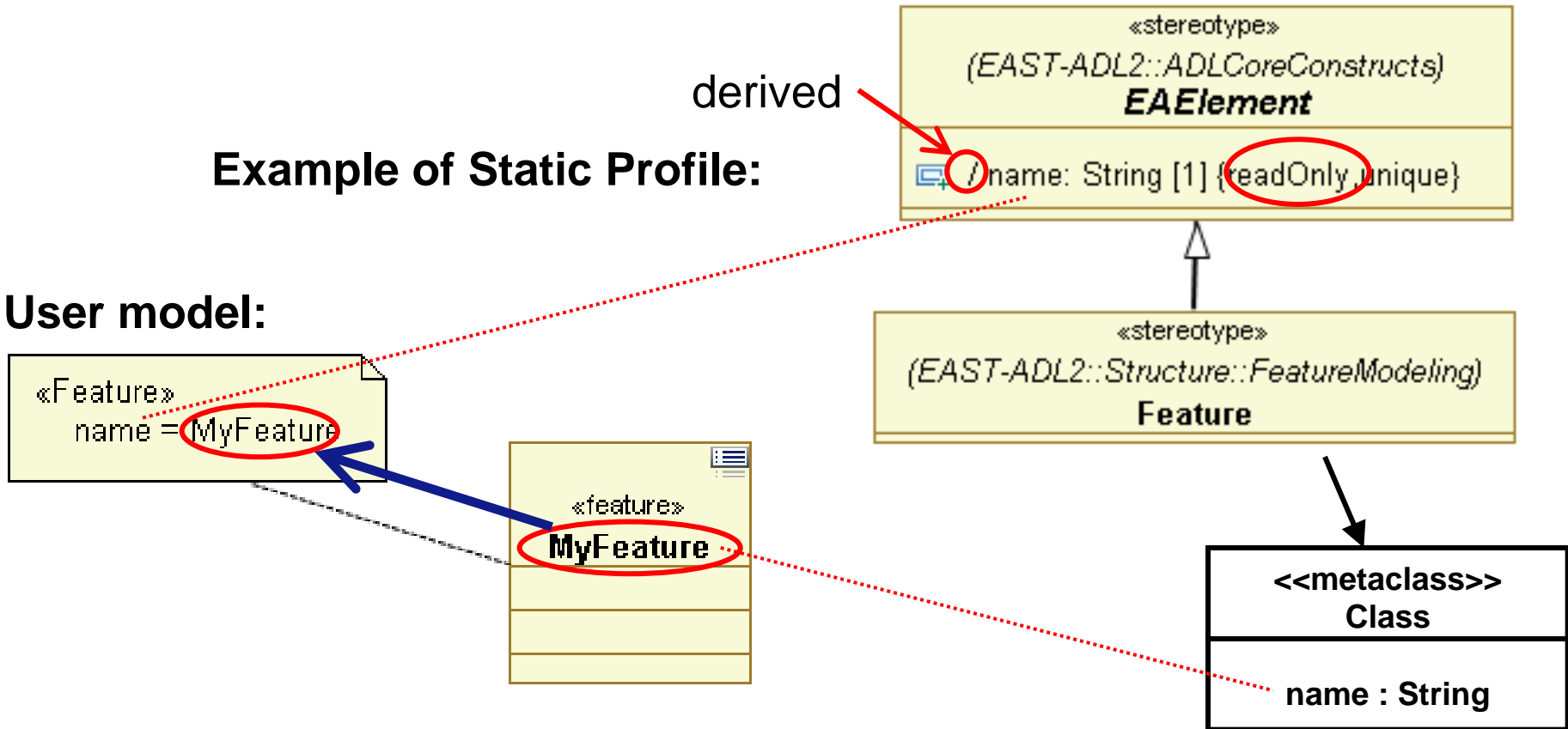
	Property	Value
General		
Profile	<input type="checkbox"/> Dependency property	
	name	FromMyFcnToMyReq
	qualifiedName	Example::FromMyFcnToMyReq
	visibility	public
Appearance		
Advanced	<input type="checkbox"/> References	
	client	[MyFcn]

UML2 view

Profile - static

Example of Static Profile:

User model:



SystemModel

- AA AnalysisLevel
- DA DesignLevel
- IA ImplementationLevel

FeatureModeling

- VFM VehicleLevel
- Feature
- VehicleFeature

FunctionModeling

- AnalysisFunctionType
- AnalysisFunctionPrototype
- DesignFunctionType
- DesignFunctionPrototype
- FunctionalDevice
- FunctionBehavior
- FunctionTrigger
- FunctionConnector
- FunctionAllocation
- PortGroup
- FunctionFlowPort
- FunctionPowerPort
- FunctionClientServerPort

Palette

Timing

- PrecedenceConstraint
- ExecutionTimeConstraint
- PeriodicEventConstraint
- InputSynchronizationConstraint
- OutputSynchronizationConstraint

ErrorModeling

- ErrorModelType
- ErrorModelPrototype
- ErrorBehavior
- FaultInPort
- FailureOutPort
- FaultFailurePropagationLink

VerificationValidation

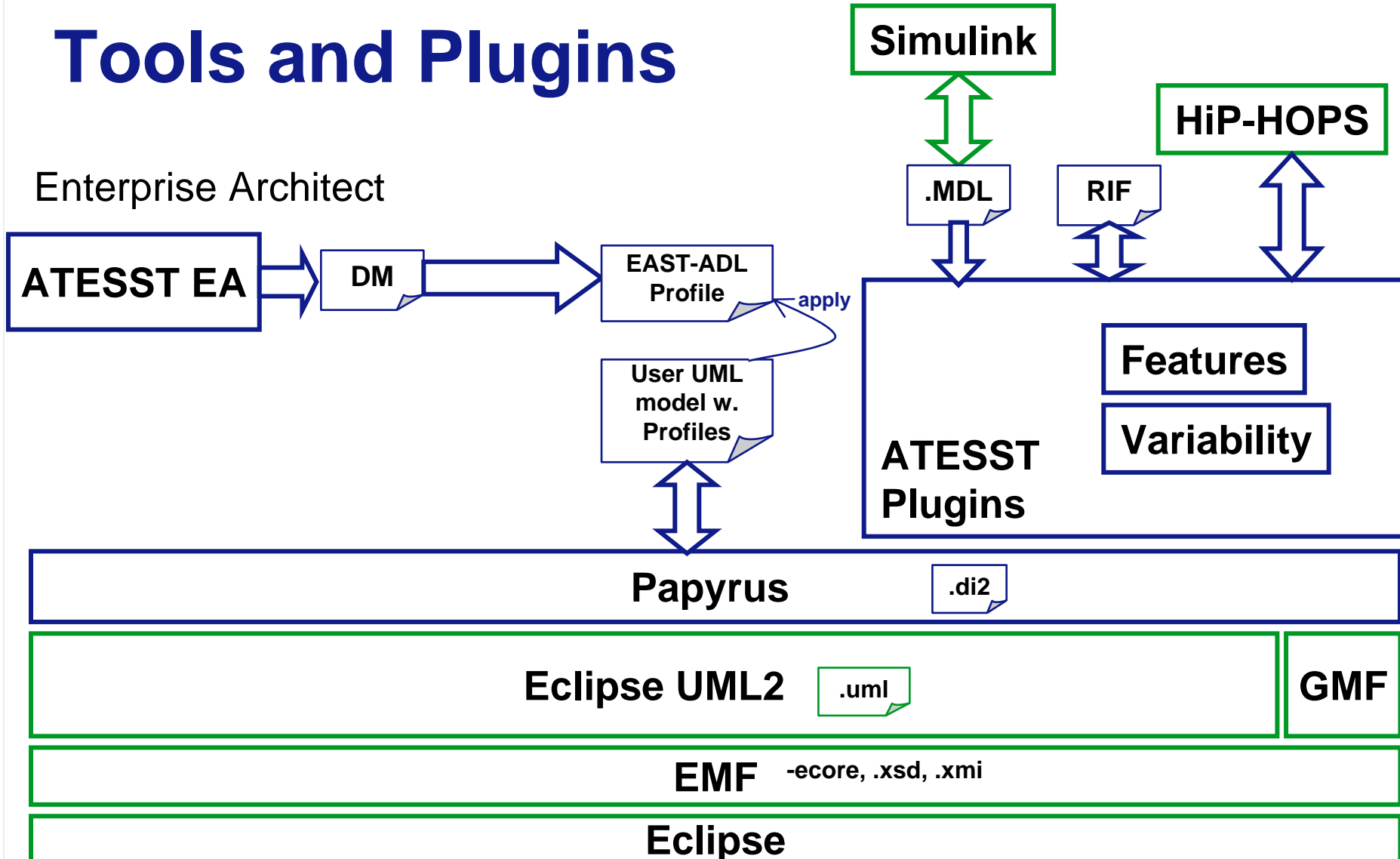
- VVCase
- VVProcedure

HardwareModeling

- Sensor
- Actuator
- Node
- PowerSupply
- HardwareComponentPrototype
- LogicalBus
- IOHardwarePin
- CommunicationHardwarePin
- PowerHardwarePin
- HardwareConnector
- Requirements
 - Requirement
 - RequirementsRelatedInformation
 - QualityRequirement
- Realization
- Satisfy
- Verify
- DeriveRequirement
- Refine

Tools and Plugins

Enterprise Architect



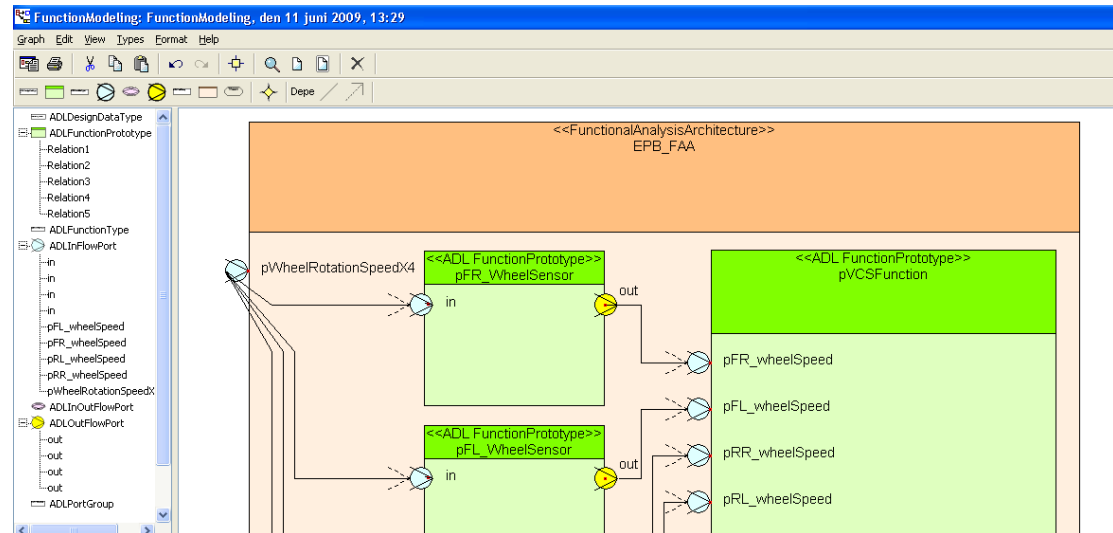
Other ways to use the Domain Model

ATESST2 has defined a Domain Model, and has used this as a specification for a UML Profile for use in UML Tools.

The Domain Model can also be used as a specification for other tools

Examples

- Topcased
- MetaCase MetaEdit+
- Mentor Graphics VSA



References

ATESST Domain Model and Profile documentation,
<http://www.atesst.org/>

Papyrus and EAST-ADL profile, www.papyrusuml.org

Eclipse, <http://www.eclipse.org>

OMG, <http://www.omg.org>

- UML2, <http://www.uml.org/>
- SysML, <http://www.omg.sysml.org/>