Assurance of Conventional and Machine Learning Systems

Alwyn Goodloe

NASA Langley Research Center

So You Want to Build an Airplane

- Form a startup and start hacking - just like Silicon Valley, right?
  - Not so fast!
- Process starts off with a notification of intent to the FAA
  - A minuet begins between the company and the regulators
  - For a Part 25 aircraft they will tell you over 1500 safety criteria you must meet
    - Autos and medical devices are easy in comparison
    - DoD aircraft not subject to these regulations
- The FAA must certify the aircraft
  - Designated Engineering Representative (DER)
- The cyber-physical component is one of the largest risk factors
- You can choose to do things your own way and make an argument to the FAA that the aircraft is safe OR you can follow approved guidelines
  - Very process oriented
  - Overarching properties will be another path to assurance in the future

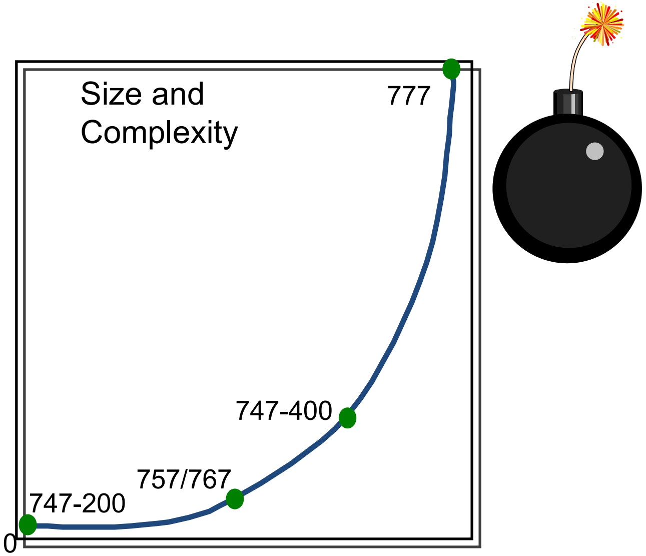
Ultra-Reliability is Hard

We are very good at building complex software systems that work 95% of the time---but not as good at building complex software systems that are ultra-reliably safe.

What has saved us in the past?

- Minimal amount of software that is safety-critical
- Simple designs with predictable behavior
- Enormously expensive verification and certification processes
- Backups that are not software, e.g.
  - Hardware interlocks
  - Human intervention

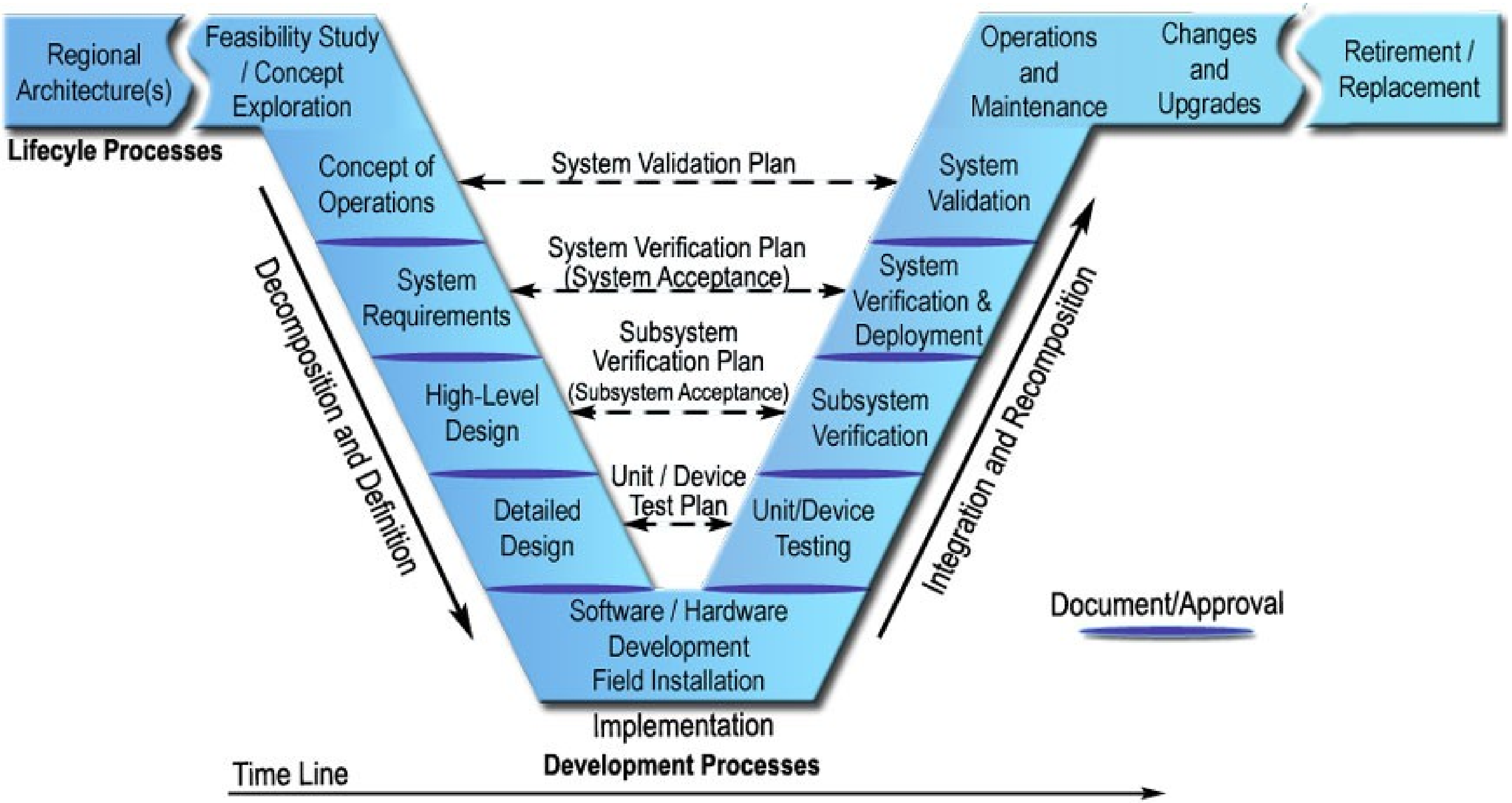
All sectors of aerospace are increasingly relying on software to perform safety-critical functions



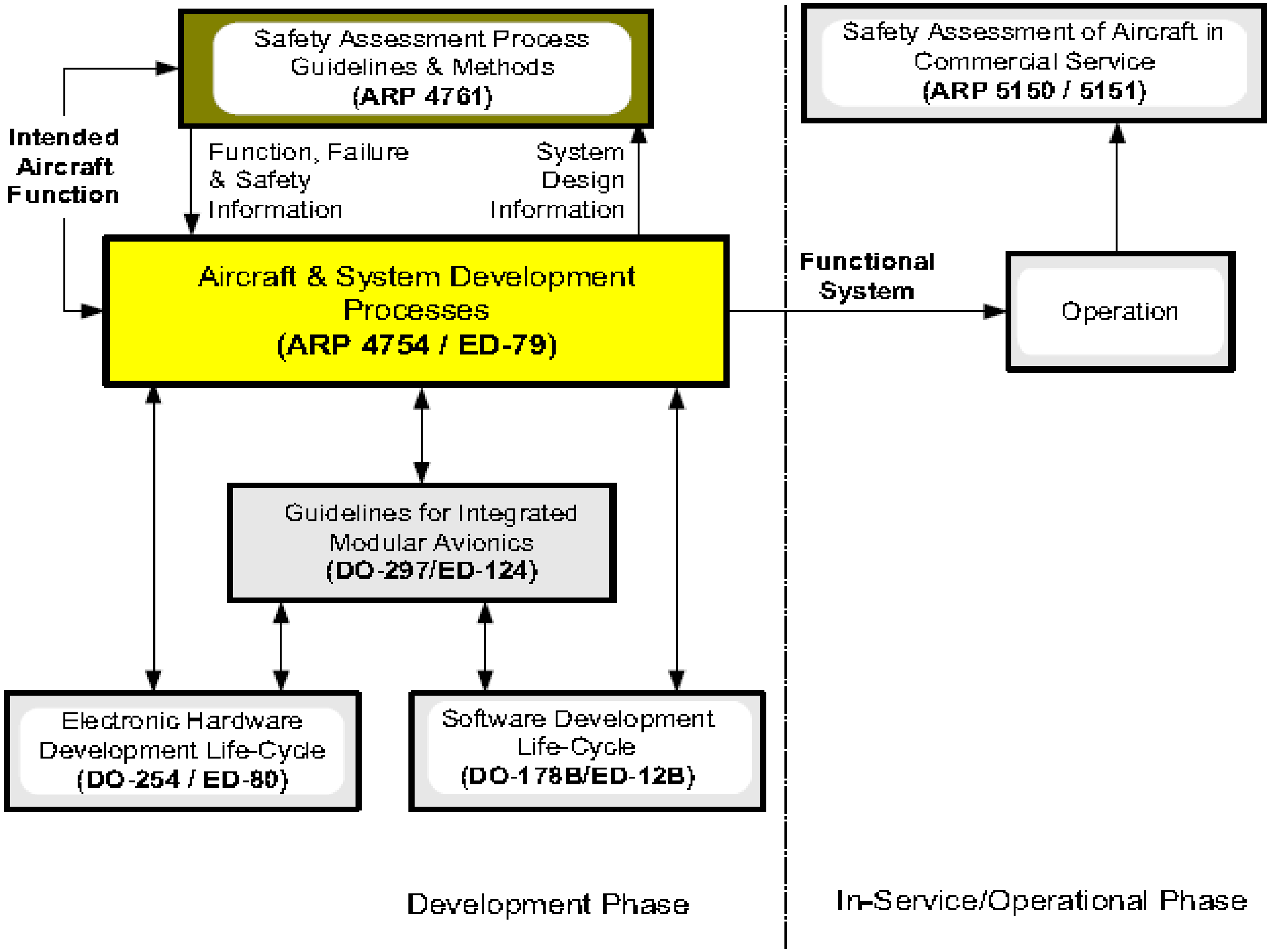
Eliminating Common Mode Errors

- Independence – A concept to minimize the likelihood of common mode and cascade errors
- Diversity
  - Hardware and software
- Redundancy
  - Triple redundancy
  - Com/Mon
- Can mix techniques
  - Dissimilar com/mon
- When all these things fail, well-trained pilots do an amazing job to save the day

Traditional Systems Engineering Process



Guideline Documents



Safety Assessment Process Guidelines & Methods **(ARP 4761)**

Intended Aircraft Function

Function, Failure & Safety Information

System Design Information

Aircraft & System Development Processes **(ARP 4754 / ED-79)**

Guidelines for Integrated Modular Avionics **(DO-297/ED-124)**

Electronic Hardware Development Life-Cycle **(DO-254 / ED-80)**

Software Development Life-Cycle **(DO-178B/ED-12B)**

Functional System

Operation

Safety Assessment of Aircraft in Commercial Service **(ARP 5150 / 5151)**

Development Phase

In-Service/Operational Phase

Central Role of Requirements

- Emphasis on getting the requirements correct
  - Requirements get refined into specifications
- Many analysis techniques are applied to validate the requirements
- Verification focuses on assuring that the system behaves as the specification indicates and does not exhibit unintended behavior
- Implementations need to show traceability to the requirements

ARP 4761

- Aerospace Recommended Practice for performing safety assessments on civil aircraft
- Guidelines and methods of performing the safety assessment
- Functional Hazard Analysis (FHA)
- Preliminary System Safety Assessment (PSSA)
- System Safety Assessment (SSA)

ARP 4761 Contd.

- Safety assessment process
- Safety assessment overview
- Detailed method guidelines
  - Functional Hazard Assessment (FHA)
  - Fault-Tree Analysis (FTA)
  - Failure Modes and Effects Analysis (FMEA)
  - Common Mode Analysis (CMA)
  - Zonal Safety Analysis (ZSA)

Functional Hazard Analysis

- Identifies and classifies the failure conditions associated with the aircraft functions and combinations of aircraft functions
    - Classification Levels: Minor (D), Major (C), Hazardous (B), Catastrophic (A)
    - Classifications establish safety objectives
    - Output starting point for generation and allocation of safety requirements

Preliminary System Safety Assessment

- Systematic examination of proposed system architecture
  - Used to complete the failure conditions list and the safety requirements
- Identify how failures lead to the hazards identified in FHA
  - Suggested analysis techniques such as FTA
- How FHA requirements can be met
  - Identify protective strategies
    - Partitioning, dissimilarity, etc.

System Safety Assessment

- Systematic examination of system, architecture, and installation to show compliance with safety requirements
- A SSA done for each PSSA
- Verification that the design requirements established at system level
- Verification that safety requirements derived from requirements are met
- Verification design requirements in CCA met
- Linkage system level SSA to aircraft level FHA

4754A

- Input is the function, failure, and safety info from 4761
- Iterative process as design is refined and the analysis process prescribed by 4761 is repeated
- Functional Design Assurance Levels (FDAL) assigned
- FDAL assigned to systems from aircraft architecture based on Preliminary Aircraft Safety Assessment (PASA)
- Item Design Assurance Level (IDAL) done in refinement

4754A Contd.

- FDAL considers functional independence of aircraft/system functions
- IDAL considers design independence of items
- Assertion of independence must be substantiated
  - Verify no common mode introduced
- IDALs are assigned to items then fed back to analysis
- During allocation of top-level function into two or more independent sub-functions
  - One sub-function cannot itself cause top-level hazard

Independence Can Be Your Friend

- Architectural strategies incorporating independence, redundancy, and dissimilarity can be a powerful means of reducing the potential for errors in requirements or in design implementation
- The people writing the standards have built these architectures
  - They do it for Boeing, Airbus, etc.
- The justifications and arguments for safety are found in certification documents
- Engineers and certification bodies lack guidelines and examples

Yet ...

- The effectiveness of particular architectural strategies, introduced to allow the allocation of lower item risk level, generally cannot be quantified
  - John Downer’s work redundancy in engineering
- As a consequence, the justification to support such allocation necessarily involves some degree of engineering judgment by the applicant and the certification authorities
- Do existing architectural patterns and arguments work on newer more complex systems

ML Assurance Problem

- We do not know how to assure machine learning (ML) enabled systems within the framework of existing methodologies used for safety-critical systems
  - Reliability, predictability, robustness to faults and failures
- The AI community have not been interested in this problem as performance is the main concern
  - ML systems “fail regularly” while the ultra safety-critical systems (aircraft, nuclear power, etc.) ideally never fail in their operating life

Machine Learning Use Cases

- Two classes of use cases for machine learning (ML)
- Conventional approaches work, but ML is cheaper, more optimal, etc.
  - Can derive specifications for what it is supposed to do or not supposed to do
- We have no idea how to build the system using conventional approaches
  - On only specification is a large high-dimensional data set
  - Runtime assurance not effective as no actionable spec is available

Current Approaches

- Many research efforts underway to verify machine learning enabled systems
- Many efforts focus on showing robustness against adversarial attacks
- Some known known approaches such as:
  - Reluplex -- SMT based approach
  - ERAN, GPUPoly, DeepPoly – Abstract Interpretation
- None of these efforts really help resolve the major challenge of assuring ML enabled systems where the spec is really a large high-dimensional data set

Questions?

Contact Information:
Alwyn E. Goodloe
+1-757-864-5064
a.goodloe@nasa.gov