Presentation of a SysML-inspired Requirements Definition and Verification Tool for AADL

Skander Turki : Lab-STICC UBS

Toulouse 17 -19 May 2010
Outline

• Introduction of the project
• Introduction of the tool
• Who uses the tool?
• The requirements metamodell
• The graphical tool
• Identifying use cases for constraints
• Questions
The Open-People Project

- Open-People is a french project intending to build a remote hardware platform for power consumption analysis.

- AADL is the chosen language for the system’s modeling.

- A demand in this project was to add to the AADL toolchain (OSATE/Adele/CAT etc.) a requirements definition and validation tool.
Presentation of the Tool

• We started by developing a graphical editor for only textual requirements (informal). Then we added the possibility to write OCL or REAL requirements in the metamodel (graphical view under development).

• We started developing the OCL constraints checker that will validate the Requirements model against the AADL model.
Who uses the tool? : AADL user

- When building a model (M1), the user can define constraints that reference elements from either M3, M2, M1 or M0: We are not talking about constraints built in the M2 or M3 level!

- Identifying constraints use cases where the actor is the AADL language user.

The AADL language engineer works here. We are not talking about this actor.

The AADL user builds his models here. We are only concerned by this actor.
For users of the AADL language, M3 level must not be used to define constraints (meatclass Class should always have name property...)

skander, 5/16/2010

For users of the AADL language, M2 level can be used to define additional constraints on model elements:
when enforcing new semantics from an application domain (a system has always at least one processor).
When enforcing specific design best practices (a thread should always be bound to a processor).
When defining requirements on the system, in this case, the requirements should be separated from the AADL model (each processor must not consume more than xx energy).

skander, 5/16/2010
This metamodel is inspired from SysML and the ISO/IEC/IEEE 15288 SE standard. But SysML is a UML extension, so we needed to adapt it (In SysML a requirements group is represented by a package instead of a class).
The Requirements graphical editor
Identifying use cases for constraints

• Identifying constraints use cases:
  
  – First use case: Constraints that reflect a requirement expressed by the system’s specification.
  
  – Second use case: Constraints that are part of the system’s model and that express a design information; They complete the AADL metamodel with additional constraints.

• In the first use case we have found that it is better to separate the requirements model from the design one.

• In the second use case we found that it is better to have the constraints embedded in the AADL model. Example of the Electrical compatibility constraint in Jerome Hugues’ use case with the REAL language:

```REAL

theorem Electrical_Compatibility

foreach e in Bus_Set do
  Cnx_Set (e) := \{x in Connection_Set | Is Accessing To (x,e) \} ;

-- Requires previous checking on number of masters (only one master allowed)
Master_Device :=
  \{ d in Device_Set | Is Accessed By (d, Cnx_Set) and
    Property Exists(d, "Electricity::Device_Type") and
    (Get Property Value (d, "Electricity::Device_Type") = "Master")
  \};
```
For users of the AADL language, M1 level can be used to define additional constraints on model elements:
when enforcing new semantics from an application domain specific to this model element (in this model, a system has always at least one processor).
When enforcing specific design choices specific to this model (in this model, a thread should always be bound to a processor).
When defining requirements on the system, in this case, the requirements should be separated from the AADL model (each processor must not consume more than xx energy).

skander, 5/17/2010
Questions

• Is there any interest for such a language in the AADL community?

• Do we need a textual concrete syntax for our requirements language (AADL Annex)?

• Would there be other specific needs for the graphical editor?

• Is the traçability enough?