AADLv3: Nested Processors, Virtual Memory, Multi-layer platforms

Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213

July 2016

Processor and Virtual Processor

Separate Hardware and Software concerns

- Processor: specification of hardware aspects (chip, core, multi-threads)
- Virtual Processor: (OS, partitions, schedulers)

Virtual Processor Binding to Processor

- V3 proposal: Virtual Processor no longer contained in a processor

Nested Processor and hardware platform

- Description of core, hyper threading, etc.
- Processor instances in processor implementation
  - Also contains memory, bus
  - Difference to system with processor, memory, bus?
- Properties to describe hardware constraints (endianness)

Concepts of hardware platform system, virtual platform system, app system

Memory & Virtual Memory

Memory

- Memory as subcomponent: currently yes.
  - Enclosing system/memory may have binding point for whole memory
  - Need for representing different section of memory addresses: binding points have properties to indicate base address and range (size)
  - Memory binding point on devices can model device registers without requiring memory subcomponents.
  - May be able to eliminate memory as subcomponent. Use system.
- Virtual memory roles
  - Represent logical addresses that are mapped to addresses in different components at the next level.
  - Represent segments of address space
  - Enforcement: property of requirement and specification of whether it is supported. Who is supporting it (processor, memory, VP?
- Virtual memory binding and subcomponents
  - Nested, in memory, virtual processor, processor?
  - Part of virtual platform system specification

System Layers (Levels) and Binding

Currently:

System contains multiple subcomponents

Binding between elements of two subcomponents that represent two layers

Semantics of Layer, Platform, Level, Tier

Restriction on accessing only same or lower layer(s)

- **Direct**
- **More than one layer**

Multi tier (aka level)

Abstraction layers (data, function : libraries)

Virtual machine layers (p-code, OS services)

OSI Layers

GOA Platform, Layer (System , Logical and Physical Resource)

Examples

Software libraries

- **Represent API to common services**
- **Shared vs. separate instances**

Multi-tier architectures

- Tiers as layers
- Each tier has SW and HW layer vs. SW tiers mapped to HW layer

OS modularity & Layers

- Memory mgnt
- Process mgnt

System and Task hierarchies

- Physical: Aircraft, brakes, engines (SAVI Tiers)
- AADL hierarchy: system, process, thread
- Control system hierarchy
- SW task hierarchy