UML/MARTE to AADL transformation process

Skander TURKI, Eric SENN, Dominique BLOUIN

AADL meeting
11 June 2009

Support for Predictable Integration of mission Critical Embedded Systems
ITEA2 European Project
Sub-Project Objectives

- Provide UML users with AADL analysis tools.
- Allow AADL users to work with well established commercial case tools.
- Accelerate AADL dissemination.
- Illustrate/test the “everything is a model” paradigm.
How to do it?

- **Language constructs:**
  - Develop a dedicated UML profile for AADL.
  - Use MARTE as an AADL profile and add missing constructs.

  Problem: Too much design possibilities.

- **Design space reduction:**
  - Provide guidelines to MARTE users.
    - Check constraints in design time.
    - Check constraints on transformation time.
Different uses of model transformations with ATL

- Usual model transformation (interoperability between tools).

- Parameterized model transformation (Add a parameters model to the input of the transformation).

- Constraints checking on models (By producing report models).

- Higher Order Transformation: Adapt a model transformation’s model to a particular situation (add rules, input/output models, etc.).

- Source compilation (ATL compiler).

- Etc.
Why a Higher Order Transformation?

In ATL the input/output models are declared in advance:
- need to add output models to the ATL model.
- Need to add appropriate production rules for each instance model.

```plaintext
module MarteToAAXL;
create OUT : AAXL
from IN : UML2, OUT_VERIF : VERIF, IN_AAXL_Properties : AAXL_PROPERTIES;
```

```plaintext
rule Module {
  from s : ATL"ATL::Module" --, u : UML!InstanceSpecification (thisModule.inElements->includes(s))
  to t : ATL"ATL::Module" (
    location <- s.location,
    commentsBefore <- s.commentsBefore,
    commentsAfter <- s.commentsAfter,
    name <- s.name,
    isRefining <- s.isRefining,
    libraries <- s.libraries,
    inModels <- s.inModels,
    outModels <- s.outModels->append(UML!InstanceSpecification.allInstances()\rightarrow collect(e | thisModule.AdditionalOclModel\(e\))\rightarrow debug('additionalInst'))),
    elements <- s.elements->append((UML!InstanceSpecification.allInstances()\rightarrow collect(e | thisModule.InstancesLazyMatchedRules\(e\))))
  )
}
```

```plaintext
unique lazy rule AdditionalOclModel {
  from s : UML"uml::InstanceSpecification"
  to t : ATL"OCL::OclModel" (
    name <- s.name,
    metaModel <- thisModule.resolveTemp(UML"uml::InstanceSpecification", 'mm')
  )
}
```
Facing problem with AADL property sets: Do we have to add an AADL profile to MARTE for each property set library?
  
  - Partial Solution: Weaving model between MARTE properties and standard AADL property sets.

  Incomplete mapping: need for specification alignments?

**Problems:**

- What about user defined property set libraries?
- Add them through an AADL profile, but then, why not directly use an independent AADL profile?
Thank You