Architecture-centric Strategies for Addressing Challenges in Software-reliant Safety-critical Systems

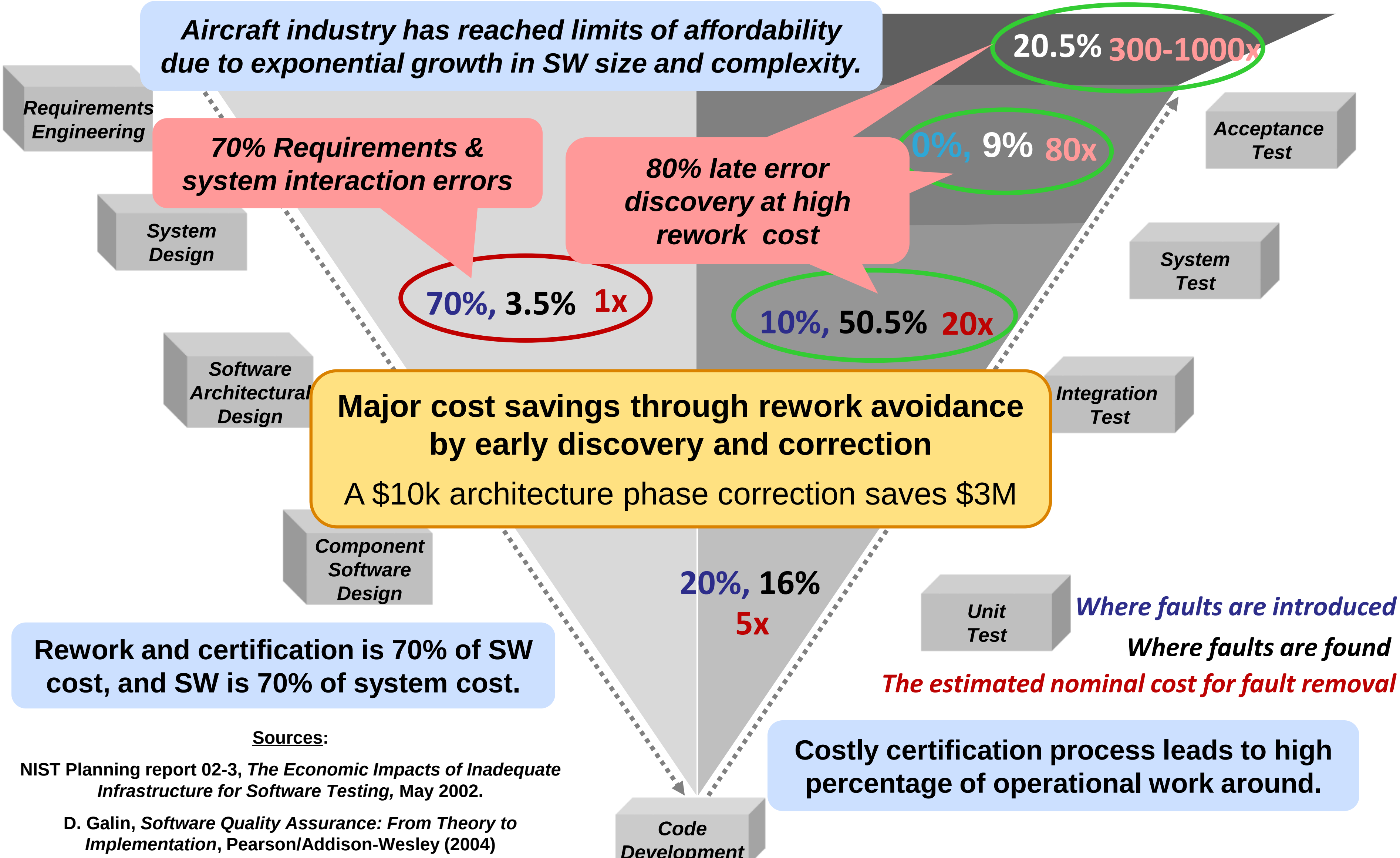
Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213

Peter H. Feiler
Oct 29, 2013

Outline

- Software Related Safety Hazards
- SAE AADL and Virtual Integration
- System and SW Architecture Fault Modeling & Analysis
- From Requirements to Managed Verification Evidence

High Fault Leakage Drives Major Increase in Rework Cost



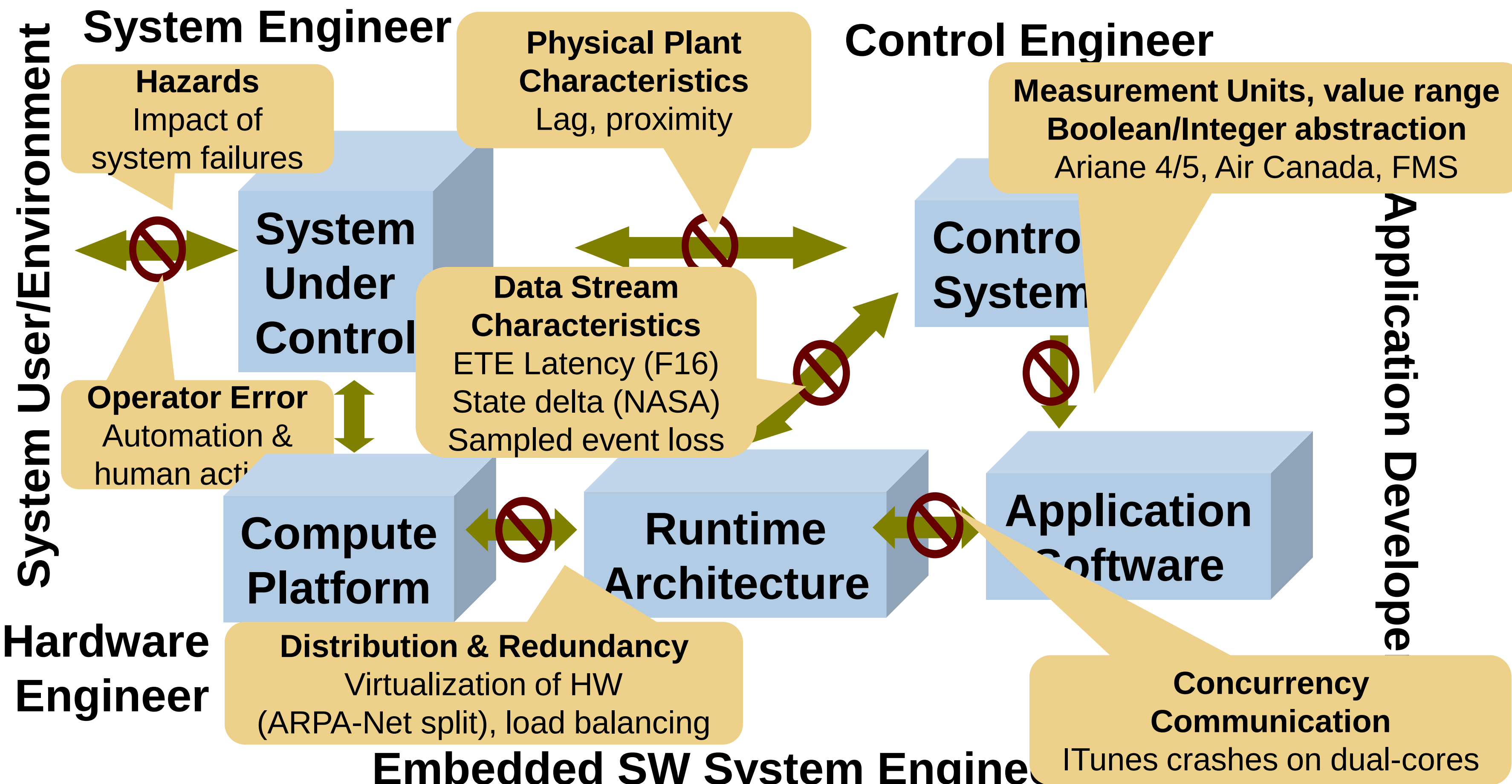
Sources:

NIST Planning report 02-3, *The Economic Impacts of Inadequate Infrastructure for Software Testing*, May 2002.

D. Galin, *Software Quality Assurance: From Theory to Implementation*, Pearson/Addison-Wesley (2004)

B.W. Boehm, *Software Engineering Economics*, Prentice Hall (1981)

Mismatched Assumptions in Embedded SW

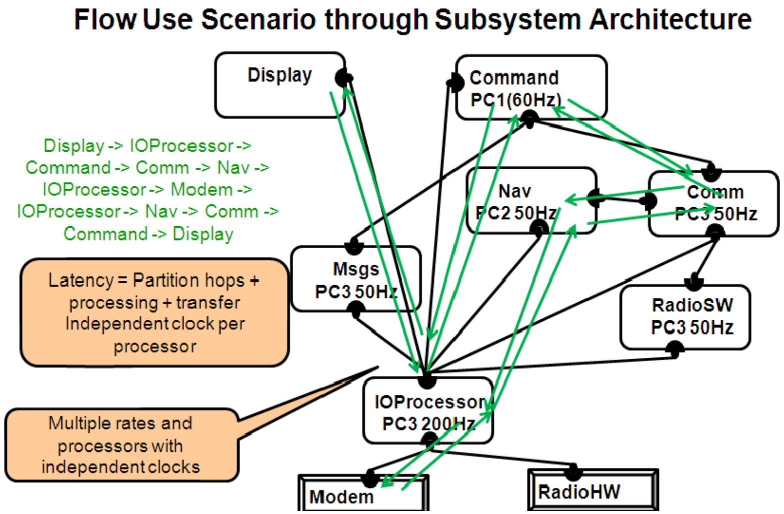


Software responsible for monitoring and managing system health
No Zero defect assumption for SW

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

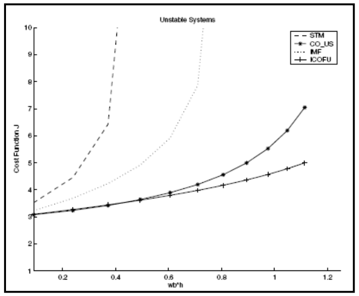
Software-Based Latency Contributors

- 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



Impact of Scheduler Choice on Control System Stability

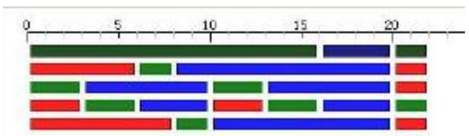
A. Cervin, Lund U., CCACSD 2006

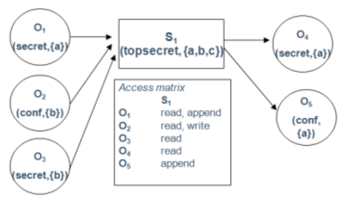


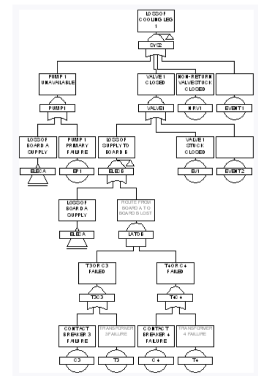
Potential Model-based Engineering Pitfalls

The system

Inconsistency between independently developed analytical models

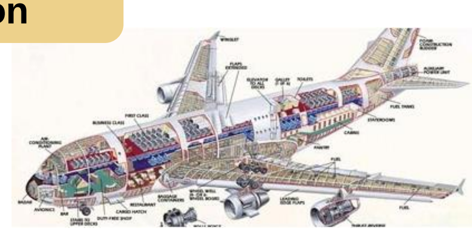






System models

Lack of confidence that model reflects implementation



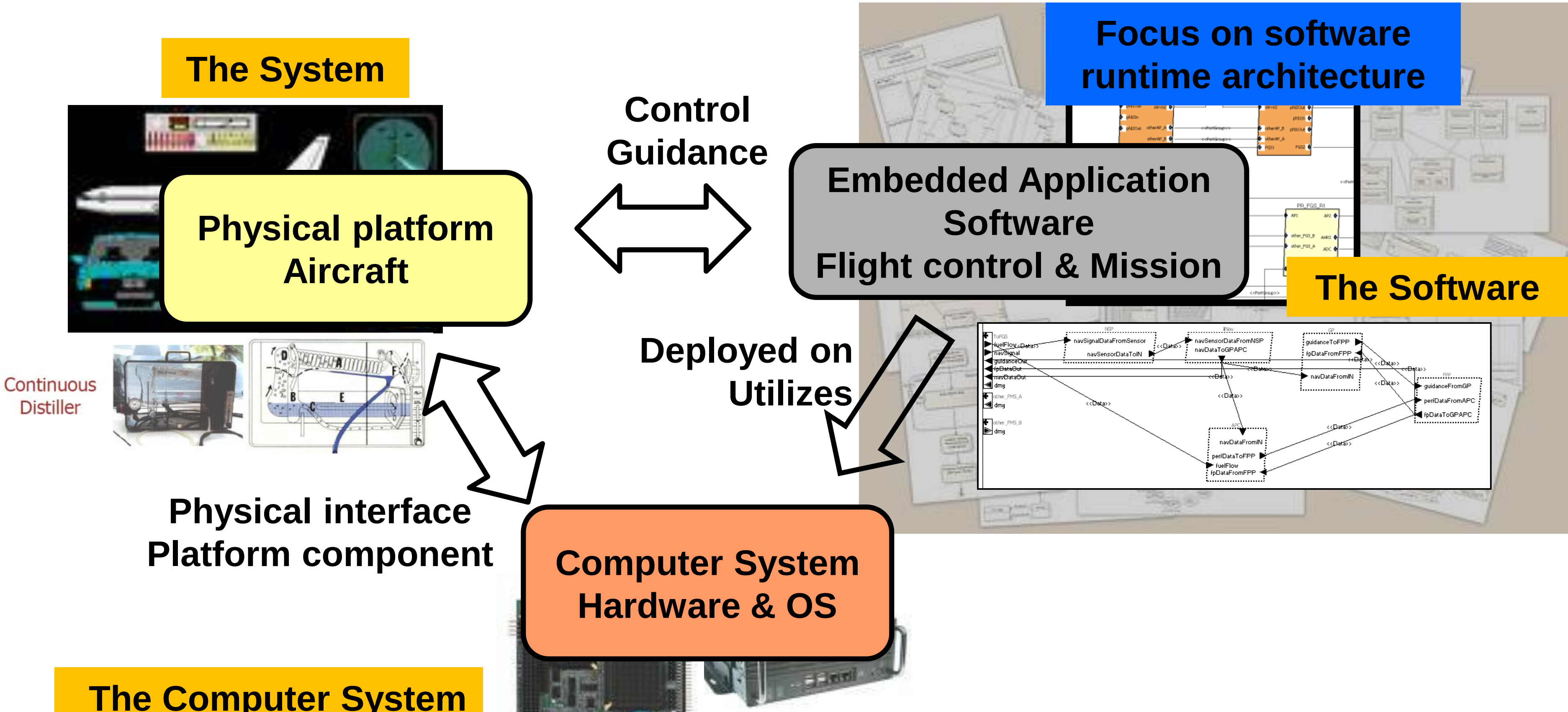
System implementation

Aircraft industry experience has led to single truth requirement in the System Architecture Virtual Integration (SAVI) initiative

Outline

- Software Related Safety Hazards
- ▶ SAE AADL and Virtual Integration
- System and SW Architecture Fault Modeling & Analysis
- From Requirements to Managed Verification Evidence

SAE Architecture Analysis & Design Language (AADL) for Software-reliant Systems



AADL focuses on interaction between the three elements of a software-intensive system based on architectural abstractions of each.

The SAE AADL Standard Suite (AS-5506 series)

Core AADL language standard (V2.1-Sep 2013, V1-Nov 2004)

- Strongly typed textual & graphical notation, Meta model & XMI interchange format
- Thread, process, system, processor, memory, bus, device, virtual processor, virtual bus
- Sampling and queuing ports, (non)deterministic sampling, end-to-end flows, modes
- Dispatch protocol, scheduling protocol, input/output timing and rates, queuing behavior
- Packages, refinement/extensions, abstract components and features, parameterization

AADL Meta model & XMI/XML standard

- Model interchange & tool interoperability

AADL Annexes (Extensions) [2006, 2012]

- Error Model Annex for dependability analysis
- ARINC653 Annex for partitioned architectures
- Behavior Annex for formal behavior specification
- Data Modeling Annex for interfacing with data models
- Requirements Definition and Analysis Annex
- Constraint Annex
- Code Generation Annex

System Level Fault Root Causes

Processing of Data Streams in Time-Sensitive Manner

- Stream miss rates, Mismatched data representation, Latency jitter & age
- Sampling, frame-level jitter, and loss of state change data/events

End-to-end latency analysis
Port connection consistency

Use of partitioned architectures (virtual machines) for fault containment

- Mixed criticality in safety and security concerns
- Logical vs. physical redundancy of resources
- Virtualization of time and time sensitive processing
- Asynchronous systems

Virtual processors & buses
Synchronization domains

Inconsistent System States & Interactions

- Modal systems with modal components
- Failure and operational modes
- Replicated, mirrored, and coordinated state machines

Fault modeling
Security analysis
Architectural redundancy patterns

Resource management

- Resource budgets for processor, memory & networks
- Mismatch of resource demand and capacity
- Unmanaged computer system resources

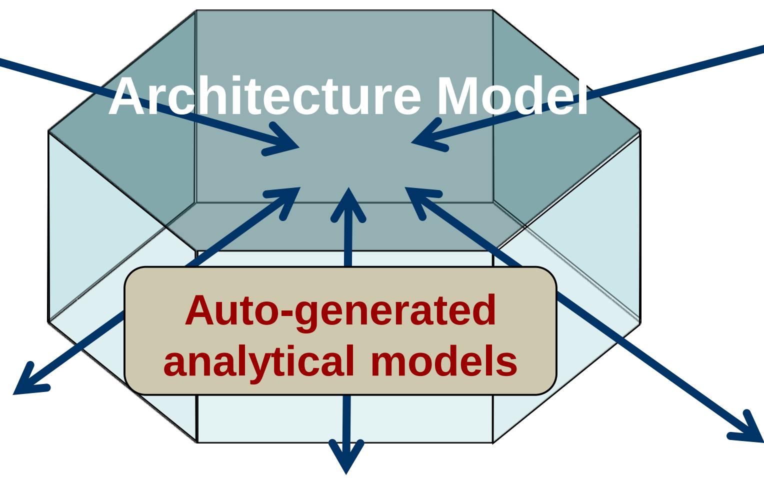
Resource budget analysis & task roll-up analysis
Resource allocation & deployment configurations

Architecture-Centric Modeling Approach

Single Annotated Architecture Model Addresses Impact Across Non-Functional Properties

Safety & Reliability
- MTBF
- FMEA
- Hazard analysis

Security
- Intrusion
- Integrity
- Confidentiality

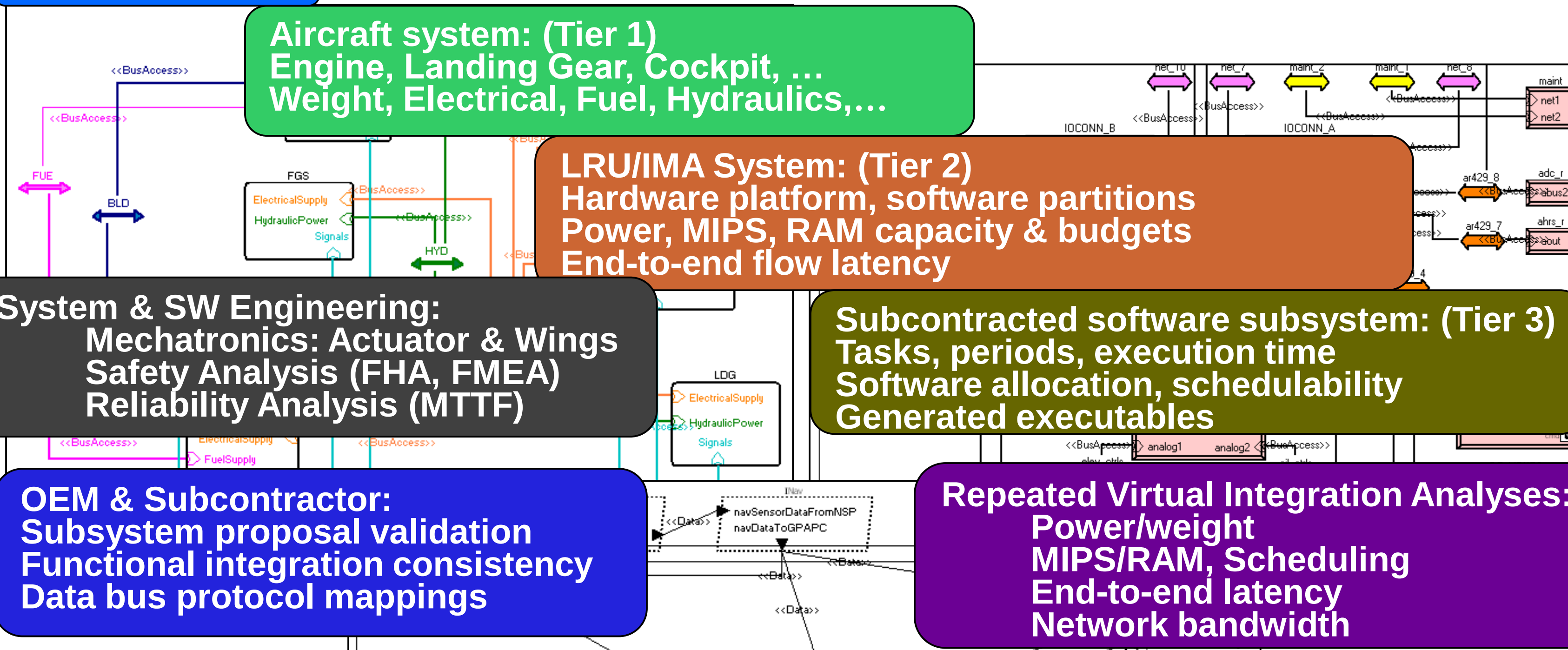


Data Quality
- Data precision/ accuracy
- Temporal correctness
- Confidence

Real-time Performance
- Execution time/ Deadline
- Deadlock/starvation
- Latency

Resource Consumption
- Bandwidth
- CPU time
- Power consumption

Early Discovery and Incremental V&V through Virtual Integration (SAVI)



Aircraft: (Tier 0)

Aircraft system: (Tier 1)
Engine, Landing Gear, Cockpit, …
Weight, Electrical, Fuel, Hydraulics,…

LRU/IMA System: (Tier 2)
Hardware platform, software partitions
Power, MIPS, RAM capacity & budgets
End-to-end flow latency

System & SW Engineering:
- **Mechatronics: Actuator & Wings**
- **Safety Analysis (FHA, FMEA)**
- **Reliability Analysis (MTTF)**

Subcontracted software subsystem: (Tier 3)
Tasks, periods, execution time
Software allocation, schedulability
Generated executables

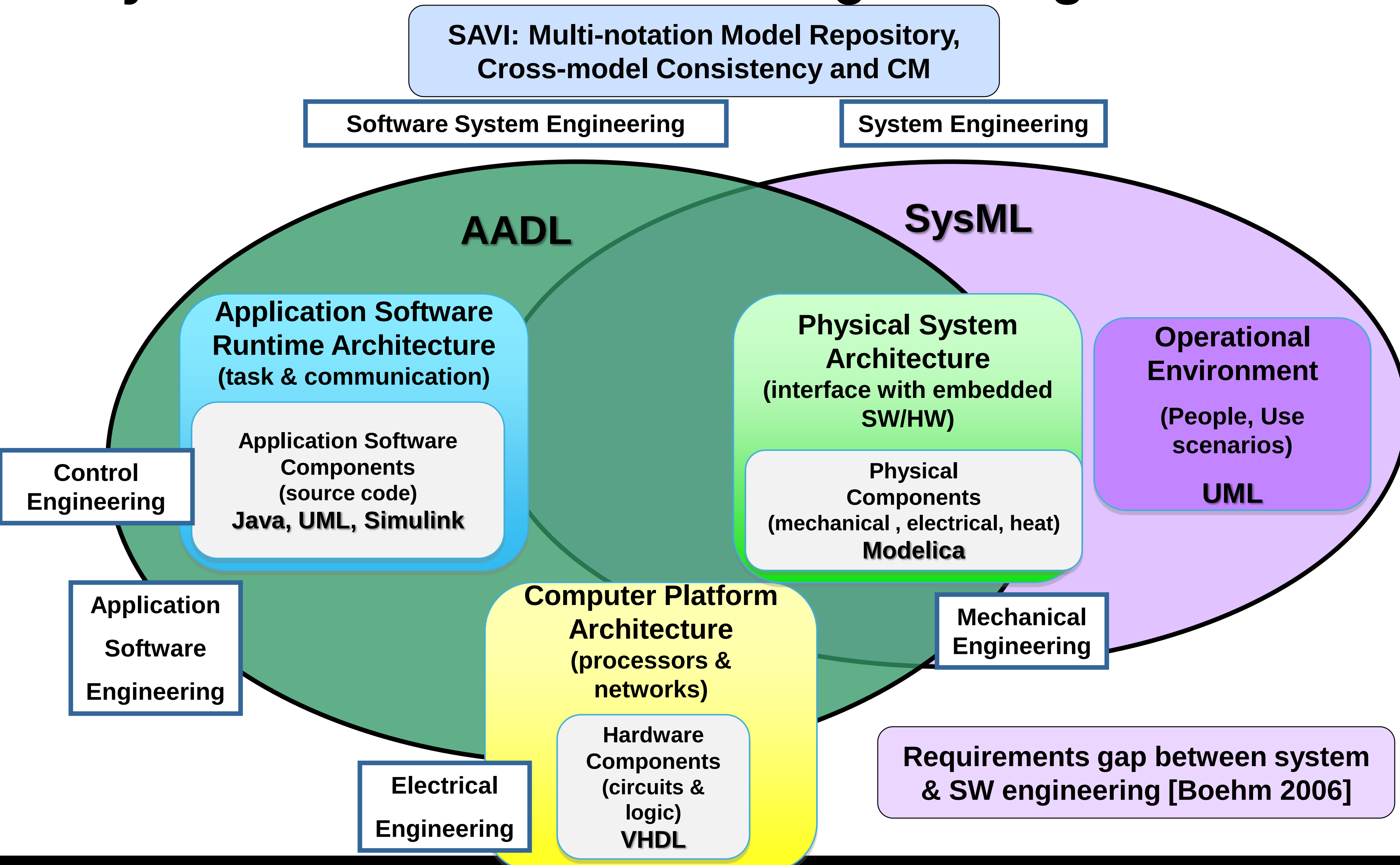
OEM & Subcontractor:
Subsystem proposal validation
Functional integration consistency
Data bus protocol mappings

Repeated Virtual Integration Analyses:
- **Power/weight**
- **MIPS/RAM, Scheduling**
- **End-to-end latency**
- **Network bandwidth**

Proof of Concept Demonstration and Transition by Aerospace industry initiative
- **Propagate requirements and constraints**
- **Higher level model down to suppliers' lower level models**
- **Verification of lower level models satisfies higher level requirements and constraints**

- Multi-tier system & software architecture (in AADL)
- Incremental end-to-end validation of system properties

System and Software Co-Engineering



Requirements gap between system & SW engineering [Boehm 2006]

Outline

Software Related Safety Hazards

SAE AADL and Virtual Integration

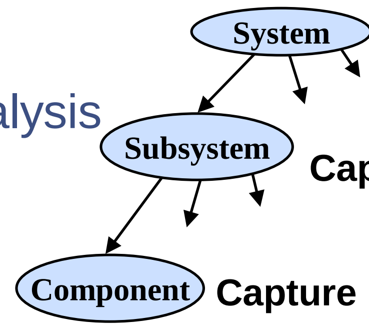
▶ System and SW Architecture Fault Modeling & Analysis

From Requirements to Managed Verification Evidence

AADL Error Model Scope and Purpose

System safety process uses many individual methods and analyses, e.g.

- hazard analysis
- failure modes and effects analysis
- fault trees
- Markov processes



Capture hazards

Capture risk mitigation architecture

Capture FMEA model

Goal: a general facility for modeling fault/error/failure behaviors that can be used for several modeling and analysis activities.

Annotated architecture model permits checking for consistency and completeness between these various declarations.

Related analyses are also useful for other purposes, e.g.

- maintainability
- availability
- Integrity
- Security

SAE ARP 4761 *Guidelines and Methods for Conducting the Safety Assessment Process on Civil Airborne Systems and Equipment*
Demonstrated in SAVI Wheel Braking System Example

Error Model Annex can be adapted to other ADLs

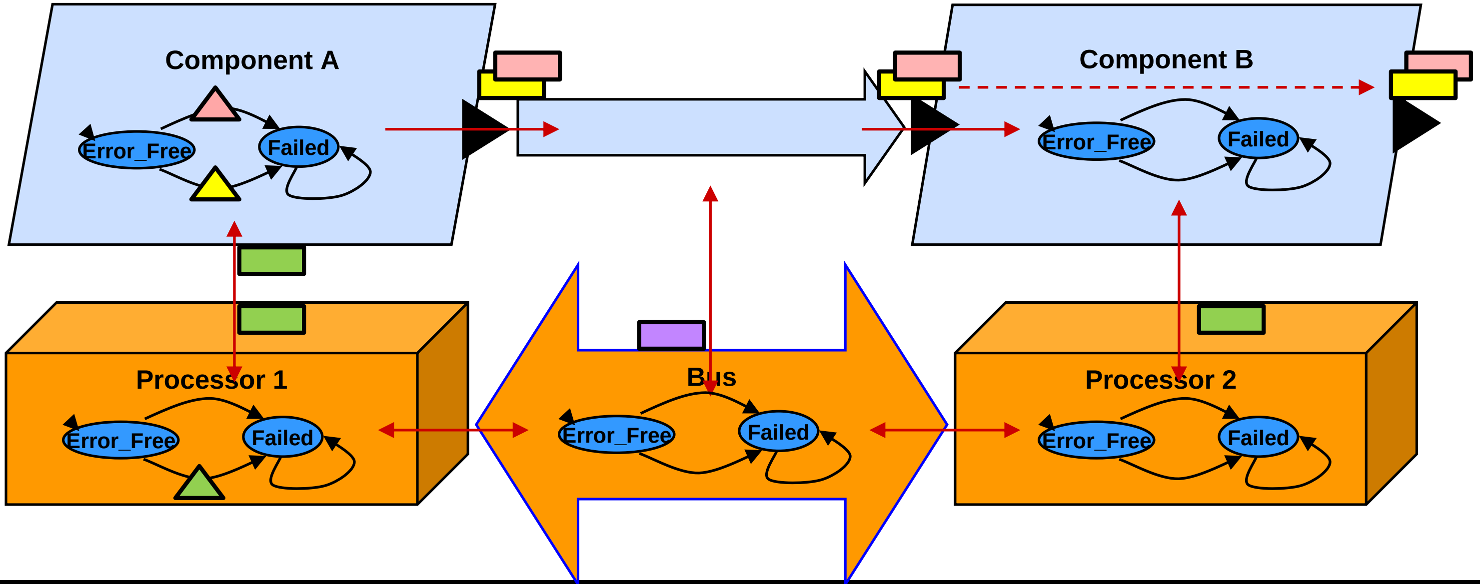
Error Model V2 Annex Overview

Three levels of abstraction:

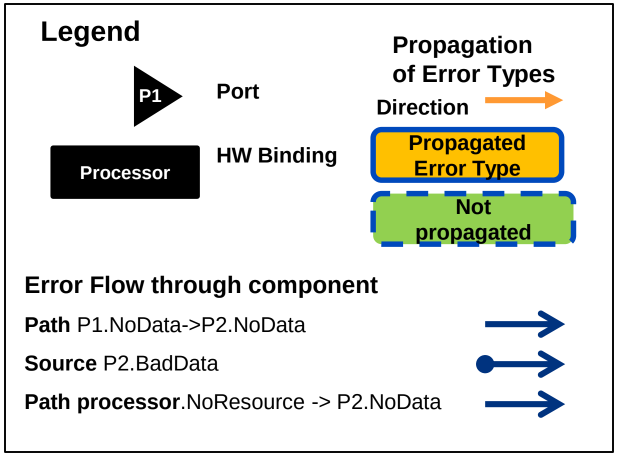
- Focus on fault interaction with other components
- Focus on fault behavior of components
- Focus on fault behavior in terms of subcomponent fault behaviors

Specification of expected fault management strategy and realization

- Voting logic, error detection, recovery, repair



Error Propagation Specification

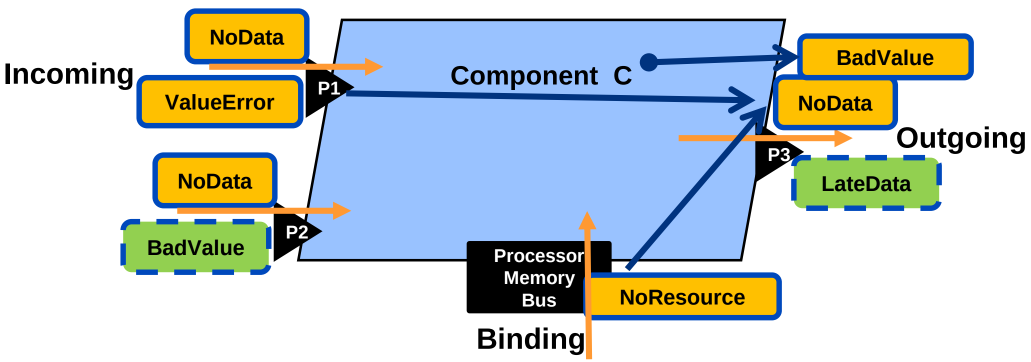


Error Flow:

Path P1.NoData->P3.NoData

Source P2.BadData;

Path processor.NoResource -> P2.NoData



"Not" indicates that this error type is not intended to be propagated.

This allows us to determine whether propagation specification is complete.

Incoming/Assumed Outgoing/Guarantee

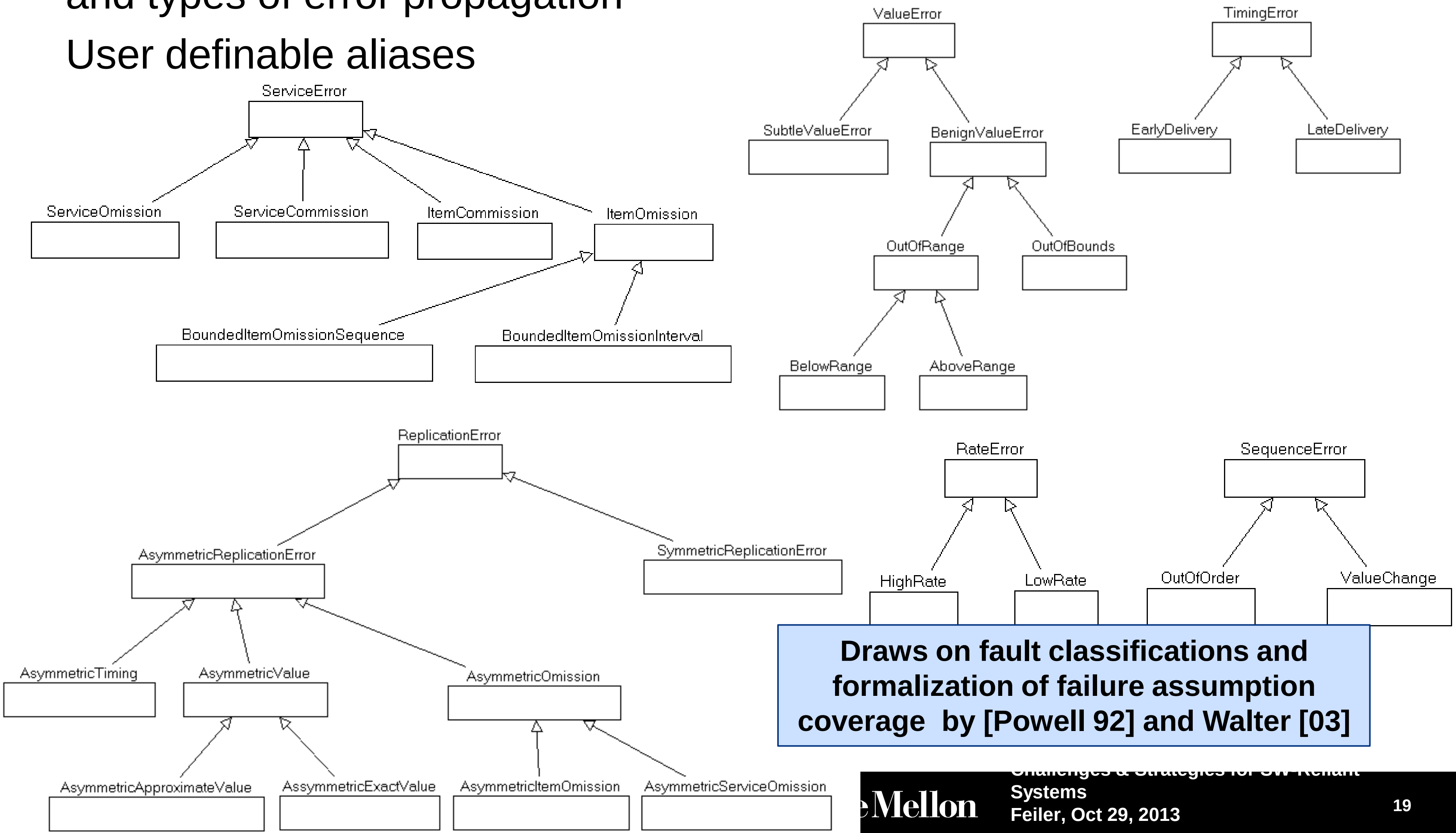
Error propagation and flow specification supports fault impact analysis based on a Fault Propagation and Transformation Calculus (FPTC)

An Extensible Set of Error Propagation Types

User definable error types, type sets, type hierarchies, and type products

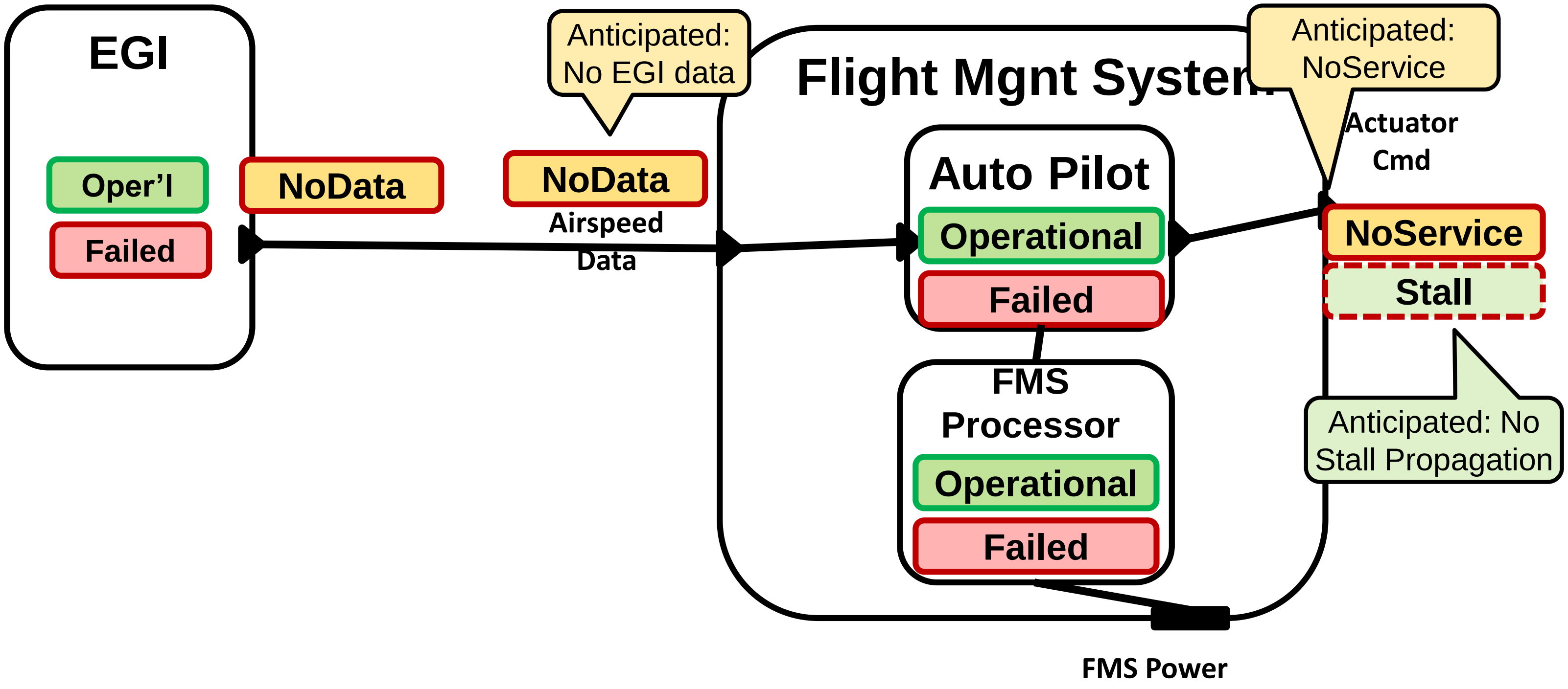
Can be combined to characterize failure modes, resulting error states, and types of error propagation

User definable aliases



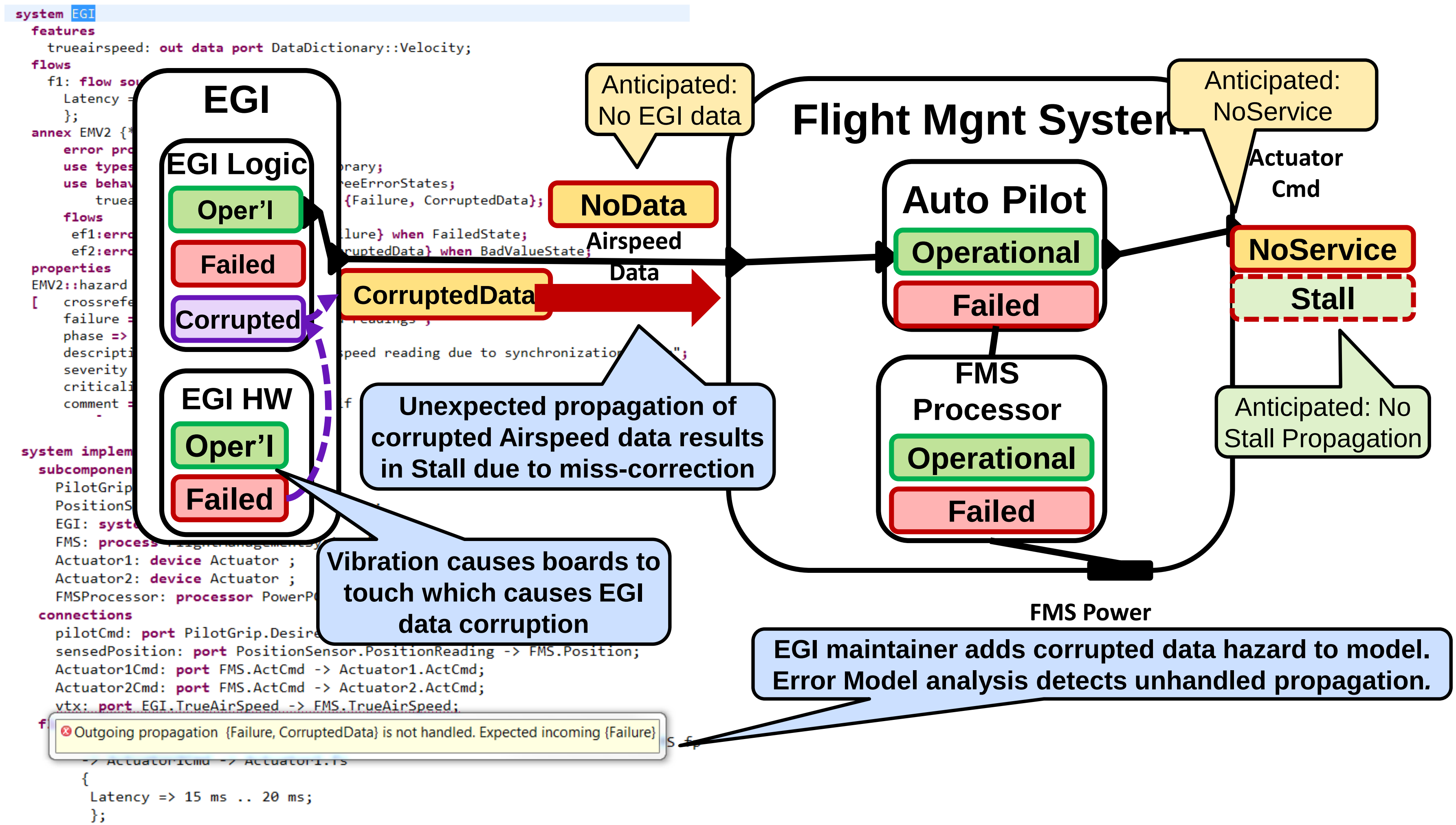
Draws on fault classifications and formalization of failure assumption coverage by [Powell 92] and Walter [03]

Original Preliminary System Safety Analysis (PSSA)



System engineering activity with focus on failing components.

Discovery of Unexpected PSSA Hazard



Recent Automated FMEA Experience

Failure Modes and Effects Analyses are rigorous and comprehensive reliability and safety design evaluations

- Required by industry standards and Government policies
- When performed manually are usually done once due to cost and schedule
- If automated allows for
  - multiple iterations from conceptual to detailed design
  - Tradeoff studies and evaluation of alternatives
  - Early identification of potential problems

ID	Item	Initial State	Initial Failure Mode	1st Level Effect	Transition	2nd Level Effect	Transition	3rd Level Effect	Severity	M
1	Sat_Bus	Working	Failure	Failed		Failed	Recovery	Working		Workir
1	Sat_Payload	Working		Working	Bus failure causes payload transition	Standby		Standby	Bus Recovery Causes Payload Transition	Workir
2	Sat_Bus	Working		Working		Working	5			
2	Sat_Payload	Working	Failure	Failed	Recovery	Working	5			

Largest analysis of satellite to date consists of 26,000 failure modes

- Includes detailed model of satellite bus
- 20 states perform failure mode
- Longest failure mode sequences have 25 transitions (i.e., 25 effects)

Myron Hecht, Aerospace Corp.
Safety Analysis for JPL, member of DO-178C committee

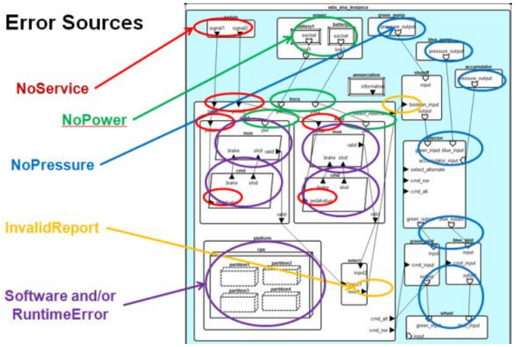
Automation of Safety Analysis Practice

A public Aircraft Wheel Brake System model

https://wiki.sei.cmu.edu/aadl/index.php/ARP4761_-_Wheel_Brake_System_%28WBS%29_Example

Use of Error-Model and ARINC653 annexes

Relevance for the avionics community



Comparative study

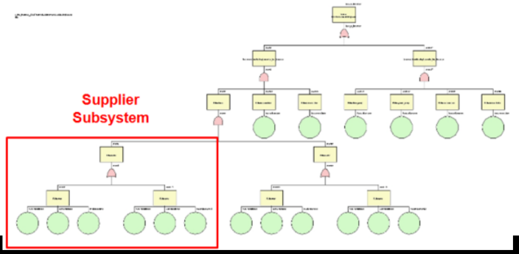
Federated vs. Integrated Modular Avionics (IMA) architecture

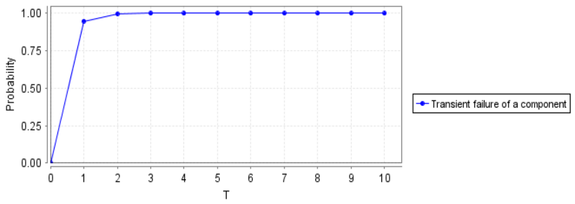
Support of SAE ARP 4761 System Safety Assessment Practice

Hazards (FHA), Fault Trees (FTA), Fault Impact (FMEA)

Reliability/Availability (Markov Chain/Dependence Diagram)

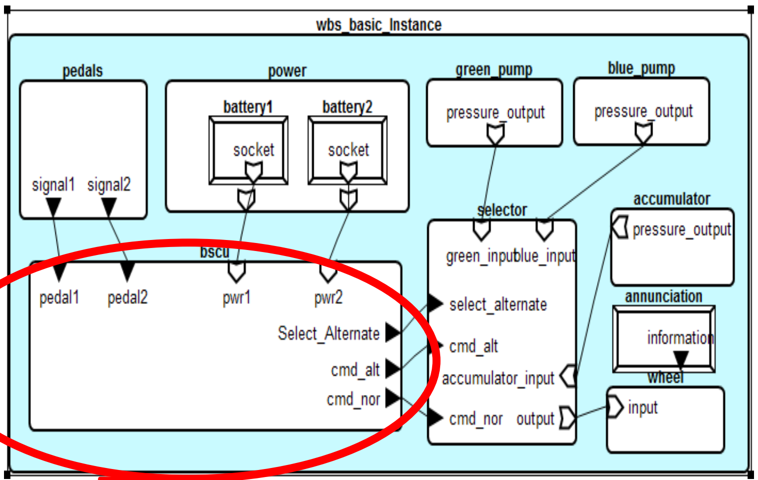
Function Name	Failure Mode	Failure Rate (E-6)	Flight Phase	Failure Effect	Detection Method	Comments
+5 Volt	+5V out of spec.	0.2143	All	Possible P/S shutdown	Power Supply Monitor trips, shuts down supply and passes "invalid power supply (P/S)" to other BSCU system	BSCU channel fails
	+5V short to ground	0.2857	All	P/S shutdown	Power supply monitor passes invalid P/S to other BSCU system	BSCU channel fails
	Loss of / reduced filtering	0.3571	All	Increase Ripple	May pass out of spec voltage to rest of BSCU if ripple is such that it is not detected by the P/S monitor	May cause spurious P/S monitor trip
	+5V open	0.5714	All	P/S shutdown	Power supply monitor passes invalid P/S to other BSCU system	BSCU channel fails
	No Effect	0.1429	All	No Effect	None/No Effect	No Effect
Total Failure Rate of +5V Supply		1.5714				



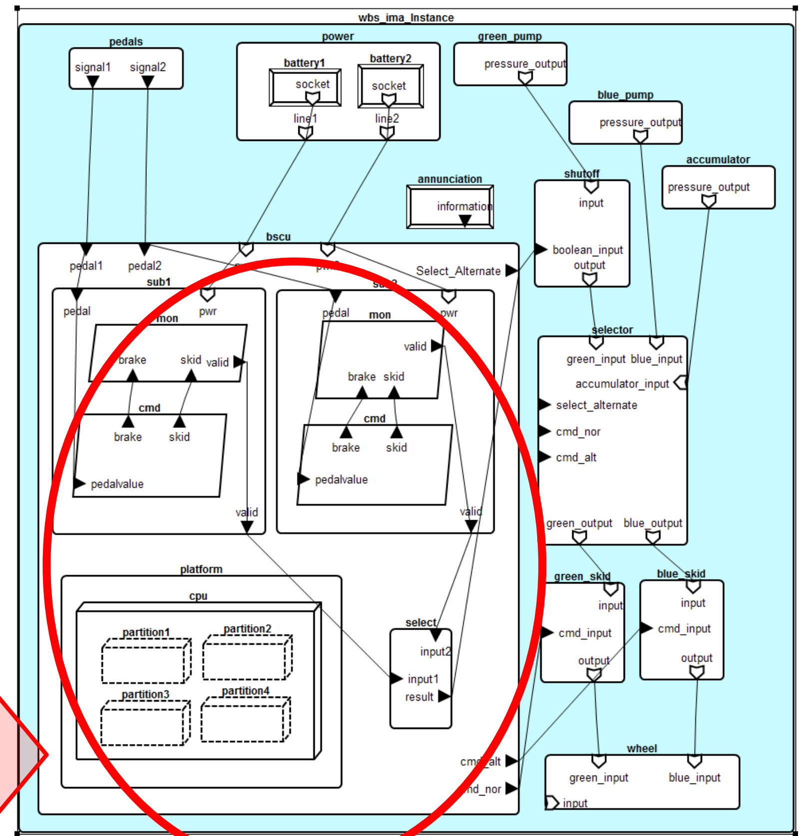


Scalability and Incremental Safety Analysis

Abstract and Composite Error Model Specification at each architecture layer



Component extension, refinement & implementation



AADL model Version n

AADL model Version n + 1

Development Process

Archetype-based Fault & Hazard Identification

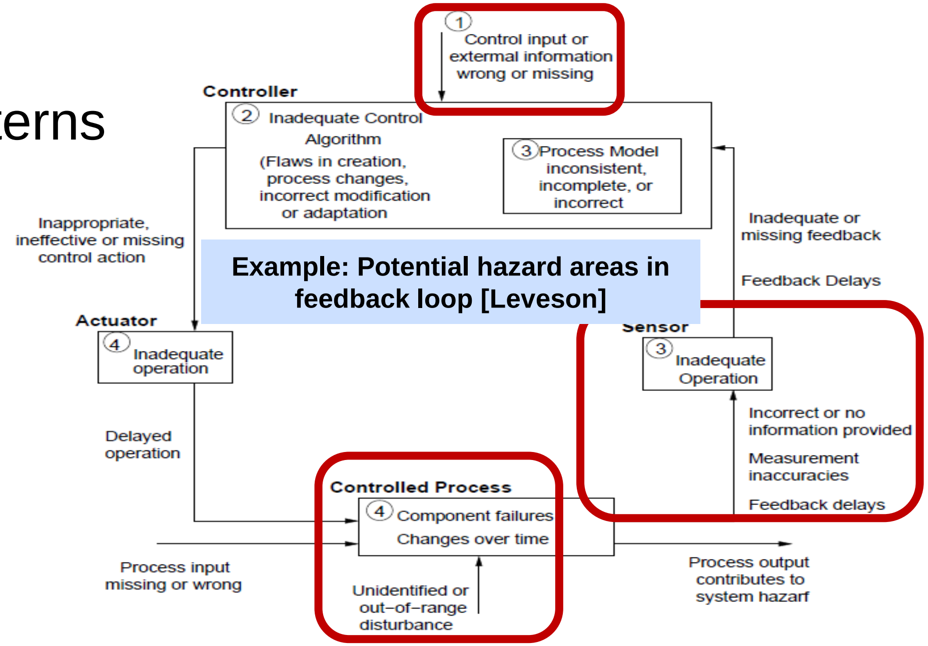
Application interaction architecture patterns

- Feedback control system
- Data, event, message, command streams
- State-based interaction protocols
- Multi-tier service layers

Fault management architecture patterns

- Redundancy
- Monitoring & recovery
- Partitions

Pre-analyzed architecture patterns enable analysis of potentially high-risk safety-criticality areas



Example: Potential hazard areas in feedback loop [Leveson]

Application as well as fault management architecture patterns have fault & hazard potential

Flow	Service		Timing		Value	
Type	*Omission*	*Commission*	*Early*	*Late*	*Subtle*	*Coarse*
Boolean	No Data	Extra Data	Early	Late	Stuck at...	N/A
Value	"	"	"	"	wrong in	out of ...ace
Complex	"					...istent

Example: Partitions limit error propagation to input/output errors [Rushby]

Fault & hazard types in common architecture patterns as starting point for FHA, FTA, FMEA, root cause analysis, and IV&V

Outline

Software Related Safety Hazards

SAE AADL and Virtual Integration

System and SW Architecture Fault Modeling & Analysis

▶ From Requirements to Managed Verification Evidence

Towards Analyzable Requirements Specification

Best practice industry study for FAA [2009]

- Primarily textual shalls , tables, and diagrams, MS Word and DOORS

FAA Requirements Engineering Management (REM) Handbook [2009]

- Draws on SpecTRM, Rockwell Collins experience with model checking
- 11 step process with avionics and medical device examples

Requirements Definition and Analysis Annex

- Separation of concerns: problem (requirements) / solution (design)
- Incorporates concepts from SysML, KAOS, URN, FAA REM Handbook
- Goals, requirements, refinement, decomposition, verification, risks
- Semantics: validation of requirement specifications, verification of formally specified requirements
- Extensible with respect to constraints, use cases, and traceability links
- Demonstrated on FAA REM Handbook process with medical device example
- Applicable to AADL and other ADLs

Formal V&V of Safety-critical System Requirements as Early Evidence

Reflected in: Requirements Engineering Management Handbook (FAA 2009)

Formalized requirements specification as best practice

- SCR (four variable model) [Parnas], SpecTRM [Leveson]
- From system to software requirements: system state behavior
- From hazards to safety requirements: intent & rationale
- Environmental assumptions, human factors

Pilots connecting safety-critical requirements to architecture & design

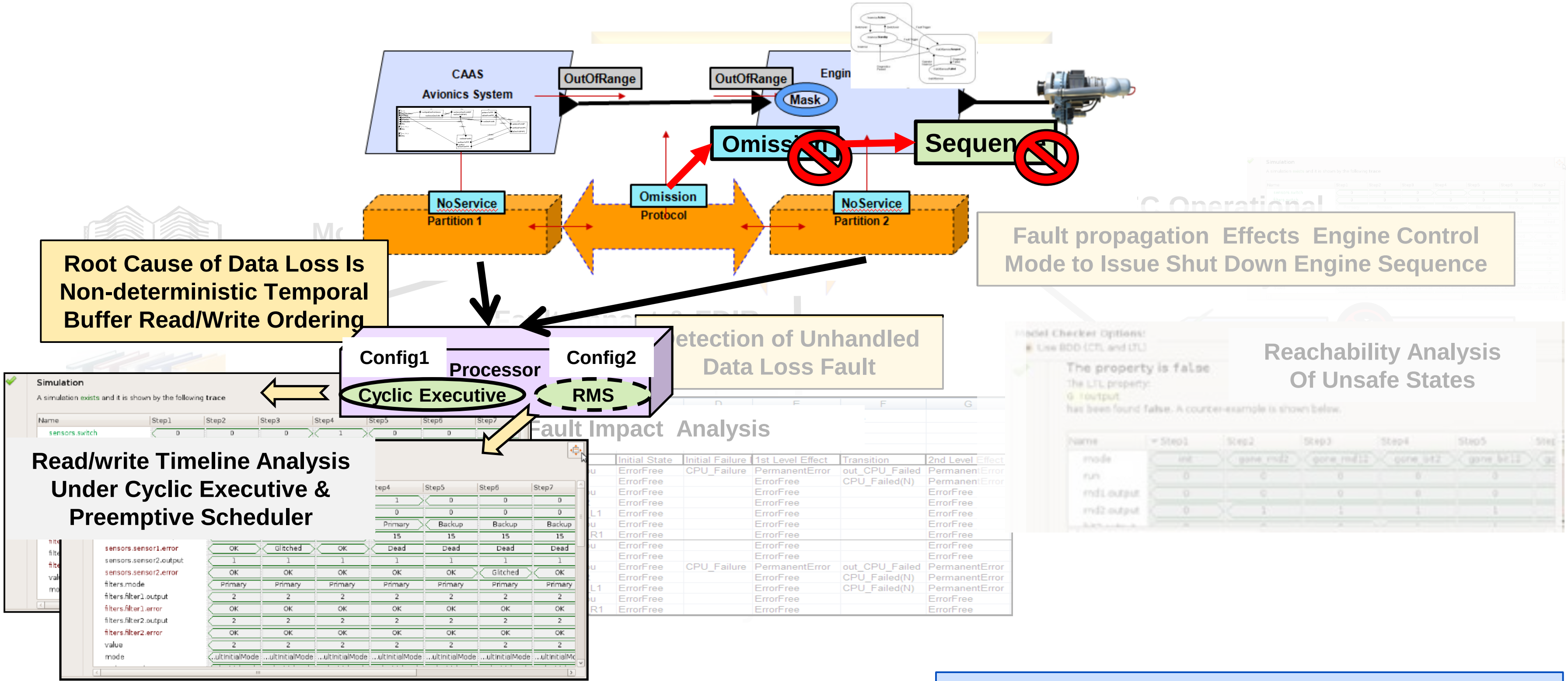
- FHA, FTA, FMEA based on AADL/Error Model Annex [Vestal, Hecht, Delange]
- SpecTRM & JPL Goal-oriented Mission State Analysis [Leveson/Weiss]
- JPL State Analysis & AADL/MBE [Weiss/Feiler NASA IV&V funded]

Pilots showing value of formal architecture-centric V&V

- Model checking to validate coverage of safe & unsafe system states by safety requirements [Tribble/Miller/Whalen, Nguyen/Noll]
- Verification of redundancy mode logic in nominal & abnormal conditions results in design revision, which introduces two new critical hazards [Miller/Whalen, DeNiz]

Understanding the Cause of Faults

Through model-based analysis identify architecture induced unhandled, testable, and untestable faults and understand root causes, contributing factors, impact, and potential mitigation options.



Demonstrated in COMPASS project

Use of text templates as formalism frontend

Stepper Motor Case Study

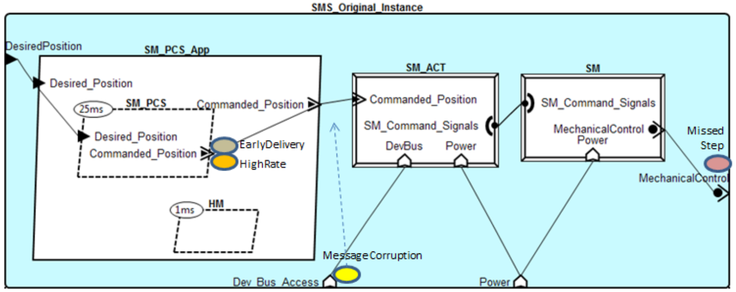
Stepper motor (SM) controls a valve

- Commanded to achieve a specified valve position
- Controller instructs the motor to move in up 15 step increments per 25ms frame
- Execution time jitter & health monitor preemption causes missed steps

Software modeled and verified in SCADE

Architecture Fault Model Analysis

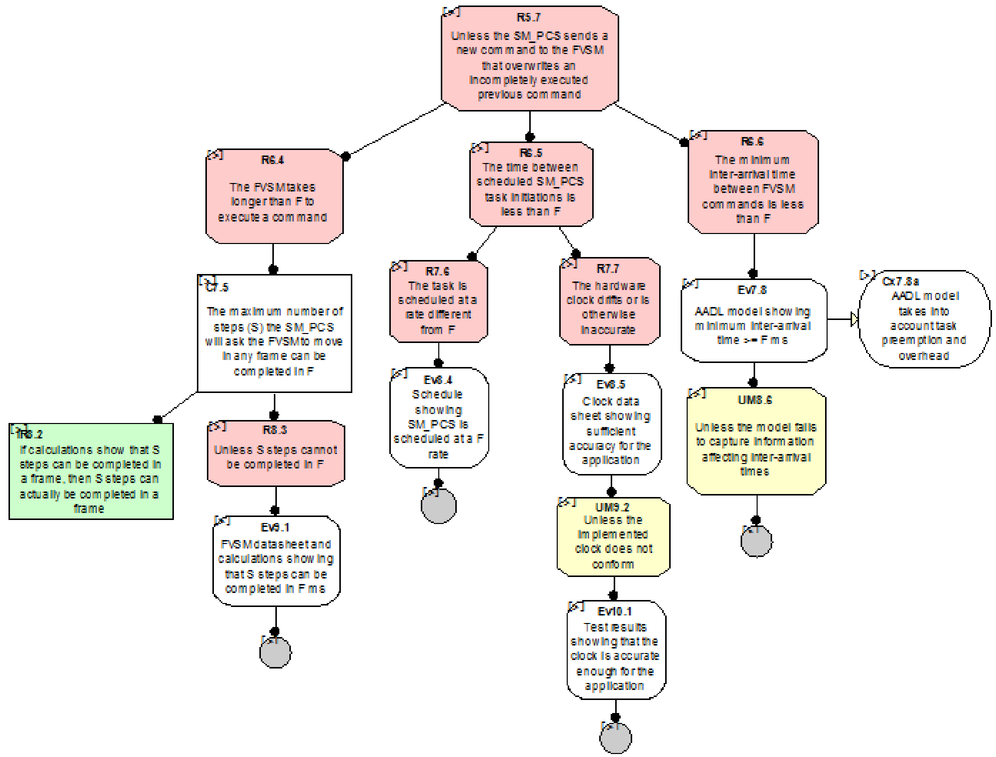
- Fault impact analysis identifies multiple sources of missed steps
  - Early arrival of step increment commands
  - Step increment command rate mismatch
  - Transient message corruption or loss
- Understanding of error cause
  - When is early too early
  - Guaranteed delivery assumption for step increment commands



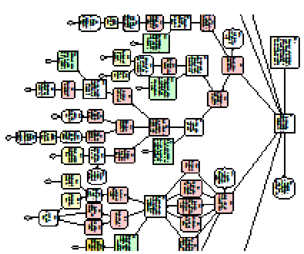
Assurance by Confidence Maps

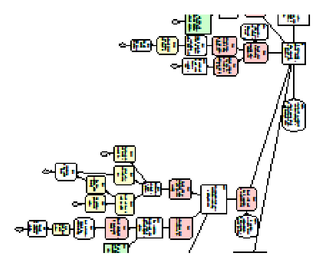
Iterative process between fault analysis and confidence mapping

- Fault analysis focuses on system hazards
- Multi-legged confidence mapping address process related defeaters as well

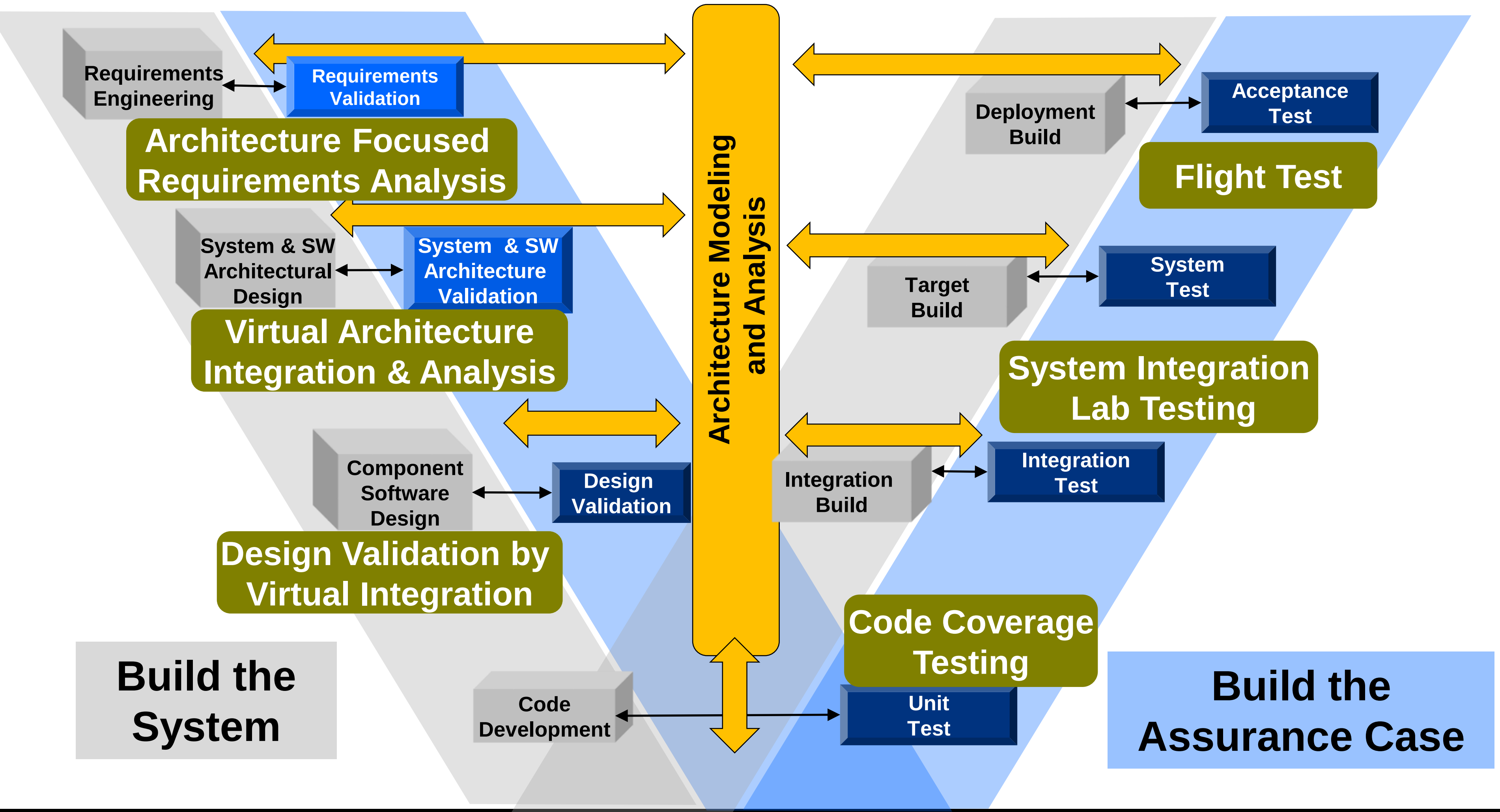


Use in comparative architecture study





Increased Confidence through Model-based Analysis and Testing Evidence throughout Life Cycle



References

Website www.aadl.info

Public Wiki https://wiki.sei.cmu.edu/aadl User Days, publications

AADL Book in SEI Series of Addison-Wesley
http://www.informit.com/store/product.aspx?isbn=0321888944

System Architecture Virtual Integration: An Industrial Case Study, SEI-2009-TR-017

Reliability Improvement and Validation Framework, SEI-2012-SR-013

Confidence-Based Architecture Fault Analysis, SEI-2013-TR-011 (in review)

Peter H. Feiler

Telephone: +1 412-268-7790

Email: phf@sei.cmu.edu