Design and Analysis of Cyber-Physical Systems: AADL and Avionics Systems

Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213

Peter H. Feiler
May 1, 2013
Outline

Software-induced Challenges in Cyber-Physical Systems
SAE AADL: an Architecture Modeling and Analysis Framework
Virtual Integration of an Avionics System
Architectural Fault Modeling of Safety-critical Systems
Conclusions
We Rely on Software for Safe Aircraft Operation

Even with the autopilot off, flight control computers still "command control surfaces to protect the aircraft from unsafe conditions such as a stall," the investigators said.

The unit continued to send false stall and speed warnings to the aircraft's primary computer and about 2 minutes after the initial fault "generated very high, random and incorrect values for the aircraft's angle of attack."

"This appears to be a unique event," the bureau said, adding that fitted with the same air-data computer. The advisory is "aimed at minimizing the risk in the unlikely event of a similar occurrence."

Embedded software systems introduce a new class of problems not addressed by traditional system modeling & analysis.

Software Engineering Institute | Carnegie Mellon University

Autopilot Off

A "preliminary analysis" of the Qantas plunge showed the error occurred in one of the jet's three air data inertial reference units, which caused the autopilot to disconnect, the ATSB said in a statement on its Web site.

The crew flew the aircraft manually to the end of the flight, except for a period of a few seconds, the bureau said.

Even with the autopilot off, flight control computers still "command control surfaces to protect the aircraft from unsafe conditions such as a stall," the investigators said.

The unit continued to send false stall and speed warnings to the aircraft's primary computer and about 2 minutes after the initial fault "generated very high, random and incorrect values for the aircraft's angle of attack."

The flight control computer then commanded a "nose-down aircraft movement, which resulted in the aircraft pitching down to a maximum of about 8.5 degrees," it said.

"Airbus has advised that it is not aware of any similar event over the many years of operation of the Airbus," the bureau added, saying it will continue investigating.
High Fault Leakage Drives Major Increase in Rework Cost

Aircraft industry has reached limits of affordability due to exponential growth in SW size and complexity.

70% Requirements & system interaction errors

80% late error discovery at high rework cost

Major cost savings through rework avoidance by early discovery and correction
A $10k architecture phase correction saves $3M

Rework and certification is 70% of SW cost, and SW is 70% of system cost.

Sources:

Resilience to SW Induced Faults
We cannot assume zero defect software
Mismatched Assumptions in Embedded SW

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

SysML does not address Embedded Software System Architecture Issues
Outline

Software-induced Challenges in Cyber-Physical Systems
SAE AADL: an Architecture Modeling and Analysis Framework
Virtual Integration of an Avionics System
Architectural Fault Modeling of Safety-critical Systems
Conclusions
SAE Architecture Analysis & Design Language (AADL) Standard for Software-reliant Systems

The System
Physical platform
Aircraft

Control Guidance
Embedded Application
Software
Flight control & Mission

The Software
Deployed on
Utilizes

Computer System
Hardware & OS

The Computer System

AADL focuses on interaction between the three elements of a software-intensive system based on architectural abstractions of each.
System Level Fault Root Causes

Violation of data stream assumptions
• Stream miss rates, Mismatched data representation, Latency jitter & age

Partitions as Isolation Regions
• Space, time, and bandwidth partitioning
• Isolation not guaranteed due to undocumented resource sharing
• Fault containment, security levels, safety levels, distribution

Virtualization of time & resources
• Logical vs. physical redundancy
• Time stamping of data & asynchronous systems

Inconsistent System States & Interactions
• Modal systems with modal components
• Concurrency & redundancy management
• Application level interaction protocols

Performance impedance mismatches
• Processor, memory & network resources
• Compositional & replacement performance mismatches
• Unmanaged computer system resources

End-to-end latency analysis
Port connection consistency

Process and virtual processor to model partitioned architectures

Virtual processors & buses
Multiple time domains

Operational and failure modes
Interaction behavior specification
Dynamic reconfiguration
Fault detection, isolation, recovery

Resource allocation & deployment configurations
Resource budget analysis & scheduling analysis

Codified in Virtual Upgrade Validation method
Architecture-Centric Modeling Approach

Single-source Annotated Architecture Model Addresses Change Impact Across Analytical Models and Non-Functional Properties

Safety & Reliability
- MTBF
- FMEA
- Hazard analysis

Security
- Intrusion
- Integrity
- Confidentiality

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

Resource Consumption
- Bandwidth
- CPU time
- Power consumption

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

Data Quality:

- Data precision/accuracy
- Temporal correctness
- Confidence

Resource Consumption:

- Bandwidth
- CPU time
- Power consumption

Analysis Architecture Fault Models
Feiler, Jan 8, 2013
© 2013 Carnegie Mellon University
Outline

Software-induced Challenges in Cyber-Physical Systems
SAE AADL: an Architecture Modeling and Analysis Framework
Virtual Integration of an Avionics System
Architectural Fault Modeling of Safety-critical Systems
Conclusions
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

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

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

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

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
End-to-end Latency in Control Systems

Operational Environment

System Engineer

System Under Control

Control Engineer

Control System

- Processing latency
- Sampling latency
- Physical signal latency

Impact of Software Implemented Tasks

Jitter affects stability of control behavior (subtle value error)

AADL immediate & delayed connections specify deterministic sampling

Impact of Scheduler Choice on Controller Stability

A. Cervin, Lund U., CCACSD 2006
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
Outline

Software-induced Challenges in Cyber-Physical Systems
SAE AADL: an Architecture Modeling and Analysis Framework
Virtual Integration of an Avionics System
Architectural Fault Modeling of Safety-critical Systems
Conclusions
Error Model and the Architecture

Propagation of errors of different types from error sources along propagation paths between architecture components.

Error flows as abstractions of propagation through components.

Component error behavior as transitions, out propagations, and detection based on event, state and incoming propagation conditions.

Composite error behavior in terms of component error behavior states in support of compositional abstraction.
Discovery of Unexpected PSSA Hazard through Virtual Integration

**EGI Logic**

- **Oper'l**
- **Failed**
- **Corrupted**

**EGI HW**

- **Oper'l**
- **Failed**

**CorruptedData**

Unexpected propagation of corrupted Airspeed data results in Stall due to miss-correction

**Oper'l Failed**

**Oper'l Failed**

**NoData**

Anticipated: No EGI data

**FMS Processor Operational**

**FMS Power**

**Auto Pilot Operational**

**NoService**

**Stall**

**Anticipated: No Service**

**Actuator Cmd**

**FMS Power**

**Auto Pilot**

**Oper'l Failed**

**Oper'l Failed**

**Corrupted Data**

**NoService**

**Stall**

**Anticipated: No Stall Propagation**

**Transient hardware failure corrupts EGI data**

**EGI maintainer adds corrupted data hazard to model. Error Model analysis detects unhandled propagation.**
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

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
Impact of Deployment Configuration Changes on Availability

FMS Failure on 2 or 3 processor configuration (CPU failure rate = $10^{-5}$)

<table>
<thead>
<tr>
<th>FMS Failure Rate</th>
<th>0</th>
<th>$5 \times 10^{-6}$</th>
<th>$5 \times 10^{-5}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>MTTF – One CPU operational</td>
<td>112,000</td>
<td>67,000</td>
<td>14,000</td>
</tr>
<tr>
<td>MTTF – Two CPU operational</td>
<td>48,000</td>
<td>31,000</td>
<td>7,000</td>
</tr>
</tbody>
</table>

Side effects of design and deployment decisions on availability predictions

Workload balancing of partitions later in development affects reliability

3 processor configuration can be less reliable than 2 processor configuration

Example: Replicated AP and FG channel (re)distributed across two processors
Outline

Software-induced Challenges in Cyber-Physical Systems
SAE AADL: an Architecture Modeling and Analysis Framework
Virtual Integration of an Avionics System
Architectural Fault Modeling of Safety-critical Systems
Conclusions
Virtual Upgrade Validation Method

Early discovery of technical risks

Capture Embedded Software System Architecture
Software runtime, computer hardware, mechanical system architecture in same SAE AADL model

Utilize knowledge of potential mismatched assumptions
Codified in Virtual Upgrade Validation method

Analyze multiple operational quality attributes
Utilize SAE AADL single model truth approach and well defined semantics

Sample Findings
Potential inconsistency and lack of integrity of recorded aircraft health data
Ambiguous task and communication architecture has priority inversion potential under fault conditions
Corrupted airspeed data not considered as hazard
Increased Confidence through End-to-end Virtual Integration and Testing Evidence

- Requirements Engineering
  - Architecture Focused Requirements Analysis
  - System & SW Architectural Design
  - Component Software Design
- System & SW Architecture Validation
- Design Validation
- Integration Build
- Target Build
- Deployment Build
- Acceptance Test
- Flight Test
- System Test
- System Integration Lab Testing
- Integration Test
- Code Coverage Testing
- Unit Test
- Build the System
- Build the Assurance Case

- Architecture Modeling and Analysis
- Virtual Architecture Integration & Analysis
- Design Validation by Virtual Integration
- Build the System

Increased Confidence through End-to-end Virtual Integration and Testing Evidence
Resources

Website www.aadl.info

Public Wiki https://wiki.sei.cmu.edu/aadl

AADL Book in SEI Series of Addison-Wesley