推荐
modeler.org.cn

AADL和MBSE实践探讨

推荐
modeler.org.cn

议程

1. AADL简要介绍

2. SysML和AADL混合MBSE流程思考

3. SCADE AADL方案

4. 例子演示

推荐
modeler.org.cn

议程

1. AADL简要介绍

2. SysML和AADL混合MBSE流程思考

3. SCADE AADL方案

4. 例子演示

推荐
modeler.org.cn

Introduction

- **ADL, Architecture Description Language:**
  - Goal : modeling software and hardware architectures to master complexity … to perform analysis
  - Concepts : components, connections, deployments.
  - Many ADLs : formal/non formal, application domain, …
- **AADL -- Architecture Analysis and Design Language is an ADL for real-time critical systems**
- **AADL objectives are “to model a system”**
  - With analysis in mind – “certification credit”
  - To ease transition from well-defined requirements to the final system : modeling, refinement, analysis, code production, …

推荐
modeler.org.cn

AADL: Architecture Analysis & Design Language

- **Standard promoted by SAE International, AS-2C committee, as AS-5506A**
  - Version 1.0 published in 2004, v2 in 2009, v2.1 in 2012
  - Committee driven by inputs from the avionics and space industry
  - Academics drive analysis capability, to ensure they match with modeling patterns
  - AADLv3 effort just started
- **http://aadl.info list all resources around AADL**
  - Public wiki with lot of resources: https://wiki.sei.cmu.edu/aadl/index.php/Main_Page
  - Include link to most research activities around AADL
- **AADL is dedicated to real-time embedded domain**
  - Modeling software and hardware resources for V&V
  - Extension & refinements concept to iterate down to generation
- **Different representations**
  - Graphical: high-level view of the system
  - Textual: to view all details
  - XML: to ease processing by 3rd party tool

推荐
modeler.org.cn

AADL Different Representations

推荐
modeler.org.cn

AADL Components

Port Group (as a feature of a thread)	
Port Group Connection (between two port groups that are each a feature of system)	
Port Group Bundle (mixed directions and ports)	Port group

推荐
modeler.org.cn

Data Connection Policies

1. **Sampling connection: takes the latest value**

   Problem: data consistency (lost or read twice) !

2. **Immediate: receiver thread is immediately awaken, and will read data when emitter finishes**

3. **Delayed: actual transmission is delayed to the next time frame**

推荐
modeler.org.cn

AADL model elements

Package
. Public decl..
. Private decl.

Component type
. Identifier
. Extends
. Prototypes
. Features
. Flows
. Properties
. Annex

. Ports
. Access
. Subprogram
. Parameter
. Feature

Category
. Data
. Subprogram
. Thread (group)
. Process
. Memory
. Device
. (virtual) processor
. (virtual) bus
. System
. Abstract

"references"

Property sets
. Units
. Property type
. Property definition
. Constants

Component implementation
. Identifier
. Extends
. Subcomponents
. Connections
. Call sequences
. Modes
. Flows
. Properties
. Annex

. Ports
. Access
. Parameter

. Modes
. transitions

AADL

推荐
modeler.org.cn

AADL – an example

Architecture helps you focusing on the actual system

Link to code/model
Workflow with SysML,
Executable models (SCADE)
Code (Ada, C, lua, …)

Architectural patterns

```
--  Textual AADL

thread Sender_Thread
features
  Data_Source : out event data port Record_Type.Impl;
properties
  Dispatch_Protocol => Periodic;
  Period            => 1 Sec;
annex real_specification {**
--  Contract to be enforced
**};
end Sender_Thread;
```

Non-functional properties

推荐
modeler.org.cn

Component type

```
<category> foo [extends <bar>]
features
  -- list of features, interface
  -- e.g. messages, access to data, etc.
properties
  -- list of properties, e.g. priority
end foo;
```

Inherit features and properties from parent

Some properties describing non-functional aspect of the component

```
 --  Model a sequential execution flow
subprogram Spg                              --  Spg represents a C function,
features                                    --  in file "foo.c", that takes one
  in_param : in parameter foo_data;         --  parameter as input
properties
  Source_Language => C;
  Source_Text => ("foo.c");
end Spg;

-- Model a schedulable flow of control
thread bar_thread                            --  bar_thread is a sporadic thread :
features                                     --  dispatched whenever it
  in_data : in event data port foo_data;     --  receives an event on its port
properties
  Dispatch_Protocol => Sporadic;
end bar_thread;
```

推荐
modeler.org.cn

Component implementation

foo.i implements foo

```
<category> implementation foo.i [extends <bar>.i]
subcomponents
  -- internal elements
connections
  -- from external interface to internal subcomponents
properties
  -- list of properties
end foo.i;
```

```
  -- Model a schedulable flow of control
  thread bar_thread                          --  bar_thread is a sporadic thread :
  features                                   --  dispatched whenever it
    in_data : in event data port foo_data;   --  receives an event on its port
  properties
    Dispatch_Protocol => Sporadic;
  end bar_thread;

  thread implementation bar_thread.impl      --  In this implementation, at each
  calls                                      --  dispatch we execute the "C" call
    C : { S : subprogram spg; };             --  sequence. We pass the dispatch
  connections                                --  parameter to the call sequence
    parameter in_data -> S.in_param;
  end bar_thread.impl;
```

推荐
modeler.org.cn

AADL Property sets

- **Property: Typed attribute, associated to components**
- **Property sets: group property definitions.**
  - Property sets part of the standard, e.g. `Communication_Properties`
  - Or user-defined, e.g. for new analysis

```
process MFDProcess
  features
    MCPaltitude: out data port scade_real;
    MCPspeed: out data port scade_real;
    AutoPilot: out data port scade_bool;
  flows
    f0: flow source MCPaltitude {Latency => 5 ms .. 10 ms;};
    f1: flow source MCPspeed {Latency => 5 ms .. 10 ms;};
    f2: flow source AutoPilot {Latency => 5 ms .. 10 ms;};
  properties
    Period => 25 ms;
end MFDProcess;
```

推荐
modeler.org.cn

AADL Property sets

```
property set AADL_Projects
is Time_Units: type units (
    ps,
    ns  => ps  * 1000,
    us  => ns  * 1000,
    ms  => us  * 1000,
    sec => ms  * 1000,
    min => sec * 60,
    hr  => min * 60);
-- …
end AADL_Projects;
```

```
--AADL2
--SAE Aerospace Standard AS5506B
--Appendix A: Predeclared Property Sets

property set Communication_Properties is
 Time: type aadlinteger units Time_Units;
 Time_Range: type range of Time;
 Latency: Time_Range
    applies to (flow, connection, virtual
bus, bus, processor, virtual processor,
device, system, feature, memory);
-- …
```

```
-- …
flows
    f0: flow source MCPaltitude {Latency => 5 ms .. 10 ms;};
    f1: flow source MCPspeed {Latency => 5 ms .. 10 ms;};
    f2: flow source AutoPilot {Latency => 5 ms .. 10 ms;};
```

推荐
modeler.org.cn

About AADL annexes

- ***AS5506/2 (January 2011)***
  - **Data Modeling Annex** provides guidance on a standard way of associating data models expressed in other data modeling notations (C or ASN.1) with architecture models expressed in AADL,
  - **Behavior Annex** enables modeling of component and component interaction behavior in a state-machine based annex sublanguage,
  - **ARINC653 Annex** provides guidance on a standard way of representing ARINC653 standard compliant partitioned embedded system architectures in AADL models.
- ***AS5506/1A (October 2015)***
  - **Code generation Annex defines language-specific rules for source text to be compliant with an architecture specification written in AADL;**
  - **Error Model Annex defines features to enable the specification of redundancy management and risk mitigation methods in an architecture, and enable qualitative and quantitative assessments of system properties such as safety, reliability, integrity, availability, and maintainability.**

推荐
modeler.org.cn

Model-based Assurance

Predictive Analysis Across Perspectives

Availability & Reliability

- MTBF
- FMEA
- Hazard analysis

Security

- Intrusion
- Integrity
- Confidentiality

Data Quality

- Data precision/ accuracy
- Temporal correctness
- Confidence

Real-time Performance

- Execution time/ Deadline
- Deadlock/starvation
- Latency

Resource Consumption

- Bandwidth
- CPU time
- Power consumption

- **Integration to a process: with SysML, SCADE, Simulink**
- **Architectural pattern checks:**
  - MILS, ARINC, Ravenscar, Synchronous
- **Model checking:**
  - Timed/Stochastic/Colored Petri Nets
  - Timed automata et al.: UPPAAL, Versa, TASM
- **Scheduling: MAST, Cheddar, CARTS**
- **Performance evaluation: real-time and network calculus**
- **Fault analysis: COMPASS, Stochastic Petri Nets, PRISM**
- **Simulation: ADeS, Marzhin**
- **Energy consumption of SoC: OpenPeople project**
- **Code generation: SystemC, C, Ada, RTSJ, Lustre**
- **WCET analysis: mapping to Bound-T**

推荐
modeler.org.cn

议程

1. AADL简要介绍

2. SysML和AADL混合MBSE流程思考

3. SCADE AADL方案

4. 例子演示

Mismatched Assumptions

Impact – AADL integrates allowing analysis

Why do system level failures still occur despite fault tolerance techniques being deployed in systems?

推荐
modeler.org.cn

Cooperative Engineering of System

推荐
modeler.org.cn

System Engineering Process of INCOSE Systems Engineering

推荐
modeler.org.cn

MBSE Process
OOSEM Approach - Selected Artifacts

Major SE Development Activities

Analyze Needs
- Causal Analysis
- Top Level Domain Block Definition Diagram
- Mission Use Case Diagrams

Define System Requirements
- System Context
- System Scenario
- Input/Output Definitions

Define Logical Architecture
- Logical Hierarchy
- Logical Interconnection
- Logical Scenarios

Optimize & Evaluate Alternatives
- Trade studies
- Parametric Diagrams

Synthesize Allocated Architecture
- Allocations to HW,SW, Data Architecture

Manage Requirements
- Spec Tree
- Reqt's
- Traceability

Support Validation & Verification
- Test cases
- Test procedures

Common Subactivities

Copyright © 2006-2009 by Object Management Group.
© 2011 by JHU/APL. All Rights Reserved.
APL

推荐
modeler.org.cn

Cooperative Engineering of SysML and AADL

推荐
modeler.org.cn

议程

1. AADL简要介绍

2. SysML和AADL混合MBSE流程思考

3. SCADE AADL方案

4. 例子演示

推荐
modeler.org.cn

SCADE AADL Solution: Workflow

推荐
modeler.org.cn

AADL meta model via UML

- **AADL v2.2 meta model (ecore file)**
  - Loaded in SCADE Architect Configurator
- **2) AADL concepts**
  - Inherits from SCADE Architect concepts
  - Constraints allows model creation guidance

推荐
modeler.org.cn

AADL language expressiveness (& complexity)

- **AADL language**

  Object-oriented inheritance mechanism:

  Prototypes and *Abstract* components

  later extended and refined into concrete category

  Component types and *Component implementation*

  An interface definition can have multiple implementations

  But definition mandatory before specifying implementation

  Instantiation:

  Component instances are references to *component implementation*, that must be inlined for analysis

  Inlining done as an explicit tool action in *OSATE* to get an instantiated model

- **In SCADE: 2 simplifications**

  1. AADL Abstraction & Inheritance inlining
  2. AADL instance based modeling

推荐
modeler.org.cn

SCADE solution for AADL

Instance based modeling

- **Benefit from SCADE Architect: Block Replica**
  - The whole content of Block Definition is replicated in each instance (SysML parts)
- **Support for AADL "instance based modeling"**
  - AADL objects:
    - ✓ "ProcessTypes" (interface only),
    - ✓ "ProcessImplementations" (content only),
    - ✓ "ProcessSubcomponents" (empty instances)
  - Replaced by:
    - ✓ "Process" definition: interface and full content, automatically replicated in each AADL "Process instance"
  - Consequences
    - ✓ Limitation: only "one Implementation per Type"
    - ✓ But much simpler model understanding for end user

推荐
modeler.org.cn

AADL Property sets

- **Automated conversion**
  - Can be imported from <property set>.aadl
- **Benefits**
  - Reused SCADE IDE
  - Automated GUI to set properties on objects in a model

推荐
modeler.org.cn

AADL Example (Sept 2016)

推荐
modeler.org.cn

Synchronization SCADE AADL – SCADE Suite

- **SCADE Suite ←→ AADL:**
  - Selected operators ←→ choice to Thread and Subprogram (default Thread)
  - In/Out variable ←→ Port (default DataPort)
    - ✓ DataPort → In/Out variable <type>
    - ✓ EventPort → In/Out boolean variable
    - ✓ EventDataPort → In/Out variable {EDP_Event:bool, EDP_Data:<type>}
    - ✓ Port ArrayDimension N → N In/Out variable

推荐
modeler.org.cn

议程

1. AADL简要介绍

2. SysML和AADL混合MBSE流程思考

3. SCADE AADL方案

4. 例子演示

推荐
modeler.org.cn

感谢聆听