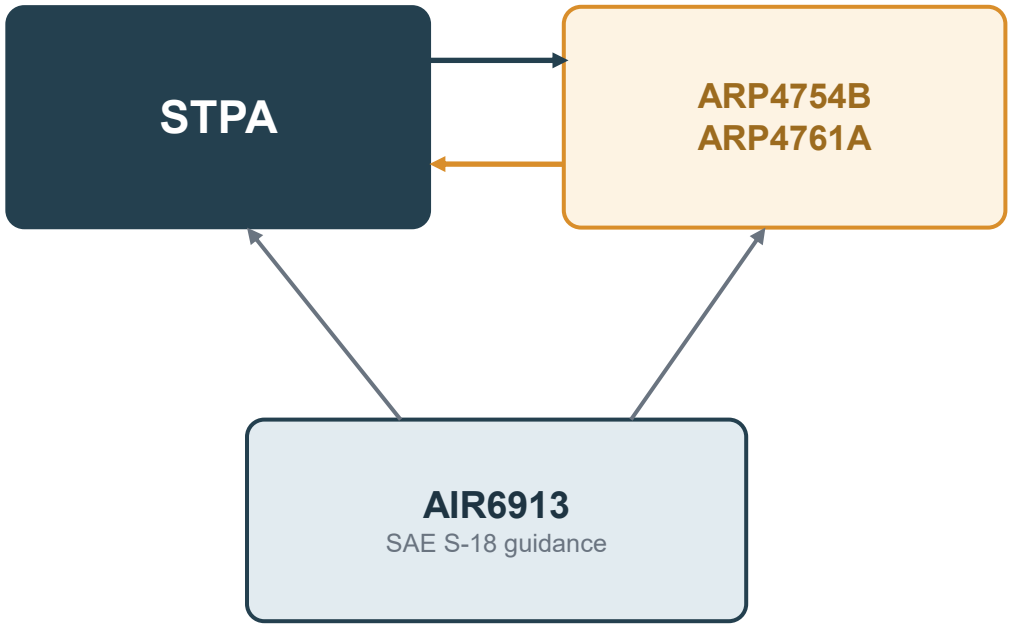
Where STPA Fits in the Aircraft Safety Assessment Process

AIR6913 – building a bridge between STPA, ARP4761A and ARP4754B

Charlie Jenkins, CEng
Dave Cummins, CEng, Intl PE



STPA and the ARP processes working together: the subject of this talk and of AIR6913.

What is AIR6913?

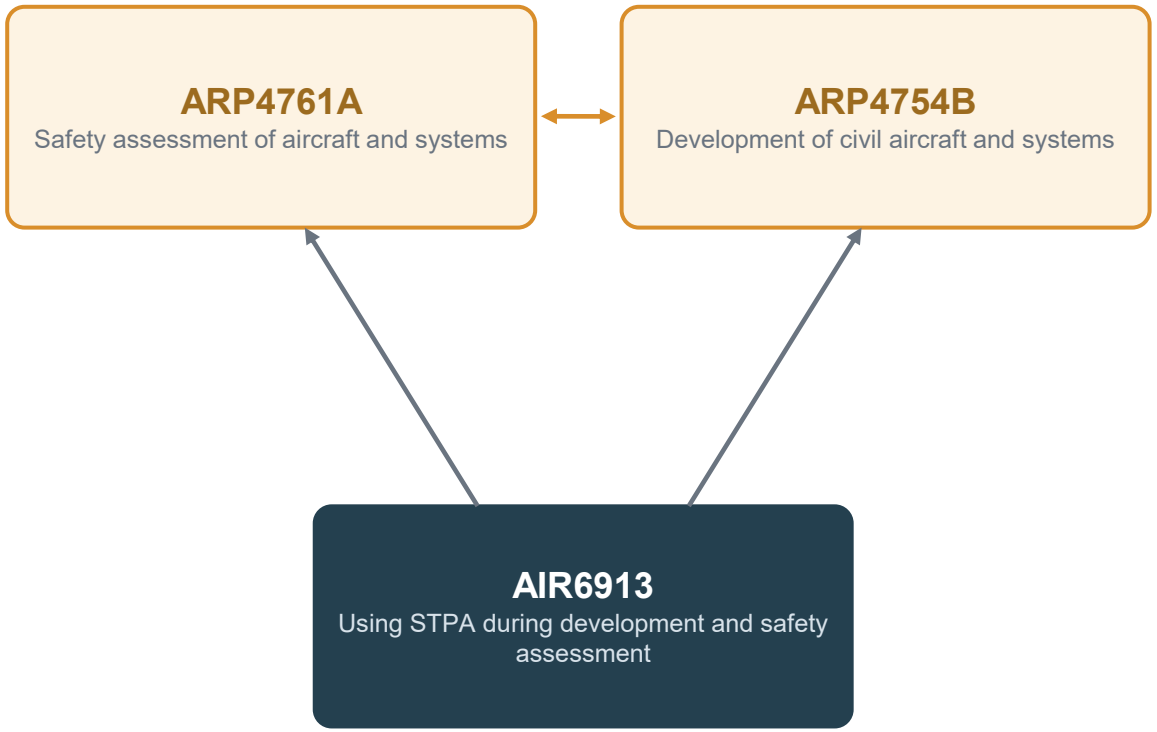
AIR6913 is an SAE **A**erospace **I**nformation **R**eport (**AIR**). An SAE AIR captures information the community has agreed on, to inform a practice without prescribing it.

Whether STPA and the ARP processes can work together has long been a point of contention.

AIR6913 builds that bridge: a shared, community-vetted account of where STPA supports ARP4754B and ARP4761A (referred to as ARP4754/61 in this presentation), the development and safety assessment processes used globally. AIR6913 is becoming part of the initial exemplar feeding the regulatory baseline for application of STPA in civil aerospace

AIR6913 is the detailed companion report for this presentation, and contains the detailed method mapping, worked example, scenario derivations and traceability.

AIR6913 is near it's second ballot, with publication targeted before the end of 2026.



Informational: it supports the application of both ARPs and adds no certification requirement.

List of common questions

Questions

1. Can STPA results be used within the established aircraft development and safety-assessment framework?
2. If so, where do the results go?
3. What does STPA add, and what remains the role of established methods?

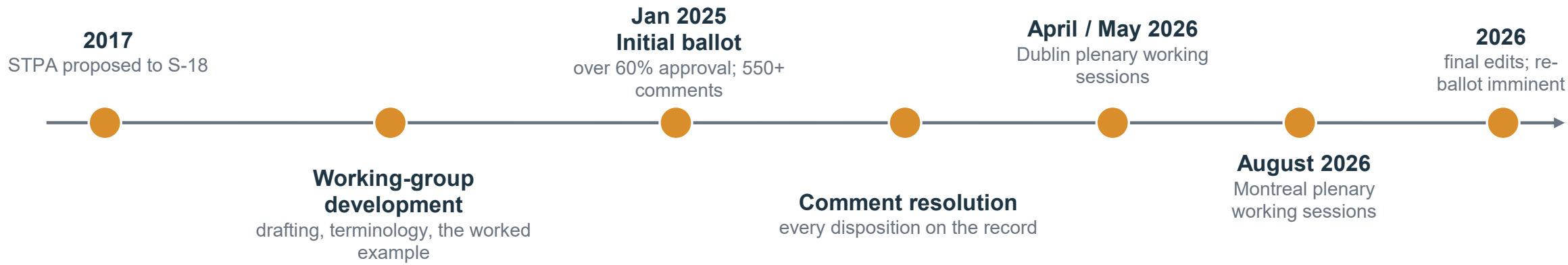
Answers

1. For years, practitioners have asked whether STPA can be used coherently within the established civil-aircraft development and safety-assessment framework alongside ARP4754/61. The short answer is yes, the long answer is very interesting and hasn't been an easy journey, see rest of presentation and AIR6913
2. AIR6913 establishes common terminology, identifies similarities and differences, and maps STPA outputs into ARP4754/61 activities
3. AIR6913 draws the clear parallels and demonstrates how STPA can work in harmony and support the ARP4754/61 process

Note: The relationship is not a simple one-to-one process substitution. STPA contributes information to safety assessment and to wider systems engineering, requirements, architecture, human factors, and verification.

AIR6913 is S-18's detailed answer to these sticking points and common questions about the relationship between ARP4754/61 and STPA

Capturing over nine years of cross community resolution



Nine years from proposal to re-ballot. The recurring arguments were conceptual and were settled on the record: whether STPA and the ARPs run as parallel tracks (**they do not** - the crossover is many-to-many).

The differences that AIR6913 had to bridge – methods that share a goal but organize analysis differently

STPA

- Defines the system as a hierarchical control structure comprising people, organizations, automation, software, hardware, control actions, and feedback
- Begins with unacceptable losses, hazards, and safety constraints, then identifies unsafe control actions
- Examines how inadequate requirements, flawed interactions, incorrect process models, timing or coordination issues, and component failures could produce unsafe control
- Uses the resulting causal scenarios to refine the architecture and derive enforceable safety requirements and design constraints

ARP4761

- Based on functional failure conditions with hazardous effects; layers of defence are added until an accepted risk level is met
- Decades of method, precedent and certification evidence stand behind it

ARP4754

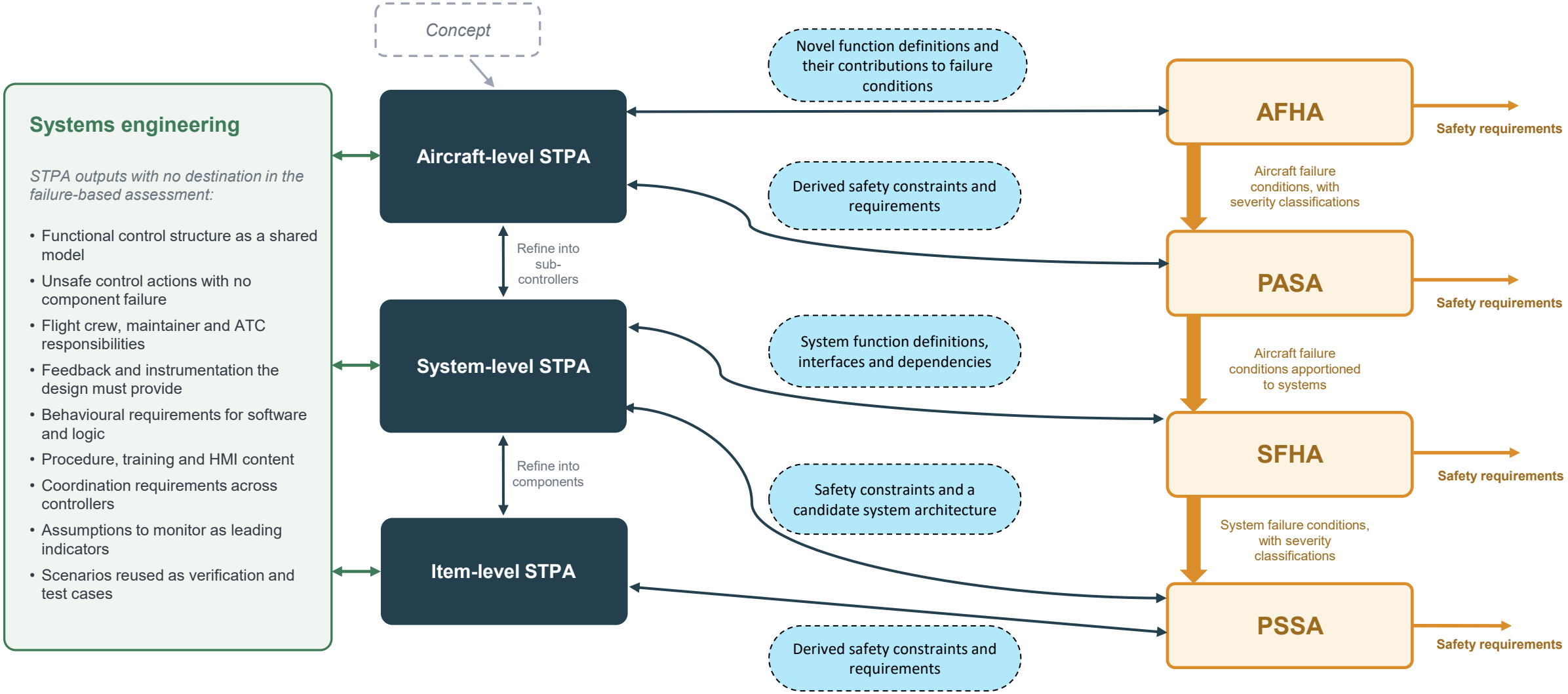
- Defines the aircraft as a hierarchy of functions and systems, progressively allocating requirements and safety objectives through the architecture
- Uses development assurance, requirements validation, and implementation verification to provide confidence that the resulting design satisfies its intended function safely

Key distinction: STPA organizes safety analysis around control and interaction, whereas ARP4754/61 generally organize it around functional failures and their effects

STPA and ARP4754/61 share the same destination, different tracks, tightly connected, driving toward a common safety goal

AIR6913 bridges STPA and the ARP safety assessment

Whether the two connect at all has long been contested. The figure maps each STPA output into the assessment.



High level STPA example included in AIR6913



The example:

A transport-category; the deceleration function, delivered by ground spoilers, wheel brakes and thrust reversers.

Deliberately high level and illustrative, not a complete worked analysis.

Only the thrust reverse controller is taken through UCAs and loss scenarios.

What the example shows STPA adds to the ARP work:

- A completeness cross-check on the AFHA and SFHA, as a many-to-many relationship - in both directions
- A functional view for PASA/PSSA allocation and fault-tree contributors
- Failure conditions, functions and feedback the decomposition potentially had not yet captured
- Four classes of loss scenario that could map onto PSSA fault-tree branches, and other ARP artefacts
- Loss scenarios with no failure which could drive architecture and requirements
- Input to requirements with traceable rationale, from loss through to verification objective

Losses, hazards, and the AFHA

STPA

ID	Loss
L-1	Loss of life
L-2	Loss of aircraft
L-3	Damage to aircraft
L-4	Loss of reputation

ID	STPA hazard (landing / take-off / RTO)	AFHA-1	AFHA-2	AFHA-3	AFHA-4	AFHA-5
H-1	Aircraft leaves the runway laterally		✓		✓	✓
H-2	Aircraft overruns the runway	✓	✓	✓	✓	
H-3	Aircraft exceeds defined structural envelopes		✓			✓
H-4	Aircraft is uncontrollable during {phase}	✓	✓	✓	✓	✓

✓ = related. – = not directly related (may warrant further examination). Every hazard traces to L-1 to L-4. STPA hazards could also link to failure effects from a given AFHA FC.

ARP4761A

ID	AFHA failure condition
AFHA-1	Loss of aerodynamic brake
AFHA-2	Asymmetric aerodynamic drag
AFHA-3	Loss of ground deceleration
AFHA-4	Asymmetric ground deceleration
AFHA-5	Uncommanded ground deceleration

AFHA-1/2: function ‘Provide controlled aerodynamic braking’. AFHA-3/4/5: ‘Provide deceleration on the ground’.

STPA hazards: aircraft states, whether or not a failure is involved. A hazard with no AFHA failure condition prompts a search for a potentially missing failure condition.

AFHA: failures of known aircraft functions. A failure condition with no STPA hazard may not be aircraft-level but is still evaluated for lower-level effects.

The relationship is many-to-many. Cross-referencing in both directions can give an auditable completeness check on both sets.

Safety constraints, the AFHA, and PASA allocation

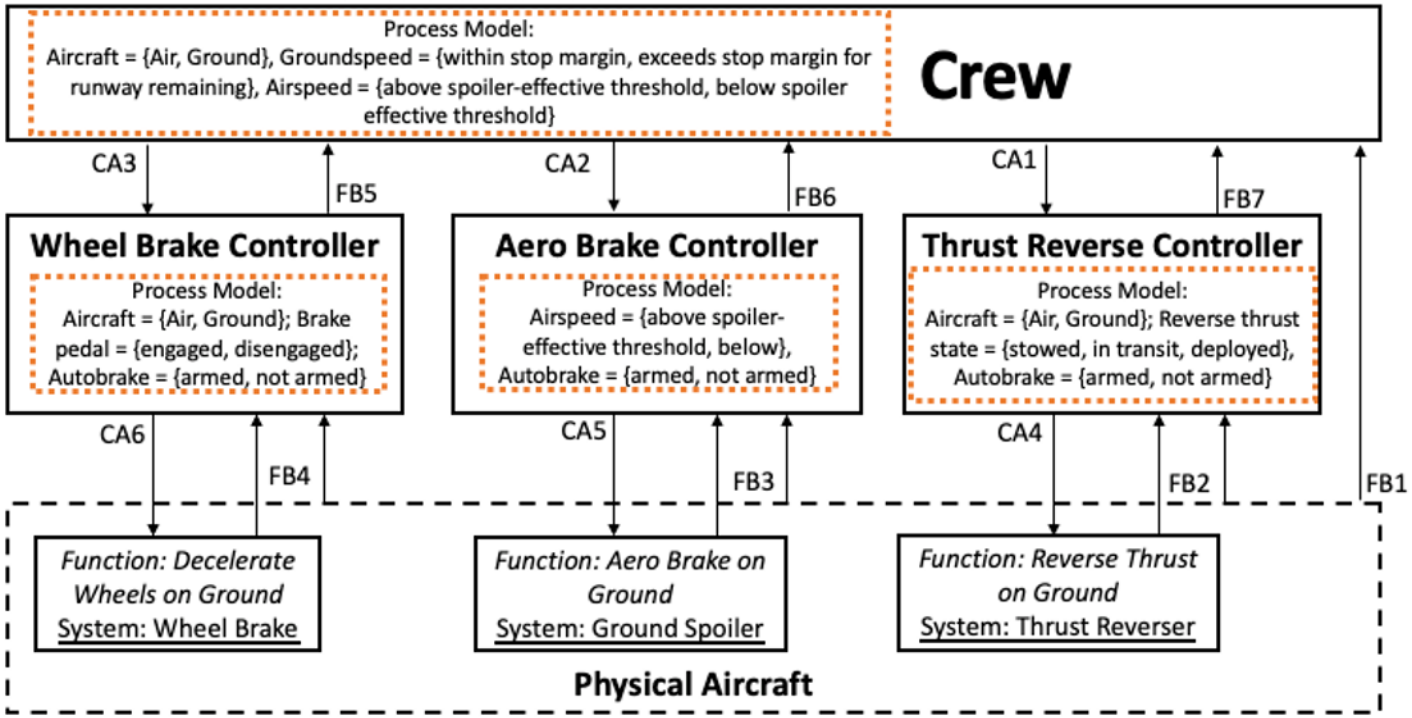
ID	STPA safety constraint (aircraft level)	From	PASA allocation: systems contributing
SC-1	Aircraft must not leave the runway laterally until a safe departure speed has been achieved	H-1	Aero brakes (AB), Wheel brakes (WB), Reverse thrust (RT)
SC-2	Aircraft must not overrun the runway	H-2	AB, WB, RT
SC-3	Aircraft must not exceed the defined structural envelope	H-3	AB, WB, RT
SC-4	Aircraft must be controllable during {phase}	H-4	AB, WB, RT

Each constraint is **derived from its hazard**, and inherits the hazard's many-to-many relationship to the AFHA failure conditions.

The allocation lists every system that contributes to the constraint; each **must satisfy it as a requirement**.

> **Setting safety constraints is the first step of requirements derivation: through the PASA they are levied on systems for further decomposition and analysis, alongside the failure-condition safety objectives.**

Control structure, PASA, and PSSA



The functional view could be used to support multiple design and safety assessments;

- Crew as the highest-level controller; three sub-controllers acting on the physical aircraft, each holding a process model of aircraft state.
- Shows which inputs each controller depends on (groundspeed, WoW, wheel speed, crew selections) and how they flow through to the landing / RTO behaviour.
- An aid to ARP4761A Appendix D.3: mapping contributors to a failure condition by functional decomposition, and a **completeness check for the fault tree**.
- A failure condition also depends on the **data each controller receives**: STPA feeds both failure identification and error identification (e.g. the CMA) in the PASA and PSSA.

STPA control actions, process models, and feedback

Controller	Responsibility / goal	Control actions (CA) → target	Process model variables	Feedback (FB) examples
Flight crew	Ensure the aircraft stops within the runway distance available	**CA1** Activate thrust reverse; modify autobrake mode → thrust reverse controller **CA2** Apply aero brake; modify autobrake mode → aero brake controller **CA3** Apply wheel brakes; modify autobrake mode → wheel brake controller	Aircraft = {air, ground} Groundspeed = {within, exceeds} stop margin for runway remaining Airspeed = {above, below} spoiler-effective threshold	**FB1** Main-gear WoW; groundspeed; runway distance remaining; deceleration achieved **FB2** Reverser state per engine (stowed / in transit / deployed); fault or disagree annunciation; reverse thrust achieved **FB3** Spoiler deployment confirmation; fault or asymmetry annunciation; armed status **FB4** Brake-applied confirmation; anti-skid active; autobrake armed / engaged / disarm; brake fault or overtemperature
Thrust reverse controller	Ensure reverse thrust contributes adequate ground deceleration	**CA4** Deploy thrust reverser(s) → physical aircraft	Aircraft = {air, ground} Reverse thrust state = {stowed, in transit, deployed} Autobrake = {armed, not armed}	**FB5** Air/ground determination (WoW per main gear; wheel spin-up above threshold; radio altitude; thrust levers at idle); reverser position and lock status per engine, incl. disagree; engine parameters for reverse thrust achieved (N1 / EPR)
Aero brake controller	Ensure aerodynamic braking contributes adequate ground deceleration	**CA5** Deploy / retract spoilers → physical aircraft	Aircraft = {air, ground} Airspeed = {above, below} spoiler-effective threshold Autobrake = {armed, not armed}	**FB6** Air/ground determination; airspeed (dynamic pressure / effectiveness); spoiler position per panel incl. asymmetry; hydraulic pressure (actuation power); electrical supply (control power)
Wheel brake controller	Ensure wheel braking contributes adequate ground deceleration	**CA6** Apply wheel brakes → physical aircraft	Aircraft = {air, ground} Brake pedal = {engaged, disengaged} Autobrake = {armed, not armed}	**FB7** Gear down-and-locked; WoW; wheel speed per wheel; brake pressure per wheel or truck; skid / lock detection; hydraulic availability; brake temperature; tyre pressure; selected autobrake setting

- **STPA makes analysing the feedback mandatory:** can each responsibility be fulfilled with the feedback available?
- **Are all of these responsibilities covered by the ARP functional decomposition?**
- **Step 4 then asks which signals maintain each process-model variable,** and what happens when they are missing, delayed or wrong.

Step 3: unsafe control actions for the thrust reverse controller

Control action	Not provided	Provided when not needed	Wrong timing or sequence	Wrong duration
Deploy thrust reverser(s)	**UCA-1: Thrust reverse controller does not deploy thrust reverser(s) when aircraft is on ground, reverse thrust selected, and current deceleration inadequate for the runway remaining** [H-2].	**UCA-2 A:** Thrust reverse controller deploys thrust reverser(s) when aircraft is airborne [H-1, H-3, H-4]. **UCA-2B:** Thrust reverse controller deploys thrust reverser(s) when aircraft is accelerating for takeoff [H-4].	**UCA-3A: Thrust reverse controller deploys thrust reverser(s) too late after touchdown such that insufficient runway remains** [H-2]. **UCA-3B:** Thrust reverse controller deploys the thrust reverser on one engine too early (>TBD s) before the other engine's reverser has deployed (resulting in out of sync reverser deployment) [H-1, H-4]. **UCA-3C:** Thrust reverse controller deploys thrust reverser(s) too early (>TBD s) before ground spoilers have deployed (potentially resulting in deployment with excessive lift, reduced ground traction, or undesired pitch motion) [H-1, H-4].	**UCA-4A:** Thrust reverse controller continues thrust reverser(s) deployment too long after groundspeed has reduced below the minimum speed (for safe reverse thrust operation, risking exhaust reingestion and engine surge) [H-3]. **UCA-4B:** Thrust reverse controller stops deploying thrust reverser(s) too soon before full deployment, leaving thrust reverser(s) partially deployed [H-2].
Stow thrust reverser(s)	**UCA-5:** Thrust reverse controller does not stow thrust reverser(s) when groundspeed is below the runway exit speed [H-3, H-4].	**UCA-6:** Thrust reverse controller stows thrust reverser(s) while engines are providing thrust above idle [H-2].	**UCA-7A:** Thrust reverse controller stows thrust reverser(s) too early before groundspeed reduces to safe speed (at which the remaining runway distance is sufficient to stop without reverse thrust contribution) [H-2]. **UCA-7B:** Thrust reverse controller stows thrust reverser(s) too late after groundspeed falls below the runway exit speed [H-3, H-4].	**UCA-8A:** Thrust reverse controller continues to command stow too long after stow travel is complete, locking out thrust reverser availability for any subsequent deployment demand (e.g., a stow command that persists into the next flight inhibits deployment when next commanded) [H-2]. **UCA-8B:** Thrust reverse controller stops commanding stow too soon before stow travel is complete, leaving reverser(s) partially deployed [H-1, H-3].

Step 3: UCAs, safety constraints, and the SFHA

STPA UCA (thrust reverse controller)	Safety constraint derived from the UCA	Relationship to the SFHA failure conditions
UCA-1 Does not deploy when on ground, reverse selected and current deceleration inadequate for the runway remaining [H-2]	**C-1** Shall deploy thrust reverser(s) when on ground, reverse thrust selected and current deceleration inadequate for the runway remaining.	**FC-1** Total loss of thrust reverser(s) deployment (landing / RTO) **FC-8** Unannunciated total loss of deployment (landing): **added to the SFHA from the Class 2 scenario**
UCA-2A Deploys while airborne [H-1, H-3, H-4]	**C-2A** Shall not deploy thrust reverser(s) when the aircraft is airborne.	**FC-3** Uncommanded thrust reverser(s) deployment (flight); UCA-2B is the same FC in the take-off phase
UCA-3A Deploys too late after touchdown, so insufficient runway remains [H-2]	**C-3A** Shall deploy within a defined maximum time after touchdown such that adequate deceleration is achieved within the runway available. The value is a design decision: validate it across the operational envelope.	**FC-4** Delayed thrust reverser(s) deployment (landing / RTO): a timing FC that failure-type iteration does not always reach
UCA-3C Deploys > TBD s before the ground spoilers have deployed [H-1, H-4]	**C-3C** Shall not deploy more than TBD s before the ground spoilers have deployed. A cross-system sequencing constraint; allocation may sit at aircraft level.	**No corresponding failure condition in the SFHA.** Each per-system assessment sees its own function operating correctly.
UCA-4A / UCA-5 Deployment continued below the minimum safe reverse speed / not stowed below the runway exit speed [H-3, H-4]	**C-4A / C-5** Shall not continue deployment below the minimum speed for safe reverse; shall stow when groundspeed is below the runway exit speed.	**FC-6** Inability to stow thrust reverser(s), **but only if a stow command has been given:** the SFHA is carrying an unstated assumption.

- **Many UCAs have potential links to failure conditions that conventional decomposition also produces;** the mapping then could inform the analysis of that FC in terms of which inputs and contexts to consider.
- **Where a UCA has no clean failure-condition pair, ask why:** no failure-based cause, or functionality that the system description has not yet expressed? (next slide)
- **Rephrased as safety constraints, UCAs become traceable building blocks for requirements:** comparing a design requirement such as R-1 ‘stow only below X kt’ back to C-7A exposes the assumed runway state, brake performance, mass, distance remaining and groundspeed source behind X.

UCAs with no clean “pairing”

UCA-3C → a candidate aircraft-level failure condition

Reverser deployed before the spoilers. Each per-system SFHA sees its own function working correctly, and no failure type applied to a single function produces ‘correct operation in the wrong order’. It only becomes visible with the three deceleration means in one control structure.

Candidate AFHA-6: Deceleration devices operate out of intended sequence.

C-3C is a cross-system sequencing constraint. Whether it is carried as an aircraft FC, a derived requirement or an operational consideration is the applicant’s decision through the ARP process. STPA’s value is that it asks the question systematically.

UCA-4A and UCA-5 → an unstated assumption

Both UCAs are stated against a speed. The only stow function in the SFHA is ‘Stow when commanded’, and both map to FC-6 ‘Inability to stow’. That mapping only works if someone has commanded stow.

ASM-1: crew recognise the speed cue and command stow within TBD s if there are no competing alerts. Human factors to validate.

…or a missing function: ‘Stow thrust reverser(s) automatically’ below a defined groundspeed, with the failure conditions that belong to it.

UCA-3C → feedback the controller does not have

To prevent UCA-3C the controller needs to know the spoiler state. FB5 in Table 7 lists air/ground, reverser position and lock, and engine parameters. Nothing about the spoilers.

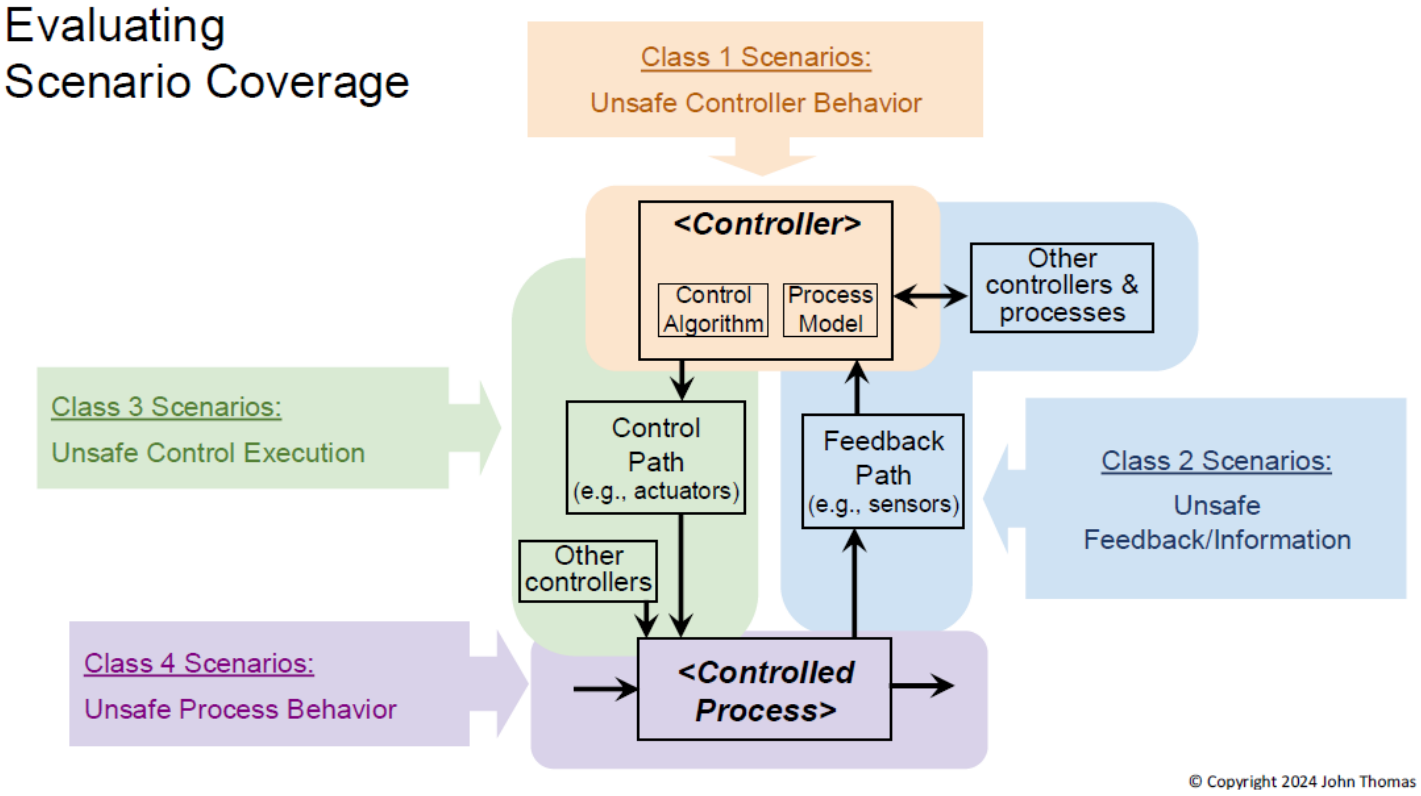
Candidate function: Provide indication of ground spoiler deployment state.

Allocating reverser operation to the crew instead moves the responsibility, not the need: the crew would need the same indication.

Comparing the UCAs against the function list, and against the feedback each controller receives can act as a completeness check on the functional decomposition.

Step 4: Scenario Classes

Figure 21 from the AIR6913 draft (© John Thomas, 2024).



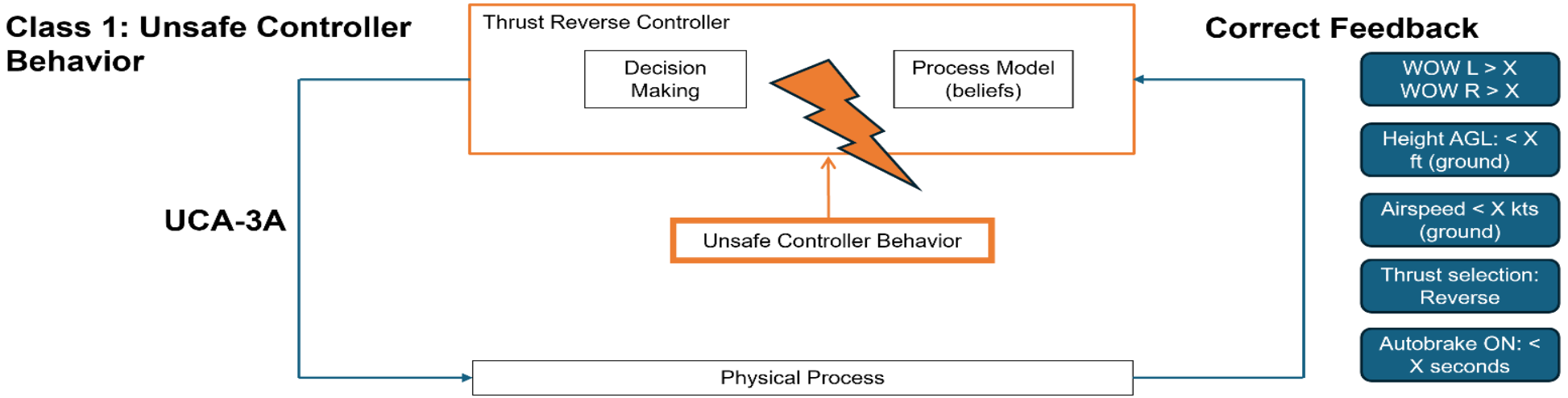
Each class interrogates one part of the control loop.

- **Class 1 Unsafe controller behaviour** → UCA-3A: deploys too late (decision logic and timing)
- **Class 2 Unsafe feedback / information** → UCA-1: does not deploy (the controller believes 'airborne')
- **Class 3 Unsafe control execution** → UCA-4B: stops deploying before full travel (command path)
- **Class 4 Unsafe process behaviour** → UCA-1: correct command, the process does not respond

Scenarios are built from the UCA, the control structure, the controller's process model and decision making, and the feedback; 'why' narratives add operational context.

Systematically deriving STPA loss scenarios and stress testing the control structure can expose gaps and strengthen ARP outputs.

Class 1, unsafe controller behaviour: asking ‘why’ of UCA-3A



UCA-3A: Thrust reverse controller deploys thrust reverser(s) too late after touchdown such that insufficient runway remains [H-2].

WHY?

DM-C1-1: If aircraft is on the ground, and reverse thrust selected, deploy thrust reverser(s) after X seconds (where X is a design defined safety delay to prevent in-air deployment).

DM-C1-2: If autobraking is selected, deploy thrust reverser(s) Y seconds after touchdown (where Y is the autobrake-specific delay, which may differ from X).

DM-C1-3: If the safety delay timer has not expired, do not deploy thrust reverser(s) regardless of other conditions.

WHY?

PM-C1-1: The thrust reverser controller correctly identifies that reverse thrust has been selected by the crew.

PM-C1-2: The thrust reverser controller correctly identifies the aircraft as on-ground.

PM-C1-3: The thrust reverser controller incorrectly identifies the aircraft as airborne.

PM-C1-4: The thrust reverser controller believes the safety delay the safety delay timer has not yet expired, even though the aircraft has been on-ground long enough for deployment to be both safe and operationally necessary.

WHY?

Class 1 scenarios for UCA-3A and uses in ARP

UCA-3A: Thrust reverse controller deploys thrust reverser(s) too late after touchdown such that insufficient runway remains [H-2]. The feedback saying ‘on ground’ was correct and on time.

W-C1-1 The delay parameter

Safety delay X guards against in-air deployment but was validated for nominal landings. For a short, contaminated, heavy or tailwind landing the same X is too large.

Finding and feedback to ARP: validate X across the operational envelope; consider adaptive delay logic or operational limits.

W-C1-2 WoW voting delays the timer

In an asymmetric or crosswind touchdown only one gear declares ground; the voting waits for agreement, so the timer, and deployment, start late relative to the real touchdown. Every WoW signal is correct: it is the controller’s own processing that defers the call.

Finding and feedback to ARP: voting logic drives downstream timing; a PSSA architectural consideration.

W-C1-3 Two timers, two ground calls

Autobrake runs its own timer Y from its own ground-contact confirmation, possibly from different sensors or thresholds. If the two controllers declare ground at different times, reverse comes later than the crew expect.

Finding and feedback to ARP: a coordination requirement between autobrake and reverser logic, to be examined in the PSSA.

W-C1-4 Timer implementation error

A counter overflow, wrong clock source or rounding error extends the delay beyond the specified X seconds.

Finding and feedback to ARP: a specific timing verification objective for the SSA, linked to DO-178C.

Four scenarios for one UCA, each ending in a specific, testable finding. Three of the four need nothing to fail: the design works as intended and the outcome is still unsafe in context. The same delay that prevents UCA-2A creates UCA-3A; STPA makes the trade-off explicit and bounds the parameter.

Classes 3 and 4: where STPA meets the fault tree

Class 3 Unsafe control execution: UCA-4B

Stops deploying before full travel, leaving the reverser(s) partially deployed [H-2]. The controller's process model and decision are correct and the deploy command is issued; it is the command that arrives that differs.

W-C3-1 The command to one engine's reverser is blocked, delayed or pre-empted by a conflicting command handled at higher priority: incomplete, asymmetric net reverse thrust.

W-C3-2 The command is degraded in transmission (drift, corruption, partial channel loss) and the position feedback that would contradict it is missing, so the controller still believes 'deployed' and the discrepancy persists undetected.

Into the PSSA: the fault tree for FC-2 (insufficient deployment) must carry these command-path modes, and the availability of position feedback is a PSSA question.

Class 4 Unsafe process behaviour: UCA-1

Does not deploy when on ground, reverse selected and deceleration inadequate [H-2]. Process model, decision and command delivery are all correct; the controlled process does not produce the commanded effect.

W-C4-1 Actuator receives the command but the blocker doors do not reach full deployment: hydraulic pressure loss or excess demand, mechanical binding or obstruction, actuator degradation over time.

W-C4-2 Ice or slush in the reverser cowl after operations on a contaminated surface obstructs door travel with a healthy actuation system: the same operating context as W-C2-4, reached from the physical side.

Into the PSSA / SSA: closest to conventional failure causes; contributors for FC-1 / FC-2 and end-to-end actuation requirements.

Class 1 → controller logic and timing. Class 2 → feedback. Class 3 → control path. Class 4 → physical process. Together they are a systematic sweep of the control loop that maps directly onto PSSA fault-tree branch identification.

Is this insight useful during the safety assessment?

- **Specific and testable** - delay ranges, voting logic, sensor sourcing, timer coordination: the findings feed the PSSA, the SSA and verification directly, with a DO-178C objective at the end of one thread.
- **Trade-offs made explicit** - the safety delay exists to prevent in-air deployment (UCA-2A) and creates late deployment (UCA-3A). STPA bounds the acceptable range for the parameter, feeding requirements and verification criteria.
- **Non-failure scenarios** - the controller operates as designed and the outcome is still unsafe in context, with nothing to annunciate. STPA produces this category systematically; the failure-based analysis does not naturally reach it.
- **Reaches across system boundaries** - one control structure for three deceleration means gave a candidate aircraft-level failure condition (AFHA-6), a cross-system sequencing constraint (C-3C), an unstated assumption (ASM-1) and two candidate functions the decomposition had not captured.
- **Interdisciplinary by design -** pilots, systems engineers and human factors specialists each strengthen the scenario set, where the conventional assessment is sometimes produced by one discipline working alone.

Traceability: one thread from loss to ARP artefact

STPA artefact	This example	ARP process link
Loss	L-1: Loss of life	Failure condition effects (AFHA)
Hazard	H-2: Aircraft overruns the runway [L-1, L-2, L-3, L-4]	AFHA failure condition identification
Safety constraint	SC-2: Aircraft must not overrun the runway [H-2]	PASA architectural allocation
Control structure	Thrust reverse controller with WoW feedback	PASA / PSSA functional decomposition (cf. ARP4761A Appendix D.3)
UCA	UCA-3A: deploys thrust reverser(s) too late after touchdown such that insufficient runway remains [H-2]	(SFHA) FC-4: Delayed thrust reverser(s) deployment (landing / RTO)
Safety constraint from UCA	C-3A: deploy within a defined maximum time after touchdown such that adequate deceleration is achieved within the runway available	Requirements derivation (ARP4754B §5); verification objective (DO-178C)
Loss scenario, Class 1	Safety delay too long for the context; WoW voting delays the timer; autobrake timer coordination; software timer error	PSSA fault-tree contributors; SSA verification requirements; DO-178C timing verification objectives
Loss scenario, Class 2	Misleading WoW feedback: the controller believes the aircraft is airborne	PSSA fault-tree contributors for FC-1; human factors assessment input; WoW debounce requirements
Loss scenario, Class 3	Deploy command interrupted in the control path, not executed by the actuation system	PSSA / SSA fault tree for FC-2
Loss scenario, Class 4	Actuator fails to deploy after a correct command (hydraulic loss, binding, ice)	PSSA / SSA fault-tree contributors for FC-1 / FC-2; end-to-end actuation requirements

> **A second thread runs through the feedback path: H-2 → UCA-1 → C-1 → Class 2 scenario → PSSA fault-tree contributors for FC-1, plus the human-factors inputs. Every STPA artefact has a documented destination in ARP4754B / ARP4761A.**

What comes next within SAE

The direction indicated within S-18: subject to approval of the AIR, the next step would be an STPA appendix to ARP4761 and an STPA reference in ARP4754.

What that step would mean

- STPA guidance would move from a standalone informational report into **the recommended practices that most civil programmes start from**
- **Substantially more uptake and attention for STPA** across the industry, including organisations that adopt methods once the ARPs reference them
- **A recognised route** for applicants already using STPA to present that work within their certification framework

Where this leaves us

What we are asking of this room

- **Read it when it is published.** Watch the SAE S-18 page for the release once the AIR completes re-ballot.
- **Apply it alongside the ARPs.** Civil, military derivative or novel programmes; where the ARPs are tailored to the context, the guidance is written to be tailored with them.
- **Feed the experience back through S-18.** The appendix step depends on application experience: what worked, what needs extending, where the mapping ran out.
- **Contribute public examples.** Programmes outside the ARP4761 context are applying STPA with almost nothing published; the next guidance needs that record.
- **Contribute to S-18.** We are always in need of individuals that are willing to support and engage the work of the volunteer SAE committees.

S-18 is a volunteer committee, and it is open to the people in this room – please scan the QR code to join us to work on industry shaping topics like AIR6913.

Thank you

Special thanks to the other significant collaborators for AIR6913:

John Thomas
Kyle Ryan
Rodrigo Rose
Philip Adrian
Leo Rodrigues

And to the wider participants of the SAE S-18 committee for their continued contributions to the development of AIR6913.