Architecture-Centric Virtual Integration Practice with AADL

Peter Feiler

Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8702-15-D-0002 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

DM18-1319

BLUF

Architecture-Centric Virtual Integration Practice (ACVIP) addresses Cyber Physical Systems Issues: A Digital Engineering Contribution

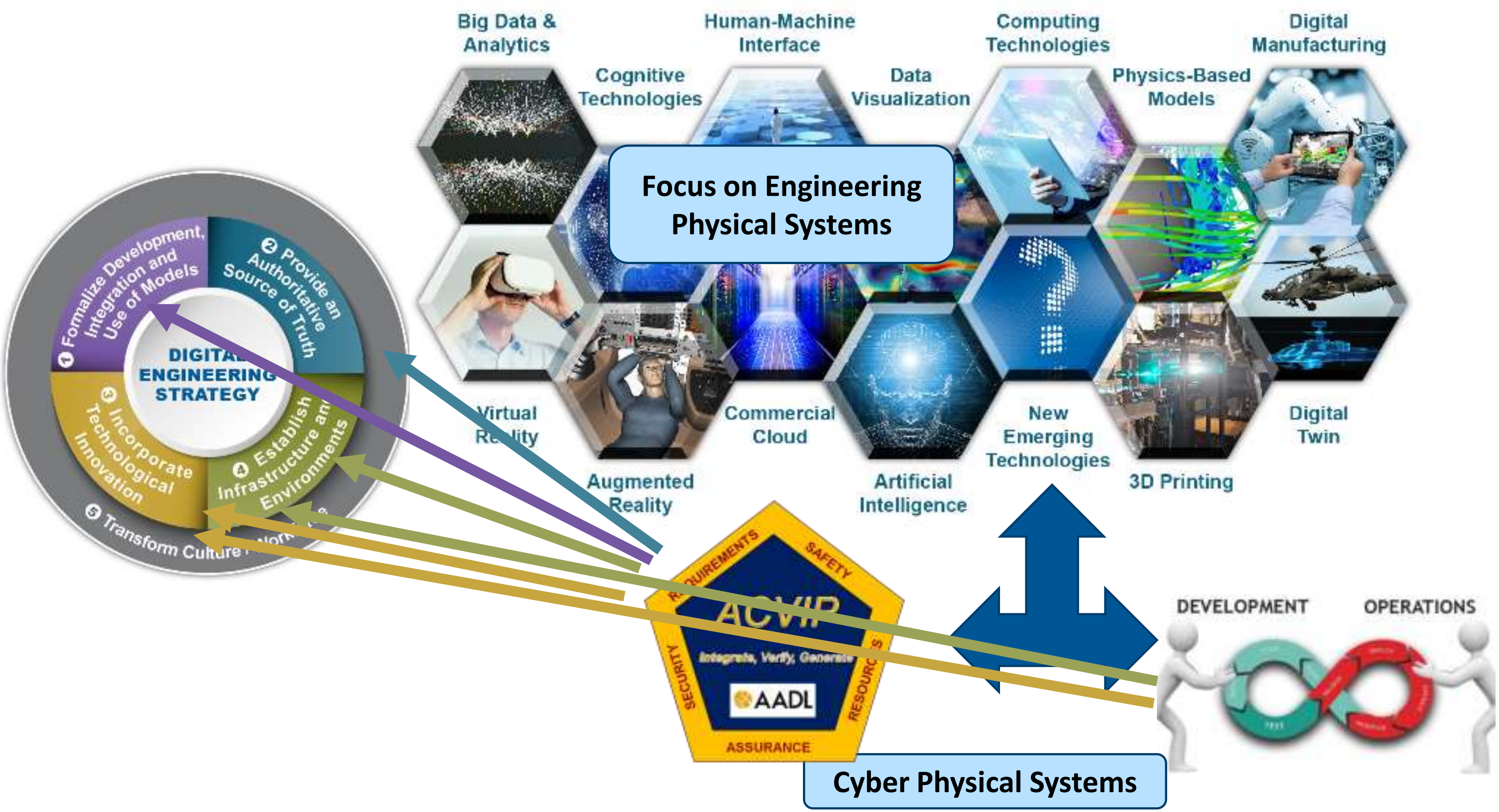
Growth in complexity and late error discovery is driving affordability in embedded software systems.

Trial use has demonstrated that ACVIP and the SAE International Architecture Analysis and Description Language (AADL) standard suite dramatically reduce late error discovery, lowering cost and accelerating the deliver of trusted capability.

ACVIP can be a key contributor to the DoD Digital Engineering Strategy:

- Standard as a foundation for a commercial tool marketplace
- Maturation and commercialization of technology
- Contribution to authoritative source of truth
- Leverage infrastructure and environments
- Transformation of workforce and culture

DoD Digital Engineering Strategy:
ACVIP as Key for Cyber Physical Systems



Architecture-Centric Virtual Integration Practice

ACVIP and Emerging Engineering Practice

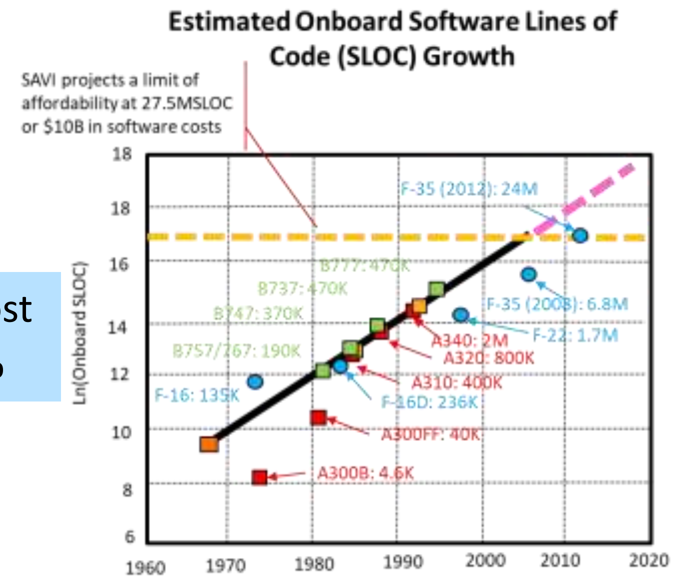
Accelerating the Adoption of ACVIP

Problem: Growth in Complexity and Late Error Discovery in Cyber Physical Systems is Driving Affordability

Cyber Physical Systems, especially Aviation Systems, are reaching a software affordability limit, impacting the amount of new functionality we can integrate.

Software as % of total system development cost
1997: 45% ➞ 2010: 66% ➞ 2024: 88%

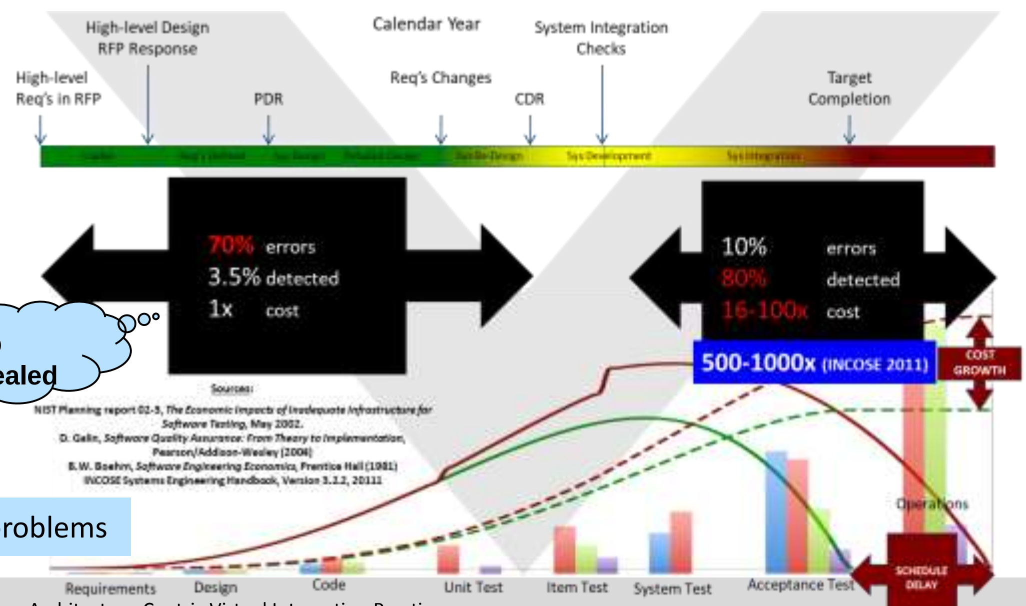
Post unit test software rework currently ~50% of total system development cost



A Commercial Aviation Industry Consortium

This represents a significant opportunity for cost reduction and functional enhancement by discovering issues early through virtual integration and analysis of embedded software system models and synthesis of implementation from verified models.

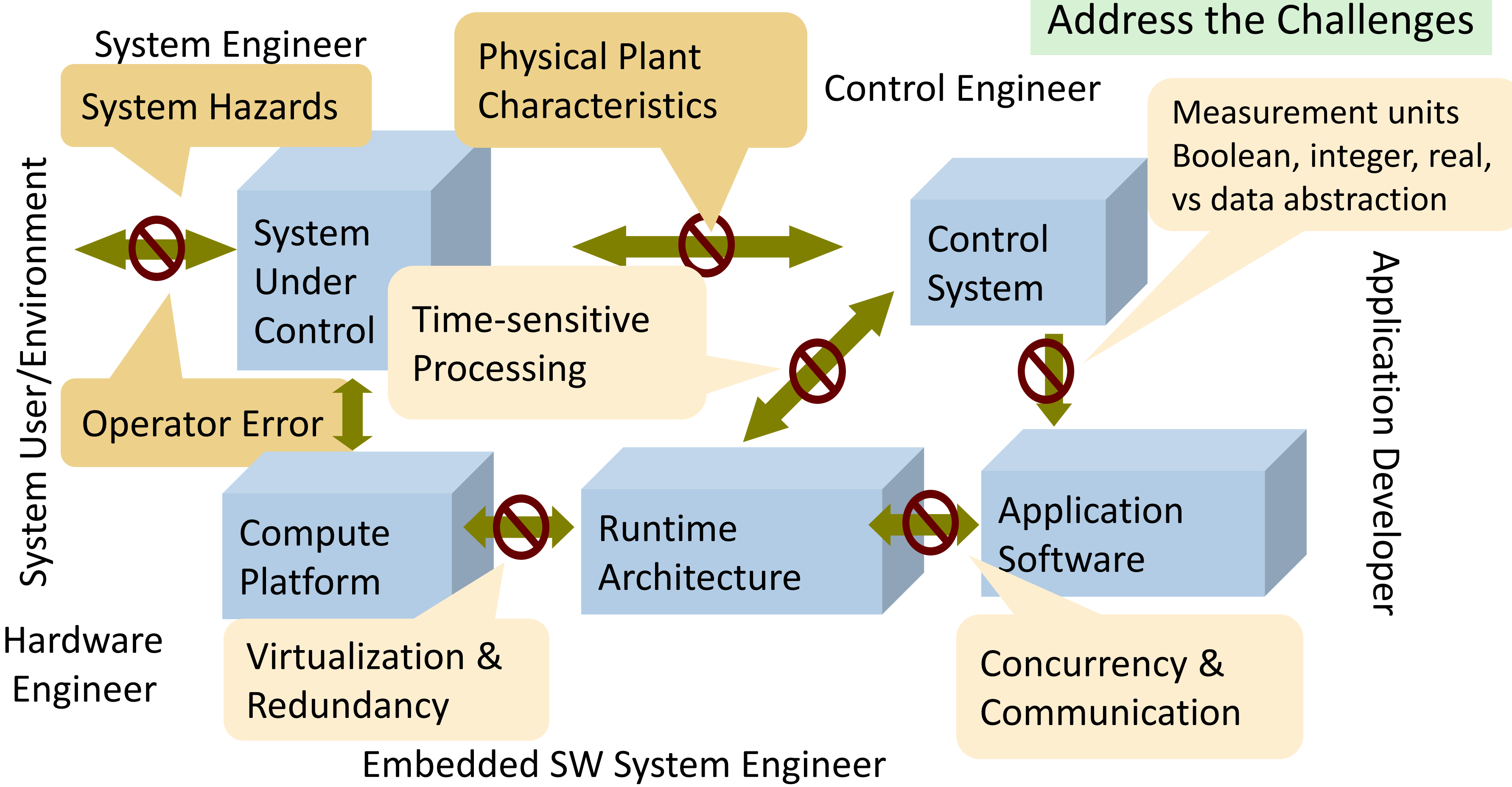
Gap revealed



F35 generated software had integration problems

Technical Challenges in Safety-Critical Embedded Software Systems

AADL Semantics Address the Challenges



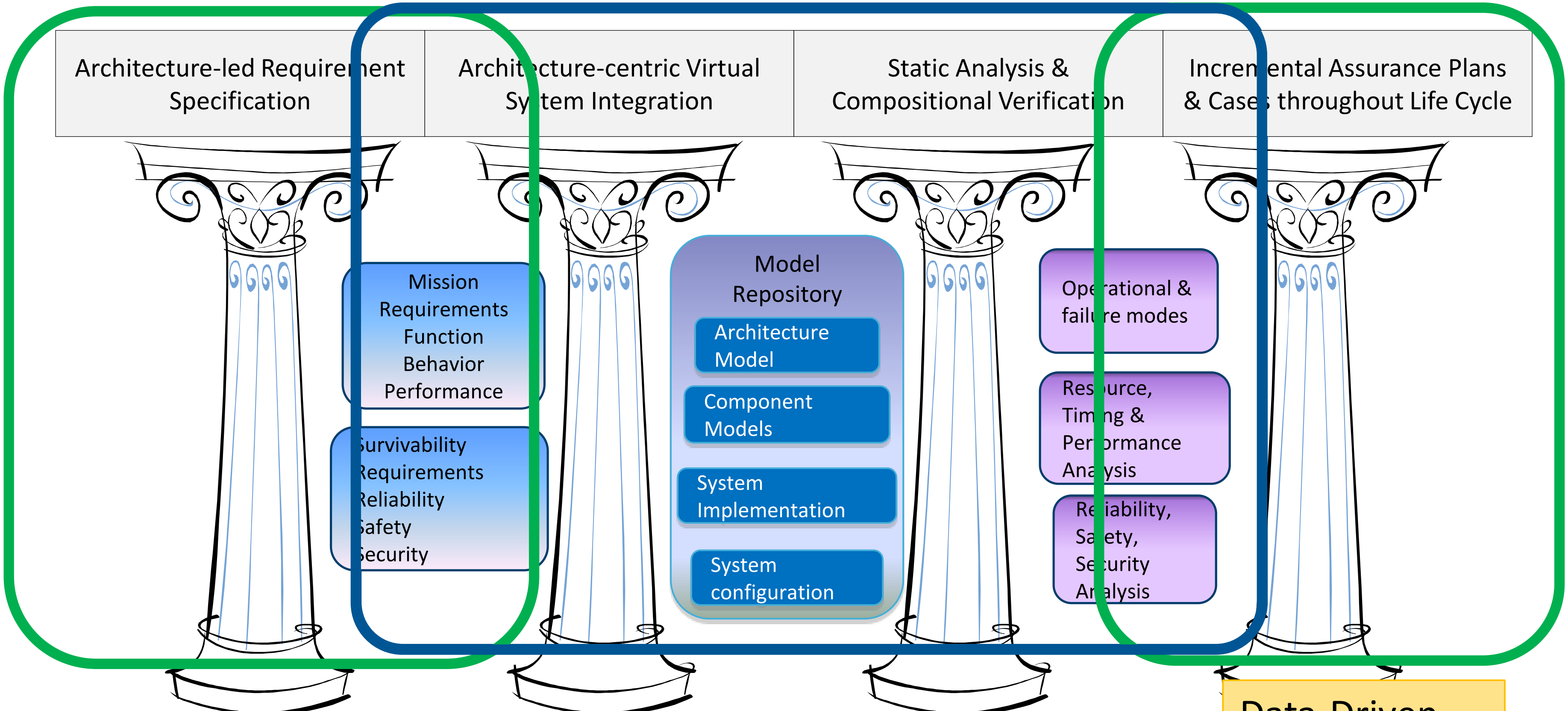
Why do system level failures still occur despite best safety practices?

*Embedded software systems have become a major **safety** and **cyber security** risk*

Assurance & Qualification Improvement Strategy

Assurance: Sufficient evidence that a system implementation meets system requirements

2010 SEI Study for AMRDEC
Aviation Engineering Directorate

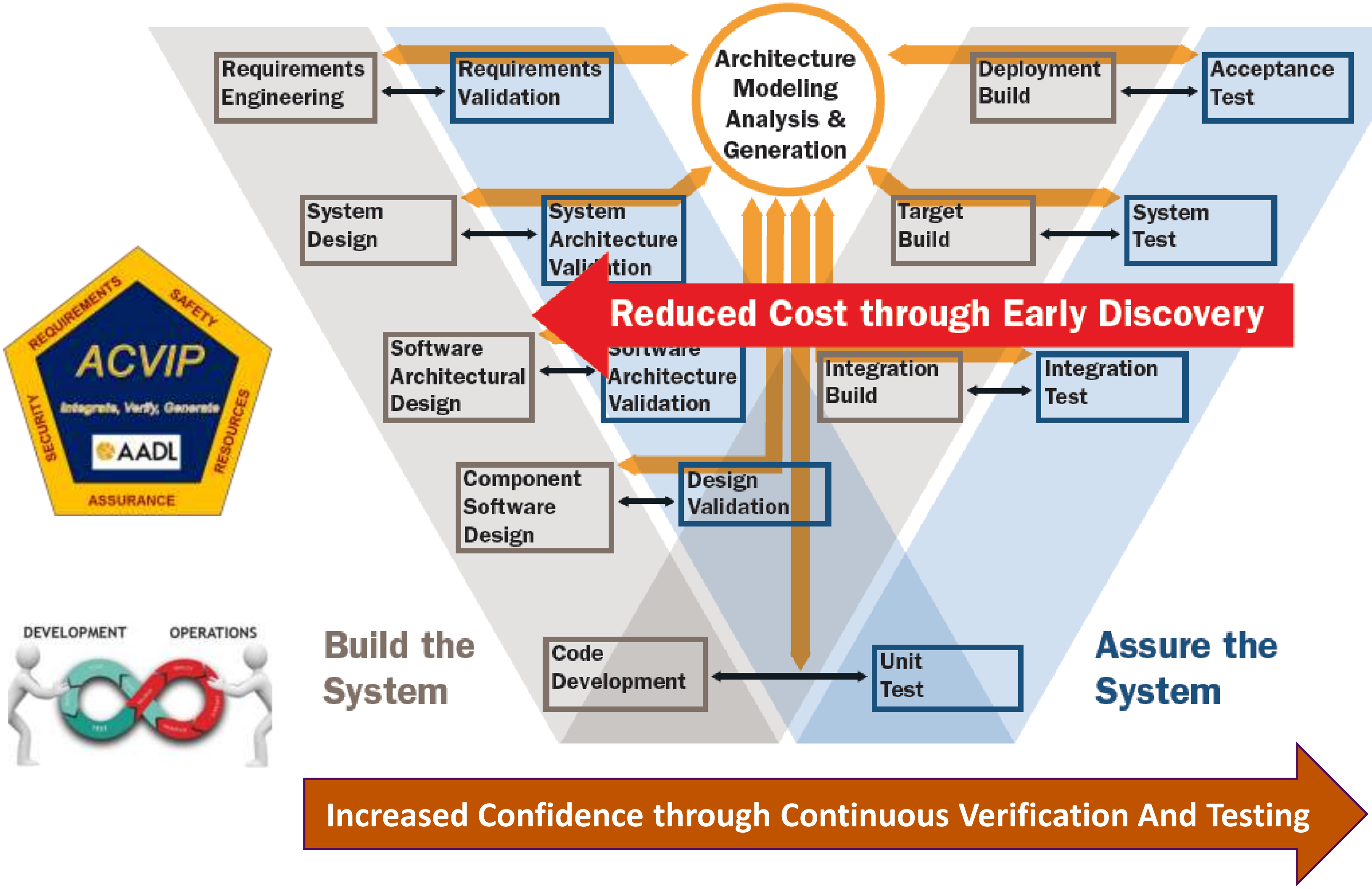


Architecture Centric Virtual System Integration Practice (ACVIP)

Architecture Led Incremental System Assurance (ALISA)

Data-Driven
High Leverage
Cost Effective

Benefits of Virtual System Integration & Continuous Lifecycle Assurance



Architecture Analysis & Design Language (AADL) Standard Targets Embedded Software Systems

The Physical System

Aircraft, Car, Train

Command & Control

The Software System

Embedded Operational Avionics & Mission Software

SW Design & Runtime Architecture

Physical Interface Platform Component

Deployed on Utilizes

The Computer System

Computer System Hardware & OS

SAE International

AS 5506 Standard Suite

Standards provide long-term industry-wide solutions to support multi-organization model-based engineering

SAVI System Architecture Virtual Integration

In 2008 Aerospace industry initiative chose AADL over SysML and other notations as it specifically addresses embedded software systems

AADL captures mission and safety critical embedded software system architectures in virtually integrated analyzable models to discover system level problems early and construct implementations from verified models

SAE International AADL Standard Suite (AS-5506 series)

Core AADL language standard [V1 2004, V2 2012, V2.2 2017]

- Focused on embedded software system modeling, analysis, and generation
- Strongly typed language with well-defined semantics for execution of threads, processes on partitions and processor, sampled/queued communication, modes, end to end flows
- Textual and graphical notation
- Revision V3 in progress: interface composition, system configuration, binding, type system unification

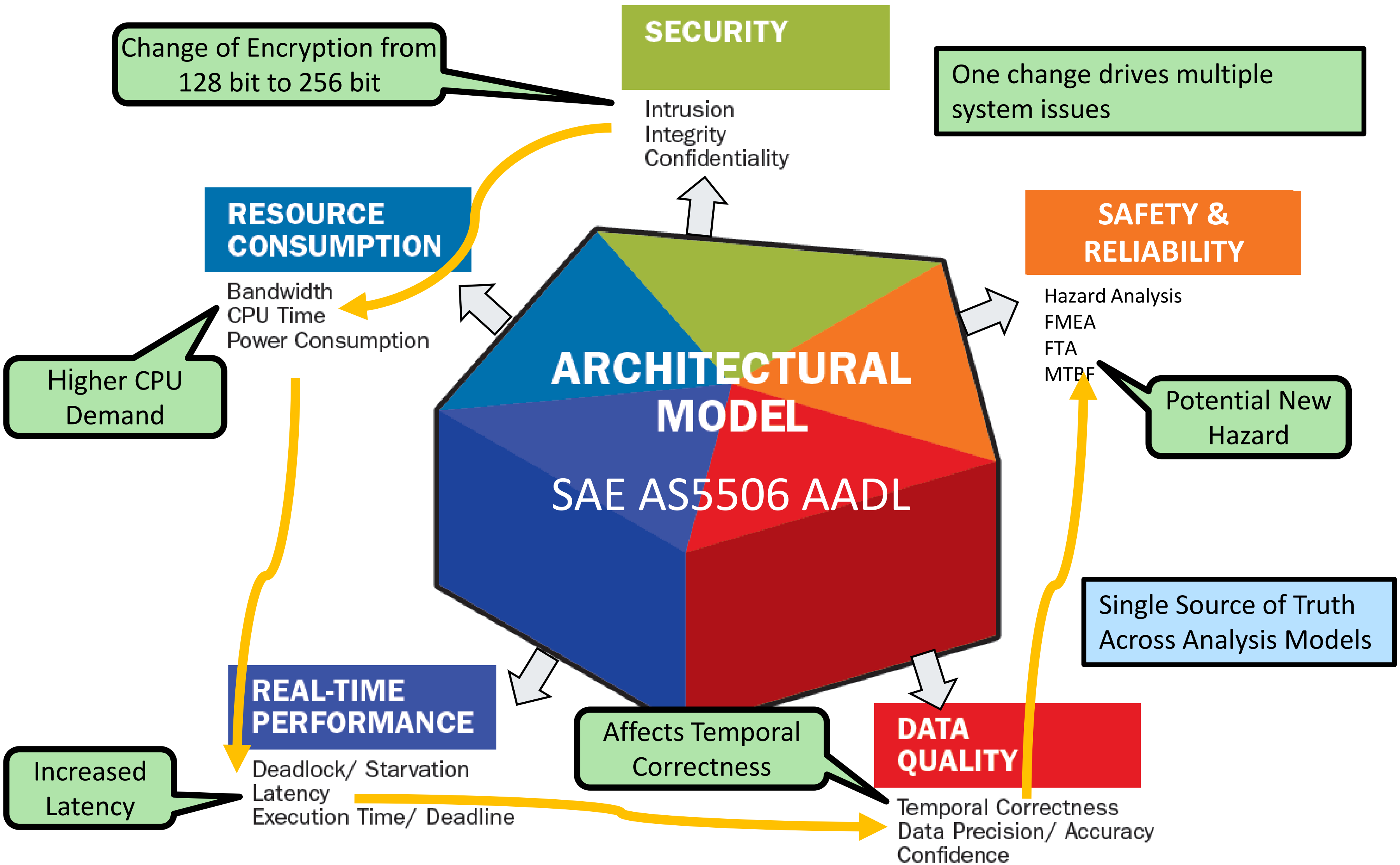
Standardized AADL Annex Extensions

- Error Model language for safety, reliability, security analysis [2006, 2015]
- ARINC653 extension for partitioned architectures [2011, 2015]
- Behavior Specification Language for modes and interaction behavior [2011, 2017]
- Data Modeling extension for interfacing with data models (UML, ASN.1, …) [2011]
- AADL Runtime System & Code Generation [2006, 2015]

AADL Annexes in Progress

- Network Specification Annex
- Cyber Security Annex
- FACE Annex
- Requirements Definition and Assurance Annex
- Synchronous System Specification Annex

Analysis of System Properties via Architecture Model
A Contribution to Single Source of Truth



Demonstrations of Effectiveness in use of ACVIP with AADL

Finding Problems Early (AMRDEC/SEI)

- Summary: 6 Week Virtual Integration on CH47 using AADL
- Result: Identified 20 major integration issues early
- Benefit: Avoided 12-month delay on 24-month program

CH47 Chinook

Transforming procurement (Joint Multi-Role)

- Summary: Industry/DoD mission system architecture demonstrations using ACVIP
- Result: Pre-integration fault identification
- Benefit: 10X reduction integration test cost

Improving System Security (DARPA / AFRL)

AADL applied to Unmanned Aerial Vehicles & Autonomous Truck

Result: AADL models enforced security policies and were used to auto build the system

Benefit: Combined with formal methods verification, prevented security intrusion by a red team

Unmanned Quadcopter

DARPA

High Assurance Cyber Military Systems (HACMS)

TARDEC Autonomous Truck

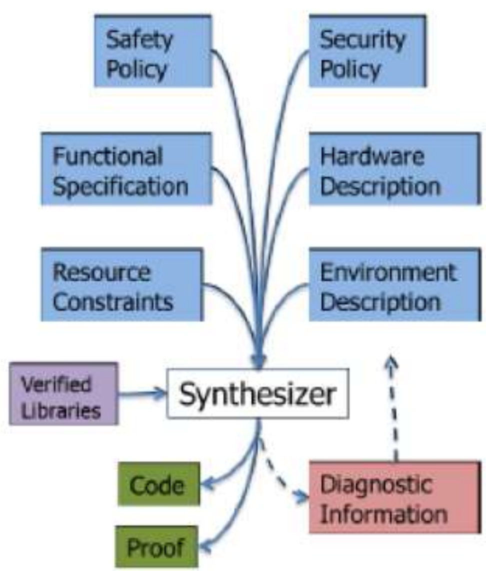
Unmanned Little Bird

Synthesize & Verify High-Assurance Systems

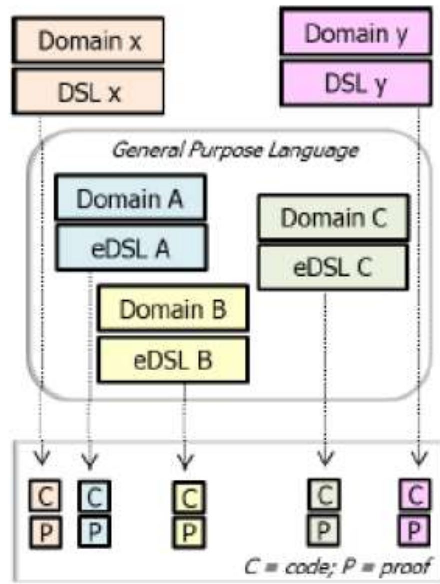
High Assurance Cyber Military Systems

Dr. Lindermann April 2018 Keynote

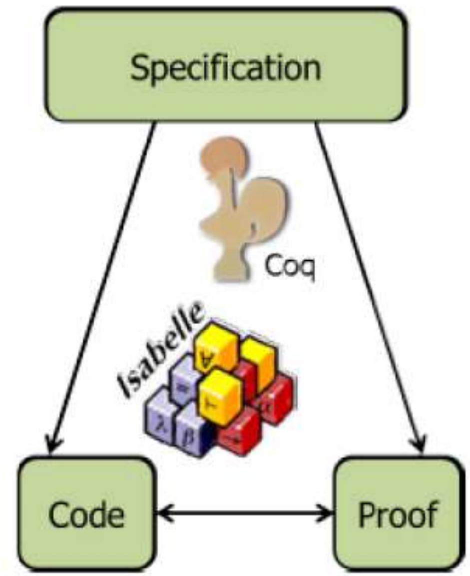
Code Synthesis



Domain Specific Languages (DSLs)

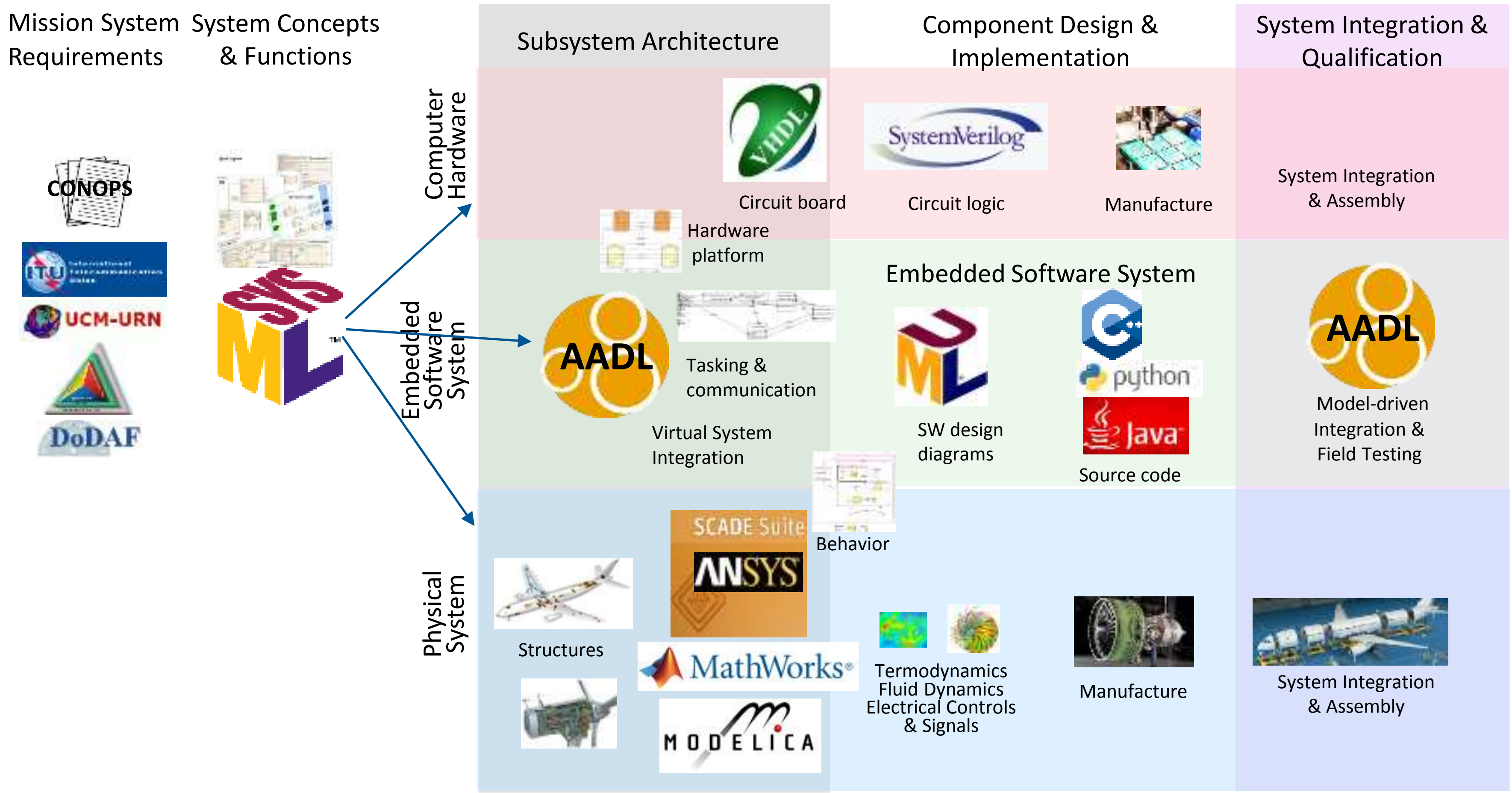


Interactive Theorem Prover as PL



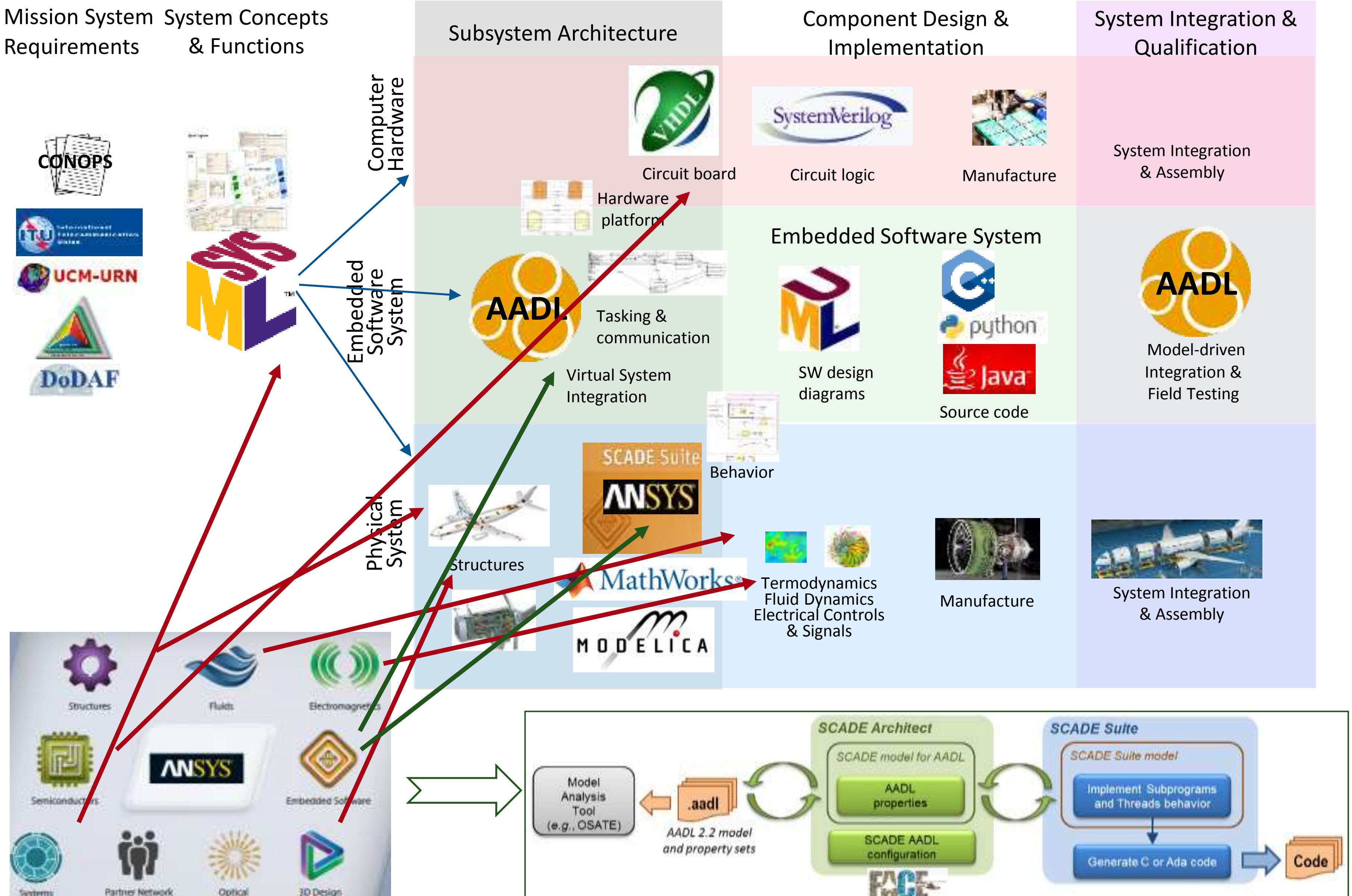
- **Research Challenges**
  - Synthesis of attack-resilient control systems
  - Synthesis of operating systems code
  - Specification languages: function, environment,
  - hardware, resources
  - Composition/Proof engineering
  - Scaling
  - Attack/fault response
  - V&V of complete system

Multiple Languages and Tools to Meet Users Needs



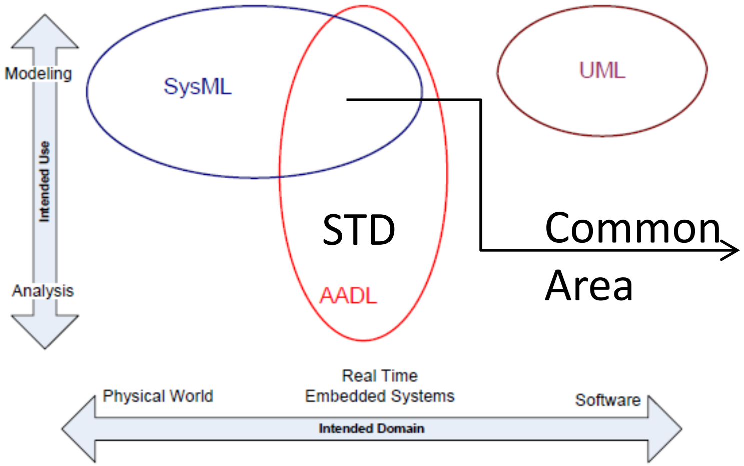
Filling the Modeling and Analysis Gap for Embedded Software System

ANSYS: A Commercial Tool Vendor Strategy

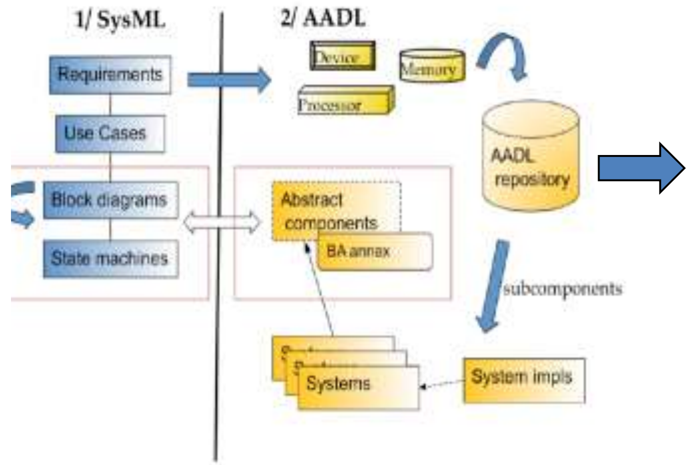


SysML & AADL Comparison

USE



TRANSLATE



INTEGRATE THEN BUILD

AADL
INCREMENTAL
REFINEMENT
ANALYSIS &
INTEGRATION
(SW/HW/SYS)

Modeling Language	UML	SysML	AADL
Standards Org	OMG	OMG	SAE International AS 5506
Purpose	Object Oriented Program Modeling	Larger Systems Modeling & Analysis	Embedded Software Systems Modeling & Analysis
Constructs/ Views	Class Diagram, Block Diagram, Sequence, Activity, State Machine	Use-Case, Block Diagrams, Internal Block Diagrams, Rqmts, Sequence, Activity, State Machine, Parametric	RT Components (Abstract, Processor, Memory, Bus, System, Threads…) State Machines (Modes, Behavior, Error) Flows, Bindings, connections
Practice / Methodology	Object Oriented	OOSEM	Virtual System Integration, ACVIP
Tools (Examples)	Rhapsody, SCADE, Sparx EA, MagicDraw, etc.	Rhapsody, SCADE, Sparx EA, MagicDraw, etc.	OSATE, Adventium, ANSYS SCADE, ElliDiss, Dassault, WW Technology Group
Practitioners	Commercialized	Commercialized	R&D platform, S&T, commercial tools available

Architecture-Centric Virtual Integration Practice

ACVIP and Emerging Engineering Practice

Accelerating the Adoption of ACVIP

Emerging Engineering Practice and ACVIP

HACMS

Emerging engineering practices emphasize:

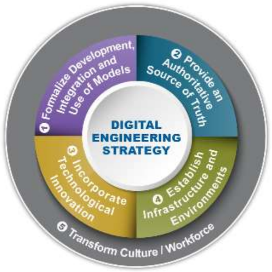
- Rapid delivery of capability using iterative, incremental development paradigms, e.g. MBE, Agile-DevOps and Software Factory.
- Multi-discipline MBE and modular open systems concepts, e.g. Digital Engineering Strategy.
- Synthesis of complex systems from unambiguous descriptions that support analysis and proof/evidence of correctness, e.g. HACMS

ACVIP delivers many of these capabilities for embedded systems.

- Rapid delivery with continuous virtual integration
- Single source of truth HW/SW architecture model repository
- Proof of correctness through semantically precise architecture descriptions
- Automated (Trusted) build capability for embedded systems architecture

Yields => Rapid integration of components into verified open architecture with automated prototyping and testing to final trusted embedded system build

DoD Digital Engineering Strategy and ACVIP Alignment



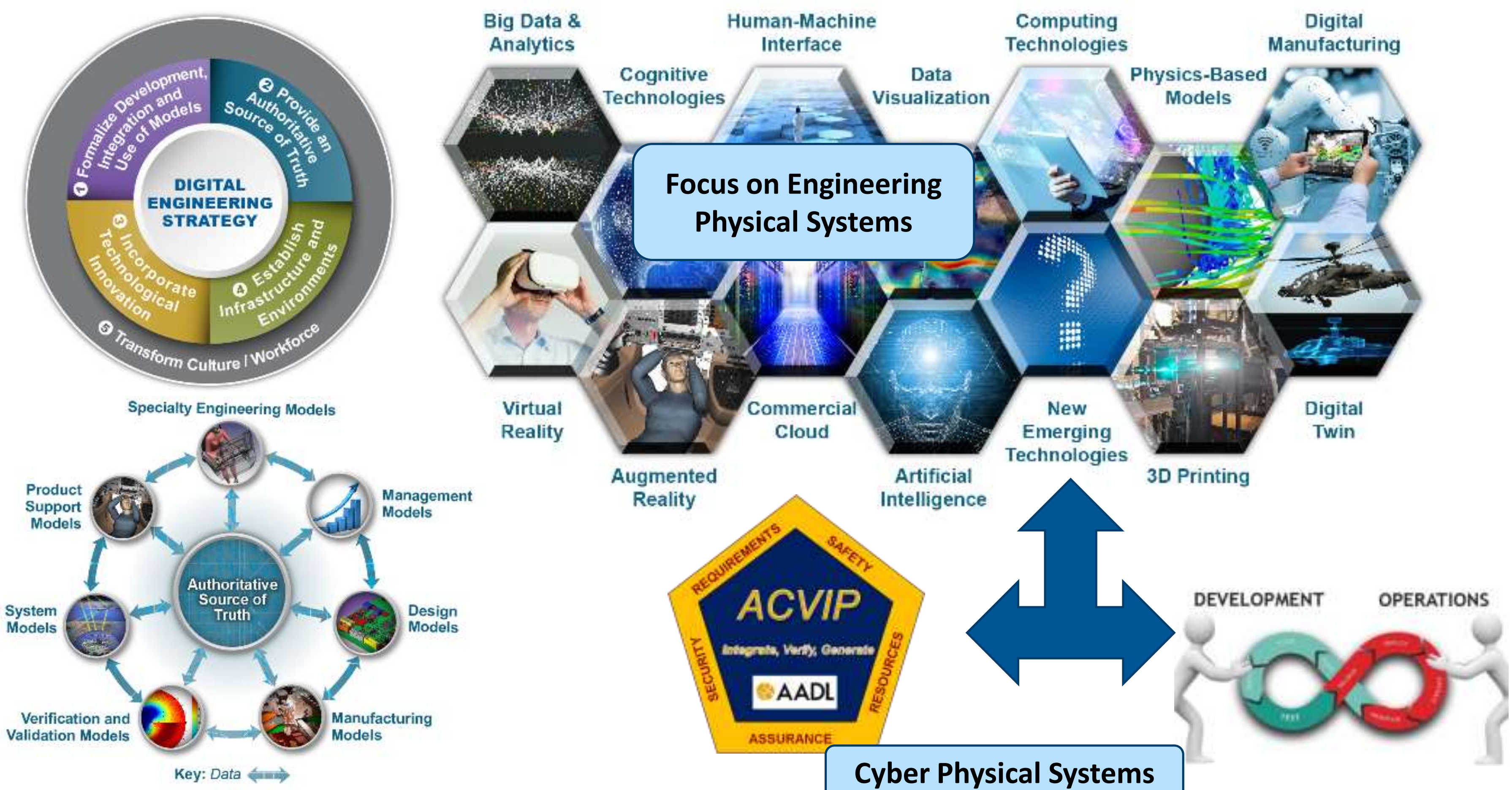
Goals

1. Formalize the development, integration, and use of models to inform enterprise and program decisions
2. Provide and enduring, and authoritative source of truth
3. Incorporate technological innovation to improve the engineering practice
4. Establish a support infrastructure and environments to perform activities, collaborate, and communicate across stakeholders
5. Transform the culture and workforce to adopt and support digital engineering across the lifecycle

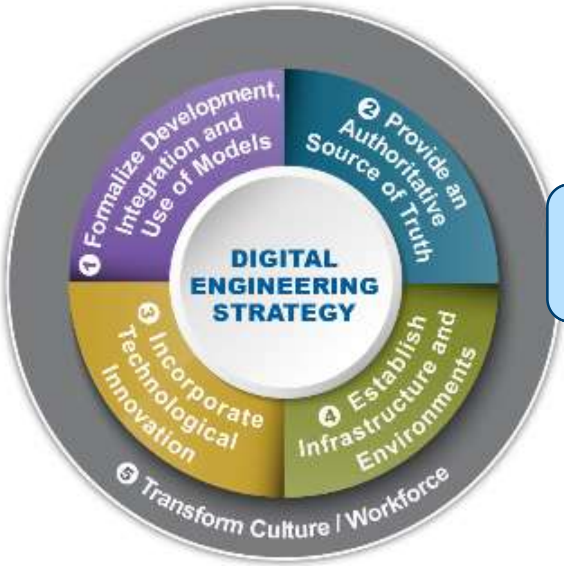
Existing and planned ACVIP capabilities address many Digital Engineering Strategy elements for embedded systems.

- ✓ *"…model based engineering… to facilitate systems engineering and decision making across the lifecycle"*
- ✓ *"…authoritative source of truth … to access, manage, protect, and analyze…data and models"*
- ✓ *"…DoD should develop, mature, and use digital engineering methodologies … "*
  - ✓ *"methods and processes to support digital engineering"*
  - ✓ *"…tools…data and interface standards…"*
  - ✓ *"…visualization, analysis, model management and interoperability, workflow, collaboration, and extensions/customization…"*

DoD Digital Engineering Strategy: Cyber Physical Systems as Key Component

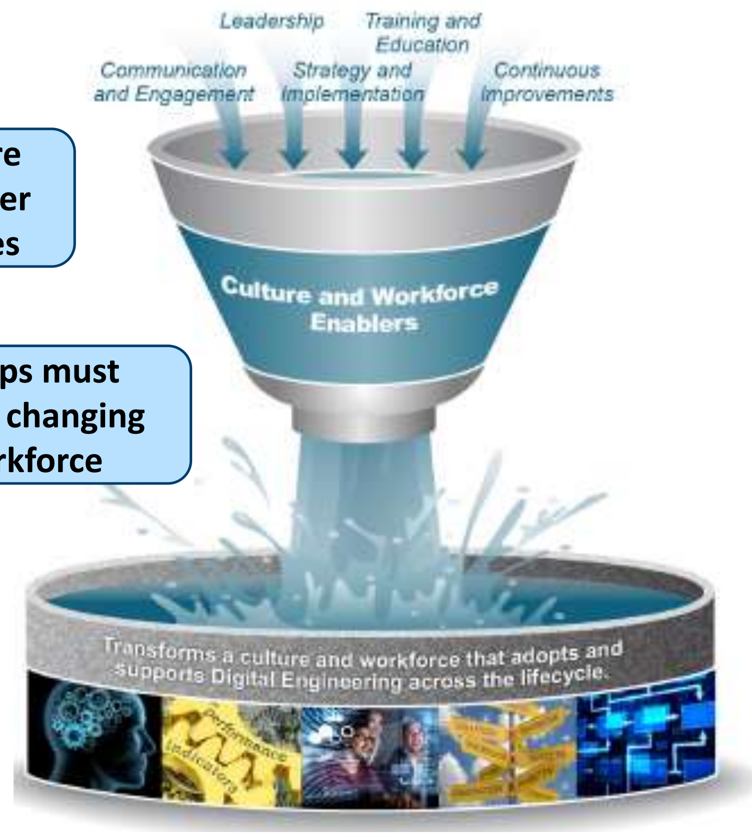


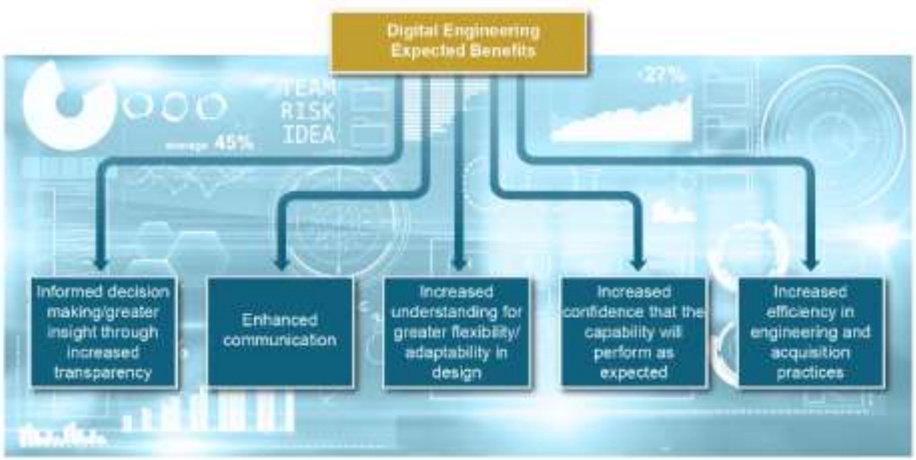
ACVIP and DevOps must be/are Part of the DoD Digital Engineering Maturation and Transition



ACVIP and DevOps are key to addressing Cyber Physical System Issues

ACVIP and DevOps must leverage efforts in changing culture and workforce





ACVIP and DevOps contribute to benefits specifically for cyber physical systems

Architecture-Centric Virtual Integration Practice

ACVIP and Emerging Engineering Practice

Accelerating the Adoption of ACVIP

ACVIP Maturation and Adoption Strategy

1. Research & Development

Mature and Extend ACVIP Capabilities including:

- Continuous Model Integration and Exchange for SSOT
- Model consistency verification framework
- Behavioral integration analysis
- Parallelized analysis to support scalability
- Cybersecurity analysis, etc.

2. Adapt/Apply

Adapt and apply ACVIP on selected legacy aviation systems

3. Community of Practice

Build an ACVIP Community of Practice with participants from Academia, Gov't, Industry, International Research agreements.

4. ACVIP Lab

Establish ACVIP lab and center of excellence to support adoption within the AMRDEC.

5. Workforce Development

Develop and support ACVIP/AADL training

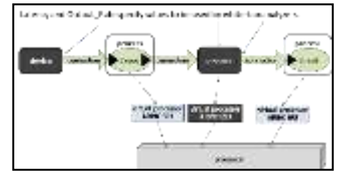
6. AADL Standards

Extend SAE AADL standard and tool support

7. Assess Cost/Benefit

Assess ACVIP/AADL Cost/Benefit

Additional High Leverage Research Opportunities

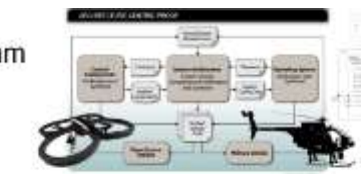


Schedulability and Latency

- End to end scheduling and latency
- Multi-core scheduling and latency
- AADL Guidance aligned with Multicore Association open standards

Safety and Security

DARPA HACMS Program

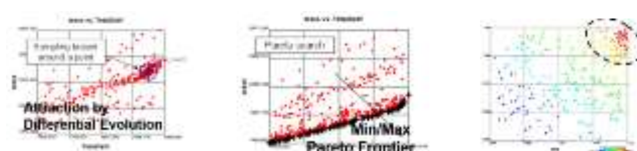


- Safety engineering of embedded software systems
- Integrated safety and security engineering approach
- AADL Security annex, revision of Error Model (safety) annex

ARP4754A
ARP4761

Design Space Exploration for Embedded Software Systems

- Impact of architecture design choices
  - sampled and message-driven task and communication architectures
  - hardware, partitioning, encryption
  - logical and physical redundancy
- AADL V3 addresses configurability of multiple dimensions



Single Source of Truth

See Synthesis and Verification of High Assurance Systems slide for other research challenges

Other S&T Research involving AADL Maturation

SAE AADL Standards Work

- Committee chair and standard suite architect
- Authoring of AADL V3 and various Annex extensions
- Prototype reference implementation

JMR Mission Systems Architecture Demonstration

- JCA Demonstration
- Architecture Implementation Process Demonstrations (AIPD)
- Capstone Demonstration

DARPA 6.2 S&T Programs using AADL

- HACMS (High Assurance Cyber Military Systems)
- CASE (Cyber Assured Systems Engineering)

Other Army Aviation 6.3-6.3 S&T Programs for potential use

- Integrated Mission Equipment (IME)
- Synergistic Unmanned Manned Intelligent Teaming (SUMIT)
- Degraded Vision Environment Mitigation (DVE-M)
- Legacy aircraft systems and subsystems

SBIR funded projects

- Incremental Partitioning to Minimize Change Impact
- Mixed Critical Cyber Physical Systems & Advanced Integrity and Safety Assurance (WWTG)
- Model Based Testing of Integrated Aviation Mission Systems (IDT)
- Rapid Configuration of Heterogeneous Collaborative System-of-Systems Simulations (Physical Optics Corp)
- State Linked Interface Compliance Engine for Data (SLICED) (Adventium)
- Unified Behavior Descriptions for AADL Architecture Models (Adventium, POC)
- Security & Safety Co-Analysis Tool Environment (SSCATE) (Adventium)
- Virtual MBSE Platform to Enable Agile Development of Secure FACE Software (DornerWorks)

Summary

Cyber Physical Systems are facing exponential growth in software development cost exceeding 70% of total system development cost

ACVIP is a set of technologies and practices that specifically have been designed to provide early detection and continuous verification throughout the life cycle

ACVIP is a key contributor to the DoD Digital Engineering Strategy

Backup Charts

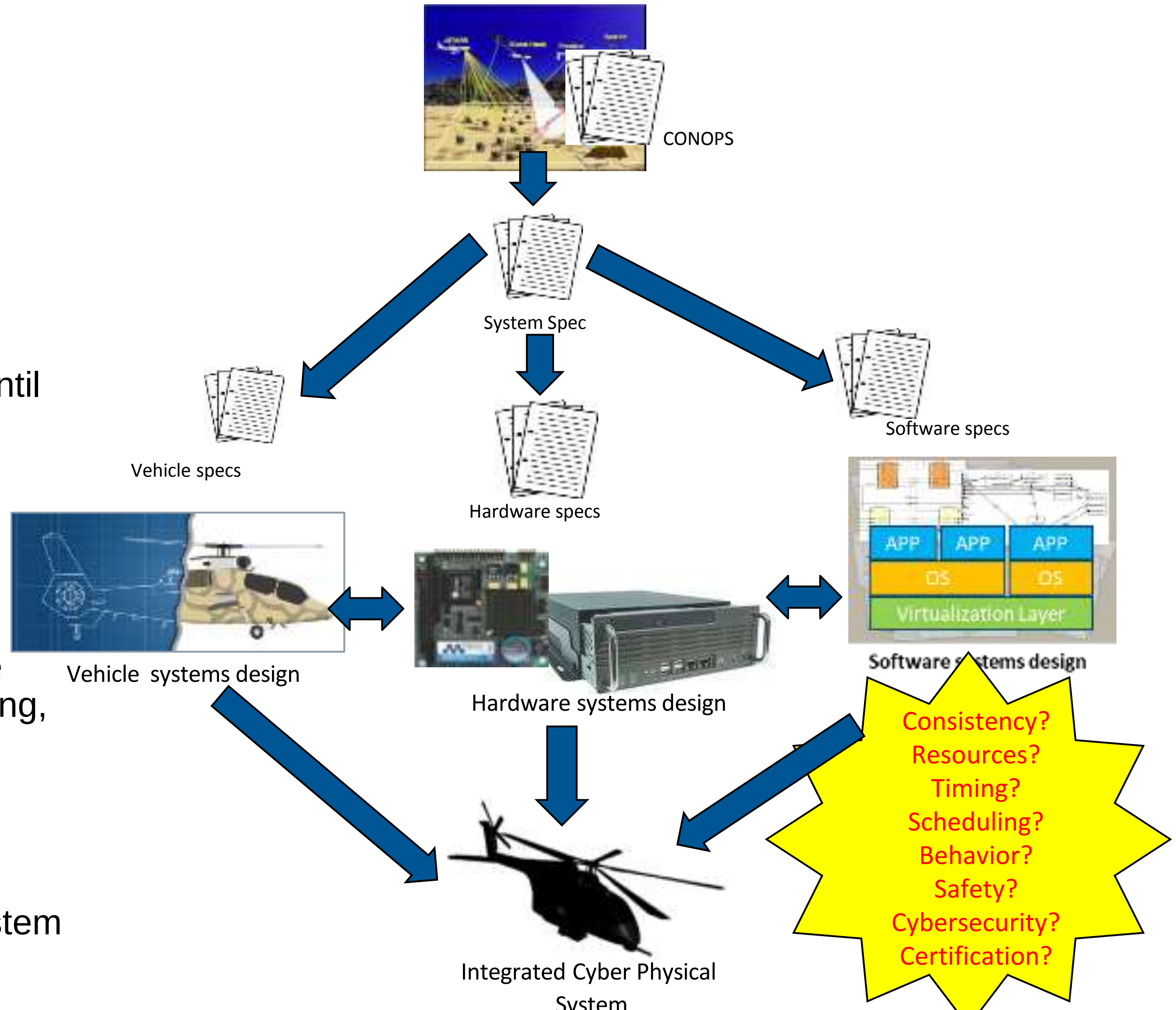
Systems Design & Challenges with Computing Systems Integration

Systems Design

- Involves multiple engineering disciplines
- Each requires different languages/tools/methods
- Most functionality deployed in software
- Software V&V does not begin until integration

Design Challenges

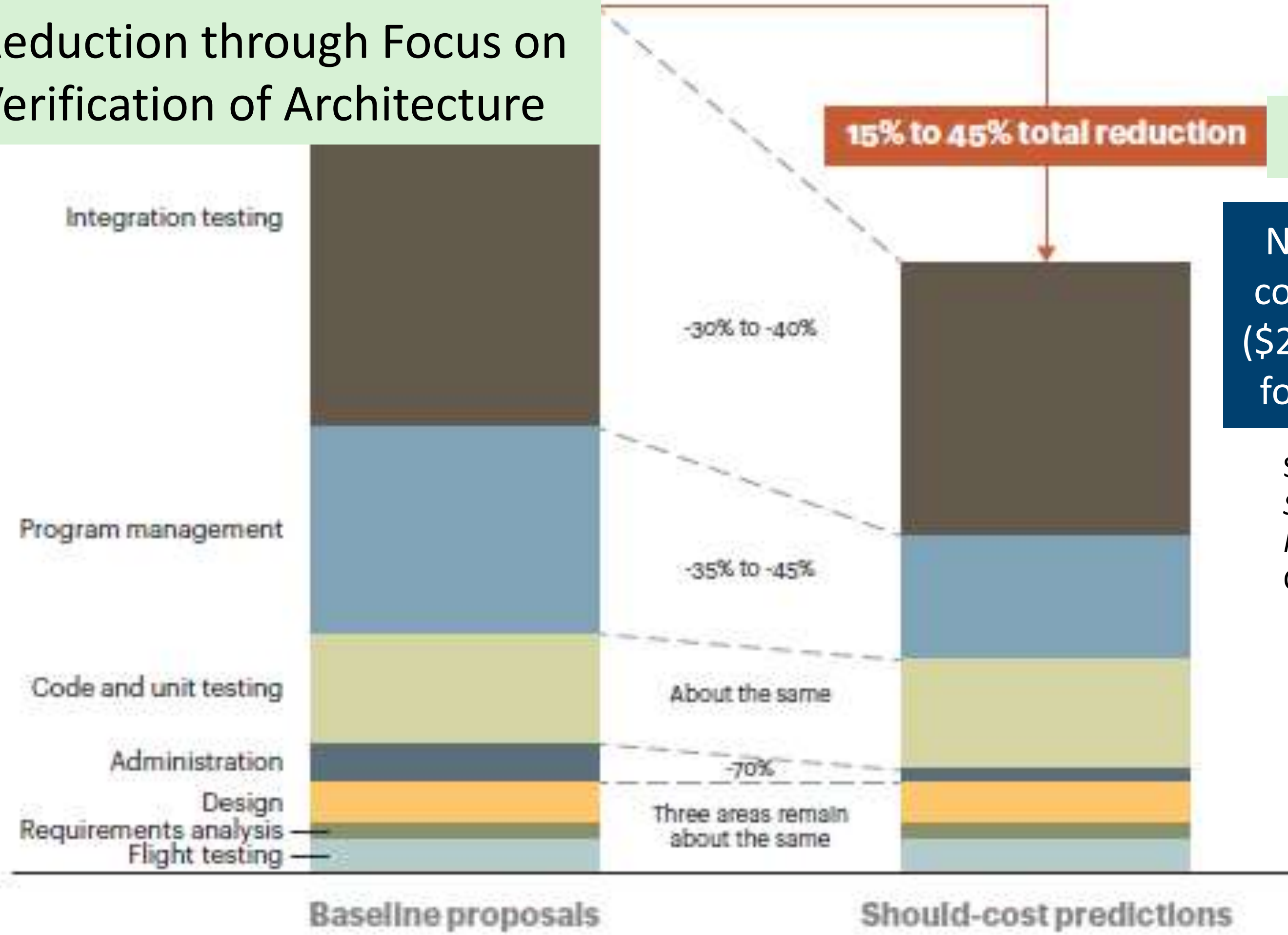
- Maintaining consistency across design elements (units, data, messages, etc.)
- Inability to detect emergent side affects of limited resources, timing, scheduling
- Predicting interaction of requirements change during development & sustainment
- Qualifying and certifying the system



The growth in system complexity is being partially addressed by various MBSE tools and methods, but there remains challenges in integration and qualification of integration with software

Cost Reduction Potential through Virtual Integration of Embedded Software Systems

Reduction through Focus on Verification of Architecture



ATKearney "Software: The Brains Behind U.S. Defenses Systems"

ROI on AADL Pilot

Nominal development cost reduction of 26.1% ($2.391B out of $9.186B) for a 27 MSLOC system

Source: *ROI Analysis of the System Architecture Virtual Integration Initiative* CMU/SEI-2018-TR-002

Previous/On-going Investment into AADL & Virtual System Integration

SAE AADL standard suite (1999-now)

- Committee work B. Lewis: chair, P. Feiler: Tech lead [AMRDEC]
- Standard document content authoring [AMRDEC, other*]
- AADL and annex concept development [SEI, AMRDEC, other*]

AADL Tooling (2004-now)

- Open Source AADL Tool Set (OSATE) development [SEI, AMRDEC**]
- AADL FACE translation tools [AMRDEC]
- Graphical OSATE support by UAH [AMRDEC]
- SBIR projects by Adventium Labs and WW Technology Group [AMRDEC]

Education and Training (2004-now)

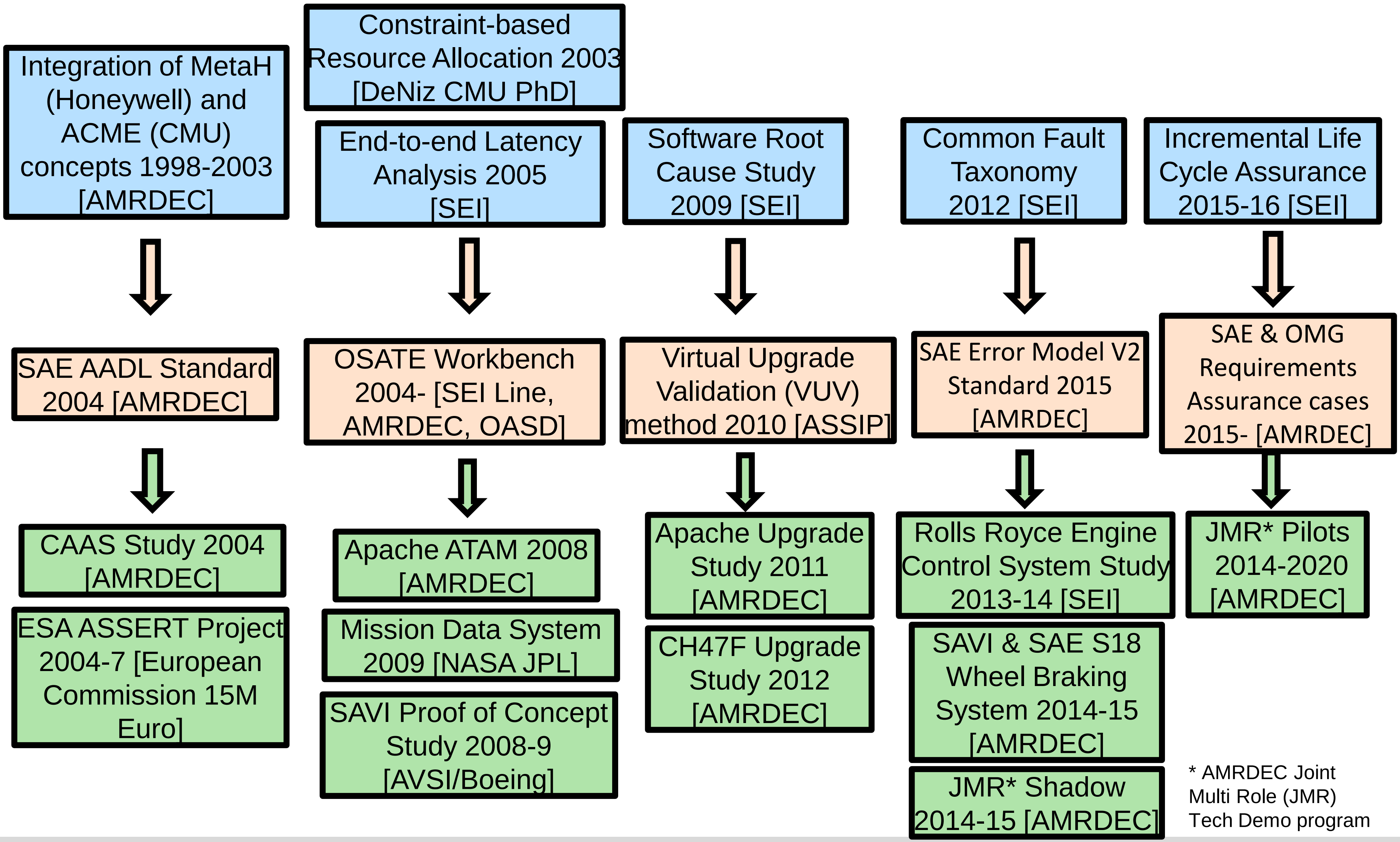
- Model-based Engineering with AADL Book [SEI]
- Virtual System Integration methods [SEI, AMRDEC]
- 5 day course on AADL, 2 day workshop [SEI]
- Joint Multi Role (JMR) ACVIP Engineering and Acquisition handbooks [AMRDEC]

Research into Architecture Analysis and Verification (2005-now)

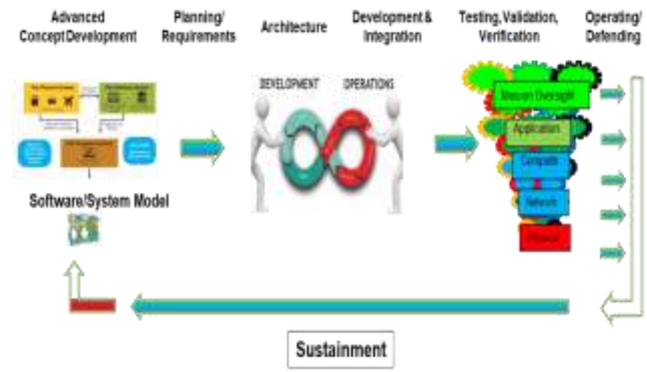
- SEI, DARPA, European Union funded projects in US and Europe
- Current SEI research
  - Integrated safety and security engineering (3 year) [SEI]
  - New concepts for AADL V3 and feasibility prototyping [SEI***]

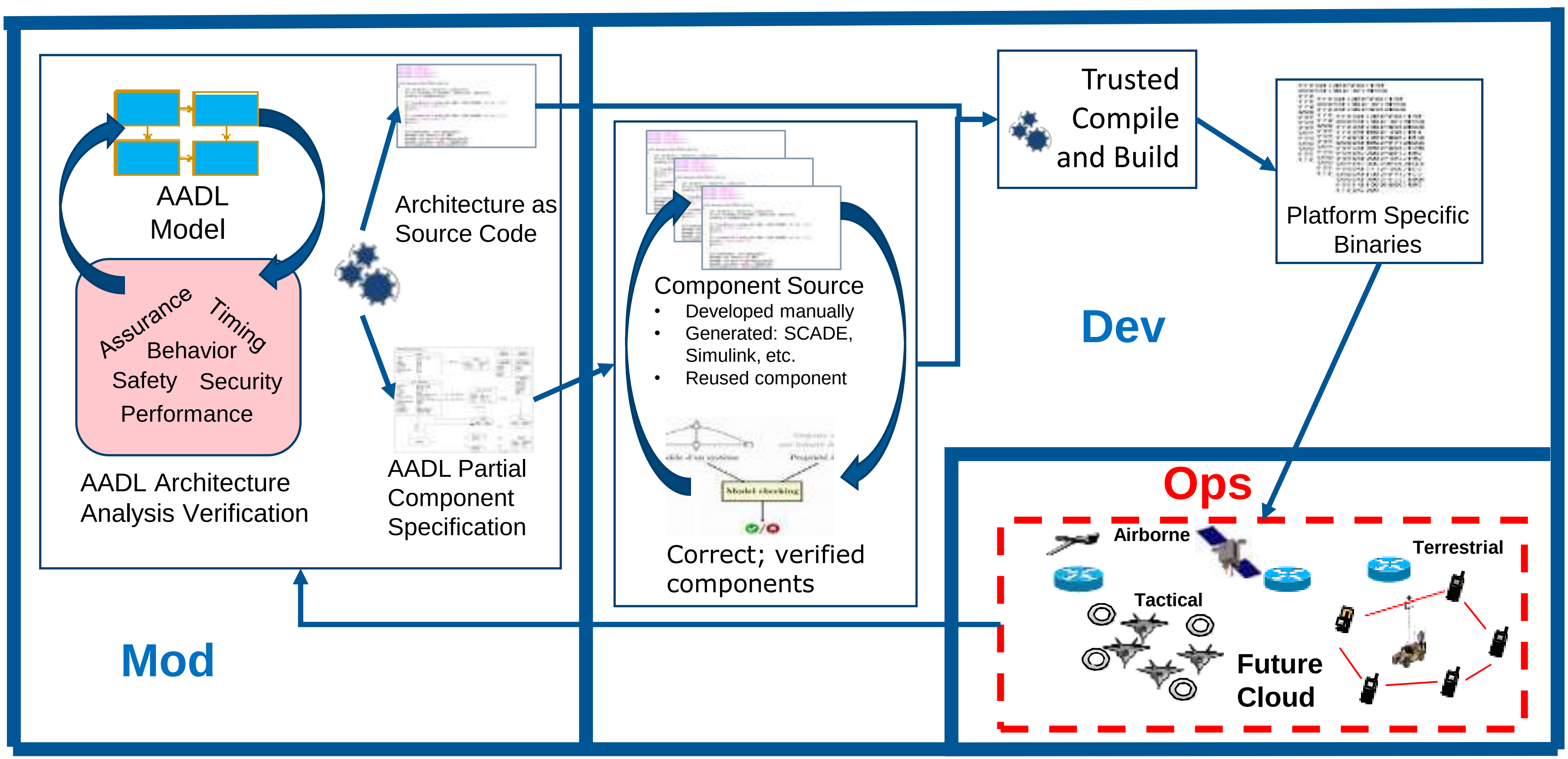
* European funding sources for European committee members
** Development SEI funded. Maturation and maintenance by AMRDEC JMR
*** Peter Feiler SEI Fellow project funds

SEI Research to AADL Standard to Pilot Projects



Model Based DevOps for Embedded Systems Using AADL/ACVIP





Note About AADL and DevOps

AADL has been designed to encompass embedded system architectural design and analysis, and later generation of complex systems.

System generation relies on code generation techniques, coupling the AADL data and code generation annexes to core language.

AADL supports the many of the central pillars of DevOps philosophy:

- coding (both through modeling and code generation)
- building the generated code
- testing it at model or code level
- packaging/releasing through regular mechanisms and configuring the running infrastructures

AADL does not address the last DevOps stack, “monitoring the deployment”, to update and improve the system overall performance, and would need to be extended to support this capability.