Carnegie Mellon University Software Engineering Institute

Applying ACVIP for Verification by Analysis during Airworthiness Qualification

MAY 12, 2023

John J. Hudak, PE
Principal Engineer

Document Markings

This material is based upon work funded and supported by the Department of Defense U.S. Army Combat Capabilities Development Command Aviation and Missile Center (DEVCOM AvMC) 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.

DM23-0459

Joint Investigation

Software Engineering Institute

- John Hudak, PE
- Dr. John McGregor

Adventium Labs, now a part of Galois, Inc.

- Bruce Lewis
- Charles Payne

US Army DEVCOM AvMC

- Alex Boydston

For copies of the report, contact Alex Boydston, (alex.k.Boydston.civ@army.mil)

Motivation and Approach

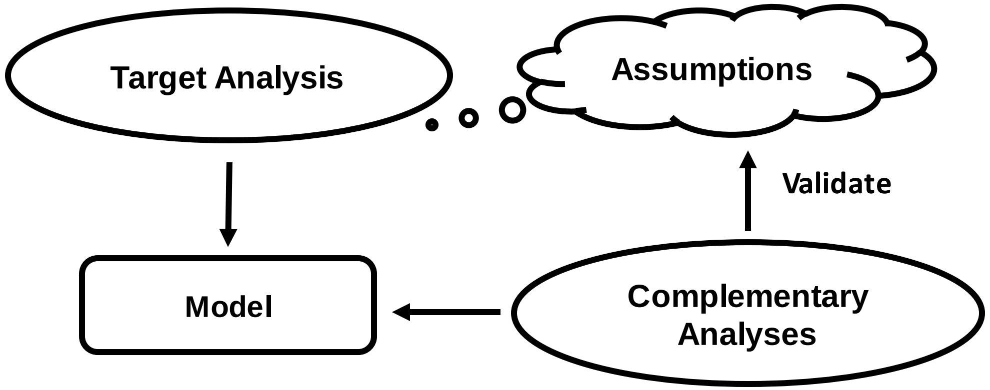
Airworthiness qualification costs continue to rise, because with increased use of software, there is a corresponding increase in software integration failures.

- To counter this trend, the Army has embraced the DoD Digital Engineering Strategy and created the Architecture-Centric Virtual Integration Process (ACVIP) to detect integration defects early.

Airworthiness authorities rely on Verification by Analysis (VbA) to detect defects prior to testing.

ACVIP enhances VbA with:

- Predictive analysis using standardized meaning
- Continuous assessment against an authoritative source of truth (model)
- Complementary analyses that validate target analysis assumptions



Description: Architecture-Centric Vertical Integration Process (ACVIP)

ACVIP is an approach used to model and analyze architectures for complex, software-intensive, embedded computing systems to reduce integration risks

ACVIP provides methods and tools to address system development where run-time sensitivity, safety, and cybersecurity are critical

ACVIP provides a virtual integration environment for early detection of defects not typically found until physical integration. This is accomplished using:

- continuous verification throughout the development lifecycle
- a consistent representation of the system by coordinating multiple models, languages, domains, and design entities
- the Architecture Analysis & Design Language (AADL) which is domain specific to embedded systems

ACVIP significantly reduces risk in embedded software / hardware integration, and increase likelihood of delivering full capabilities on schedule,within budget

ACVIP Resources

ACVIP is supported by the following resources

- The *ACVIP Overview Handbook* provides overall motivation, modeling strategies, and workflow guidance [ACVIP Overview 2019]
- The *ACVIP Acquisition Management Handbook* discusses existing acquisition strategies (including the Modular Open Systems Approach (MOSA), Future Airborne Capability Environment (FACE™) , and Comprehensive Architecture Strategy (CAS)), stakeholders, and development milestones and provides a sample workflow [ACVIP Acquisition 2020]
- The *ACVIP Modeling and Analysis (M&A) Handbook* discusses modeling goals and strategies to support ACVIP and recommends analyses for common development milestones [ACVIP M&A 2021]

Characteristic of the Army Military Airworthiness Certification Criteria (AMACC)

The AMACC establishes the airworthiness certification requirements stated in terms of criteria, standards and methods of compliance used in the determination of airworthiness of all manned & unmanned aircraft.

Airworthiness qualification (or certification) is a progressive assessment process performed at the component, subsystem, and system levels to ensure that a system meets airworthiness requirements.

The substantiation data delivered against the requirements will be used to perform an airworthiness assessment and determine if any potential hazard exists.

ACVIP in the context of the AMACC

Army Military Airworthiness Certification Criteria (AMACC)

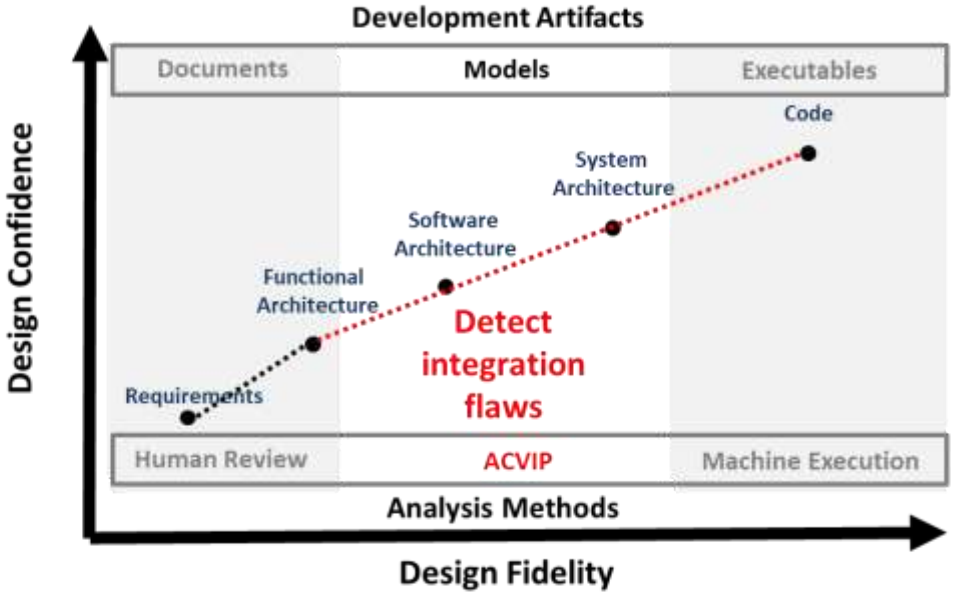
Revision A Change 2 (C2)

Prepared by:

U.S. Army Combat Capabilities Development Command

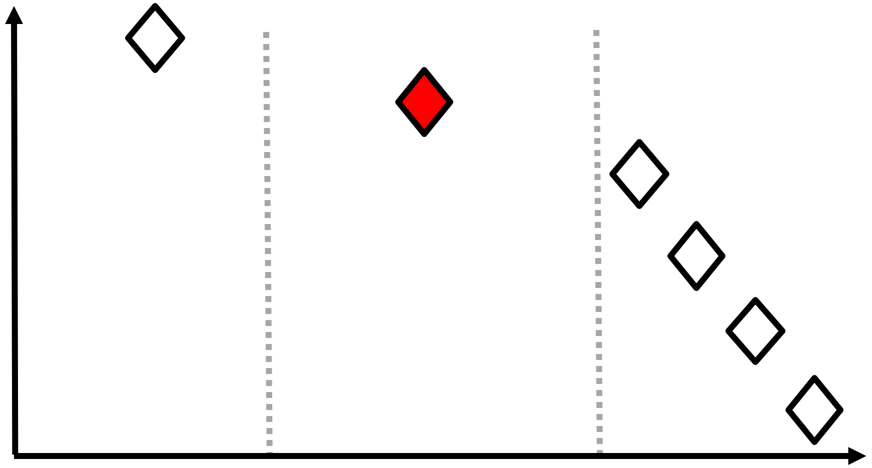
Aviation & Missile Center (DEVCOM AvMC)

Redstone Arsenal, AL 35898

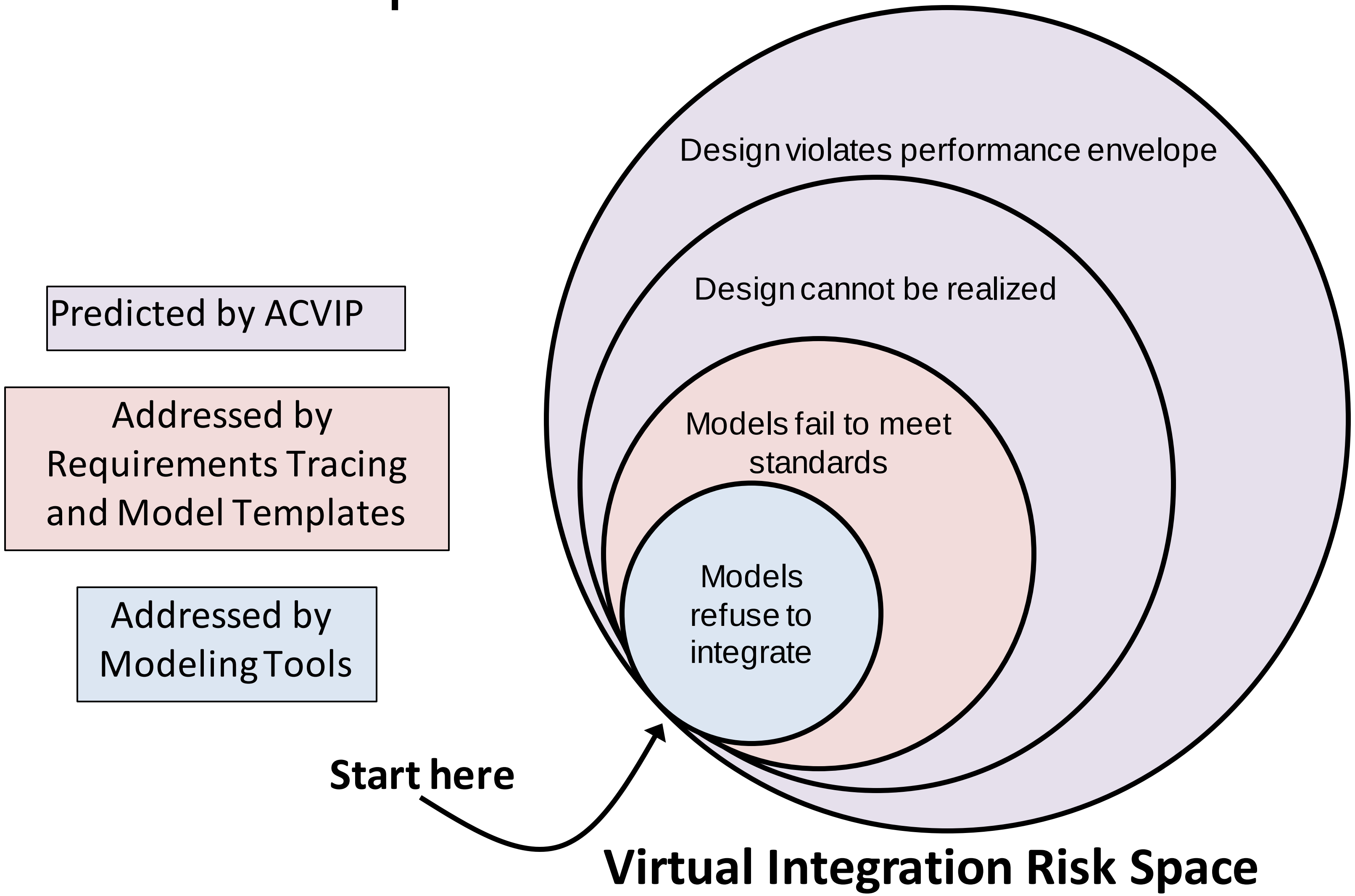


Verification Methods cited in AMACC

- Similarity
- Analysis (VbA)
- Testing
- Demonstration
- Simulation
- Inspection



ACVIP Addresses Deeper Risks



ACVIP extends the reach of conventional checks.

Applying ACVIP to an Example from AMACC

Section 9 – Human Systems Integration

AMACC Requirement 9.4.2.1: The total system latency for the presentation of primary flight information used for real-time control of an aircraft should not exceed 100 ms.

AMACC 9.4.2.1 calls for three methods of compliance:

- Verification by demonstration: “The display shall not exhibit flicker that is discernible to the eye.”
- Verification by analysis: “Document timing allocations and expected system response times” and “as the system design evolves or is modified, again analyze … based on the updated and refined timing allocations and expected system response times.”
- Verification by test: “Test that the latency budgets … are valid for all critical and safety critical tasks and functions.”

ACVIP Guidance for Latency Analysis – From M&A Handbook

At System Requirements Review (SRR)

- The SRR model should declare requirements that are allocated to the architecture and its components and are to be verified by analysis of the architecture model….
- A simple form of latency analysis is to check consistency between end-to-end flow requirements and subflow requirements derived from them. Analysis that verifies consistency between system latency requirements and derived/allocated subsystem latency requirements may be desired at SRR.

At Preliminary Design Review (PDR)

- The PDR model will be an elaboration of the SRR model that fully identifies software and hardware configuration items and their interfaces. … A PDR model may contain process, subprogram group, and data declarations (software objects); and virtual processor, processor, virtual bus, bus, device, and memory declarations (hardware objects).
- Repeat the SRR analysis on the more detailed model.

Repeat for later milestones (e.g., CDR, TRR) with higher fidelity models of the design.
Resolve issues in next phase.

Latency Result Assumptions

During latency analysis, the contractor may assume that delays will not occur due to

- resource contention
- scheduling constraints
- component deadlock
- safety faults
- cybersecurity attacks

If these conditions do exist, will the result still hold?

The airworthiness authority cannot validate analysis assumptions using the latency result itself, and the latency analysis is incomplete without that validation.

ACVIP Guidance for Complementary Analysis

Source: ACVIP Modeling and Analysis Handbook

At the SRR milestone, consider these analyses:

- Interface Behavior Consistency Analysis will detect components that could deadlock.
- Resource Loading Analysis for key performance parameters will detect components that could fail to operate within assigned performance envelope.
- Reliability, Availability, and Failure Analysis will detect components that could fail in particular states.
- Functional Hazard Analysis (FHA), Fault Tree Analysis (FTA), Failure Modes and Effects Analysis (FMEA), and System Theoretic Process Analysis (STPA) will detect hazards that could block or slow data flows.
- Cross-Domain Analysis will detect the need for cross-domain solutions (e.g., guards), which could increase latency for data flows that must traverse those guards.
- Risk Management Analysis will detect data flows that could interfere with each other.

Complementary Analysis (con't)

At the PDR milestone, add these analyses:

- Functional Hazard Assessment (FHA) to identify hazards and set criticalities and levels of rigor
- System Theoretic Process Analysis (STPA) to define control loops, identify unsafe control actions and identify mitigations/constraints
- Failure Modes and Effects Analysis (FMEA) will detect components with insufficient fault handling.
- Fault Tree Analysis (FTA) will detect components whose failures are not independent.
- Reliability Block Diagram (RBD) Analysis will detect components with unanticipated interdependencies.
- Markov Analysis will detect components that lack sufficient ability to recover from failures.

Iterate on these analyses at subsequent milestones (e.g., CDR, TRR)

ACVIP and Safety

The ACVIP M&A handbook specifies that the contractor shall use MIL-STD 882E and SAE ARP 4761A

- Defines a system of safety process that enables identification and management of hazards and their associated risks during system development and sustaining engine
- Planned ACVIP modeling and analysis activities should still align with program safety processes in order to reduce project risk and rework due to problems found during certification

Analysis Techniques Summary

The AMACC and the ACVIP M&A Handbook Identify these analysis:

– **Functional Hazard Assessment (FHA)** to identify hazards and set criticalities and levels of rigor
– **System Theoretic Process Analysis (STPA)** to define control loops, identify unsafe control actions (UCAs) and identify mitigations/constraints to those UCAs
– **Failure Modes and Effects Analysis (FMEA)** to specify error-handling capabilities that are required to mitigate risks identified by hazard assessment
– **Fault Tree Analysis (FTA)** will detect components whose failures are not independent
– **Reliability Block Diagram (RBD) Analysis** determines reliability for a capability based on the reliabilities of the other capabilities that it depends on and information about redundancy among those other capabilities.
– **Markov Analysis** applied to systems that have degraded modes of operation, suffer transient errors, or can reconfigure and recover

ACVIP Guidance for Safety and Project Milestones

System Requirements Review (SRR)

- Architecture Artifacts: Preliminary identification of all hw & sw components is completed
- Safety: Hazards have been reviewed and mitigating courses of action have been allocated
- Analysis methods: Aircraft, System FHA, STPA - Hazards associated with model components

Preliminary Design Review (PDR)

- Architecture Artifacts: Preliminary identification of all hw & sw components is completed, detail added
- Safety: Hazards have been reviewed and mitigating courses of action have been allocated
- Analysis methods: Aircraft,System FHA, STPA , FMEA, FTA, STPA - Hazards associated with model components

Critical Design Review (CDR)

- Architecture: Detailed design (hw, sw), including interface descriptions are complete and satisfy all requirements in the system functional baseline – Failure types associated with components
- Safety: Risk items/Criticality for hardware, software identified, mitigation approaches described
- Analysis: Failure Mode, Effects, and Criticality Analysis (FMECA) is complete
- At this stage FHA, STPA charts, FMEA (FMECA), FTA have been produced, refined

Test Readiness Review (TRR)

- At this stage FHA, STPA charts, FMEA (FMECA), FTA and models have been updated since CDR
- Verification of the safety related requirements is conducted and the models can be validated and updated as needed

Targeted Analysis for Different Stages of Model Maturity

Source: ACVIP Modeling & Analysis Handbook

Targeted Analysis Examples	Stages of Model Maturity: Black Box (environment, flows)	Refine to Functional (subcomponents, connections)	Refine to Software (processes, threads, messages)	Add Hardware (processors, buses, memory)
Static Consistency	Interface	Interface	Interface	Interface
Behavior Consistency	Interface	Interface	Component	Component
Resource Loading	Power, Mass	+ Utilization	X	+ Schedulability
Latency	X	X	X	X
Safety (FHA, FMEA, FTA, RBD, Markov, STPA)	FHA, STPA	+ FMEA, FTA, STPA, RBD, Markov	+ FMEA, FTA, STPA, RBD, Markov	+ FMEA, FTA, STPA, RBD, Markov
Cybersecurity (MILS, RMF, Attack Trees)	X	X	+ RMF Mixed Criticality	+ RMF Step 4
Model Checking (AGREE, Resolute)	X	X	X	X
Custom Analyses*	X	X	X	X

X = apply here

*** Integrate with User Properties or as plug-ins**

Continuous, virtual integration

Defects **Defects** **Defects** **Defects**

Mapping AMACC Methods of Compliance with ACVIP Methods -1

AMACC System Safety Elements	Applicable Standards	Method of Compliance	ACVIP Identified Methods	Supported with AADL Models & EMV2
14.2.1 System Safety Program Plan (SSPP)	MIL-STD-882E	MIL-STD-882E DO-178C DO-254	ACVIP document identifies processes and methods that map to SSPP	ACVIP Modeling & Analysis Handbook outlines applicable analysis and tools
14.2.2 Preliminary Hazard Analysis (PHA)	MIL-STD-882E	ARP 4761 FHA	Model component annotations with hazards, FHA report, STPA (control, causal scenarios)	x
14.2.3 Functional Hazard Assessment (FHA)	ARP 4761	ARP 4761 FHA Updated PHA	Model component annotations with hazards, FHA report, STPA (control, causal scenarios)	x
14.2.4 Aircraft Functional Hazard Assessment	ARP 4761	ARP 4761 FHA Updated FHA to include aircraft functions	Model component annotations with hazards, FHA report, STPA	x
14.2.5 System-Level Functional Hazard Assessment	ARP 4761	ARP 4761 FHA Updated FHA Allocate aircraft-level functions to systems	Model component annotations with hazards, FHA report, STPA	x
14.2.6 Preliminary Aircraft / System Safety Assessment (PASA/PSSA)	ARP 4761	PASA/PSSA IAW SAE ARP 4761	Model component annotations with hazards, FHA report, aircraft FTA STPA	x

X – supported in modeling language and automated analysis

Mapping AMACC Methods of Compliance with ACVIP Methods -2

AMACC System Safety Elements	Applicable Standards	Method of Compliance	ACVIP Identified Methods	Supported with AADL Models & EMV2
14.2.7 Common Cause Analysis (CCA)	ARP 4761	CCA IAW SAE ARP 4761, FTA	FTA, Minimum Cut Sets, CCA reports	x
14.2.8 Fault Tree Analysis (FTA)	SAE ARP 4761	Qualitative FTAIAW SAE ARP 4761	FTA	x
14.2.9 System Safety Assessment (SSA)	ARP 4761, Paragraph 3.4 ARP 4761, Appendix C	SSA IAW SAE ARP 4761 – Specify verification methods implementation meets design	Contributory-models reference verification artifacts and methods	x
14.2.10 Failure Mode, Effects, and Criticality Analysis	ARP 5580	FMEA IAW SAE ARP 5580	FMEA, to be augmented with criticality	x
14.2.11 Safety Assessment Report	MIL-STD-882E, Task 301	MIL-STD-882E, Task 301	Contributory – include model analysis reports	x
14.2.12 System Safety Hazard Analysis	MIL-STD-882E, Task 205	MIL-STD-882E, Task 205	Contributory – include model analysis reports	x

X – supported in modeling language and automated analysis

Summary

Next Steps

- Identify a larger set of AMACC requirements that would benefit from ACVIP
- Define levels of model maturity that align with program reviews
- Adapt Verification by Test
  - Generate test cases based on models
  - Verify model properties at runtime
- Train airworthiness authorities to evaluate ACVIP analysis results

Recommendation

We recommend that PMs require ACVIP for VbA evidence for embedded systems.

For More Information

Contact:

John J. Hudak, PE
Principal Engineer

Software Engineering Institute
4500 Fifth Avenue
Pittsburgh, PA 15213-2612

Phone: 412/268.5219 |

Web: www.sei.cmu.edu

Email: jhudak@sei.cmu.edu

References

- [ACVIP Acquisition 2020] Software Engineering Institute. Architecture-Centric Virtual Integration Process (ACVIP) Acquisition Management Handbook. CMU/SEI-2020-SR-021-Restricted. August 2020.
- [ACVIP M&A 2021] Adventium Labs. Architecture-Centric Virtual Integration Process (ACVIP) Handbooks – Modeling & Analysis with the Architecture Analysis & Design Language (AADL). March 2021. https://www.adventiumlabs.com/sites/default/files/documents/ACVIP-Modeling-%26-Analysis-Handbook_Mar2021_DistA.pdf
- [ACVIP Overview 2019] Software Engineering Institute. Architecture-Centric Virtual Integration Process (ACVIP) Handbooks – Overview with the Architecture Analysis & Design Language (AADL). April 2019.
- [Adventium 2021a] Adventium Labs. ACVIP Training. 2021. https://www.adventiumlabs.com/acvip-training
- [Adventium 2021b] Adventium Labs. Airworthiness Qualification of ACVIP Tools." August 2021. IN DRAFT.
- [AMACC 2021] U.S. Army Combat Capabilities Development Command Aviation & Missile Center (DEVCOM AvMC). Army Military Airworthiness Certification Criteria (AMACC) Revision A Change 2 (C2). April 9, 2021. https://www.avmc.army.mil/Directorates/SRD/TechDataMgmt/
- [Army 2016] Department of the Army. Airworthiness of Aircraft Systems. Army Regulation 70–62. May 2016. https://armypubs.army.mil/ProductMaps/PubForm/Details.aspx?PUB_ID=1000692
- [Boydston 2019] Boydston, Alex K.; Feiler, Peter H.; Vestal, Steve; & Lewis, Bruce. Architecture Centric Virtual Integration Process (ACVIP): A Key Component for the DoD Digital Engineering Strategy. 22nd Annual Systems and Mission Engineering Conference. September 2019. https://www.adventiumlabs.com/publication/architecture-centric-virtual-integration-process-acvip-key-component-dod-digital
- [Delange 2014] Delange, Julien; Feiler, Peter; Gluch, David; & Hudak, John. AADL Fault Modeling and Analysis Within an ARP4761 Safety Assessment. CMU/SEI-2014-TR-020 . Software Engineering Institute, Carnegie Mellon University. 2014. http://resources.sei.cmu.edu/library/asset-view.cfm?AssetID=311884
- [Feiler 2015] Feiler, Peter H. & Hudak, John J. Potential System Integration Issues in the Joint Multi-Role (JMR) Joint Common Architecture (JCA) Demonstration System. CMU/SEI-2015-SR-030. Software Engineering Institute, Carnegie Mellon University. December, 2015. https://resources.sei.cmu.edu/library/asset-view.cfm?assetid=447176
- [Feiler 2015a] Feiler, Peter; Weinstock, Charles; Goodenough, John; Delange, Julien; Klein, Ari; & Ernst, Neil. Improving Quality Using Architecture Fault Analysis with Confidence Arguments . CMU/SEI-2015-TR-006. Software Engineering Institute, Carnegie Mellon University. 2015. http://resources.sei.cmu.edu/library/asset-view.cfm?AssetID=435051
- [Feiler 2016a] Feiler, Peter; Hudak, John; Delange, Julien; & Gluch, David. Architecture Fault Modeling and Analysis with the Error Model Annex, Version 2. CMU/SEI-2016-TR-009. Software Engineering Institute, Carnegie Mellon University. 2016. http://resources.sei.cmu.edu/library/asset-view.cfm?AssetID=464380
- [Feiler 2016b] Feiler, Peter H. & Delange, Julien. Automated Fault Tree Analysis from AADL Models. ACM High Integrity Language Technology International Workshop on Model-Based Development and Contract-Based Programming (HILT). Pittsburgh, PA. 6-7 October 2016. https://resources.sei.cmu.edu/library/asset-view.cfm?assetid=499346
- [Hansson 2018] Hannsson, Feiler and Helton. ROI Analysis of the System Architecture Virtual Integration Initiative. SEI Technical Report CMU/SEI-2018-TR-002, 2018. https://resources.sei.cmu.edu/library/asset-view.cfm?assetid=517157
- [SAE AADL 2017] Society of Automotive Engineers. Architectural Analysis and Design Language. SAE AS 5506C-2017. 2017.
- [SEI 2021] Software Engineering Institute. Modeling System Architectures Using the Architecture Analysis and Design Language (AADL) – eLearning. 2021. https://www.sei.cmu.edu/education-outreach/courses/course.cfm?courseCode=V40

End of Presentation

Backup Slides