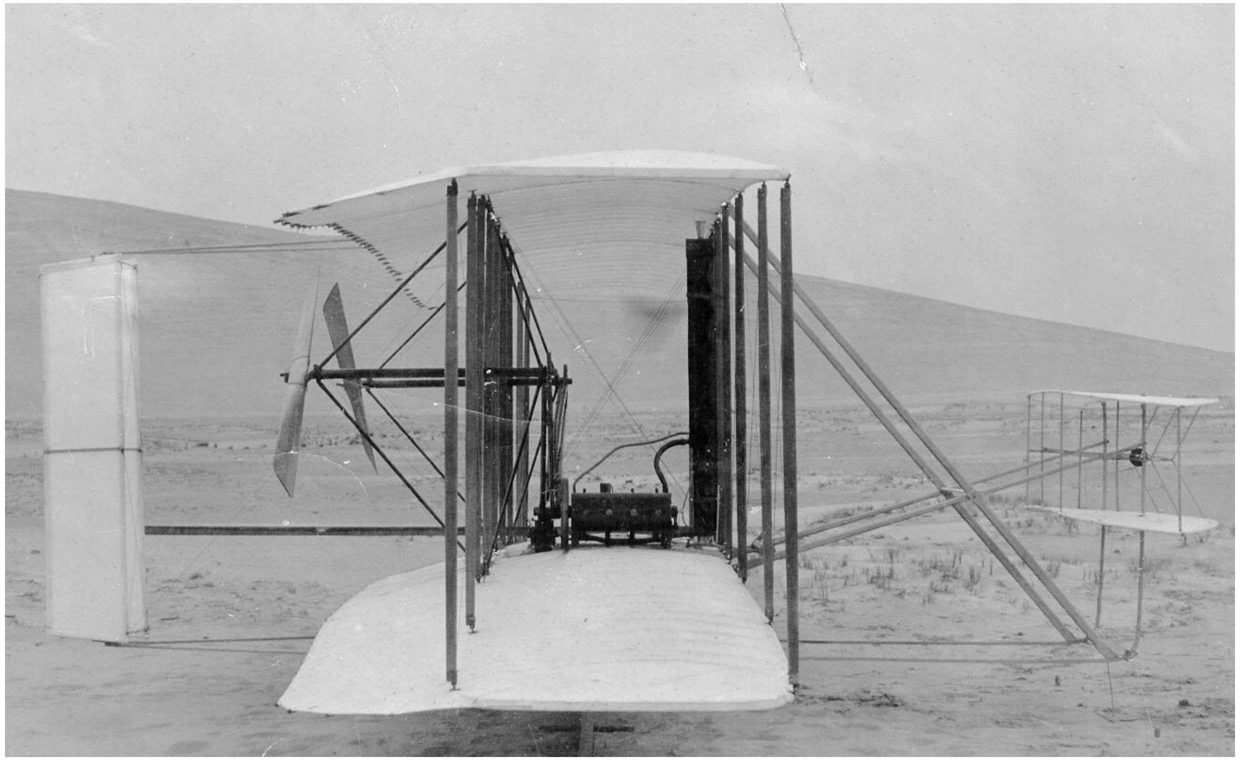
Safety expertise matters more than you might think

Dr. Mallory Suzanne Graydon
NASA Langley Research Center
Hampton, VA, USA

First, some aviation safety history …

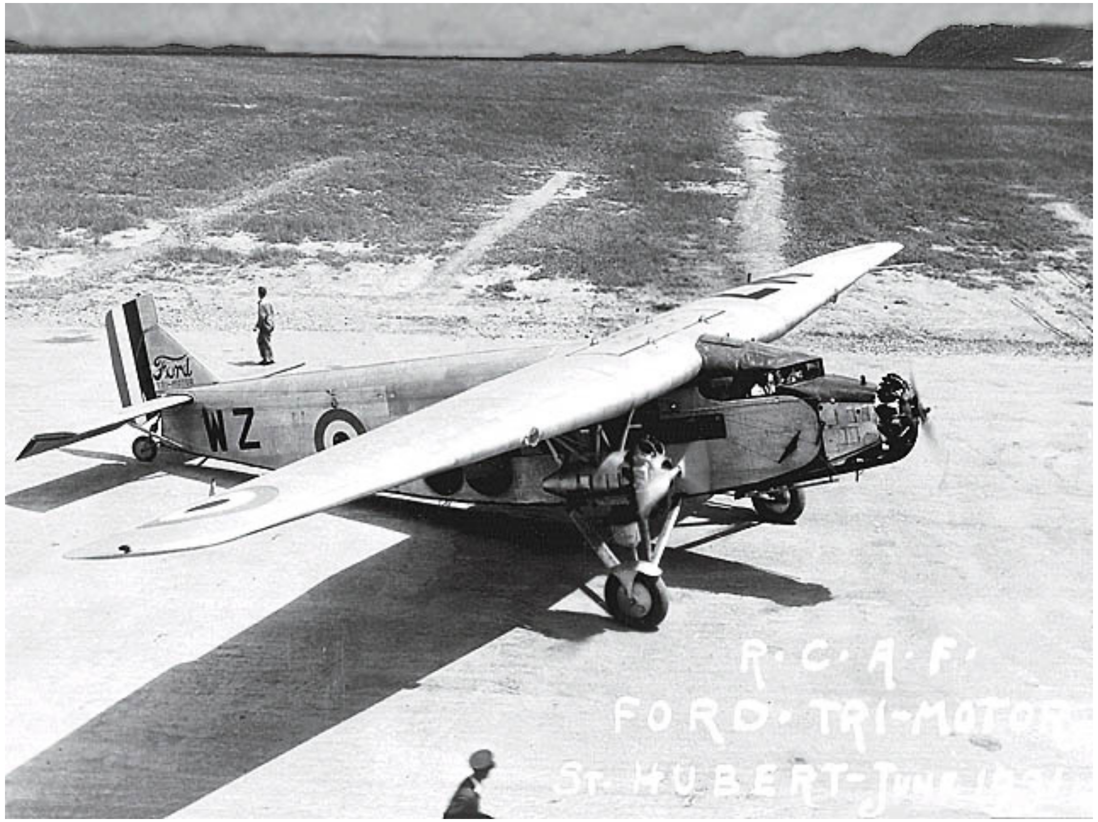
- First powered aircraft fatality (1908): Wright Flyer
  - A propeller failed
  - Fragments damaged structure / flight controls
  - The crash injured Orville Wright and fatally injured Lt. Thomas Selfridge
- What to do about this?
  - Make better parts, of course!



https://commons.wikimedia.org/wiki/File:1903-12_Wright-Flyer-side-view.jpg

Hmmm. That's not enough. More parts!

- Perfect parts are not possible: stuff is going to break
- Solution: redundancy!
  - Piston radial engines failed with distressing frequency
  - But if you've got more than you need, and one fails …



https://commons.wikimedia.org/wiki/File:Ford_Trimotor.jpg

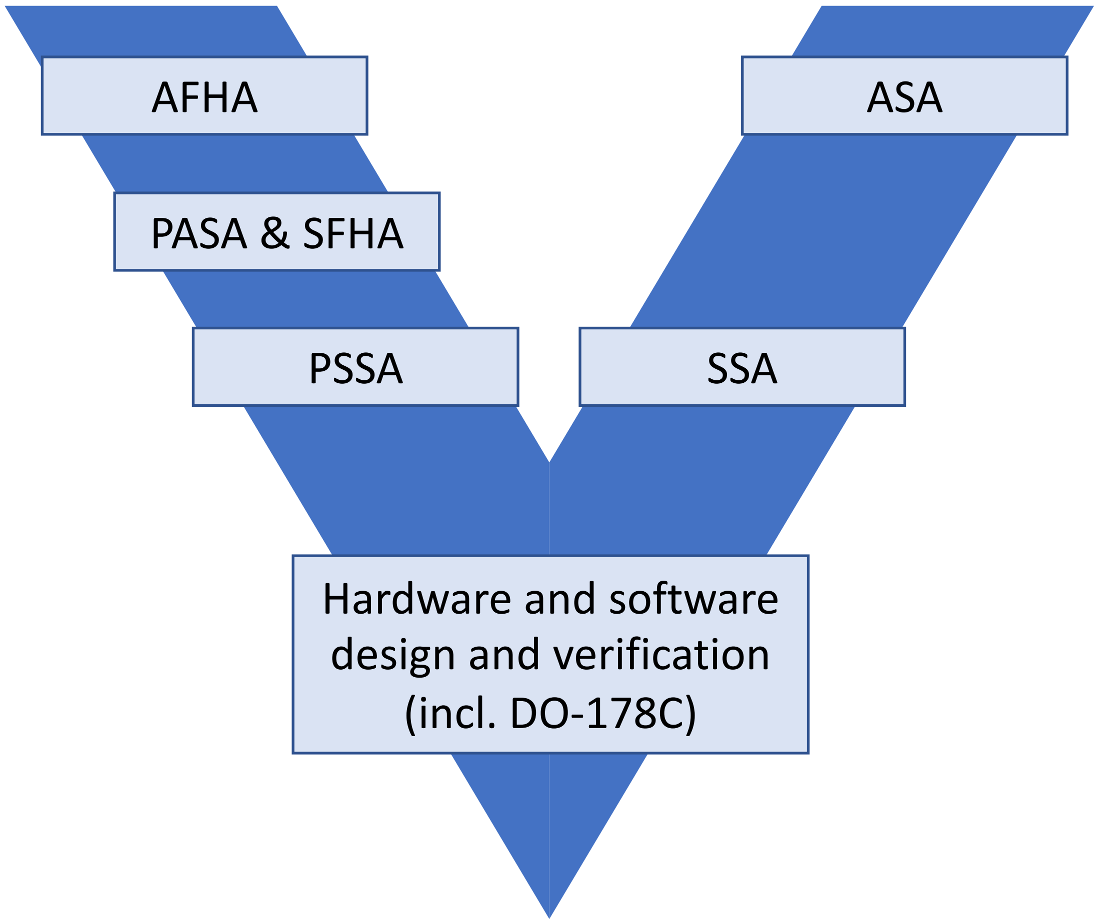
But what if they all fail at the same time?

- We've seen redundant things fail simultaneously
  - UAL232: Engine debris disables three redundant hydraulic systems
  - CI202: Voting logic fails three lanes of main/monitor computer pairs
- ARP4754B/ARP4761A process
  - *Common cause analysis* (CCA)
  - One *particular risks analysis* (PRA) identifies vulnerability to damage from uncontained engine debris

https://en.wikipedia.org/wiki/United_Airlines_Flight_232#/media/File:UA232precrash.gif

Aircraft safety engineering process

- SAE ARP4754B (soon!) defines the overall process:
  - Functional hazard analysis (*x*FHA)
  - Preliminary safety analysis (P*x*SA)
  - Development assurance
  - Safety analysis (*x*SA)
- ARP4761A (soon!) defines the analyses that ’54 calls for



Note: Some process steps are done at both the aircraft (A) and “system” (S) levels.

Safety expertise is needed at all stages

- Functional hazard assessment:
  - Identify failure conditions (FCs)
    - Drives safety requirements
  - Determine possible effects
  - Classify those effects
    - Drives development assurance levels

- Effects & classifications often come from expert judgment
  - History of pilot training and action
  - History of classifications
  - Can check *some* (not all!) flight crew responses in a simulator

Function	Failure condition	Flight Phase	Effects	Classification
Decelerate on ground	Loss of ability to decelerate with crew aware	Takeoff	FC: Aware of condition, crew will choose suitable location & minimize airspeed & weight. Excessive crew workload.	Catastrophic

Safety expertise is needed at all stages

- Zonal safety analysis (ZSA):
  - Divide the aircraft into zones
  - Identify equipment in zone
  - Prepare checklist, e.g., look for:
    - Drainage & accumulation
    - Clearances around hoses
    - Potential for damage due to maintenance activities
  - Identify unexpected interactions
- Checklists are driven in large part by lessons learned

- Common mode analysis (CMA):
  - Performed at both P*x*SA and *x*SA
  - Helps to define requirements from independence principles & verify satisfaction of those requirements
  - Again, based on checklists:
    - Errors in software tooling?
    - Errors in common software libraries?
    - Errors in software function (e.g., aircraft dynamics models)?
- Again, expertise features heavily

Process and intelligence are not enough

- Hazard analysis is *guided enumeration*
  - Systematic, piece-by-piece examination of a system asking ‘what-if’ questions
    - FHA iterates over functions
    - Hazard Operability Study (HazOp) iterates over flows in a plant schematic
    - System Theoretic Process Analysis (STPA) iterates over controllers and control actions
  - Systematic, piece-wise analysis helps ensure every corner is searched
  - But analysts may not see what they don’t know to look for
- Planning/ensuring sufficient mitigation requires judgment (expertise)
  - If you think a 15m tsunami is not credible, you don’t build for it
  - If you think Byzantine faults are vanishingly rare, you don’t build in Byzantine fault tolerance

History reveals the unknowns to us

- 1972 Eastern 401: Crew resource management is essential
- 1982 British Airways 9: Volcanic ash is really bad for turbine engines
- 1982 Air Florida 92: Engine pressure probe icing creates false thrust reading
- 1988 Aloha 243: Short cycles in humid, salty air accelerates fatigue
- 1988 TACA 110: Engines react differently to hail than to rain
- 1989 United 232: Uncontained engine debris can fail triply-redundant hydraulics
- 2008 British Airways G-YMMM: "Sticky ice" can clog fuel systems
- 2009 Air France 447: Training for high-altitude stall is necessary
- 2020 United N16009: "Repeat clearance" beats "confirm"
- 2020 Titan Airways G-POWN: Kathon overdose can lead to dual engine failure

But all that’s about systems, not software …

- Planes aren’t falling out of the sky over misplaced semicolons
  - DO-178C might not be infallible, but it works … for now
- In accidents, software usually performed per its spec.
- And where the specs are wrong, it’s often about management of fault cases
  - And sometimes human factors …
- 2007 Boeing 777 9M-MRG: Fault management logic puts a known-faulty accelerometer back into service
- 2011 Airbus A330 VH-QPA: Fault management logic can’t handle spiky angle-of-attack data
- 2020 Airbus A330 B-18302: Rudder oscillation at touchdown fails all 3 (main-mon.) flight computers

Safety expertise is accumulated wisdom

- We learn from stuff going wrong
  - Not always in accidents, and not always published
    - Things get caught at the design stage …
- We learn from being continually curious and humble
  - *"The best designers … are never not thinking about product safety. [They] recogniz[e] fallibility … as hard-wired in humanity. [They] are thus always prepared … to uncover potential threats to safety, often subtle and seemingly implausible threats, and to chase them to bitter ends."* — Frank McCormick
- We learn from each other
  - Accident/incident reports are remarkably open/transparent

There’s no substitute for expertise

- You can’t test your way to perfect requirements
  - Pilot-in-the-loop simulation is too expensive to test every crew reaction
  - Can’t test everything; you need to know which axes/variables might matter
- You can’t just simulate your way to perfect requirements
  - Simulations only reflect the parts of reality they are created to reflect
    - Do road simulations for car vision systems include 7-foot fuzzy pink werewolves?
    - Do the images show effects from dead pixels, gunk on lenses, dust catching light, etc.?
    - Do the images have ramen shop logos that look like wrong-way symbols?
- You can’t calculate your way to perfect requirements
  - Need to know which formulae hold where, which data is applicable, etc.

Big questions turn on safety expertise

- There is a pronounced split on interpretation of the requirement that no single failure will result in a catastrophic failure condition
  - Some folks maintain that “no single failure” implies “no single error”
  - Some interpret this as requiring mitigations such as dissimilar architecture
  - Some folks insist dissimilar architecture is not always required or even helpful
- A lot of the debate turns on expertise derived from limited evidence
  - Understandings of the kinds of failures that happen and *could* happen
  - Experience of having deployed various kinds of redundancy
  - A lot of this is company proprietary data

Robust monitoring and transparency are key

- Civil aviation has a long and robust practice of accident and incident investigation
  - Investigators work with airframers and engine manufacturers
  - Aircrews and maintainers report, e.g., in the Aviation Safety Reporting System (ASRS)
  - This is something new sectors would do well to emulate

- Civil aviation safety culture is remarkably open & transparent
  - Accident reports reveal detail that folks would prefer not to share
  - No one likes bad news … but we have anonymous reporting (e.g., in ASRS)
  - *The benefits of learning from each other are seen as worth protecting*

Novelty must be approached cautiously

- Don’t embrace novelty for novelty’s sake
  - Even when it doesn’t cost lives, lessons can be expensive
    - E.g., Boeing 787 fleet grounded after lithium-ion battery fires
- Try out novelty in safer / more risk-tolerant applications
  - Cautious buildup of experience with turbines is how we worked up to today’s long overwater flights in twin-engine aircraft
  - A novel autonomous crop duster crashing in an unpopulated field is better than a self-flying robotaxi crashing in Manhattan
  - Autonomous monitoring of wildland fires might provide benefit worth the unknown risk of deploying untrusted technology

Safety expertise must be cultivated

- Expertise must be passed down
  - No textbook holds all the expertise in the minds of good engineers
  - People retire
  - People quit
  - People die
  - Young engineers don't know what they don't know
  - Promotion process matters
  - Mentorship matters
- Expertise must be brought in where it is needed
  - New ventures may lack an experienced 'old guard'
  - Different kinds of expertise …
    - Crop dusters will tell you about flying near power lines
    - Maintenance folks know how design choices affect maintainability
    - Etc.
  - There is a market for ex-DERs …

Implications for safety reasoning

- Reaching agreement requires shared understanding
  - When a regulator and developer disagree, it can be over background
    - Understanding of how likely circumstances are to arise
    - Understanding of failure modes of technology
    - Understanding of when prior experience or common wisdom isn’t relevant
  - *Dialogic* argument is good at unpacking positions and finding the disparities
    - But you only need this where you need it!