Integrated Safety and Security Engineering for Mission-Critical Systems

Dr. Sam Procter
Copyright 2019 Carnegie Mellon University.

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.

The view, opinions, and/or findings contained in this material are those of the author(s) and should not be construed as an official Government position, policy, or decision, unless designated by other documentation.

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.

[DISTRIBUTION STATEMENT A] This material has been approved for public release and unlimited distribution. Please see Copyright notice for non-US Government use and distribution.

This material 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.


DM19-1076
Making Critical Systems Safer and More Secure

- Modern embedded systems – such as those found in the CH47F Chinook, TARDEC Autonomous Truck, and Little Bird – need to be both safe and secure, but too often, a system’s safety is designed and assessed separately from its security.

- The pace and scale of these systems’ development are such that traditional analysis cannot keep up. We’re developing software and processes that use a system’s architecture to support developer intuition and improve safety and security.

- But AADL – the internationally standardized Architecture Analysis and Design Language – is for more than research: Alex Boydston will talk about how the U.S. Army is using prior research in model-based engineering to build systems that are safer and less expensive.
Integrated Safety and Security Engineering

AADL Overview
Like a lot of models that engineers draw every day on their whiteboards, AADL consists of boxes and lines.

The difference between AADL and a whiteboard is that AADL has precise semantics.

This box represents a computer process – a protected region of memory and a space where we can allocate individual threads.
AADL Overview

Those threads are also boxes – but they have very precise meanings.
We can connect the threads together using lines to represent different types of intra-process communication.

We add more semantics via properties – they are useful for both system analyses and to guide code generation.

This box shows a periodic thread – it is dispatched regularly according to some clock.

And this thread is sporadic – it is dispatched whenever a message arrives at a specified port.
Integrated Safety and Security Engineering

Transitioning Research to Practice
Research into Practice

- This project consists of a handful of tasks. Some are more theoretical and some more mature.
- All of the tasks, though, are implemented using AADL: a language already used by practitioners.
- This lets us rapidly move ideas from research – conducted here at the SEI, in academia, or in industry – to practice.
Hazard Analysis: Re-tooled for Modern System Development

- Hazard analysis – a way of assessing a system’s safety – traditionally results in a large report.
- What if that report could be split into its constituent parts and – generated dynamically based on the system architecture? – queried interactively by an auditor?

We exploit state-of-the-art data-dependence analysis (developed by colleagues at Kansas State University) to power the report.
Slang and HAMR: Verification Integrated with Code Generation

We’re working with Kansas State University on two related technologies that translate a system architecture (in AADL):

- **Slang** – an analyzable intermediate representation, and then
- **C / C++** – HAMR produces low-level source code targeted at a given platform

Derived from a model built by John Hatcliff, Kansas State University.
Safety and Security Design Patterns

Operationalize existing patterns:
- Stated in unambiguous AADL
- Machine checkable (via ALISA)

Outputs:
- A tool-supported library of patterns
- Moving through AADL standardization process

Global composition (tool: AWAS)
Contract on interface (tool: RESOLUTE)
Faulty components (AADL/EMV2)
Safeguard (AADL/BA)
Integrated Safety and Security Engineering

DoD Impact
U.S. ARMY COMBAT CAPABILITIES DEVELOPMENT COMMAND – AVIATION & MISSILE CENTER

Architecture Centric Virtual Integration on Joint Multi-Role (JMR) Mission Systems Architecture Demonstration (MSAD)

Alex Boydston, MSEE
JMR MSAD / FARA Project Engineer
CCDC AvMC

1 Oct 2019
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 Systems 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

We need tools that help do the job, not become the job!
• In 1983, the Army planned to buy 5,023 vehicles at $12.1 million/copy.
• Test schedule delays and increasing development costs scaled down the planned buy to 650 aircraft at $58.9 million/copy.
• Most testing involved integration of the complete Mission Equipment Package, which incorporated a radar, infrared, and image-intensified television sensors for night flying and target acquisition.
• Technical challenges remained in software development, integration of mission equipment, radar and infrared signatures, and radar perf.
• The first flight had been originally planned to take place during August 1995, but was delayed by a number of structural and software problems that had been encountered.
• Key program elements, including development and integration of certain software capabilities, failed to foster confidence with Army overseers; several capabilities were viewed as having been unproven and risky.
• The anticipated consumption of up to 40% of the aviation budget by the Comanche alone for a number of years was considered to be extreme.

References:
• https://en.wikipedia.org/wiki/Boeing%E2%80%93Sikorsky_RAH-66_Comanche#cite_note-26
• https://en.wikipedia.org/wiki/Boeing%E2%80%93Sikorsky_RAH-66_Comanche#cite_note-Eden_p139-9)
• Origin (2009): Aerospace Vehicle Systems Institute’s System Architecture Virtual Integration (SAVI) concept for incremental virtual integration using AADL.

• First step, embedded systems architecture modeling in AADL, a language for precisely specifying key components and properties of embedded systems.

• Virtual integration process uses AADL-enabled analyses of real-time safety- and security-critical computing systems to identify issues early before integration.

• Automated continuous virtual integration enables architecture-based incremental and compositional modeling & analysis as system evolves.

• Provides increasing assurance confidence; complements testing.

• Provides a “Single Authoritative Source of Truth.”

• Enabler of MOSA to provide a standard analyzable and processable architecture description for embedded systems.

Virtual Integration of Software, Hardware, and System supporting verification, airworthiness, safety and cybersecurity certification

“Model, Integrate, Analyze, then Build”
1ST JMR MSAD DEMONSTRATION: JCA DEMO LESSONS LEARNED

- Use of AADL for virtual integration and analysis identified >85 issues

- ACVIP analyses identified errors prior to system integration (as early as during the kickoff meetings with DCFM suppliers)

- Translators for FACE->AADL and SysML->AADL would have been beneficial to automate and reduce human error

- ACVIP training proved beneficial
  - Boeing used AADL to extend their demo for timing and control stability analysis and found issues

Architecture analysis is critical for the successful and affordable integration of systems
**Finding Problems Early Using AADL (CCDC/SEI)**

- **Summary:** 6 Week Virtual Integration of HUMS on CH47F using AADL
- **Result:** Identified 20 major integration issues early
- **Benefit:** Avoided 12-month delay on 24-month program

**Discovering Performance Issues Early Using AADL (UH-60V)**

- **Summary:** Applied AADL analysis on UH60V per AED requirement
- **Result:** Predicted multicore loading issues
- **Benefit:** Provided early performance insight and risk reduction opportunity

**Improving System Security (DARPA / AFRL)**

- **AADL applied to Unmanned Aerial Vehicles & Autonomous Truck using formal methods analysis and trusted system generation**
- **Result:** AADL models enforced security policies and were used to auto build the trusted system
- **Benefit:** Combined with formal methods verification, prevented security intrusion by a red team

**Transforming procurement supporting MBE and ACVIP (JMR MSAD)**

- **Summary:** Increasingly complex industry/DoD mission system architecture demonstrations using Model Based Engineering
- **Result:** Pre-integration fault identification, maturation of tools and processes
- **Benefit:** ~3x increase to requirements, design and analysis activities, resulted in ~10x reduction on test and integration activities

**Increased Cybersecurity**

- Makes complex capabilities possible through Agile analytic and virtual integration of real-time safety and security critical cyber physical embedded systems
Web Site
https://www.avmc.army.mil/

Facebook
www.facebook.com/ccdc.avm

Instagram
www.instagram.com/CCDC_AVM

Twitter
@CCDC_AVM

Public Affairs
usarmy.redstone.ccdc-avmc.mbx.pao@mail.mil
Integrated Safety and Security Engineering

Looking Ahead
# Looking Ahead

<table>
<thead>
<tr>
<th>NEAR</th>
<th>MID</th>
<th>FAR</th>
</tr>
</thead>
<tbody>
<tr>
<td>What other assumptions underlying various emerging technologies (e.g., ML / AI, DevOps, formal verification of behavior) would be beneficial in architectural models?</td>
<td>How can models be used at runtime?</td>
<td>To what extent can we use ML / AI to help develop models, rather than the other way around?</td>
</tr>
<tr>
<td></td>
<td>What data do we need to more effectively let systems autonomously use models of themselves?</td>
<td></td>
</tr>
</tbody>
</table>

We are also looking for sponsors to try out our tools, or just tell us their challenges with critical and embedded system development – please reach out!