



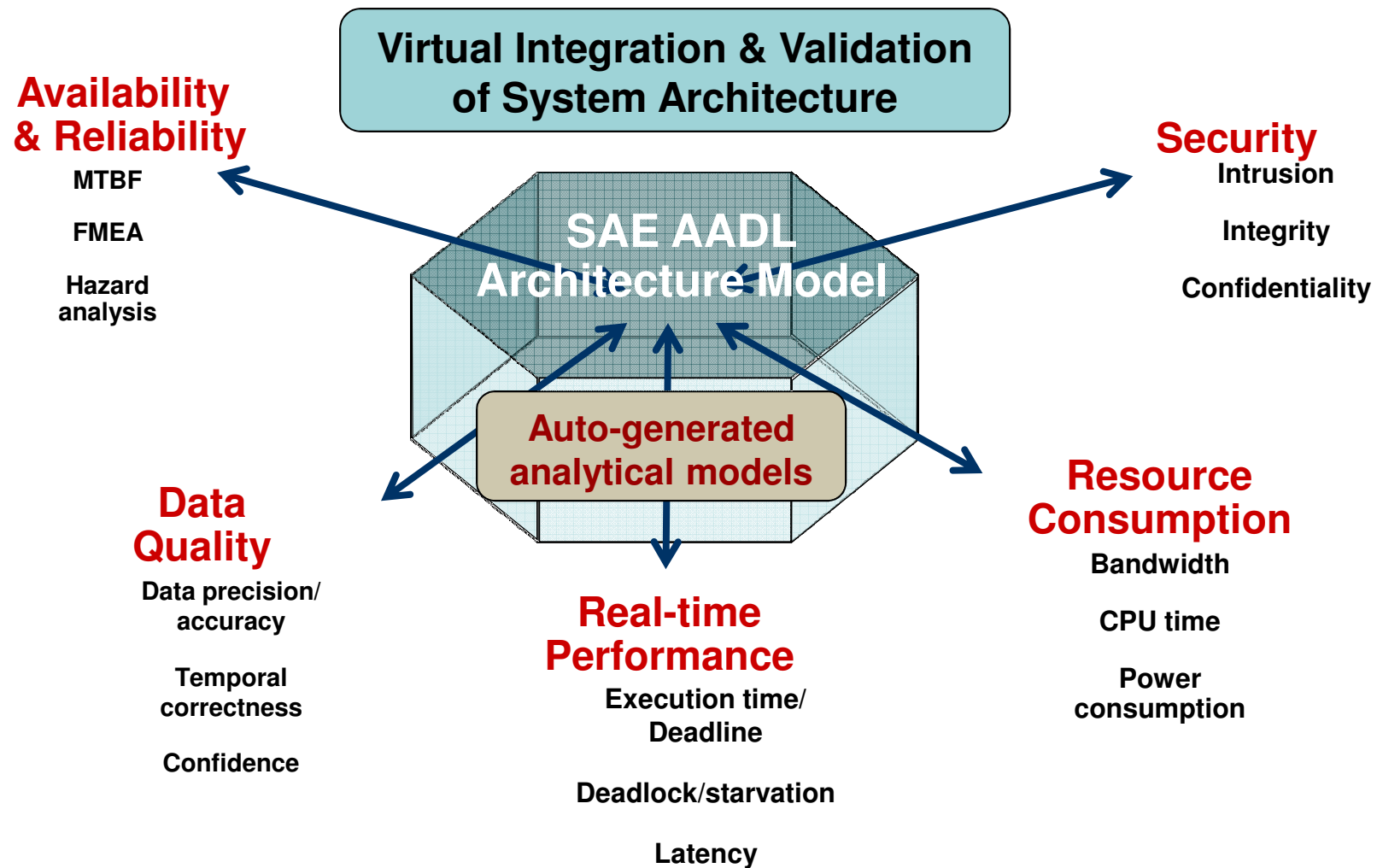
Validation of Safety-Critical Embedded Systems with AADL

Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213

Peter H Feiler
April 2009



Architecture-Centric Engineering Approach



Outline

Multi-fidelity Model-based Analysis

- Physical system analysis
- Resource budget analysis
- End-to-end latency analysis
- Scheduling analysis
- Security analysis
- Fault analysis



Engineering of Embedded/Computer System

Parts model & mass

- Processor, memory, bus/network, physical target system
- Weight limits, net weight , gross weight
- Weight budgets & recursive rollup

Electrical power

- Power system with capacity as AADL bus type
- Power supply and power budget as bus access properties
- Connected power systems
- Power system upgrade implies change in weight



Mass & Electrical Power Analysis

Hardware component specs

```

bus EtherSwitch
  features
    Power: requires bus access PowerSupply {
      SEI::PowerBudget => access 0.100 W;
    };
  properties
    SEI::PowerCapacity => 1.0 W;
    SEI::BandWidthCapacity => 100.0 Mbps;
    SEI::NetWeight => 5.0 kg;
end EtherSwitch;

processor Xeon
  features
    HS: requires bus access EtherSwitch {
      SEI::PowerBudget => access 75.0 mW;
    };
    Power: requires bus access PowerSupply {
      SEI::PowerBudget => access 4.9 W;
    };
  properties
    SEI::NetWeight => 0.2 kg;
end Xeon;

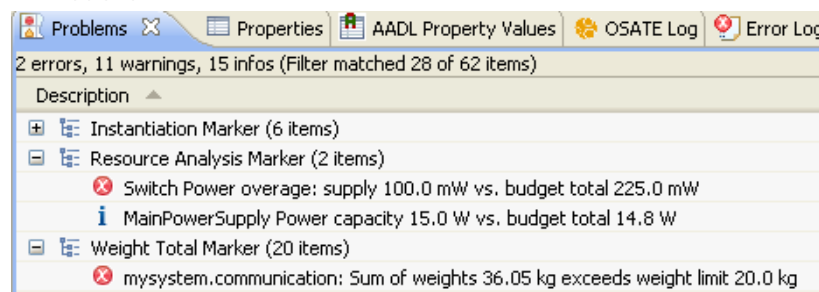
```

Early HW model excluded power supply

MissionProcessor2: Xeon.solo: Sum of weights / Gross weight 0.350 kg (no limit specified)
MissionProcessor3: Xeon.solo: Sum of weights / Gross weight 0.350 kg (no limit specified)
PilotDisplay: MFD: Sum of weights / Gross weight 5.000 kg (no limit specified)
Switch: EtherSwitch: Sum of weights / Gross weight 5.000 kg (no limit specified)
ApplicationSystem: EmbeddedApp.SubSystemParts: Sum of weights / Gross weight 5.000 kg (no limit specified)
Platform: ComputingPlatform.ThreeProcessorParts: Sum of weights / Gross weight 6.050 kg (no limit specified)
mssystem.parts: Sum of weights 11.050 kg below weight limit 20.000 kg (44.7 % Weight slack)

Increase in Switch supply overloads power supply

Power system change implies change in mass



Multi-Fidelity Resource Budgeting

Resource capacities for processors, memory, bus/networks

- Compute resources: MIPS, MB, bandwidth
- Physical resources: power

Budgets for major subsystems

- Capacity and budget totals
- Early deployment decisions & resource-specific budget totals
- Port group connections & bandwidth budgets

System decomposition & budget refinement

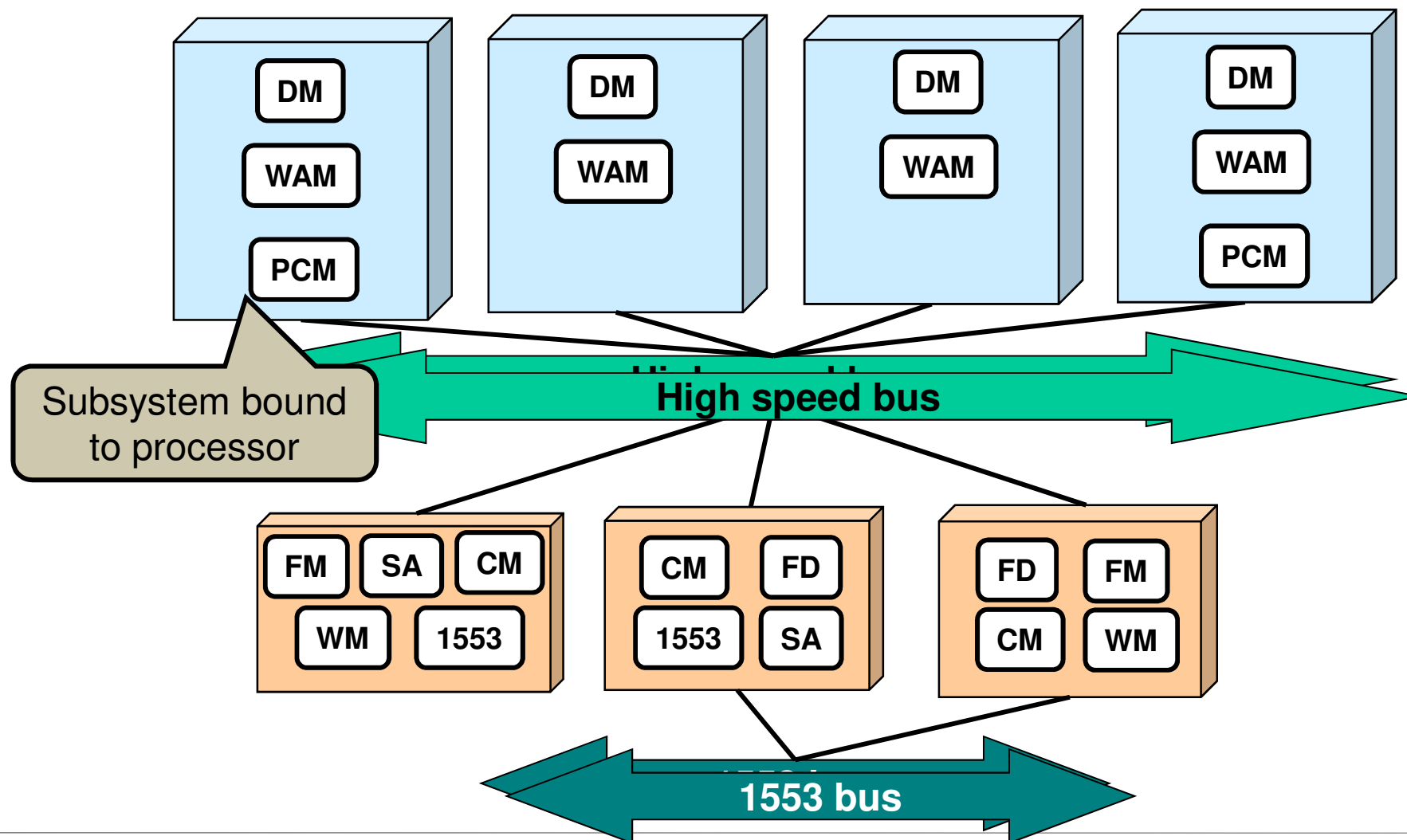
- Budget rollup & re-negotiation

Task & communication refinement

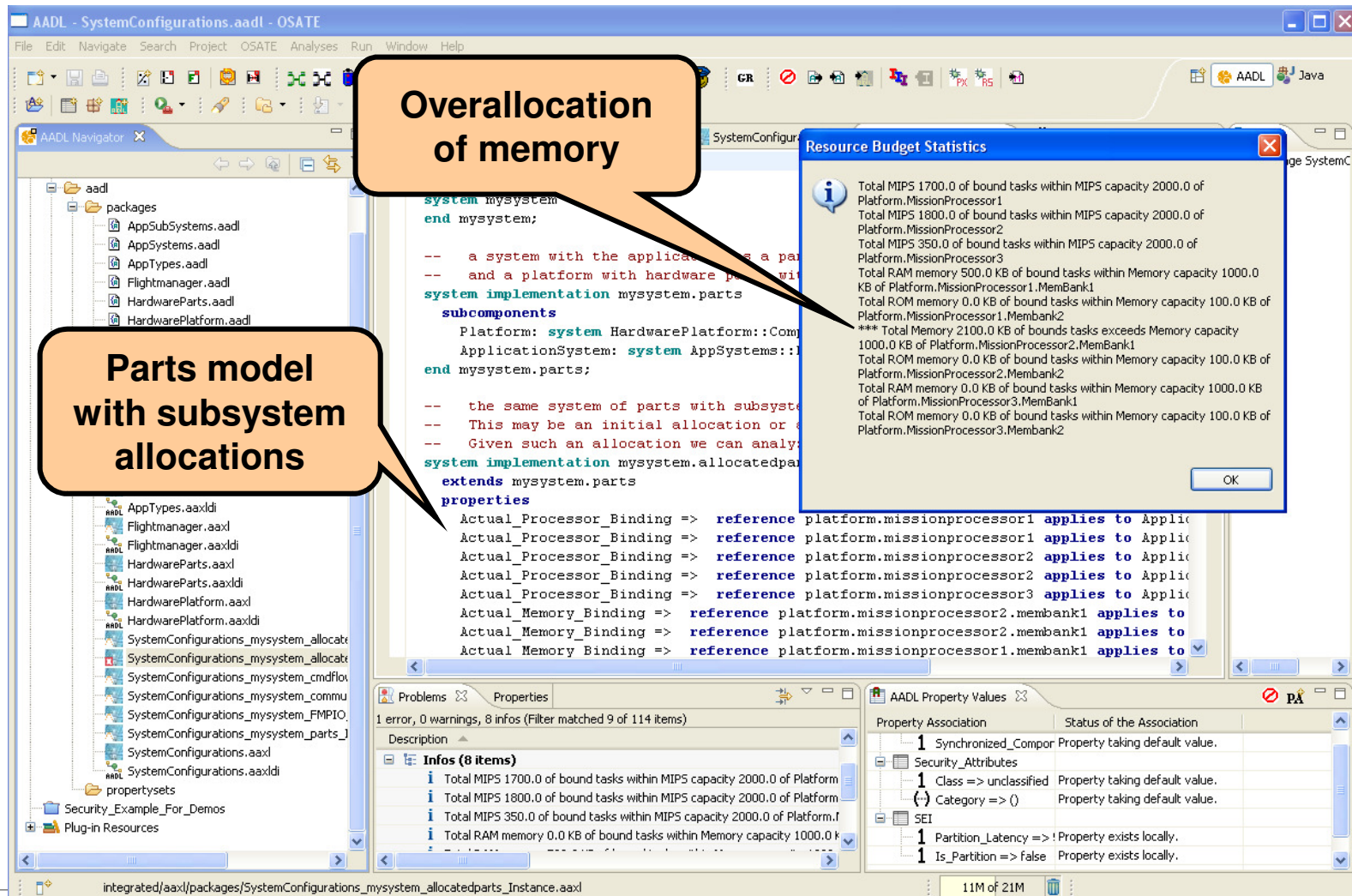
- Rates, WCET and budgets



Basic System Architecture



Initial Partition Allocations



Overalllocation of memory

Parts model with subsystem allocations

```

system mysystem
end mysystem;

-- a system with the application as a part
-- and a platform with hardware parts
system implementation mysystem.parts
subcomponents
Platform: system HardwarePlatform::Comp
ApplicationSystem: system AppSystems:::
end mysystem.parts;

-- the same system of parts with subsystem
-- This may be an initial allocation or
-- Given such an allocation we can analyze
system implementation mysystem.allocatedparts
extends mysystem.parts
properties
Actual_Processor_Binding => reference platform.missionprocessor1 applies to Applic
Actual_Processor_Binding => reference platform.missionprocessor1 applies to Applic
Actual_Processor_Binding => reference platform.missionprocessor2 applies to Applic
Actual_Processor_Binding => reference platform.missionprocessor2 applies to Applic
Actual_Processor_Binding => reference platform.missionprocessor3 applies to Applic
Actual_Memory_Binding => reference platform.missionprocessor2.membank1 applies to
Actual_Memory_Binding => reference platform.missionprocessor2.membank1 applies to
Actual_Memory_Binding => reference platform.missionprocessor1.membank1 applies to

```

Resource Budget Statistics

- Total MIPS 1700.0 of bound tasks within MIPS capacity 2000.0 of Platform.MissionProcessor1
- Total MIPS 1800.0 of bound tasks within MIPS capacity 2000.0 of Platform.MissionProcessor2
- Total MIPS 350.0 of bound tasks within MIPS capacity 2000.0 of Platform.MissionProcessor3
- Total RAM memory 500.0 KB of bound tasks within Memory capacity 1000.0 KB of Platform.MissionProcessor1.MemBank1
- Total ROM memory 0.0 KB of bound tasks within Memory capacity 100.0 KB of Platform.MissionProcessor1.MemBank2
- *** Total Memory 2100.0 KB of bounds tasks exceeds Memory capacity 1000.0 KB of Platform.MissionProcessor2.MemBank1
- Total ROM memory 0.0 KB of bound tasks within Memory capacity 100.0 KB of Platform.MissionProcessor2.MemBank2
- Total RAM memory 0.0 KB of bound tasks within Memory capacity 1000.0 KB of Platform.MissionProcessor3.MemBank1
- Total ROM memory 0.0 KB of bound tasks within Memory capacity 100.0 KB of Platform.MissionProcessor3.MemBank2

Problems
1 error, 0 warnings, 8 infos (Filter matched 9 of 114 items)

Infos (8 items)

- Total MIPS 1700.0 of bound tasks within MIPS capacity 2000.0 of Platform
- Total MIPS 1800.0 of bound tasks within MIPS capacity 2000.0 of Platform
- Total MIPS 350.0 of bound tasks within MIPS capacity 2000.0 of Platform
- Total RAM memory 0.0 KB of bound tasks within Memory capacity 1000.0 KB of Platform

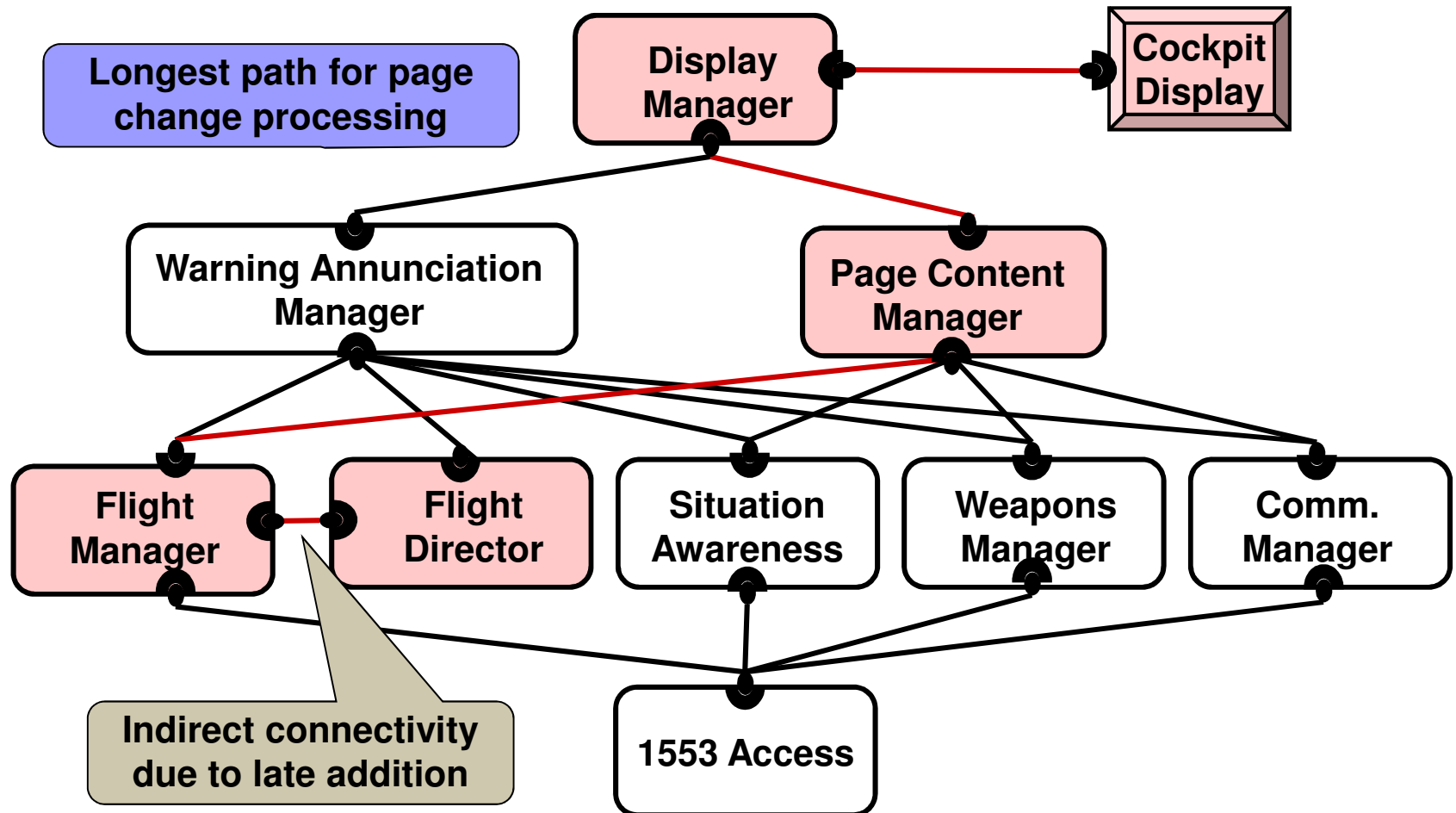
AADL Property Values

Property Association	Status of the Association
1 Synchronized_Comp Property taking default value.	
Security_Attributes	
1 Class => unclassified Property taking default value.	
(-) Category => () Property taking default value.	
SEI	
1 Partition_Latency => ! Property exists locally.	
1 Is_Partition => false Property exists locally.	



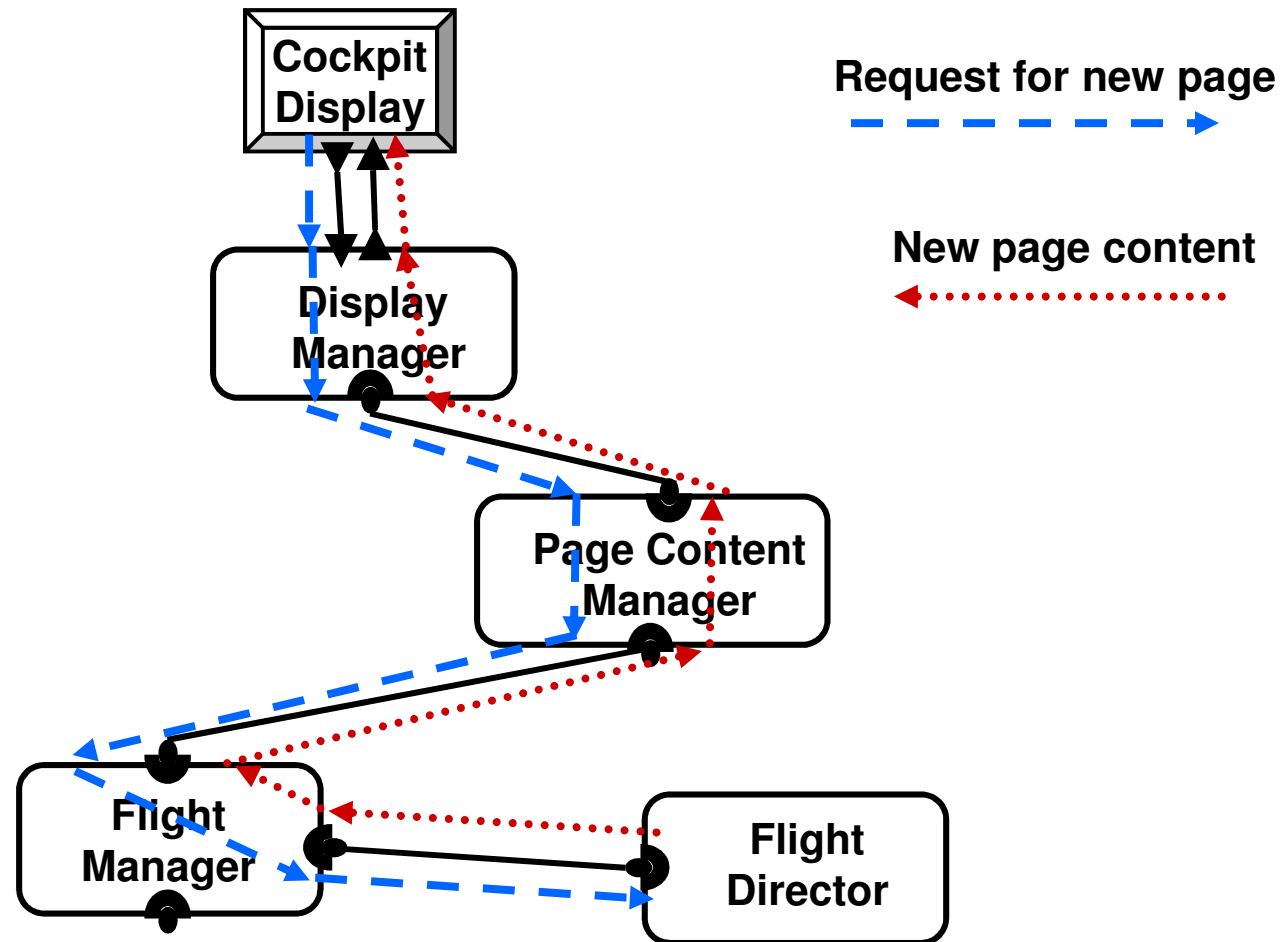


Worst Case Use Scenario

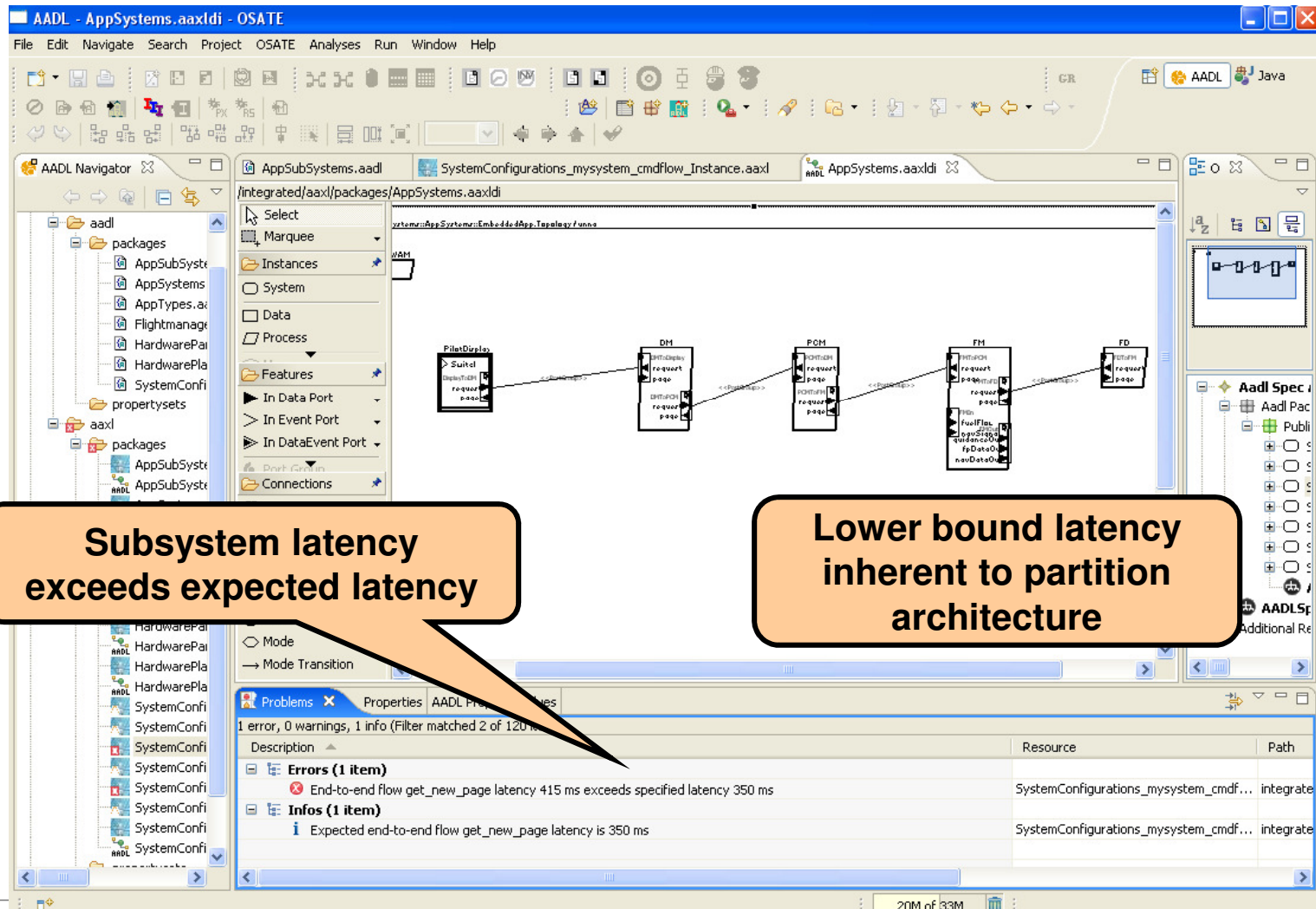




Flight Director Command Flow



Partition-Level Flow Latency



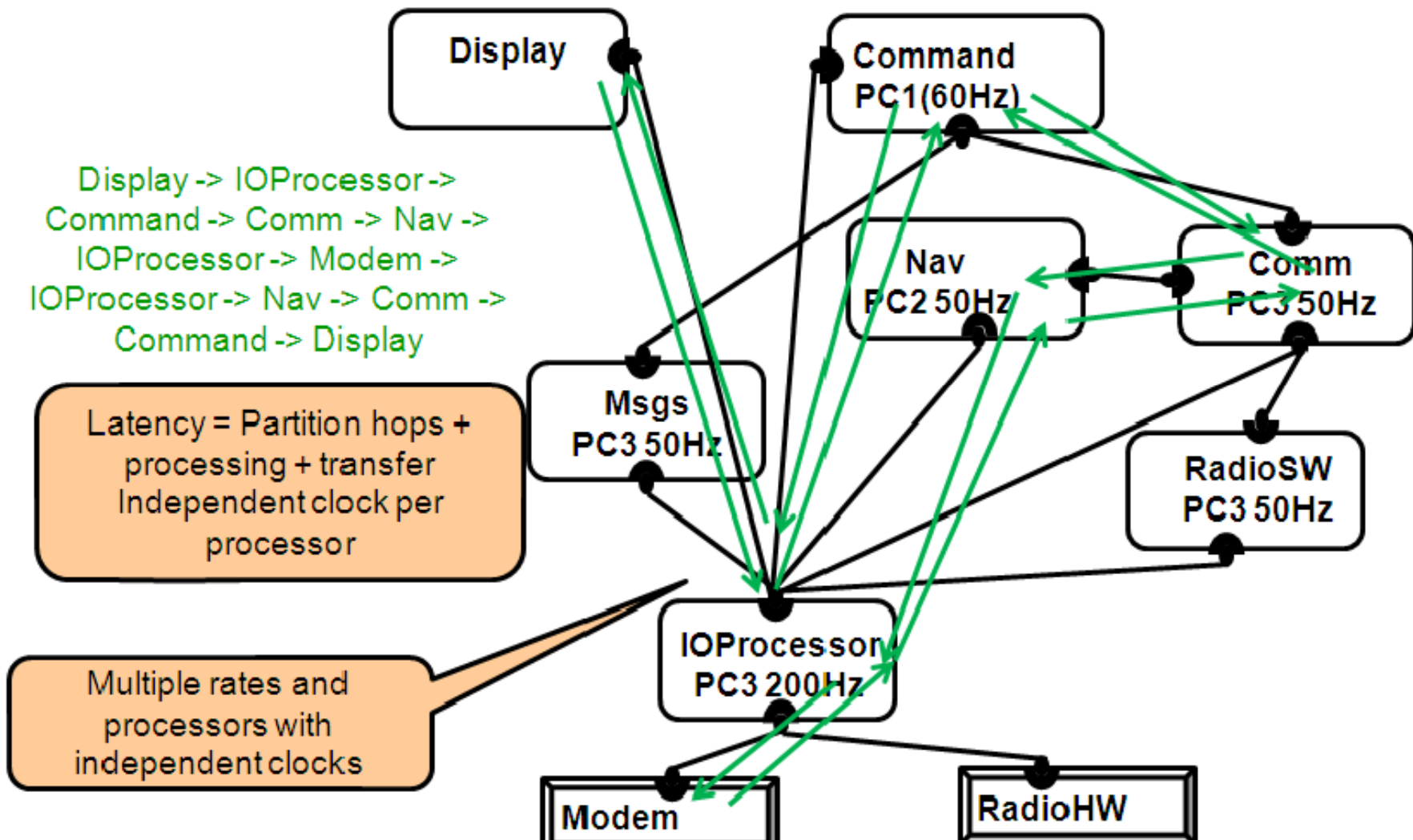
Subsystem latency exceeds expected latency

Lower bound latency inherent to partition architecture

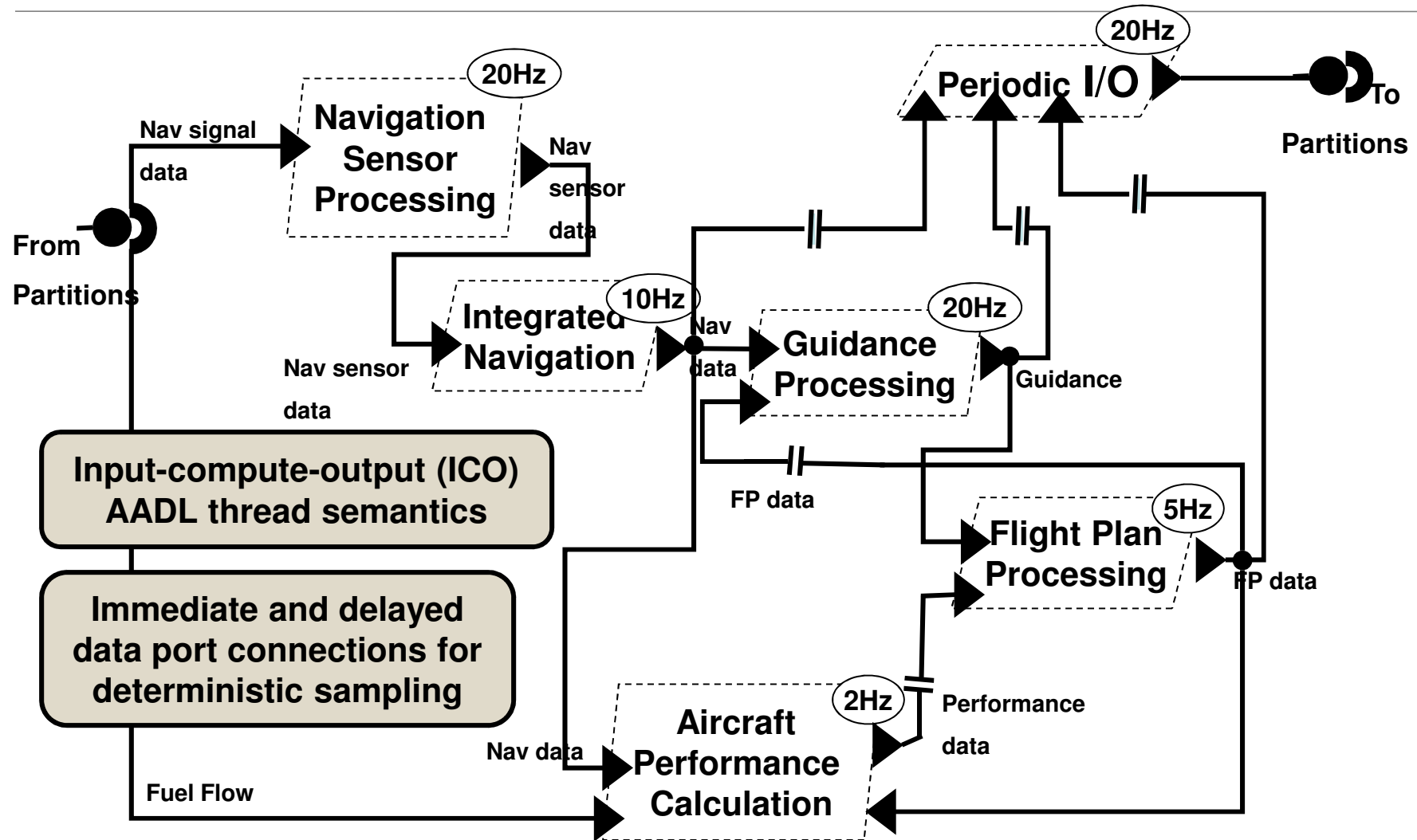
Description	Resource	Path
Errors (1 item)		
End-to-end flow get_new_page latency 415 ms exceeds specified latency 350 ms	SystemConfigurations_mysystem_cmdf...	integrate
Infos (1 item)		
Expected end-to-end flow get_new_page latency is 350 ms	SystemConfigurations_mysystem_cmdf...	integrate



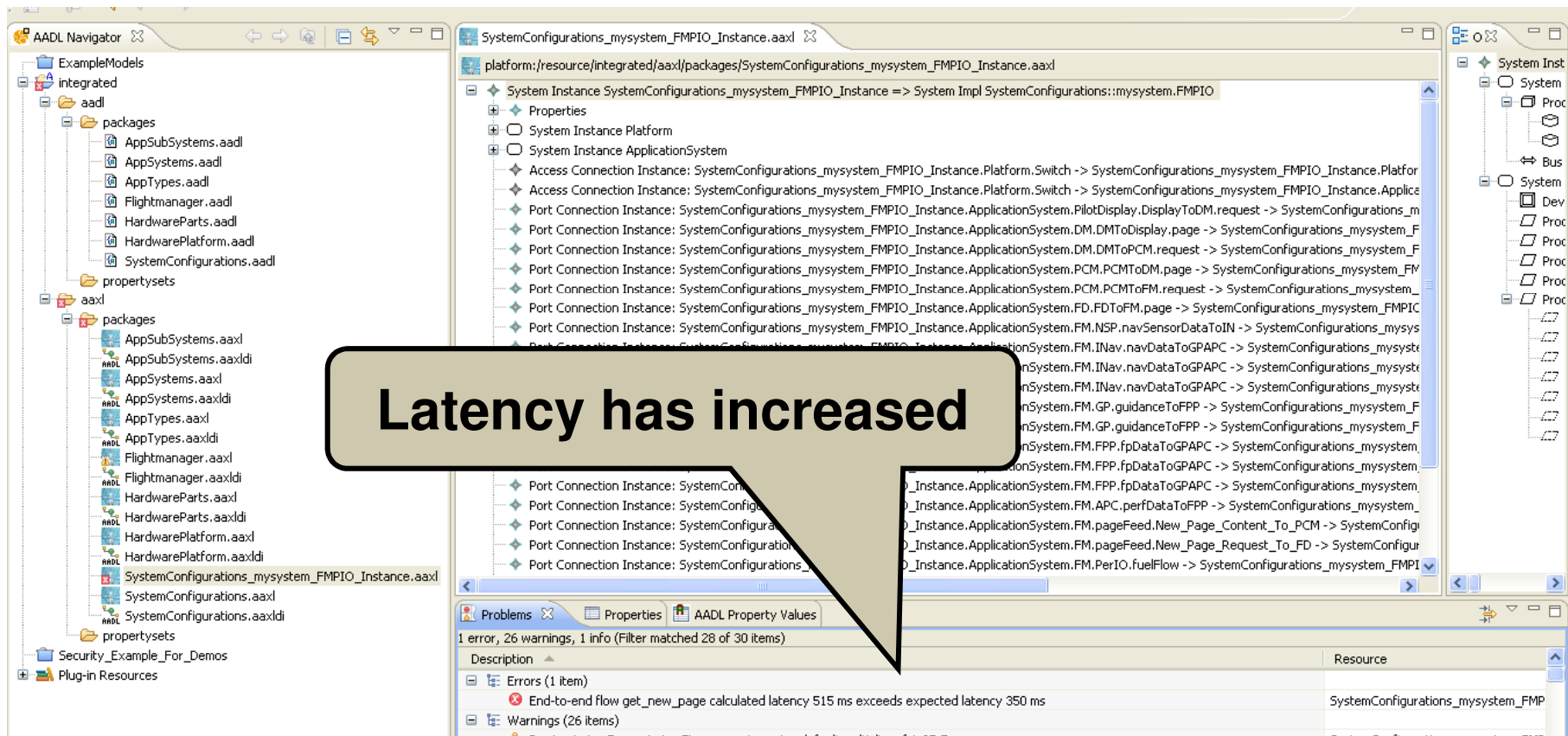
Flow Use Scenario through Subsystem Architecture



Managed Latency Jitter through Deterministic Sampling



Latency Revisited



Latency has increased

1 error, 26 warnings, 1 info (Filter matched 28 of 30 items)

Description	Resource
End-to-end flow get_new_page calculated latency 515 ms exceeds expected latency 350 ms	SystemConfigurations_mysystem_FMP
Bus Injection Transmission Time property uses default multiplier of 1.0E-5	SystemConfigurations_mysystem_FMP



Software-Based Latency & Jitter Contributors

Execution time variation: algorithm, use of cache

Processor speed

Resource contention

Preemption

Legacy & shared variable communication

Rate group optimization

Protocol specific communication delay

Partitioned architecture

Migration of functionality

Fault tolerance strategy



What If Scheduling Analysis

If a system is not schedulable

Automated allocation
based on constraints

Explore these options using AADL and analysis tools

- Leverage operational modes (higher fidelity)
- Use faster processor
- Add second processor
- Rewrite code to reduce worst-case execution time
- Consider lower signal processing rate for controller

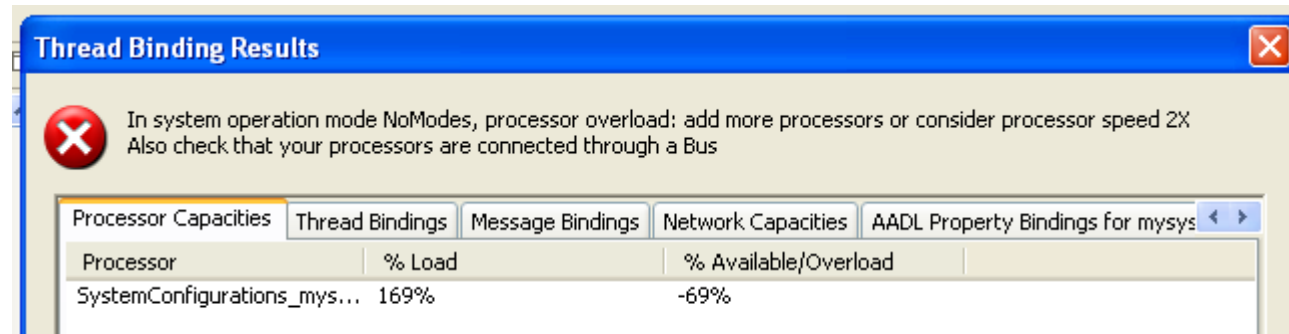
Allocation/scheduling
Binpacker (CMU)
Cheddar (U. Brest)
Versa (U.Penn)
RapidRMA (TriPacific)
In-house tools



What If Task Allocation & Schedulability

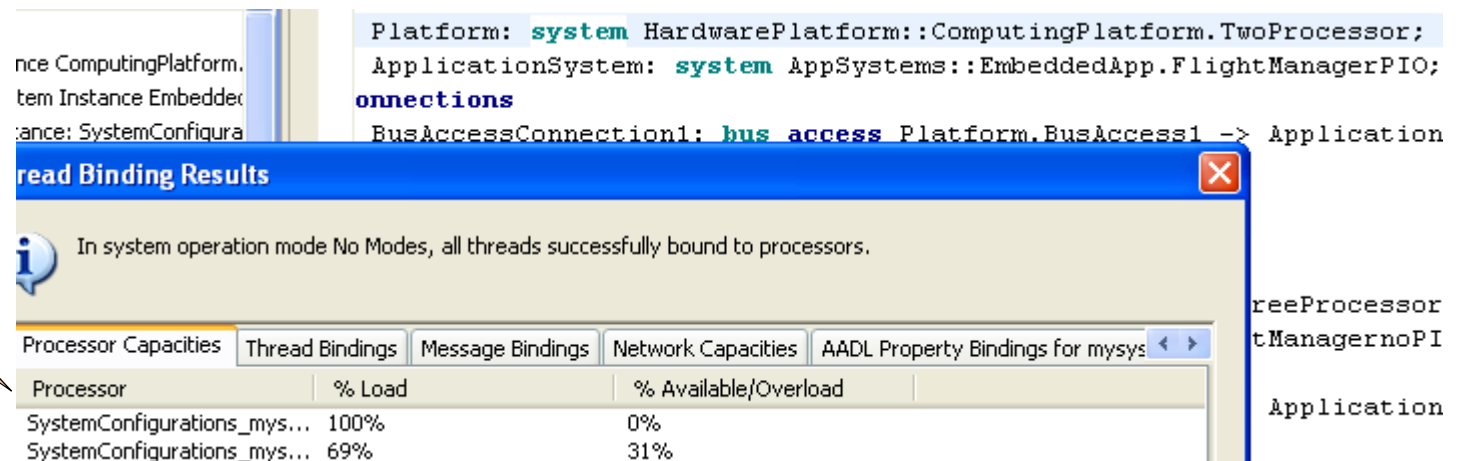
Task allocation & schedulability using binpacking technique with allocation constraints

Single processor



Processor	% Load	% Available/Overload
SystemConfigurations_mys...	169%	-69%

Connected 2 processor system



Processor	% Load	% Available/Overload
SystemConfigurations_mys...	100%	0%
SystemConfigurations_mys...	69%	31%

Extension of AADL: Security Modeling

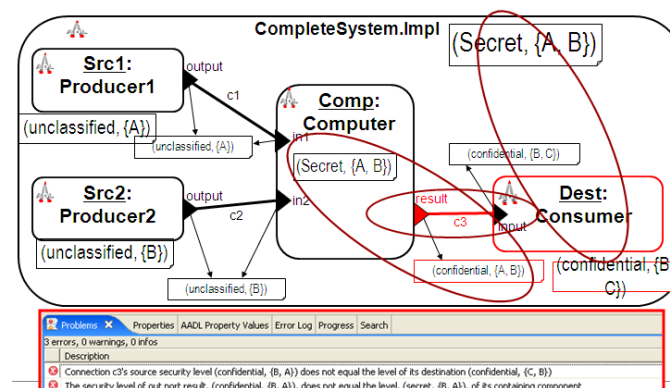
Confidentiality concerns that sensitive data should only be disclosed to or accessed/modified by authorized users, i.e., enforcing prevention of unauthorized disclosure of information.

Objective: Model security attributes for an architecture to verify that data is properly accessed and handled by users and applications.

Confidentiality frameworks

- Bell-LaPadula framework: military applications
- Chinese wall framework: commercial applications
- Access role/role-based access framework

Low fidelity consistency checking of security levels



AADL and Safety-Criticality

Fault management

- Architecture patterns in AADL
 - Redundancy, health monitoring, ...
- Fault tolerant configurations & modes

Dependability

- Error Model Annex to AADL
- Specification of fault occurrence and fault propagation information
- Use for hazard and fault effect modeling
- Reliability & fault tree analysis

Behavior validation

- Behavior Annex to AADL
- Model checking
- Source code validation

Consistency checking of safety-criticality levels

```

package errormodels
public
  annex error_model (**
    -- simple error model
  error model Basic
  features
    Failed : error event;

    Error_Free: initial error state;
    Permanent_Failure: error state;

    Visible_Failure: in out error propagation;
  end Basic;

  error model implementation Basic.Nominal
  transitions
    Error_Free -[Failed, in Visible_Failure]-> Permanent_Failure;
    Permanent_Failure -[out Visible_Failure]-> Permanent_Failure;
  properties
    Occurrence => poisson 10E-4 applies to Failed;
    Occurrence => poisson 10E-6 applies to Visible_Failure;
  end Basic.Nominal;

```

Consistency Checking of Domain Information

Data range limits and units of measurement for input & output

Setpoint constraints on data streams as bounded value deltas

Expected miss rates and miss rate contributors

State vs. state delta & guaranteed delivery





Software Engineering Institute

Carnegie Mellon

Peter H Feiler

phf@sei.cmu.edu

www.aadl.info

NO WARRANTY

THIS CARNEGIE MELLON UNIVERSITY AND SOFTWARE ENGINEERING INSTITUTE MATERIAL IS FURNISHED ON AN “AS-IS” BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT.

Use of any trademarks in this presentation is not intended in any way to infringe on the rights of the trademark holder.

This Presentation may be reproduced in its entirety, without modification, and freely distributed in written or electronic form without requesting formal permission. Permission is required for any other use. Requests for permission should be directed to the Software Engineering Institute at permission@sei.cmu.edu.

This work was created in the performance of Federal Government Contract Number FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center. The Government of the United States has a royalty-free government-purpose license to use, duplicate, or disclose the work, in whole or in part and in any manner, and to have or permit others to do so, for government purposes pursuant to the copyright license under the clause at 252.227-7013.

