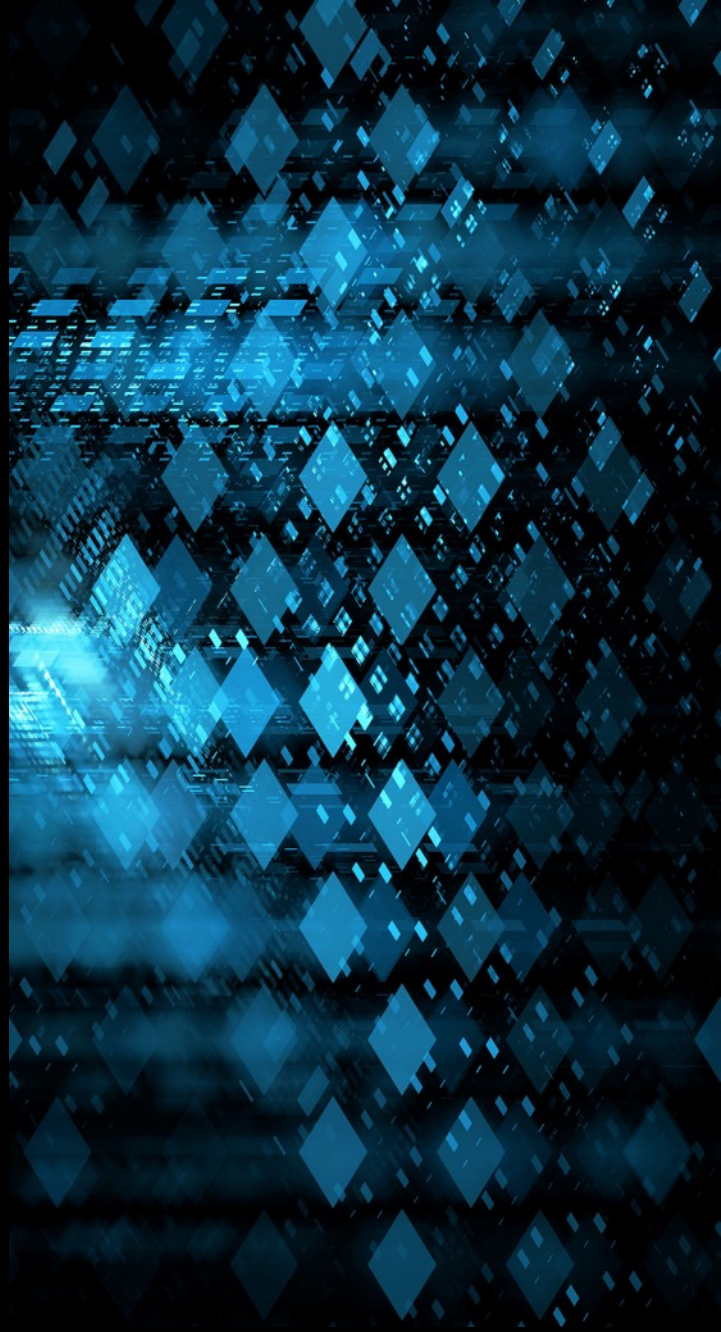
AADL and Virtual System Integration: Past, Present, and Future

Peter Feiler

March 2018



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.

DM18-0381

Virtual Integration and Assurance of Safety-Critical Software-Reliant Systems

Problem:

Software increasingly dominates safety and mission critical system development cost.

80% of issues discovered post unit test.

Solution: Early discovery of system level issues through virtual integration and incremental analytical assurance.

Approach:

International standard based technology matured into practice through pilot projects and industry initiatives.

Open source research prototyping platform continually enhances analysis, verification, and generation capabilities.

Reduced Defect Leakage through Early Analytical Assurance is Critical

We Rely on Software for Safe System Operation

Quantas Airbus A330-300 Forced to Landing - 36 Injured

Written by htbw on Oct-7-08 1:48pm

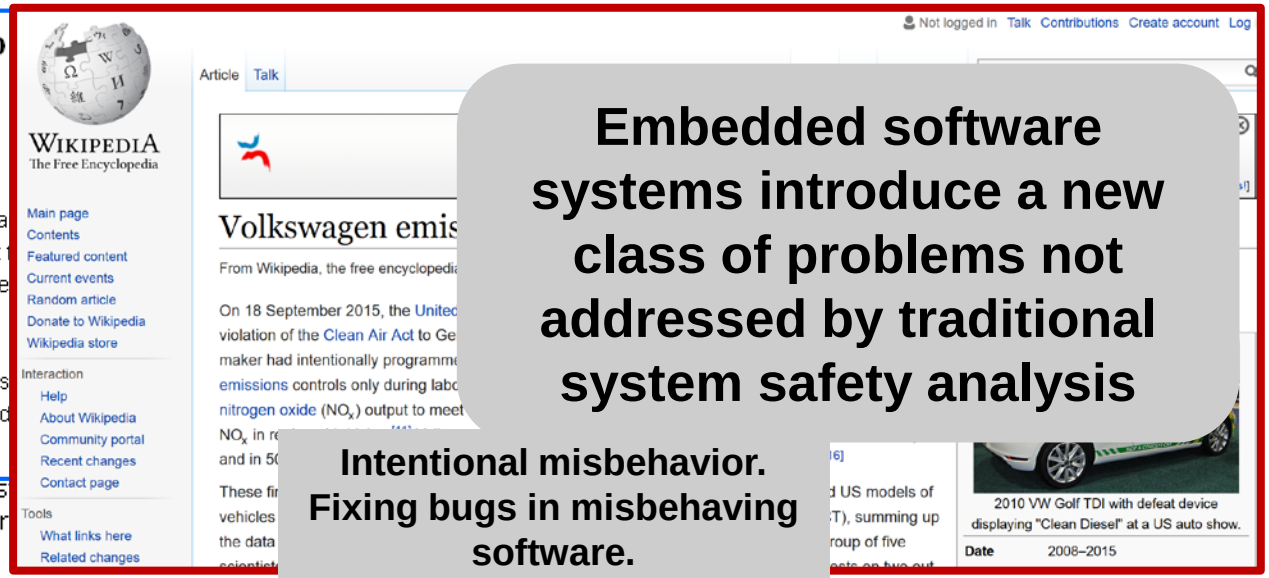
From: soyawannaknow.blogspot.com

Thirty-six passengers a
in a mid-air drama that
emergency landing, the
Tuesday.

The terrifying incident s
mayday call when it suc
from Singapore to Perth, Qantas said.

Australian Transport Safety Bureau said yes
650 feet within seconds, slamming passenger
ceiling, before the pilots regained control.

"This appears to be a unique event," the bureau said, ad



Embedded software systems introduce a new class of problems not addressed by traditional system safety analysis

Intentional misbehavior. Fixing bugs in misbehaving software.

Kenmore 4027

Expert • Independent • Nonprofit
ConsumerReports.org

This article appeared in
May 2010 Consumer Reports Magazine.

How do you upgrade washing machine software?

Many appliance
turned out to b
2010 report.
Our tests foun
ing.
When Sears,
ith LG, which makes
the washer, to
ort or missing rinse
and wash cycles, affecting wash performance. Sears and LG say they have reprogrammed the software on the models in their warehouses and on about 65 percent of the washers already sold, including the ones we had purchased.

Our retests of the reprogrammed Kenmore 4027 found that the cycles now worked properly, and the machine excelled. It now tops our Ratings (available to subscribers) of more than 50 front-loaders and we've made it a CR Best Buy.

If you own the washer, or a related model such as the Kenmore 4044 or Kenmore Elite 4051 or 4219, you should get a letter from Sears for a free service call. Or you can call 800-733-2299.

AA says software problem with oeing 787s could be catastrophic

an Catchpole
dcatchpole

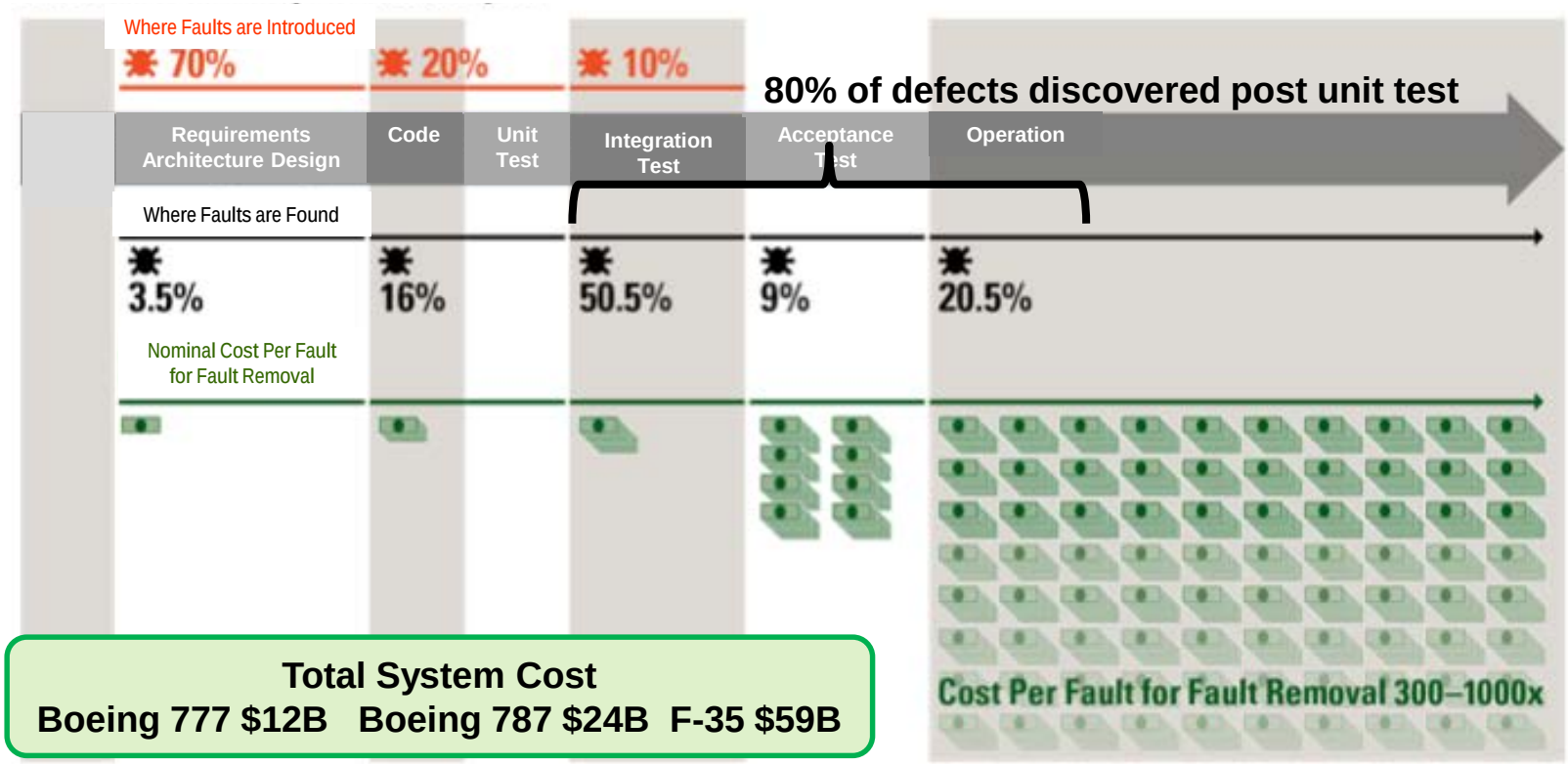
Federal Aviation Administration says a
vare problem with Boeing 787
mliners could lead to one of the
nced jetliners losing electrical power in
t, which could lead to loss of control.

- **The Buzz**: Hipster's dilemma
- Boeing & aerospace news
- Aerospace blog

FAA notified operators of the airplane Friday that if a 787 is powered continuously for 248 days, the plane will automatically shut down its alternating current (AC) electrical power.

Critical System Cost Drivers

Software Development Life Cycle



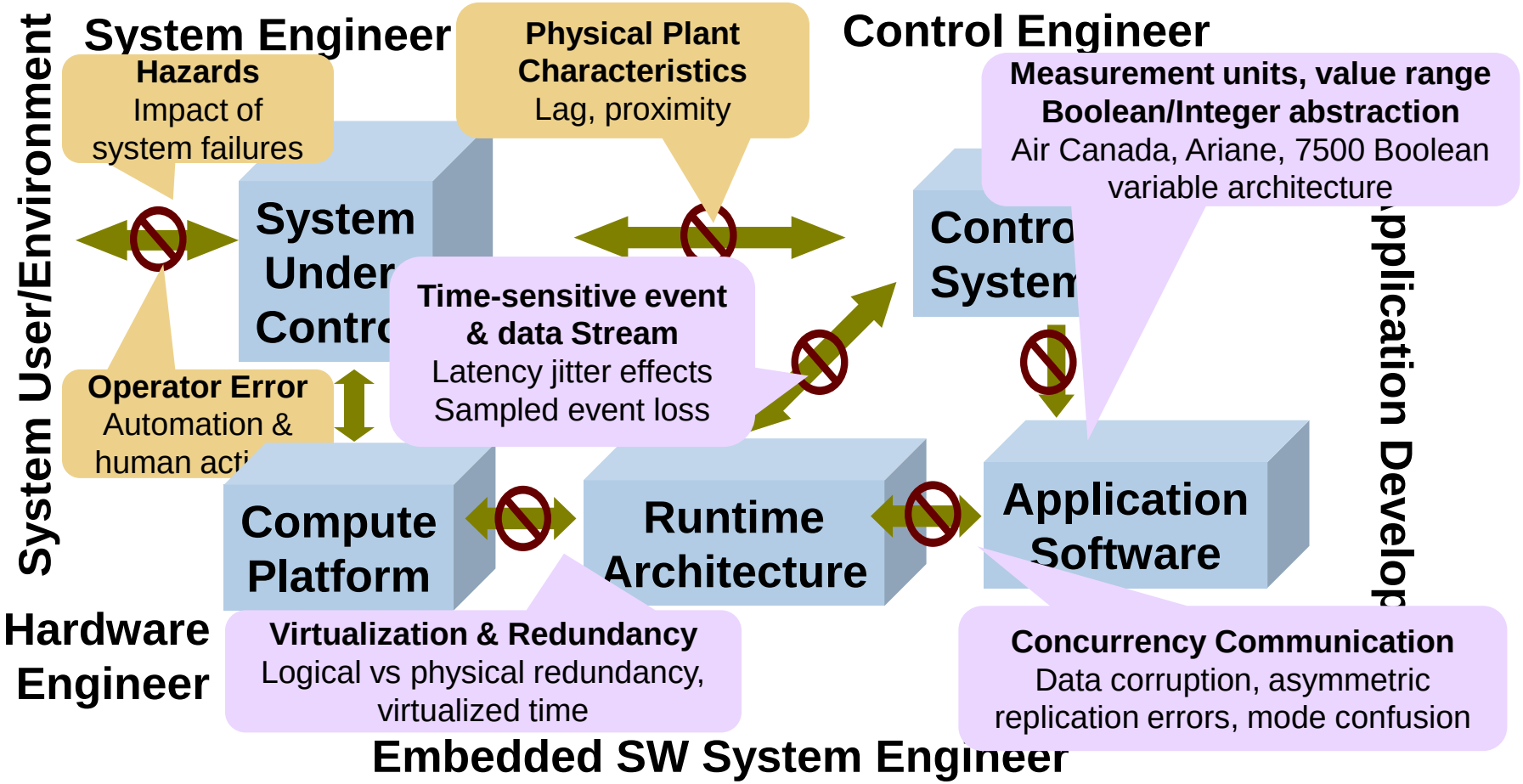
Total System Cost
Boeing 777 $12B Boeing 787 $24B F-35 $59B

Sources: *Critical Code*; NIST, NASA, INCOSE, and Aircraft Industry Studies

Post-unit test software rework cost
50% of total system cost and growing

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

Mismatched Assumptions in System Interactions



Embedded software system as major source of hazards

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

Model Based Engineering (MBE)

Modeling and Simulation have been around for a long time

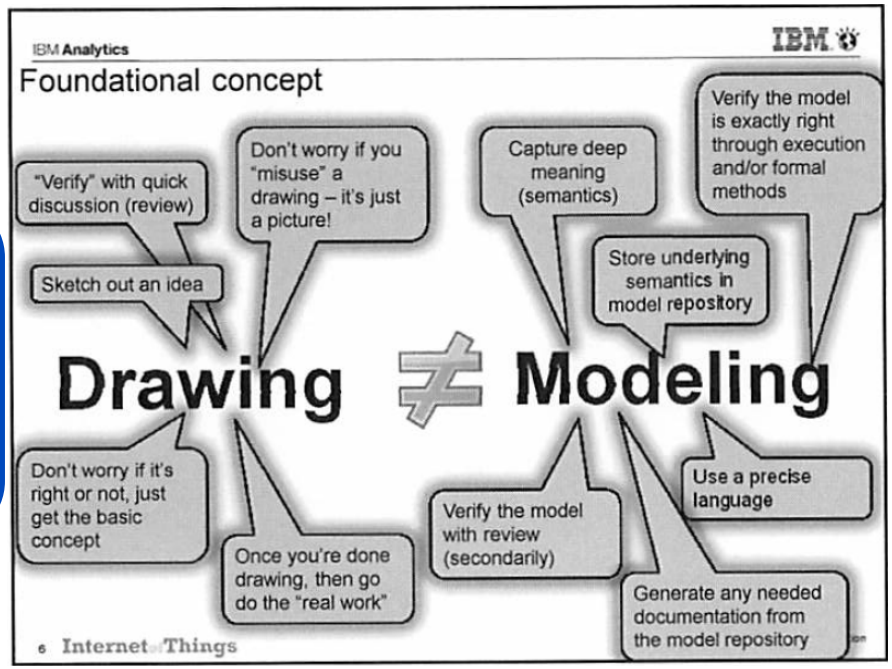
Continuous and discrete state modeling, verification and code generation of detailed design (Mathworks Simulink, ANSYS SCADE)

Software modeling through UML with limited semantics

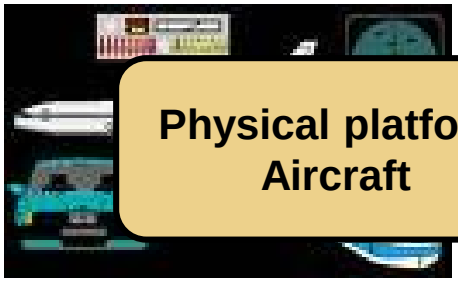
Early system modeling with SysML

Our MBE Focus

Safety-Critical Software Systems with stringent Safety, Security, and Performance Requirements



Architecture Analysis & Design Language (AADL)

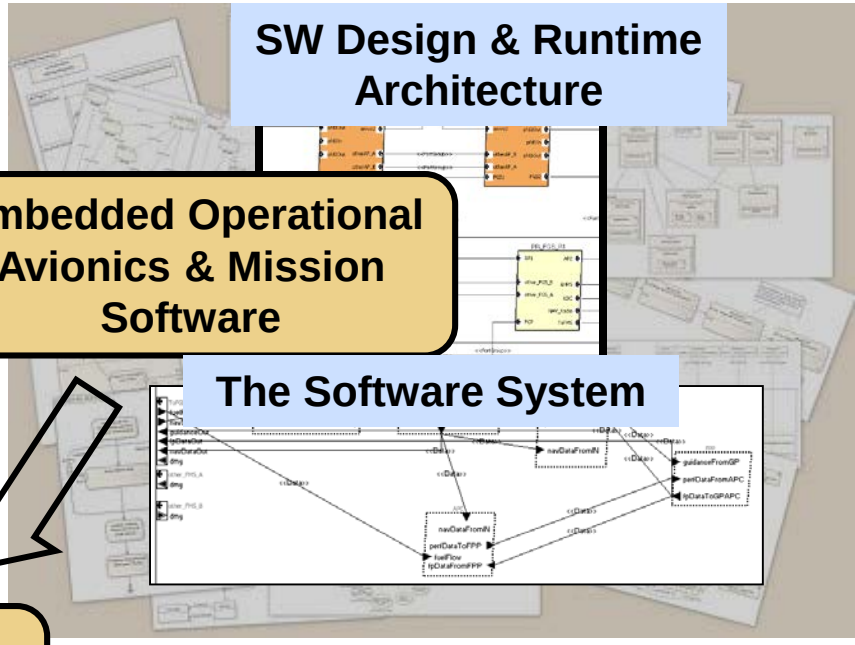


Physical platform
Aircraft

Command &
Control

Embedded Operational
Avionics & Mission
Software

SW Design & Runtime
Architecture



The Software System

Deployed on
Utilizes

Physical interface
Platform component

Computer System
Hardware & OS

The Computer System

AADL focuses on interaction between the three elements of embedded software systems.

SAE AADL Standard Suite (AS-5506 series)

Core AADL language standard (V2.2-Jan 2017, V1-Nov 2004)

- Strongly typed language with well-defined semantics
- Textual and graphical notation
- AADL V3 in progress

Peter Feiler has been the technical lead since 1999

Authored core AADL and EMV2

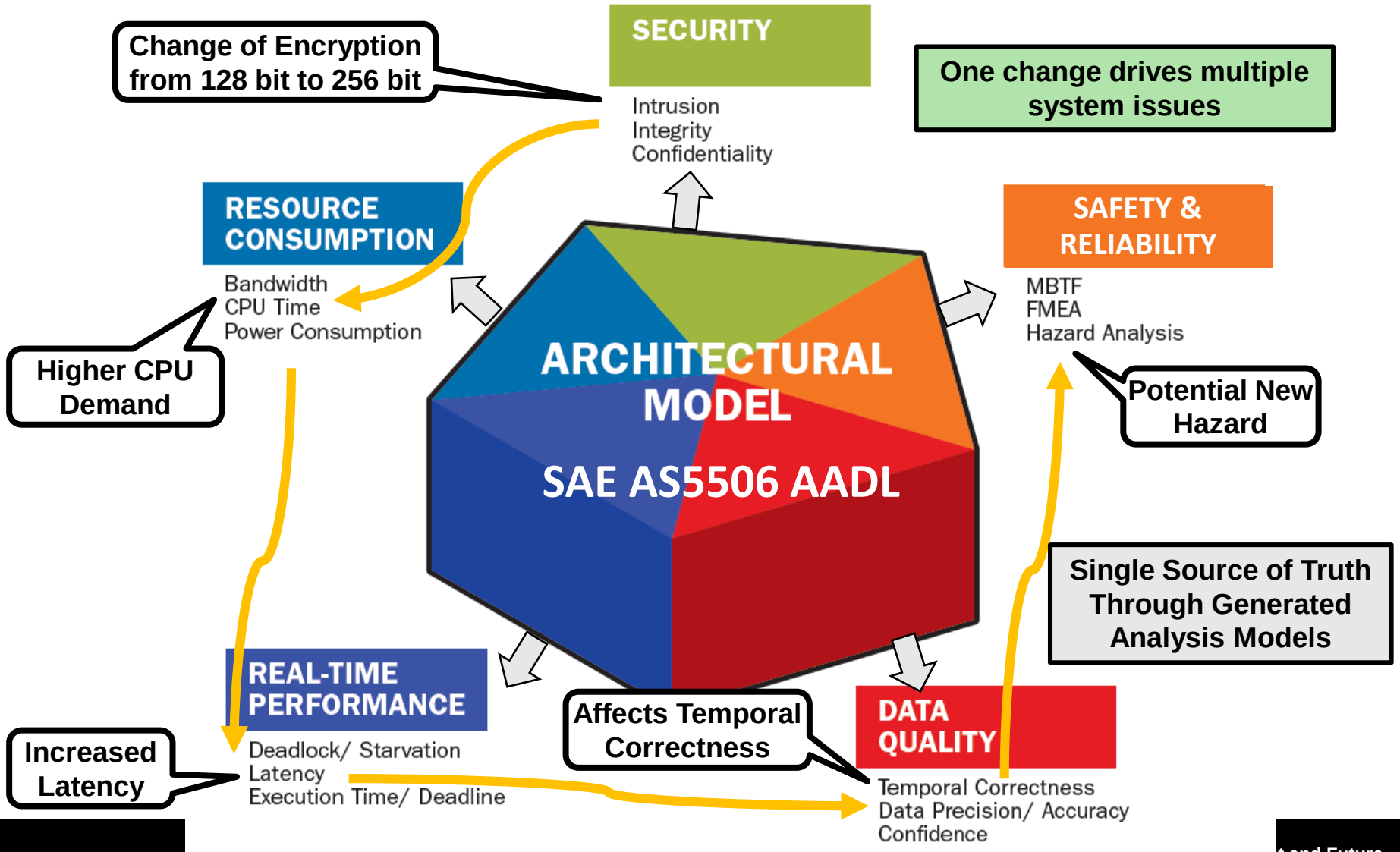
Standardized AADL Extensions

Error Model (EMV2) Annex for safety, reliability, security analysis
ARINC653 Annex for partitioned architectures
Behavior Specification Annex for modes and interaction behavior
Data Modeling Annex for interfacing with data models (UML, ASN.1, …)
Runtime System Annex for executive generation

AADL Extensions in Progress

Networking Annex
Security Annex
Requirements Definition and Assurance Language
Synchronous System Specification Language
Hybrid System Specification Language
System Constraint Specification Language

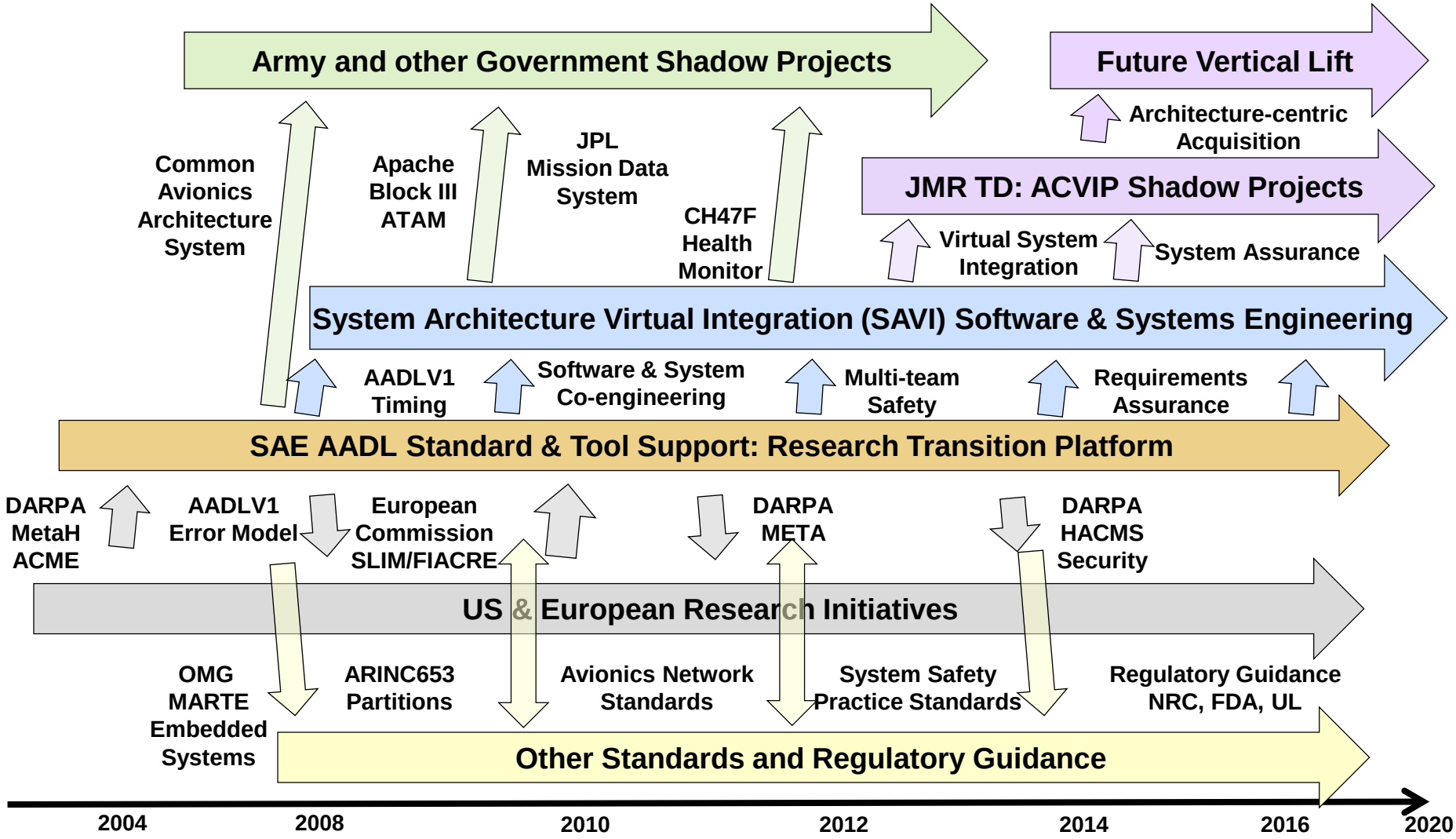
Analyzable Architecture Models Discover System Level Issues Early in Development



SAE AADL & Architecture-centric Virtual Integration

Evolution, Maturation and Transition

AMRDEC has funded AADL standards development since 1999



Army Pilot Projects with AADL

Common Avionics Architecture System (CAAS) 2004

- Modeling and analysis of reference architecture

Comparative study of 6 CAAS based helicopter systems 2007

- Single AADL model identifies commonalities and differences

Apache Block III Model-based Architecture Evaluation (ATAM) 2008

- AADL modeling during ATAM uncovers additional issues

ASSIP Real-time System State-of-the-Art 2009

- Virtual system integration as emerging technology

Virtual Upgrade Validation Method (VUV) 2010

- Codifies software system root cause areas

Apache Mission Processor Upgrade Study 2011

- Application of VUV method

Reliability Improvement Framework Study for Aviation Engineering Directorate 2011

- Four pillars of a qualification improvement strategy

Apache Flight Management System 2012

- Analyzed 3100 page requirement document with 7500 global Boolean flags

CH47-F Health Monitoring System Upgrade study 2012

- Pre-CDR discovery of system issues

AMRDEC JMR program 2014-2020

- Technology roadmap & ACVIP shadow project (FY14/15), MSAD (FY16/17), Capstone (FY18-19)

Architecture-Centric Virtual System Integration with AADL

Investment Highlights

SEI & AMRDEC SED long term strategy for maturing architecture-centric virtual system integration

International commercial aerospace industry AVSI SAVI

D-MILS Euro-MILS: Design of Secure Systems

Multiple projects in DARPA META&HACMS programs

NASA research by University Minn & Rockwell Collins

10+ Chinese patents based on papers written by others

Spreading the Word

5000+ OSATE Downloads per release

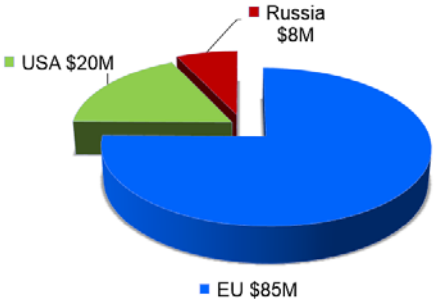
53,000 separate visitors to AADL website

AADL Book by Feiler/Gluch

20+ SEI blog posts, 40+ podcasts, 1 webinar

60+ SEI publications

1000+ publications by others



Workbenches

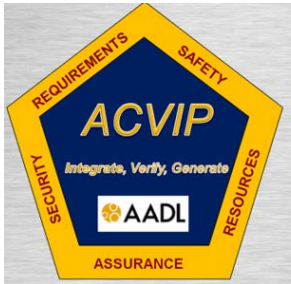
OSATE: Open source by SEI

AADL Inspector: commercial

TASTE: European Space Agency

MASIW: Russian aerospace

ANSYS: Commercial Tool Suite



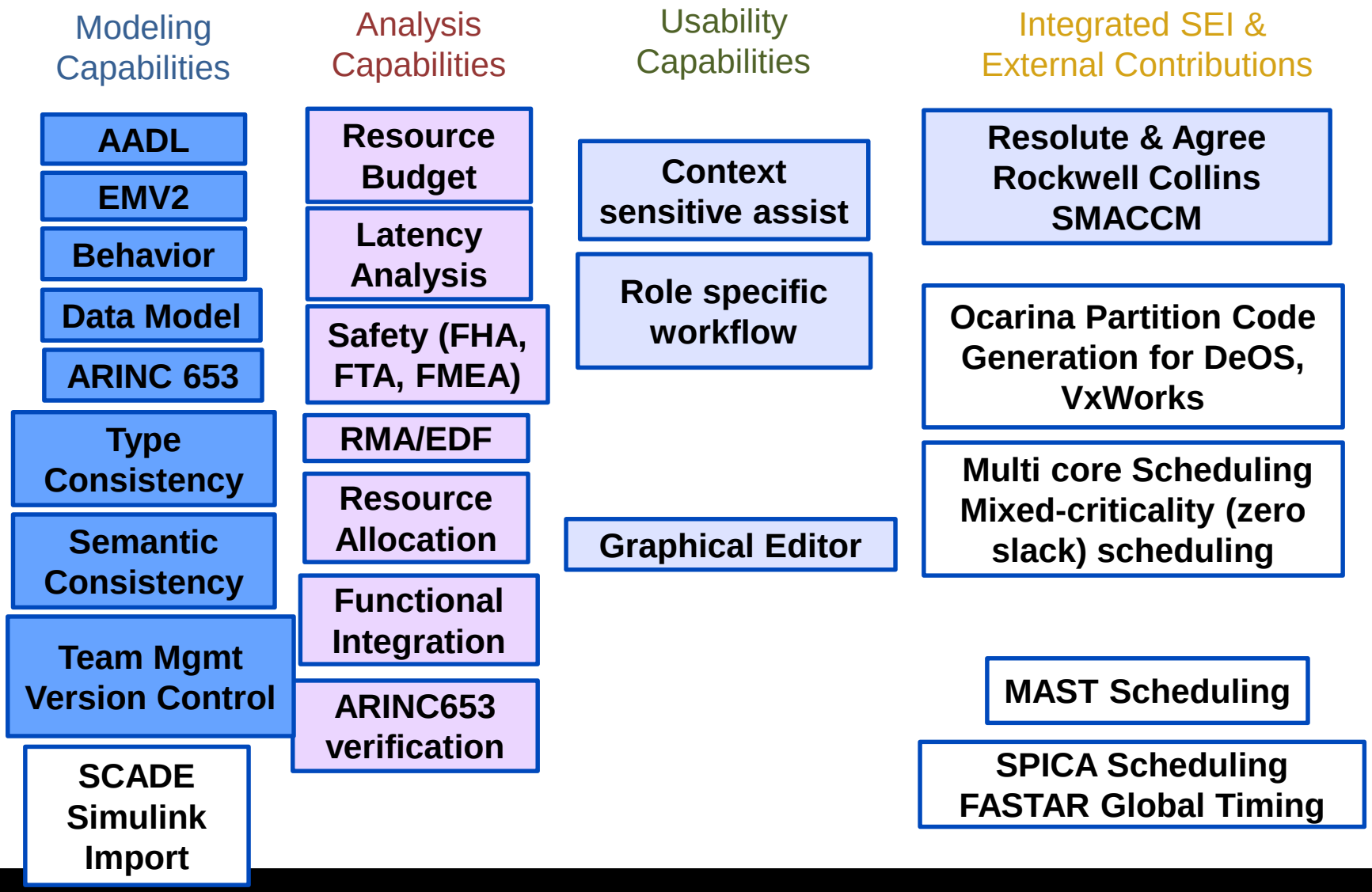
OSATE

Eclipse-based Open Source AADL Tool Environment (OSATE)

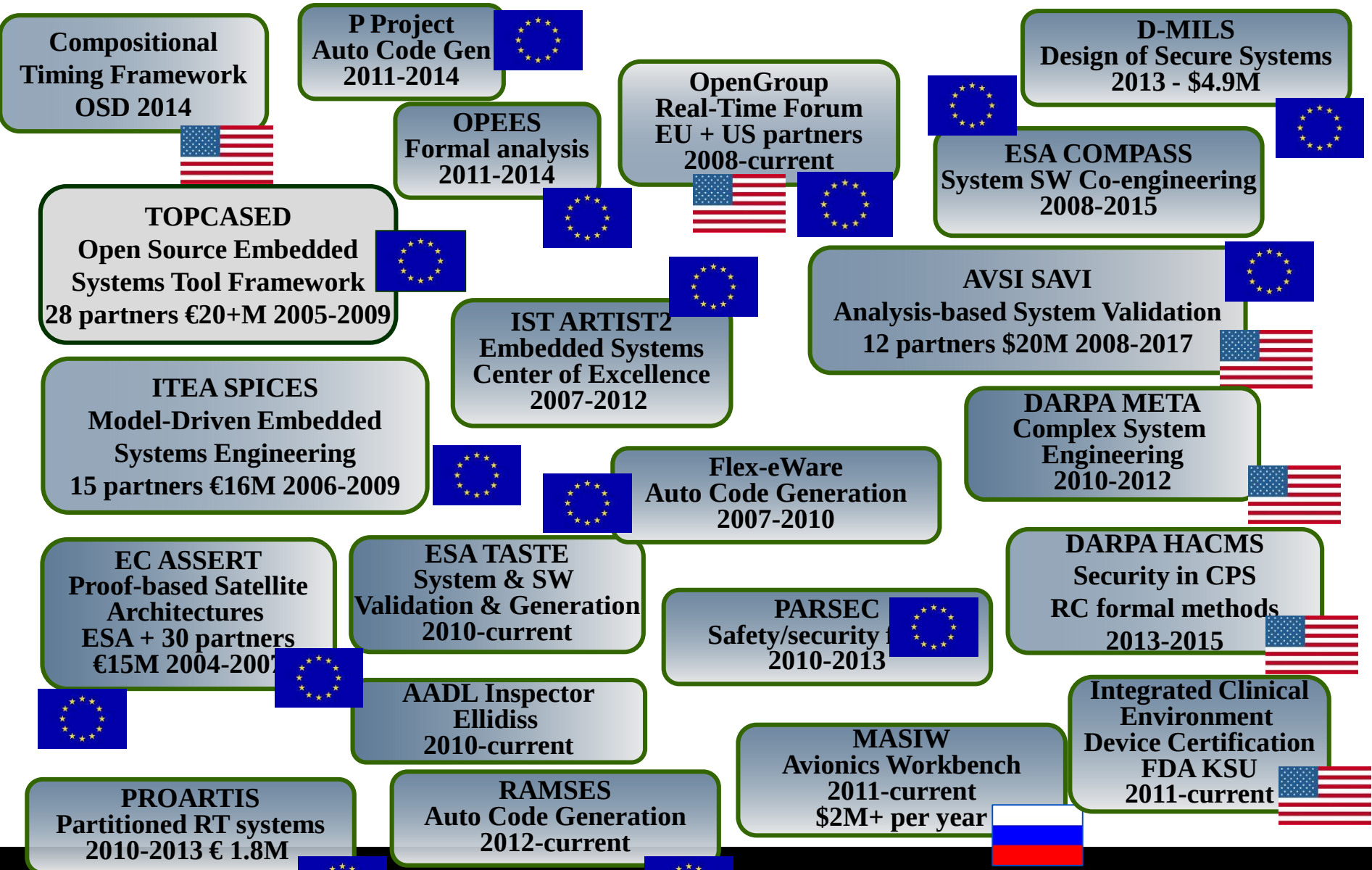
- No cost license under EPL license
- Reference implementation of core AADL and annex standards
- Original release in 2004 with publication of AADL standard
- Maintained on GitHub
- OSATE release cycle
  - Quarterly stable release, Nightly builds

- AADL/OSATE community
  - AADL Wiki
  - Aadl.info and osate.org websites
  - Discussion forums
- Used in pilot projects
  - SEI customer work, Industry and universities around the world
- Research prototyping platform
  - European projects, DARPA META and HAMCS projects

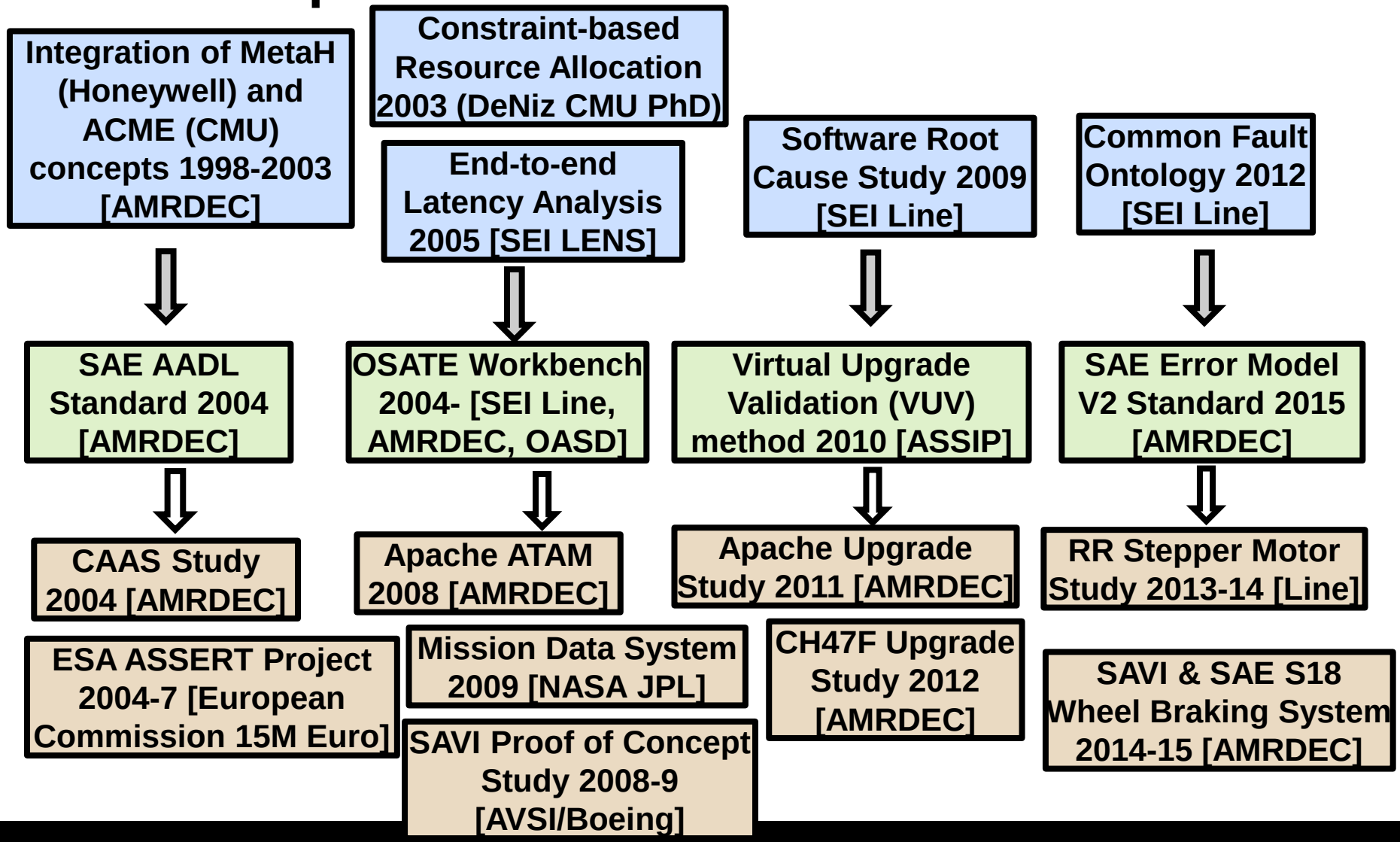
OSATE Workbench Capabilities



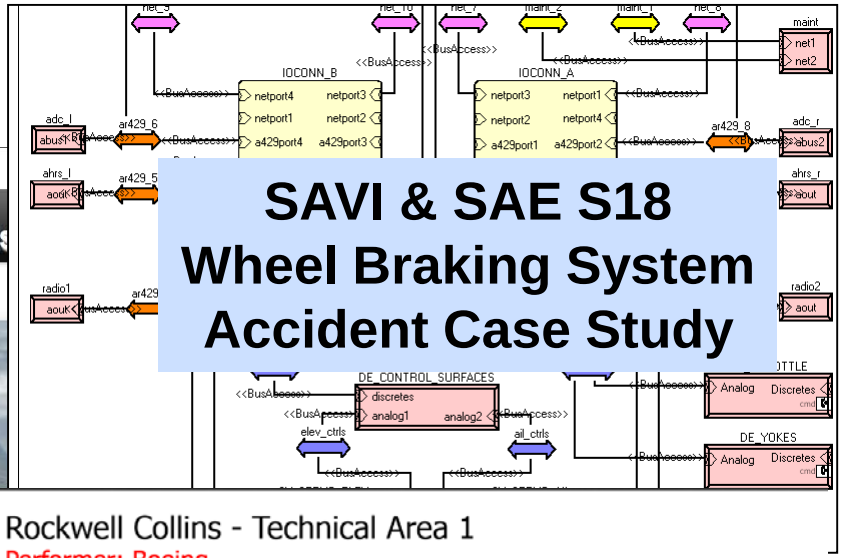
Sampling of International Efforts Leveraging SAE AADL



From Research to Practice under SEI Leadership



Studying Real Systems



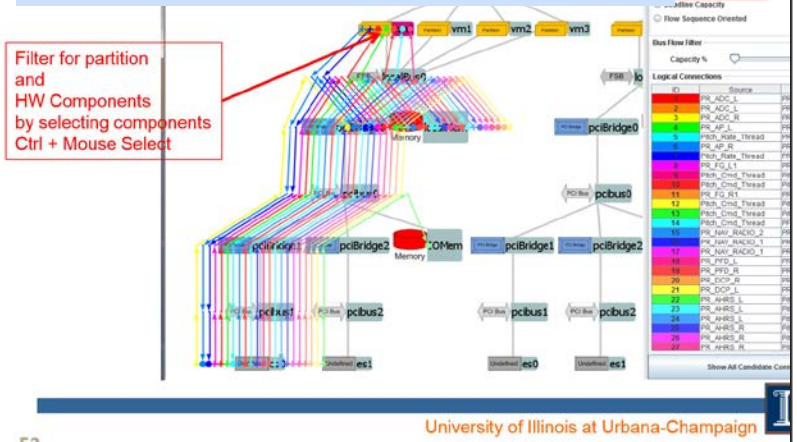
SAVI & SAE S18 Wheel Braking System Accident Case Study

Institut Supérieur de l'Aéronautique et de l'Es

Boeing 777 Accident Analysis

ASIIST Bus Delay Visualization

F-35 Performance Study



University of Illinois at Urbana-Champaign

52

DARPA

Rockwell Collins - Technical Area 1

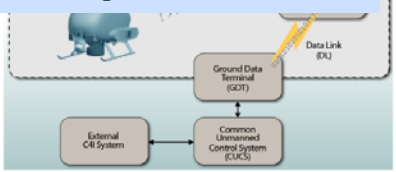
Performer: Boeing

- Real-time Flight Control Computer (FCC)
- Manages safety critical systems

- Mediates communication with CUCS
- Translates STANAG 4586 commands into vehicle specific actions
- Manages non safety critical mission equipment packages

DARPA HACMS No Security Break-in

- Pilot-able Unmanned
- **Mission-tested**
- **Multi-level challenge problems:** range from low-level embedded real-time system to higher level vehicle domain requirements
- **Unclassified Platform**
- Live **Flight Demo** in Phase 3
- **Transition Target:** NCORE common open architecture which is the basis for systems based on both AH-6 and AH-64 (Apache) airframes



Source: HACMS Rockwell Collins Proposal

Utilizes common military buses for comms:

MIL-STD-1553B,

ARINC-429,

Ethernet, VME, RS-422, and RS-232

Example: CUCS sends STANAG waypoint commands via the Data Link (TCDL); waypoint translated into vehicle specific actions by VSM and passed to FCC where it is converted into appropriate control commands.

Distribution Statement A - Approved for Public Release, Distribution Unlimited

Incremental Multi-Tier Assurance in SAVI

Aircraft: (Tier 0)

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

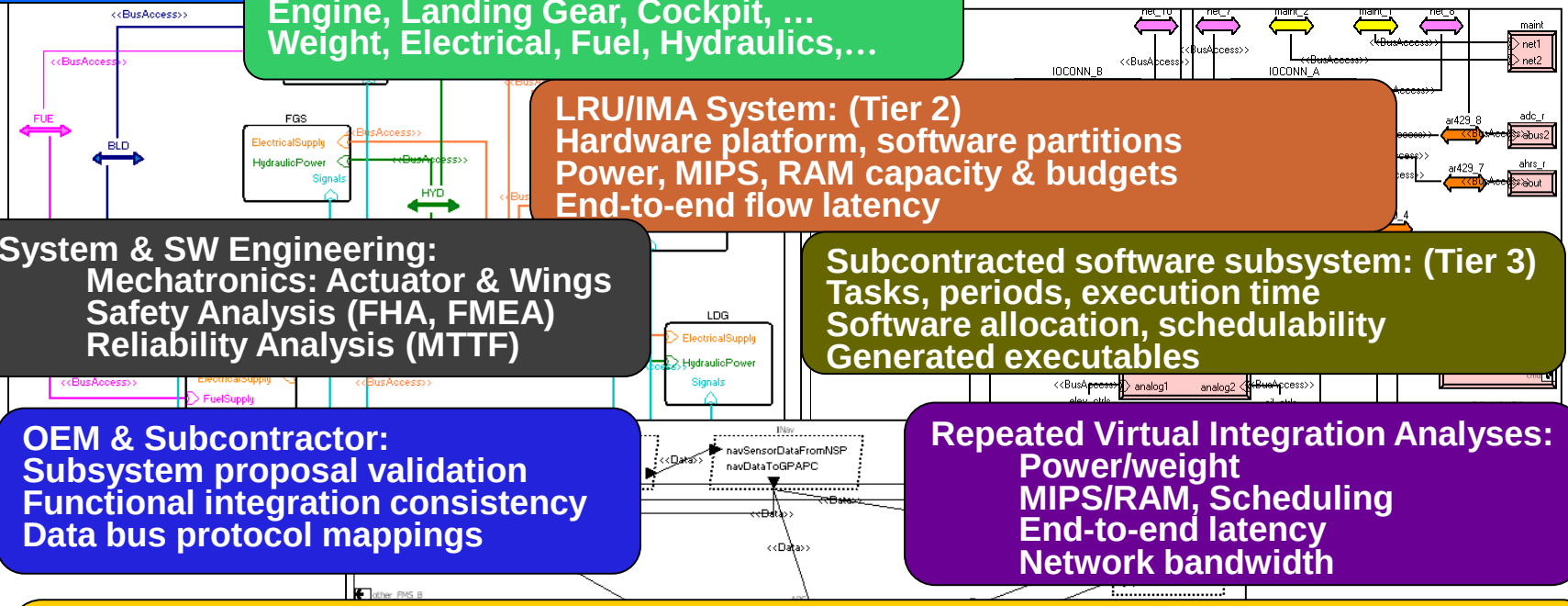
LRU/IMA System: (Tier 2)
Hardware platform, software partitions
Power, MIPS, RAM capacity & budgets
End-to-end flow latency

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

- **Architecture-centric model-based software and system engineering**
- **Architecture-centric model-based acquisition and development process**
- **Multi notation, multi team model repository & standardized model interchange**

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

Return On Investment: There is No Magic

- A typical commercial airplane development program should cost ~$10B in 2007
  - $2B/year x 5 years = $10B
  - "The development cost of the A380 had grown to €11 billion when the first aircraft was completed." http://en.wikipedia.org/wiki/A380
- In aerospace, rework is in the 50% range
- 50% rework x $10B = $5B opportunity
- About half of this opportunity, $2.5B, will be related to airplane systems
- Most of that will be related to software
- **There is a large opportunity**

Mission Systems Architecture Demo (MSAD)

- **Effective Acquisition**
  - Competitive Opportunities
  - Reduced Vendor Lock
  - Increased Affordability
- **Efficient Integration**
  - Reduced Time to Field
- **Improved Capabilities**
  - Portable / Reusable
  - Interoperable
  - Upgradeable / Resilient
  - Planned Variability
- **Efficient Qualification**
  - Safe/Secure

Boeing · Sikorsky · Rockwell Collins · Honeywell · GE Aviation · Lockheed Martin · Northrop Grumman

JCA AIPD Capstone

JMR matures the architectures, tools and processes for FVL and legacy systems

Purpose:

Investigate/Mature processes, tools and standards necessary to specify, analyze, design, implement and qualify a Mission Systems Architecture in support of emerging FVL PoR that meets Army business goals

Approach:

- Leverage or develop the standards and tools necessary to successfully implement a mission systems architecture
- Execute a series of increasingly complex demos - Learn by doing

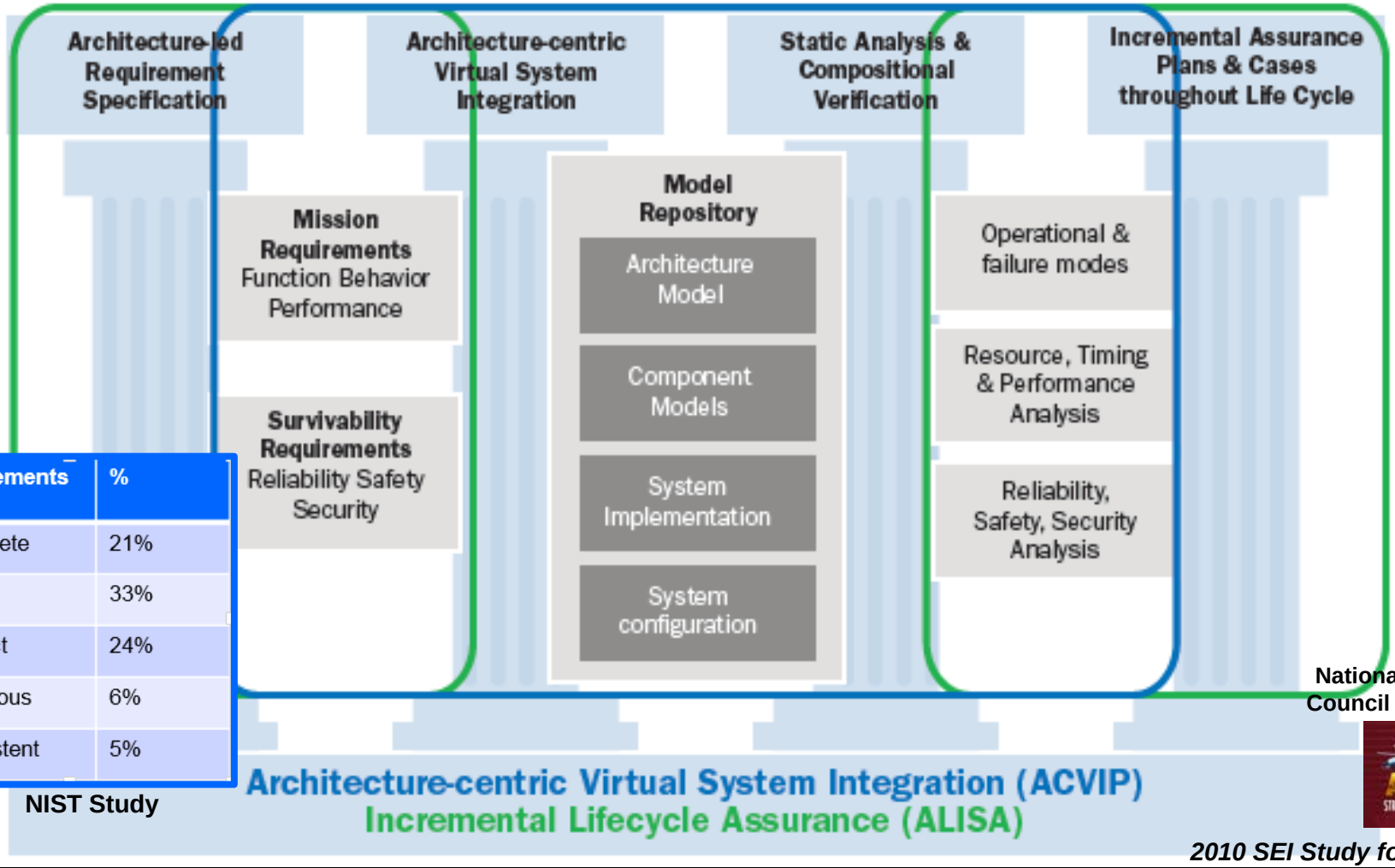
Focus Areas:

- Implementation of Open System Architectures (OSA)
  - Joint Common Architecture (JCA)
  - FACE™ Technical Standard
  - Hardware Open Systems Technologies (HOST)
- Application of Model Based Engineering (MBE)
  - Model-based specification/acquisition
- Execution of an Architecture Centric Virtual Integration Process (ACVIP)
  - Predictive performance assessment

TECHNOLOGY DRIVEN. WARFIGHTER FOCUSED.

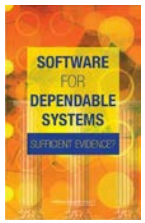
Verification: Assurance and Qualification Improvement Strategy

Assurance: Sufficient evidence that a system implementation meets system requirements



Requirements error	%
Incomplete	21%
Missing	33%
Incorrect	24%
Ambiguous	6%
Inconsistent	5%

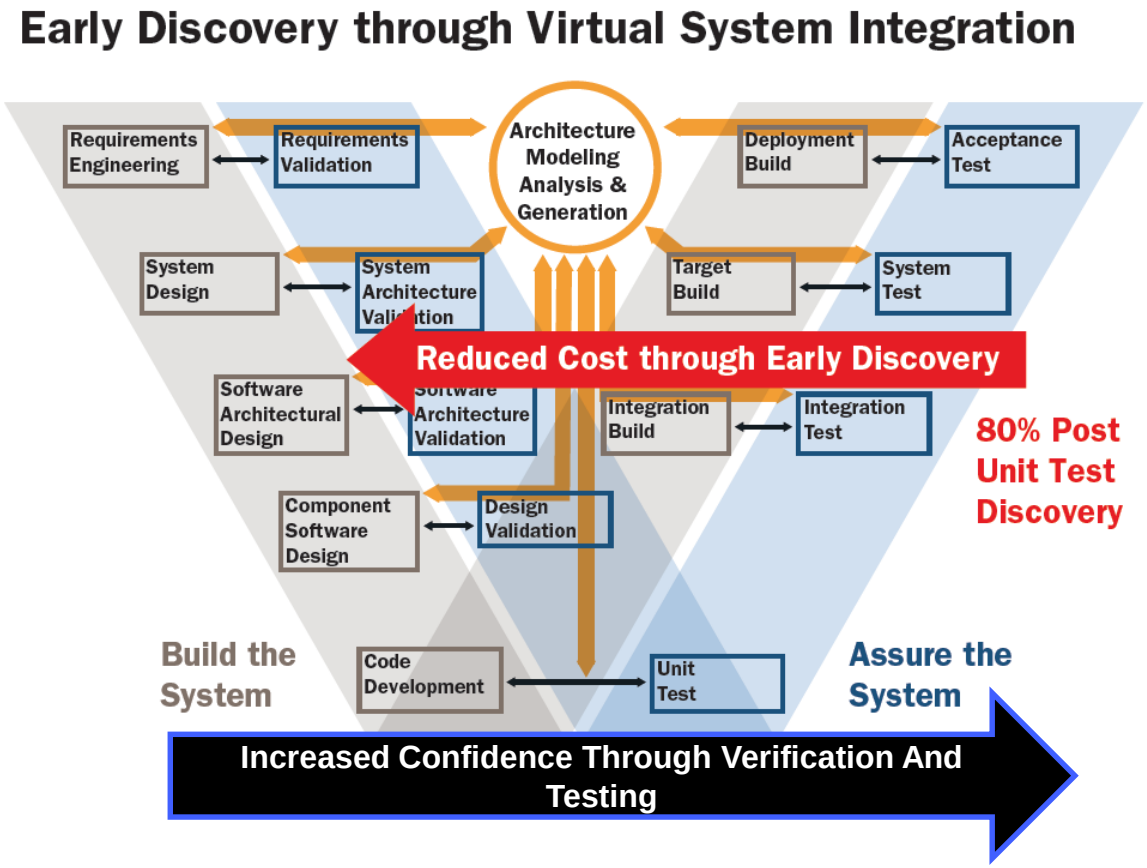
NIST Study



National Research Council Study 2007

2010 SEI Study for AMRDEC

Benefits of Virtual System Integration & Incremental Lifecycle Assurance



Current Projects

Maturing practices and raising TRL of tool support

- FY14-15: AADL Workbench (OASD funded)
- FY15-21: AMRDEC and JMR support (funded)
- FY18-19: Integration with FACE (funded)

Broadening perspective from development to assurance

- FY15-16: Incremental Life Cycle Assurance of Critical Systems
- FY16-17: Automated Assurance of Security Policy Enforcement
- FY17: Guided
- FY18-20: Integrated Safety & Security Engineering of Mission Critical System

New domains & organizations (regulatory agencies)

- FY15-17: NRC (funded)
- FY15-16: FAA (funded, collaboration)
- FY15-17: FDA/KSU (collaboration)

Looking Ahead

Technology Creation and Maturation Funding

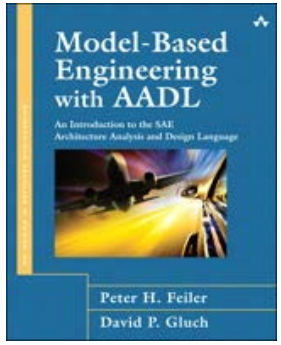
- Continued evolution of SAE AADL standard suite
- Common approach to system safety and security robustness
- Integration with other standards (network, security, safety, multi-core)
- Integration with system engineering practices

Transition of Virtual Integration Technology Funding

- Training, mentoring, community of interest lessons learned
- Cost effectiveness studies
- Transition to other programs
- Incorporation into acquisition process

This Requires Coordinated Community Investment

References



AADL Website www.aadl.info and AADL Wiki www.aadl.info/wiki

Blog entries and podcasts on AADL at www.sei.cmu.edu

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

On AADL and Model-based Engineering
http://www.sei.cmu.edu/library/assets/ResearchandTechnology_AADLandMBE.pdf

On an architecture-centric virtual integration practice and SAVI
http://www.sei.cmu.edu/architecture/research/model-based-engineering/virtual_system_integration.cfm

On a four pillar improvement strategy for software system verification and qualification
http://blog.sei.cmu.edu/post.cfm/improving-safety-critical-systems-with-a-reliability-validation-improvement-framework

Webinars on system verification https://www.csiac.org/event/architecture-centric-virtual-integration-strategy-safety-critical-system-verification and on architecture trade studies with AADL https://www.webcaster4.com/Webcast/Page/139/5357