System-Theoretic Process Analysis for Flight Test Safety

2 May 2023

Lt Col Sarah “Pancho” Summers

Lt Col Daniel “Mirf” Montes

Outline – Day 1 Introduction

- Systems Thinking Background
- STPA Basics
- UAV Example
- Test Example
- Takeaways & Risk Discussion

Lt Col Sarah “Pancho” Summers

How I got into STPA

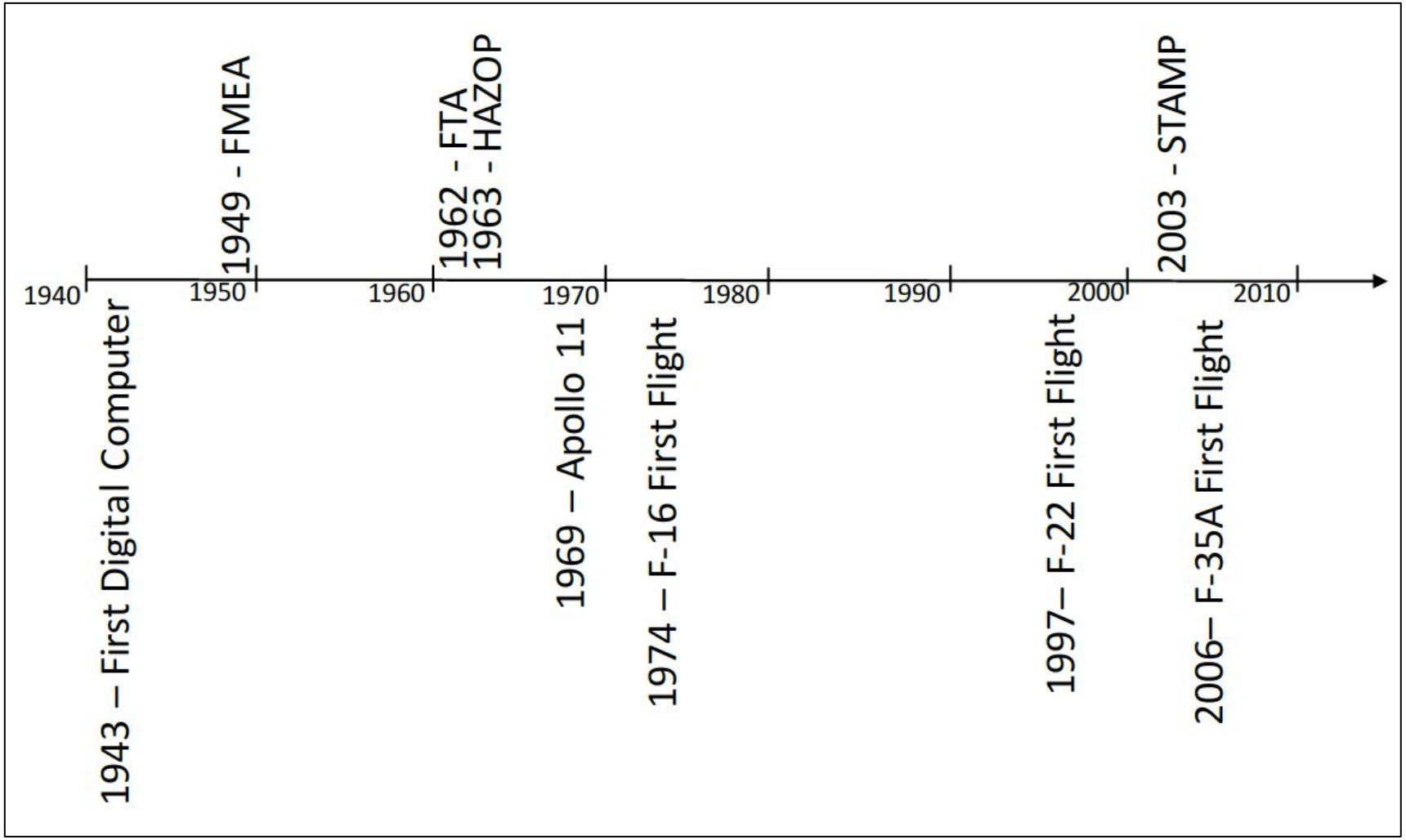
- My safety background:
  - As a Mx Officer – attended JEMIC, investigated small incidents such as engine FOD
  - In AFRL: investigated ~6 sUAS mishaps
  - In AFTC: TPS Grad, FSO, Project Safety Lead, Squadron Operations Officer
- Issues that I saw:
  - Current safety methods: Chain of Events based--don’t fully capture systemic concerns
  - We “blame the operator” vs fix the design (nor fix what leads to dangerous design)
  - Test safety planning based on knowledge/judgement from previous projects
- I took Prof Leveson’s STPA class and the lightbulb turned ON!

What gets me up every day: ensuring weapon systems we deliver to the warfighter will allow him/her to do the mission and come home safely.

SYSTEMS THINKING

Why do we need a more holistic approach?

- Traditional hazard analyses were developed before Man landed on the moon
- They are reliability-based
  - Work great for simple electromechanical systems
  - Not designed for complex software or human integrated systems
- Many aerospace accidents not due to component failure – but rather system behavior



Types of Problems

Well-structured Puzzles

- There is a standard answer

Partially-structured Design challenge

- Allocate functions & techniques against a known requirement – many answers (trade offs)

Ill-structured Messes, Dilemmas, “Wicked”

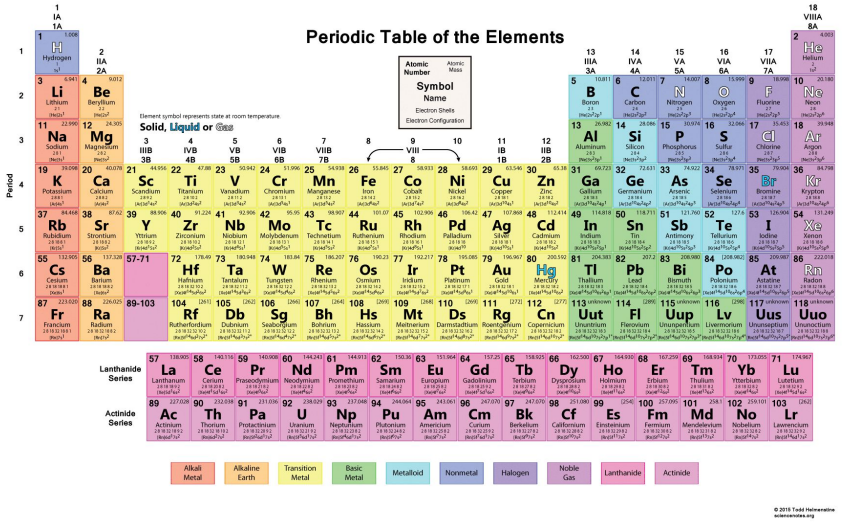
- Ambiguous goals
- Uncertainty / incomplete information
- Require collaboration and synthesis

Complexity

The interaction of systems' parts with each other in multiple ways that culminate in a higher order of emergence greater than the sum of its parts

Classic ways to deal with complexity

- Reductionism:
  - The whole may be explained as the sum of its parts
  - Divide, explain, predict phenomena at simpler levels



- Statistics:
  - The system is a structureless mass with inputs and outputs
  - Components are sufficiently regular and random in their behavior
  - Law of Large Numbers: quantify the distribution of the output and relate to the inputs

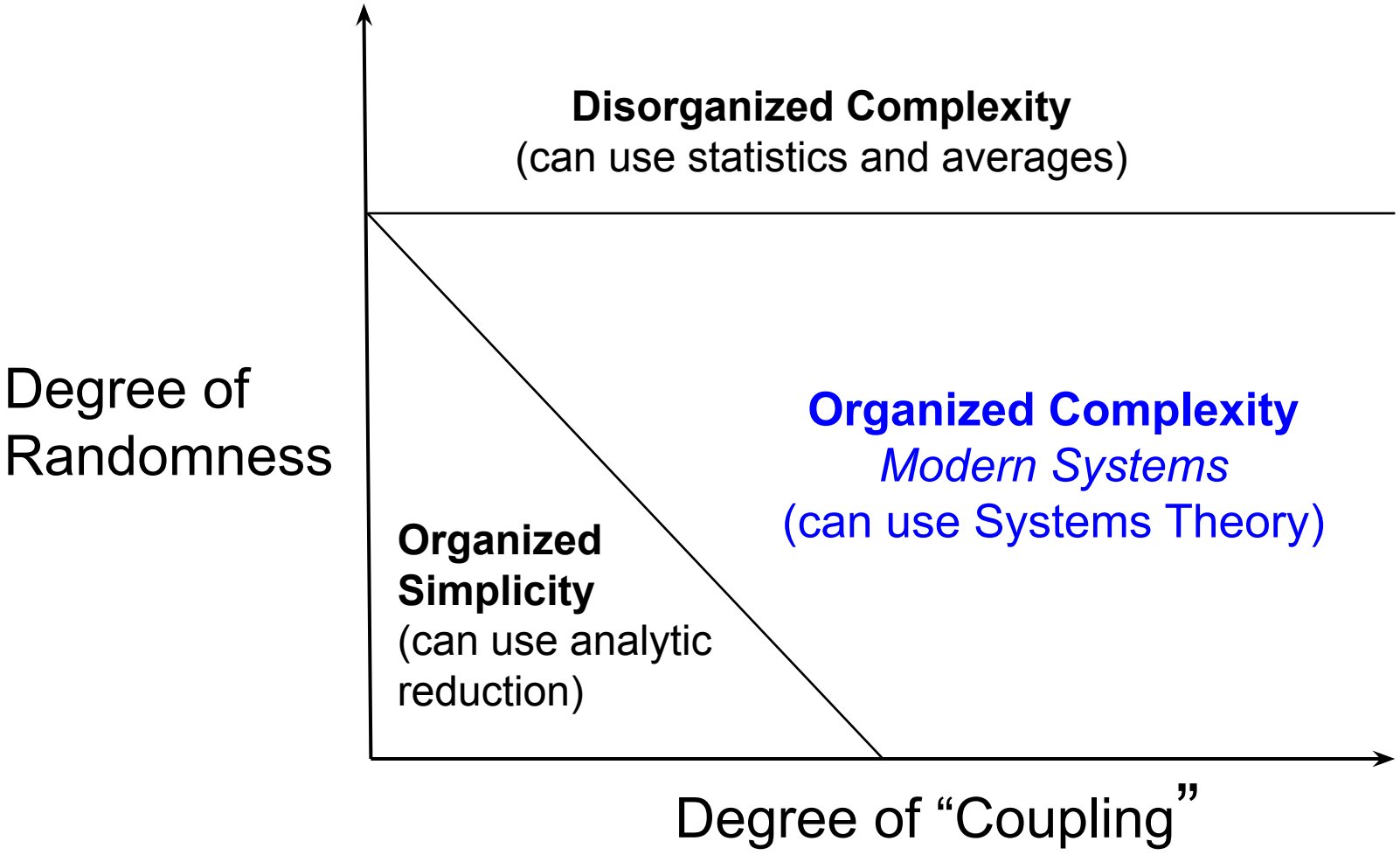
Human/software intensive systems

- Too intricate for complete physical analysis
  - Separation into (interacting) subsystems distorts results
  - System component behavior is tightly coupled
  - Decision agency: adaptable, interdependent, heterogeneous
  - The most important properties are emergent
- Too organized for statistics
  - Too much underlying structure that distorts the statistics
  - Variables receive feedback and adapt over time

[Statistics] is the core of knowledge... tells you if something is true, false, or merely anecdotal; it is the "logic of science"; it is the instrument of risk-taking; it is the applied tools of epistemology... but... let's not be suckers. The problem is much more complicated than it seems to the casual, mechanistic user who picked it up in graduate school. Statistics can fool you (Taleb, 2008)

Modern technology and society have become so complex that the traditional branches of technology are no longer sufficient; approaches of a holistic or systems–and generalist and interdisciplinary–nature became necessary. (Klir, 1972)

Methodologies (Weinberg, 1975)

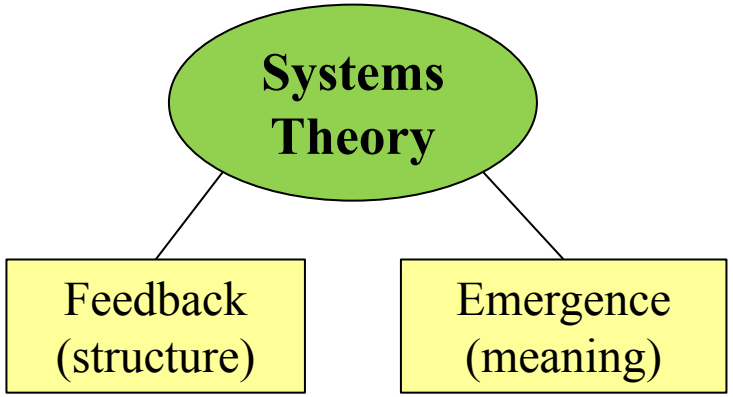


These are certainly complex problems. But they are not problems of disorganized complexity, to which statistical methods hold the key. They are problems which involve dealing simultaneously with a sizable number of factors which are interrelated into an organic whole.
(Jacobs, 1992)

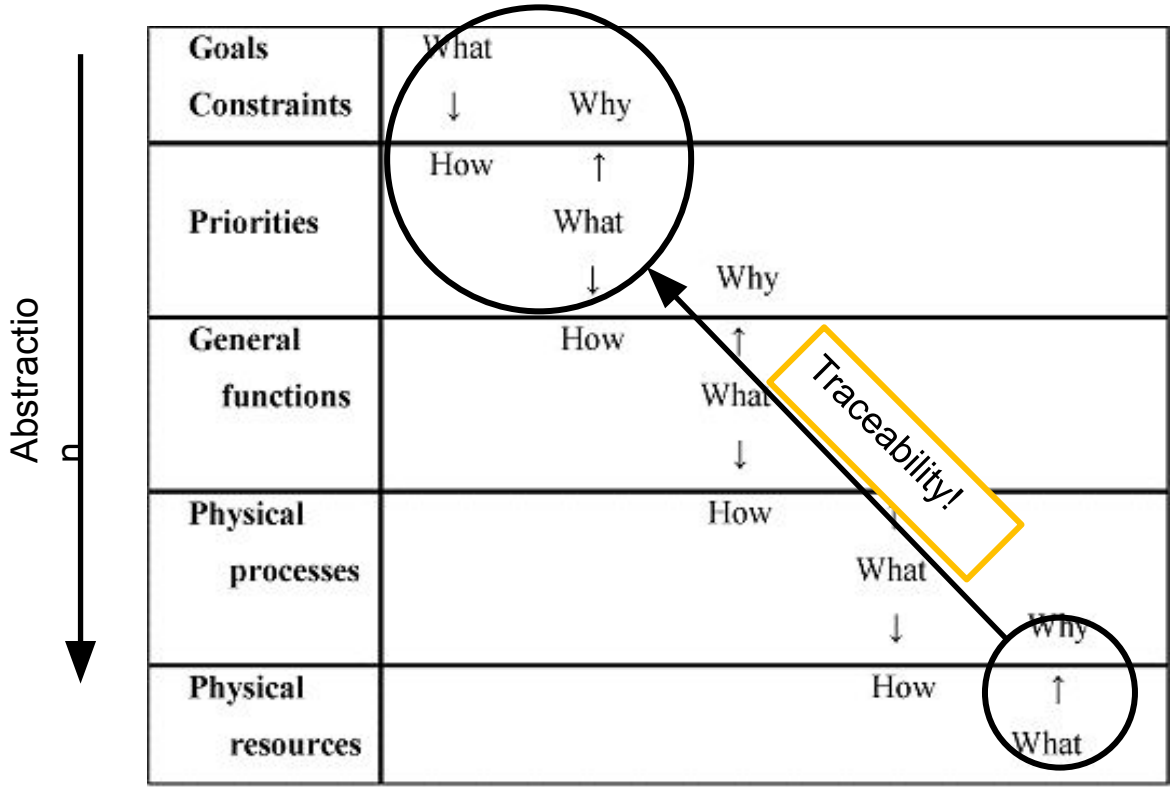
Systems Theory

- A meta-discipline and philosophy for problem solving
  - Complements the scientific method (predict-test-validate), which aims to acquire testable and refutable knowledge of the universe

- For the model-based approach, focus on feedback and emergence



Emergence (Meaning)



- System behaviors emerge from subsystem interactions and interdependencies
- What matters (the why) is different at each level
- Lower level components do not satisfy mission level requirements (i.e., ‘safety’, ‘security’)

Tracing through levels of abstraction can assist in system design validation

Structure (Feedback)

- Dynamic Systems subjected to environment disturbance
- Goal-Seeking Behaviors
  - Decisions
  - Coordination
  - Control

Authority
Responsibility
Accountability

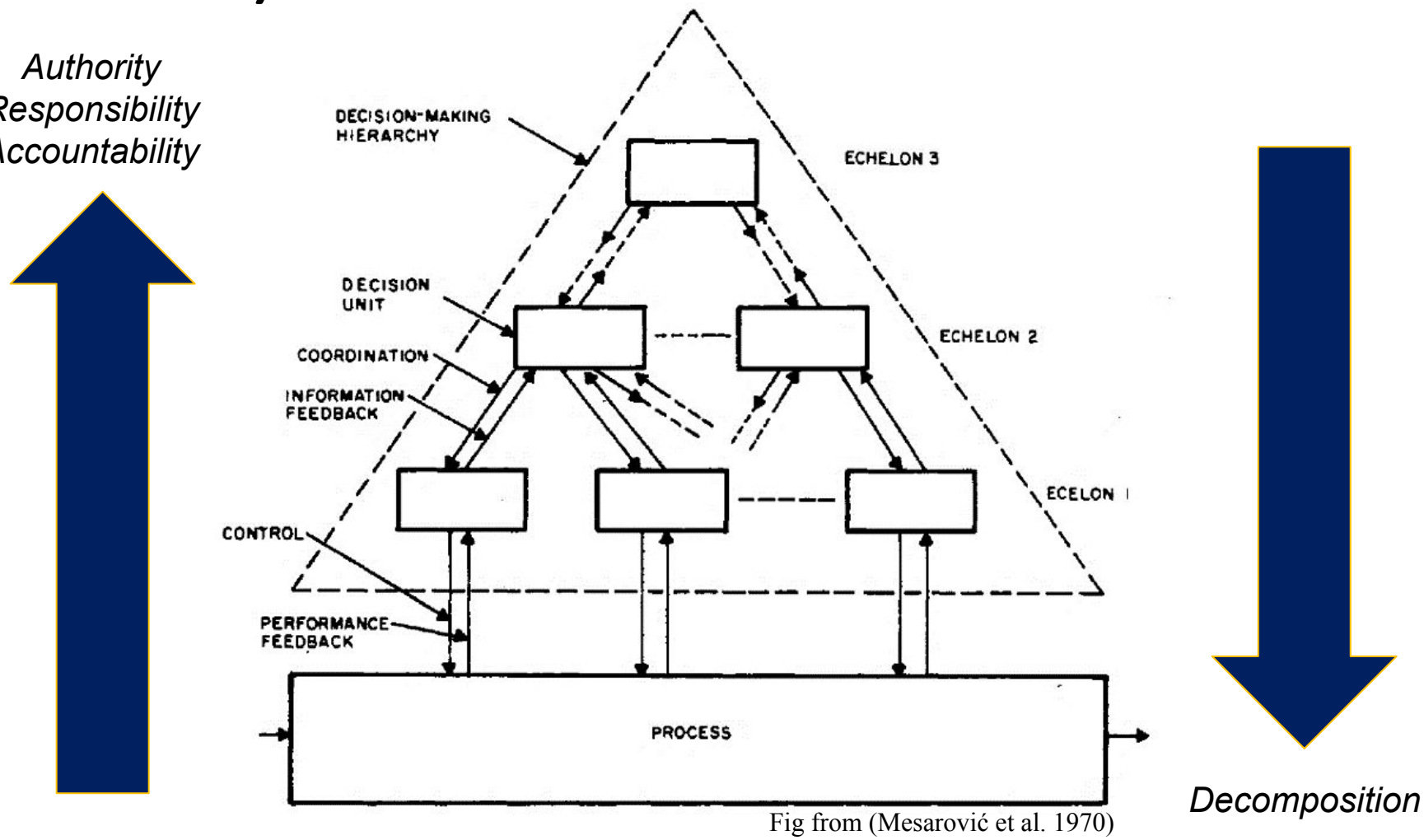
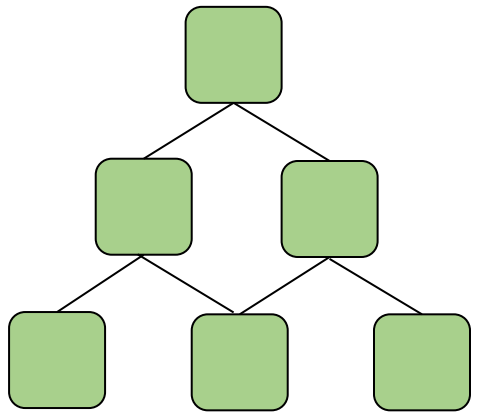


Fig from (Mesarović et al. 1970)

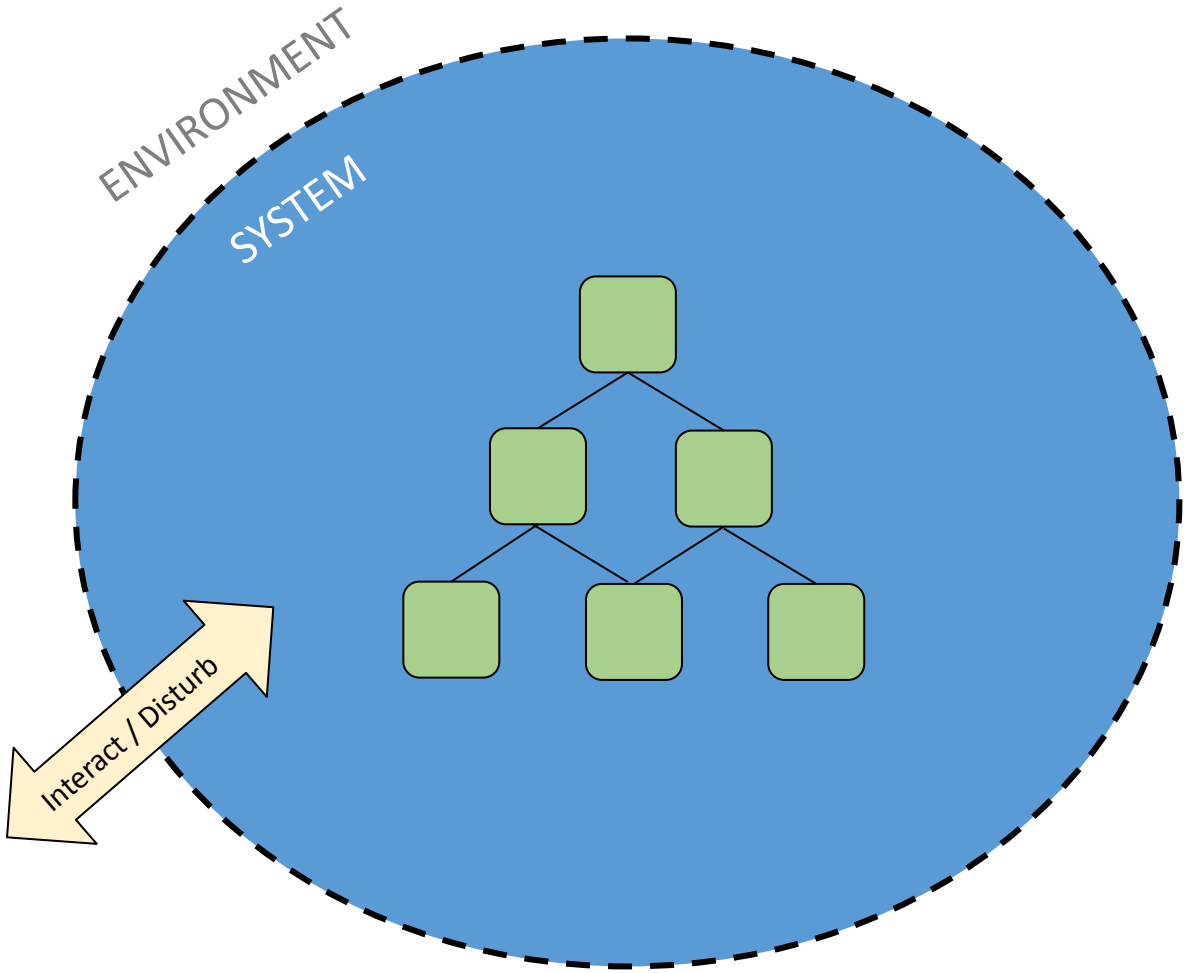
What would an orbital warfare engagement look like?

Feedback

- The transmission of information about a process to the process-regulator (controller)
- Staple of cybernetics (study of control and coordination within systems)
- The underlying notion for organized activity!
- System is functionally structured to allow for decision making based on information of lower layer



Ordered System



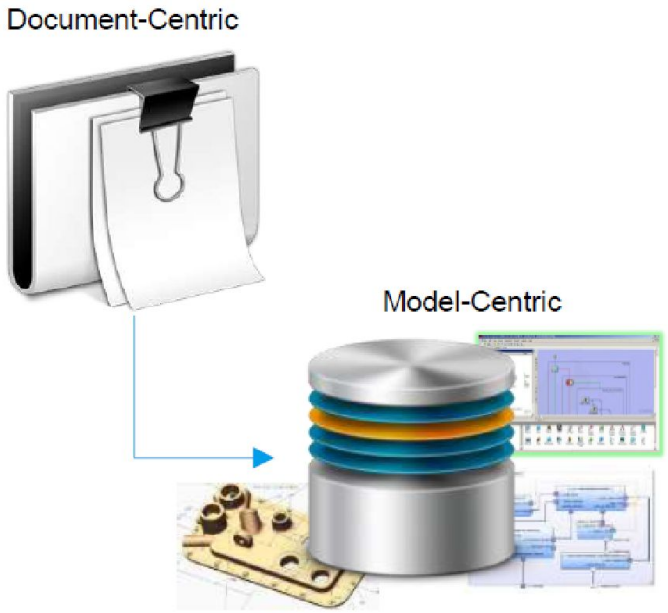
Higher-level concerns

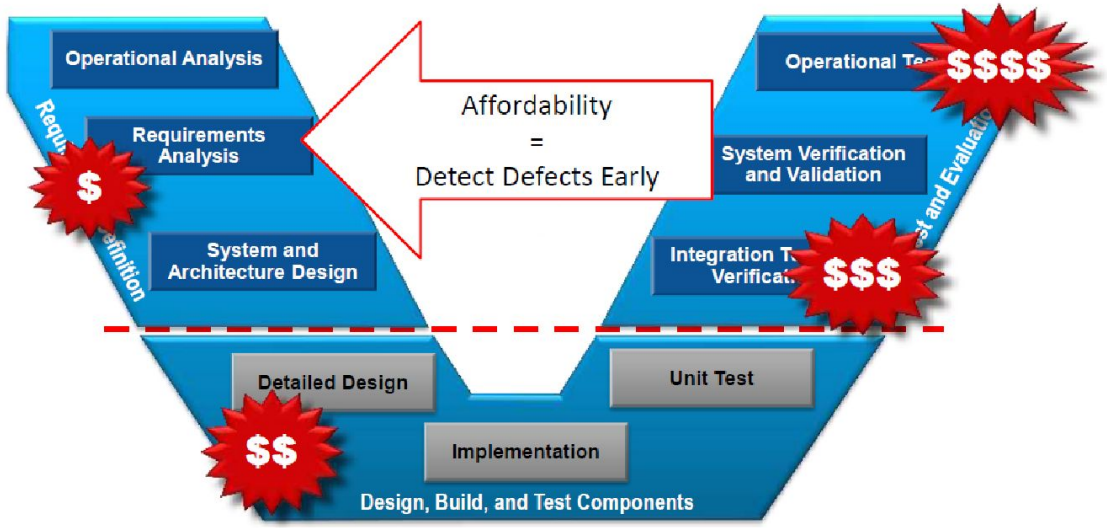
Lower-level behavior

Model-Based Systems Engineering (MBSE)

The formalized application of modeling to support system requirements, design, analysis, verification, and validation activities beginning in the conceptual design phase and continuing throughout development and later life cycle phases.

(INCOSE, 2007)





Abstraction

Structured product of the conscious mind

- Thoughts and concepts that serve a human purpose
- Math, music, art, language, other representations
- Useful for:
- Generalizing theories based on perceptions
- Representing ideas in a general form

Abstractions help us deal with complexity!

“Model”: powerful as a verb

The use of abstraction to interpret your surroundings and create a representation of the real world

- Represent curiosities and perceptions of the world
- Understand human involvement within systems
- Guide relationships
- Communicate system design, evaluation, training, operation
- All the above for sociotechnical systems

“Everything should be made as simple as possible, but not simpler”
– Albert Einstein

System-Theoretic Process Analysis (STPA)

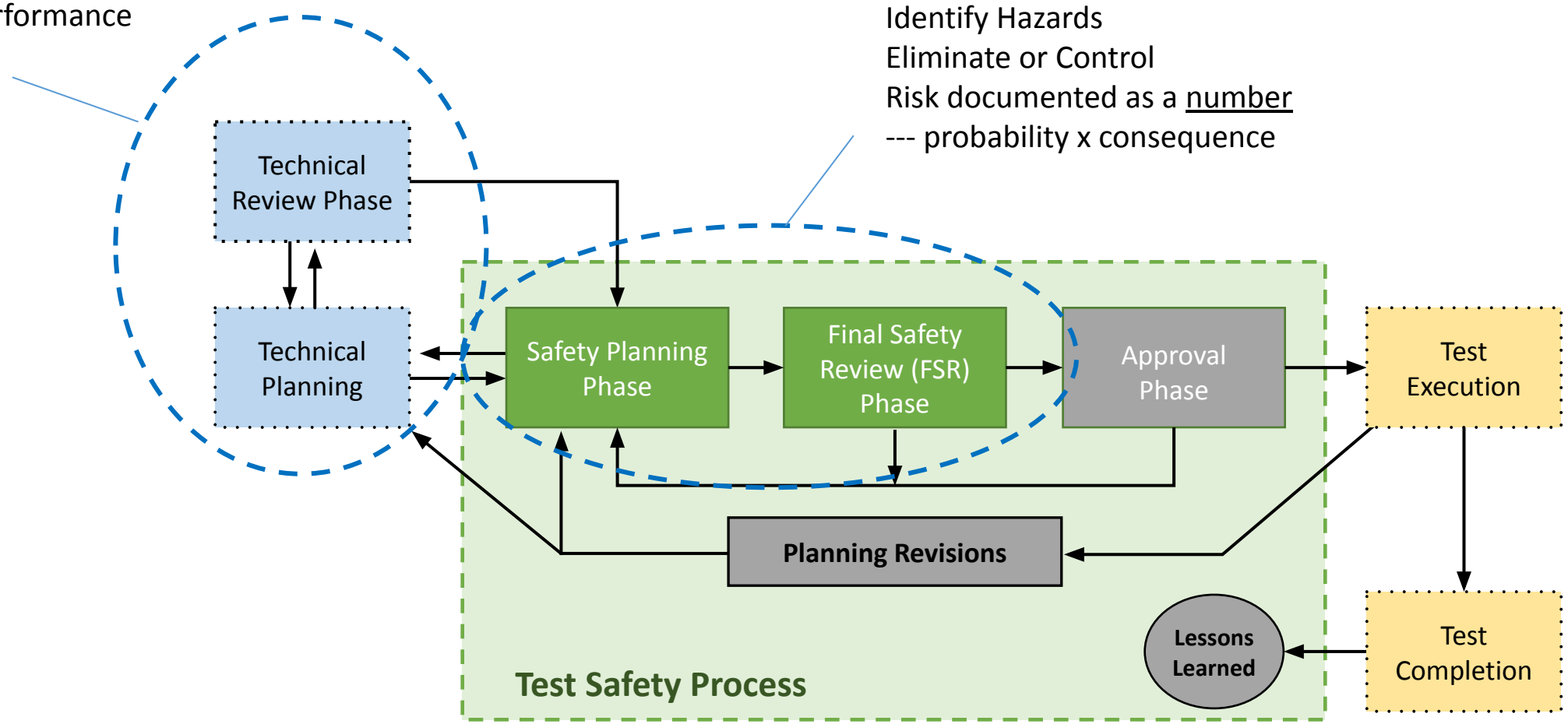
Systems Theoretic Process Analysis (STPA) Academic Discipline @MIT

- Type of MBSE – no ‘software’ required
- Engineering/ops hazard analysis technique
  - Developed by Prof Nancy Leveson after career in software safety
  - Tackles emergent problems not detectable by component failure-based analyses (e.g. fault trees)
- Systems-engineering level of discipline
  - Ties everything back to the visual model
  - Not just a documentation-based method
- Same technique can be used for any emergent property
  - Safety, security, performance → mission protection!

Safety Management - Traditional

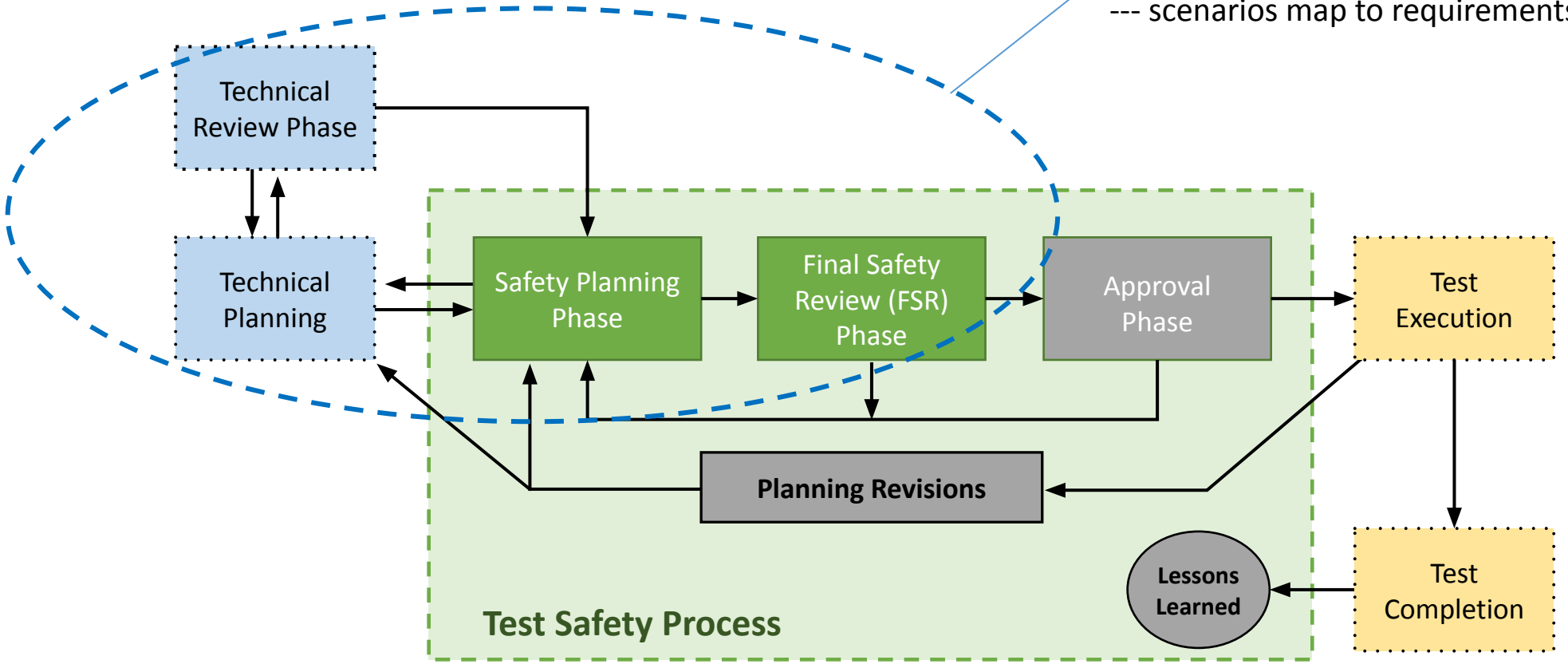
<u>Documentation Based:</u>
Technical objectives
Measures of Performance
Techniques

<u>Documentation Based:</u>
Identify Hazards
Eliminate or Control
Risk documented as a <u>number</u>
--- probability x consequence

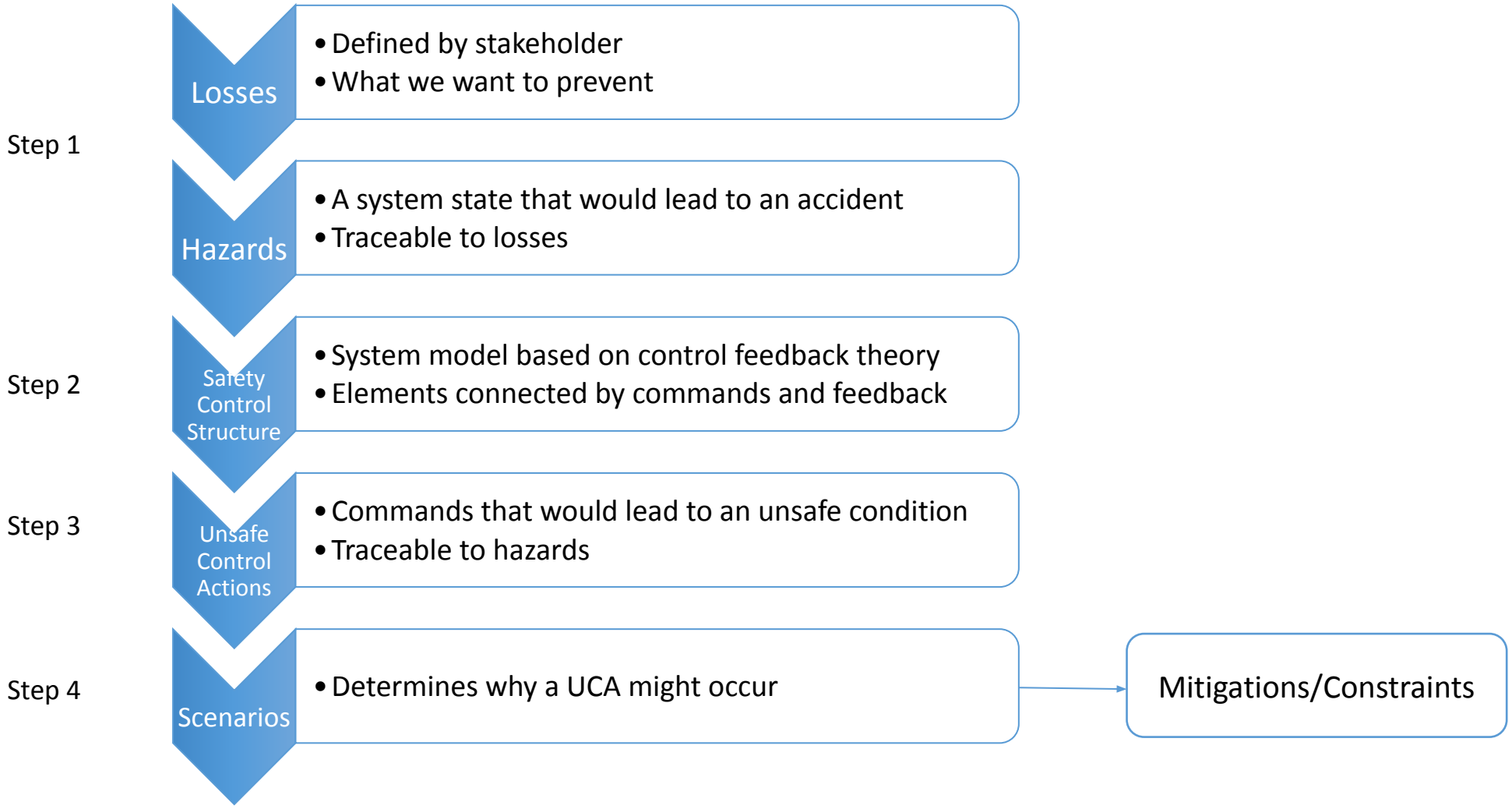


Safety Management - FMA

Model Based:
Any emergent property can be investigated
Requirements and hazards defined first
System model:
--- aligns the team's shared mental picture
--- forms common baseline for traceability
Identify scenarios behind hazards
Eliminate or Control
Risk documented as uncertainty
--- scenarios map to requirements

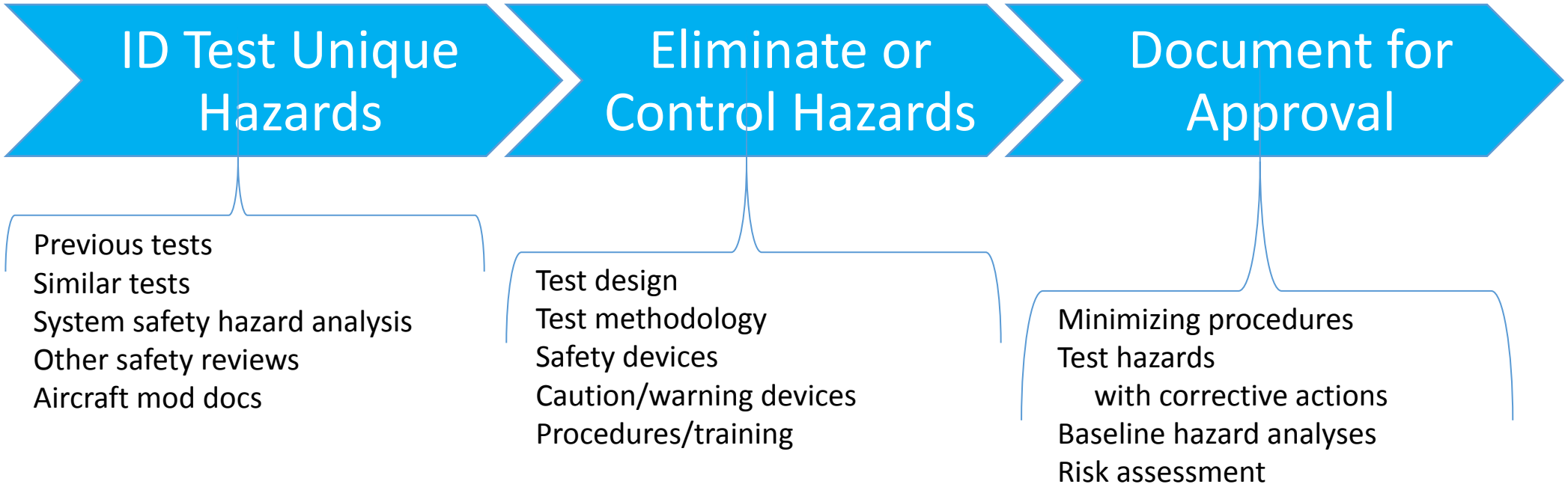


STPA Steps



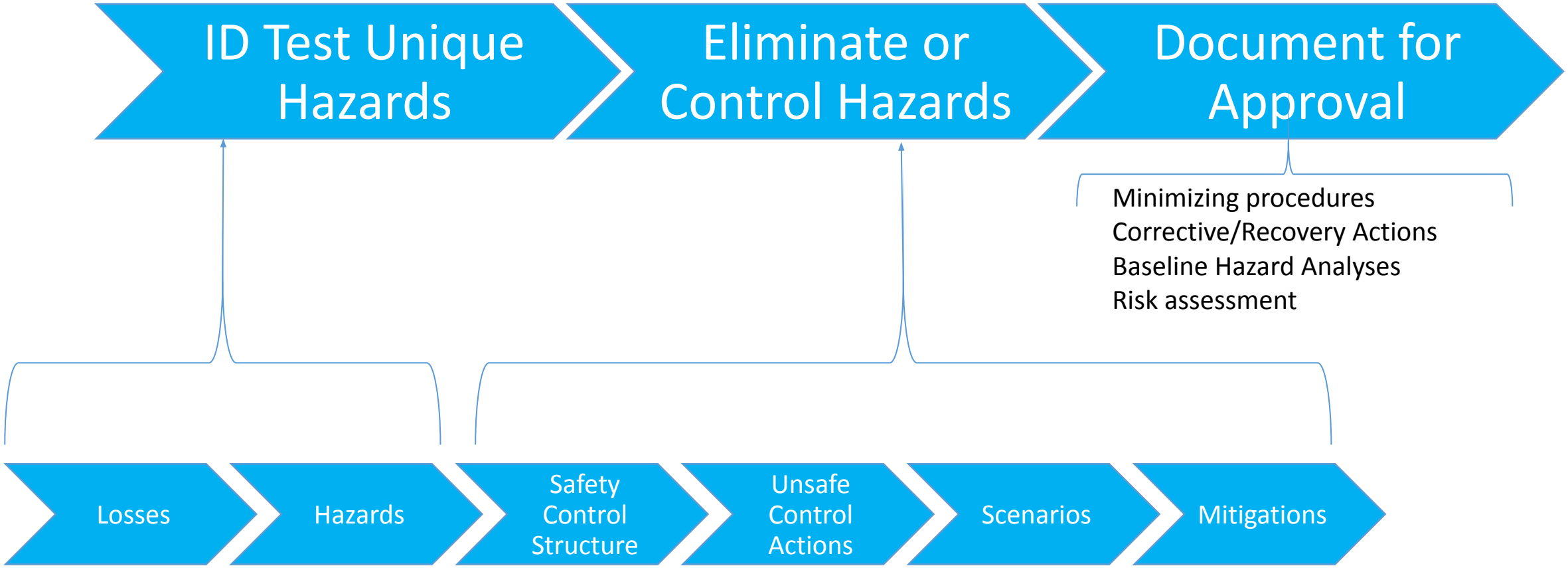
Note: Losses aren't just safety – can be mission, environmental, security, etc

Traditional Test Safety Planning



This method is effective for experienced teams with well known systems

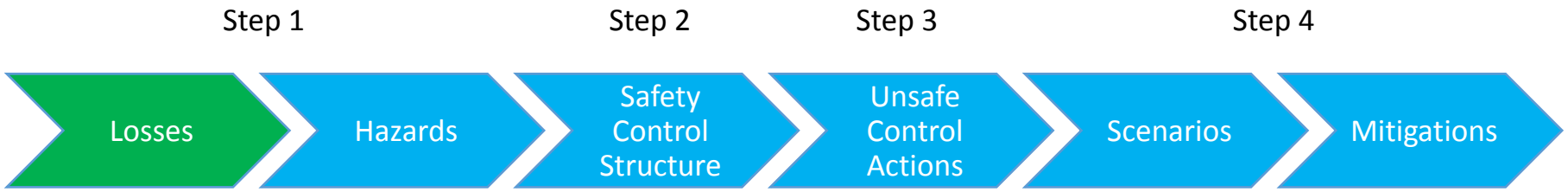
Test Safety Planning With STPA



Output of STPA can be translated to existing documentation processes

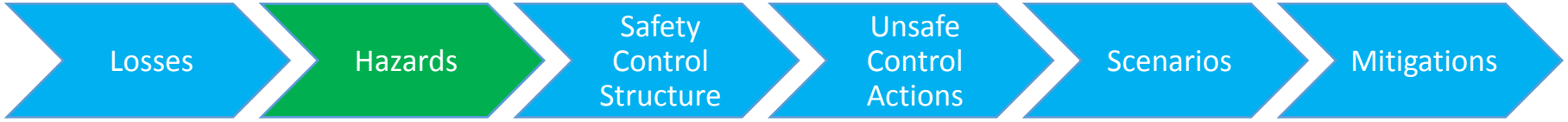
Losses (AKA Mishaps AKA Accidents)

- What we want to prevent
- For AFTC – losses are predefined:
  - Loss of life/injury to people
  - Loss of/damage to system under test
  - Loss of/damage to other infrastructure



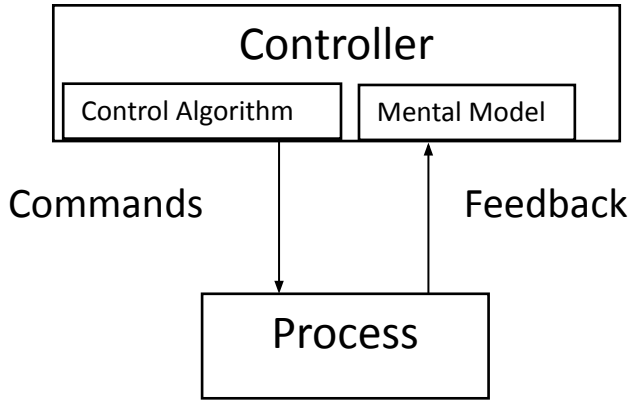
Hazards

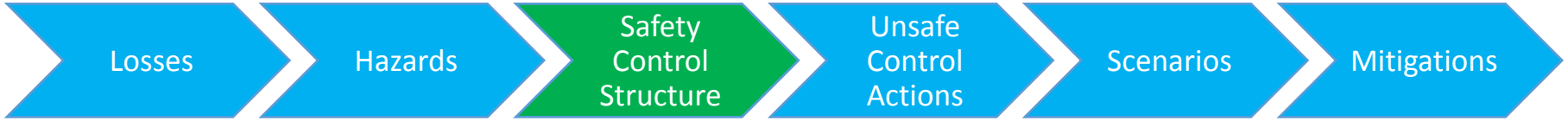
- **System-level** states or conditions that, combined with environmental factors, could lead to a loss
- Hazards trace back to losses
- Examples:
  - Controllable aircraft violates minimum separation distance to another air vehicle
  - Aircraft departs controlled flight
  - Weapon/store guides toward air vehicle or ground infrastructure other than intended target area



Functional Control Diagram

- STPA calls it the ‘Safety Control Structure’
- Backbone of the analysis
- Arranged in hierarchical manner
- Contains commands & system feedback
- May have human or automated controllers
- May have multiple controllers and processes
- Includes entire sociotechnical system
- Highest level of useful abstraction



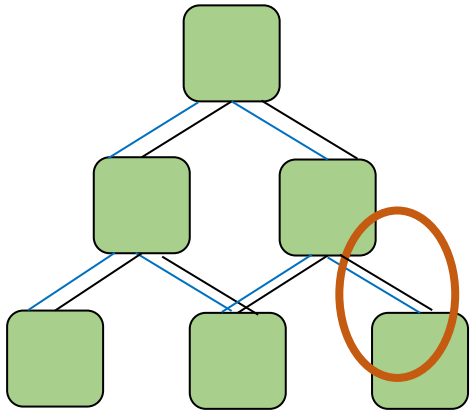


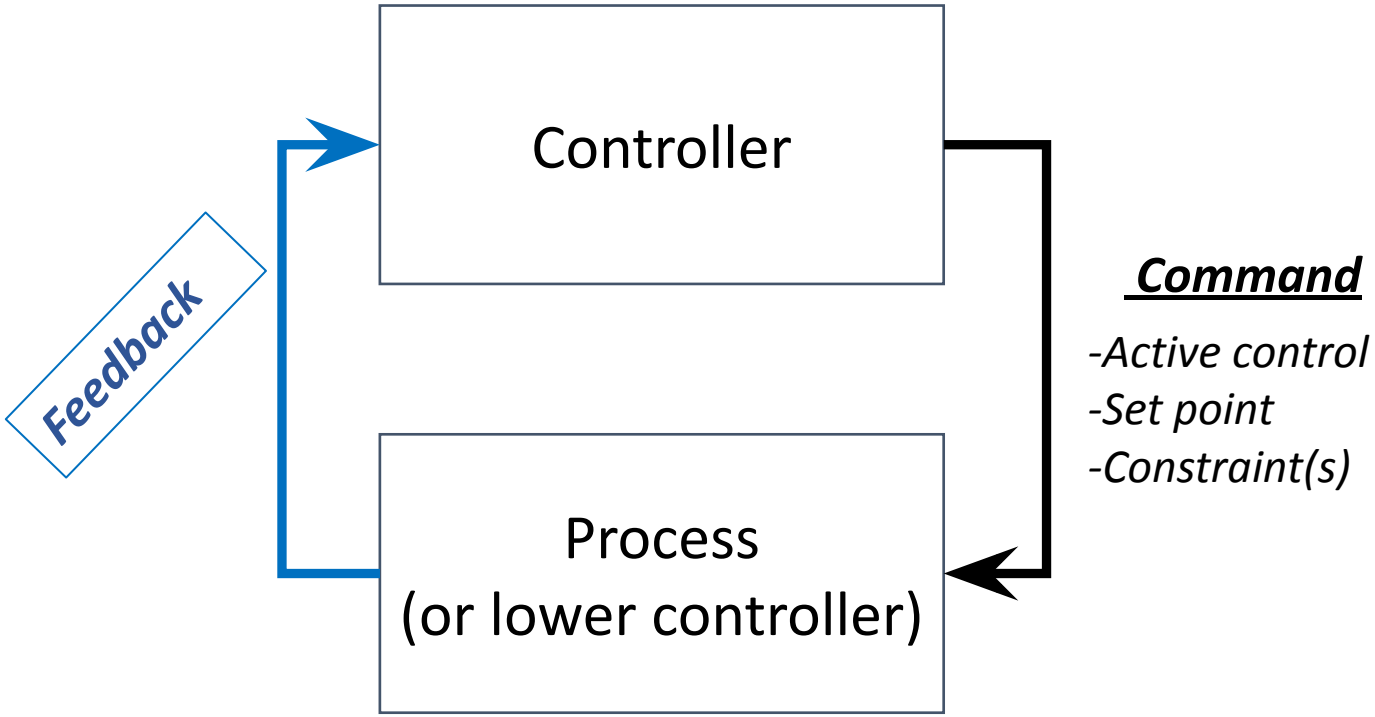
Architecting

Since all models are wrong the scientist cannot obtain a "correct" one by excessive elaboration. On the contrary following William of Occam he should seek an economical description of natural phenomena. Just as the ability to devise simple but evocative models is the signature of the great scientist so overelaboration and over-parameterization is often the mark of mediocrity..

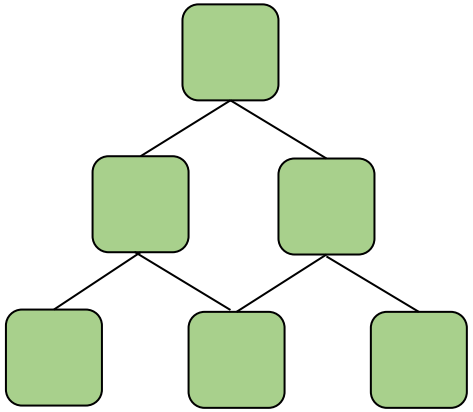
– Box (Journal of the American Statistical Association)

System-theoretic convention





Types of interfaces



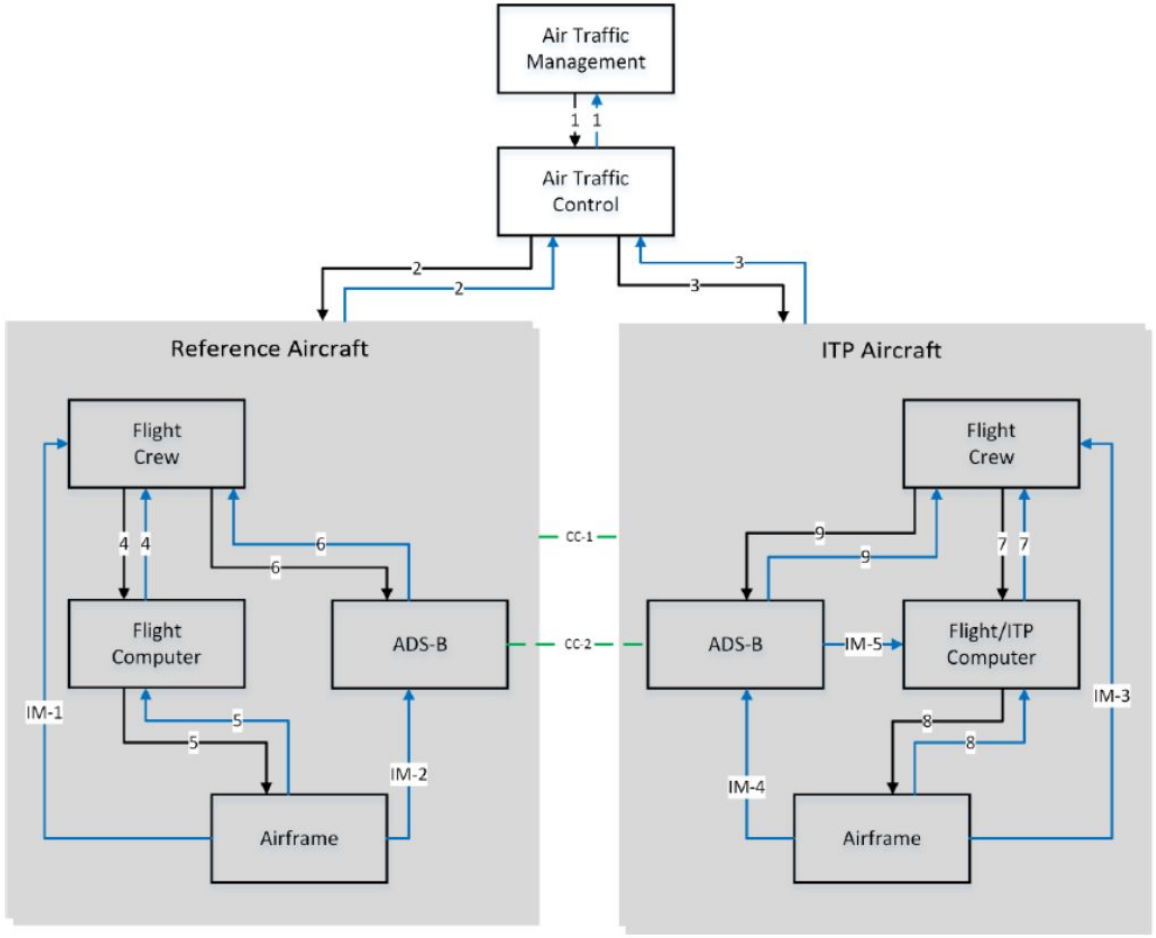
Medium	Operand	Process	Form?
Mechanical Link	Forces, Torques [N, Nm]	Force or Torque Transmitting	bolts, washers, rivets, spot welds…
Energy Flow	Work [J]	Electricity or Heat Transmitting	copper wires, microwaves, …
Mass Flow	Mass [kg]	Fluid, Gas or Solid Matter Transmitting	fuel lines, air ducts, exhaust pipes …
Information Flow	Bits [-]	Data or Command Transmitting	micro-switches, wireless RF, humans

Physical Dimension

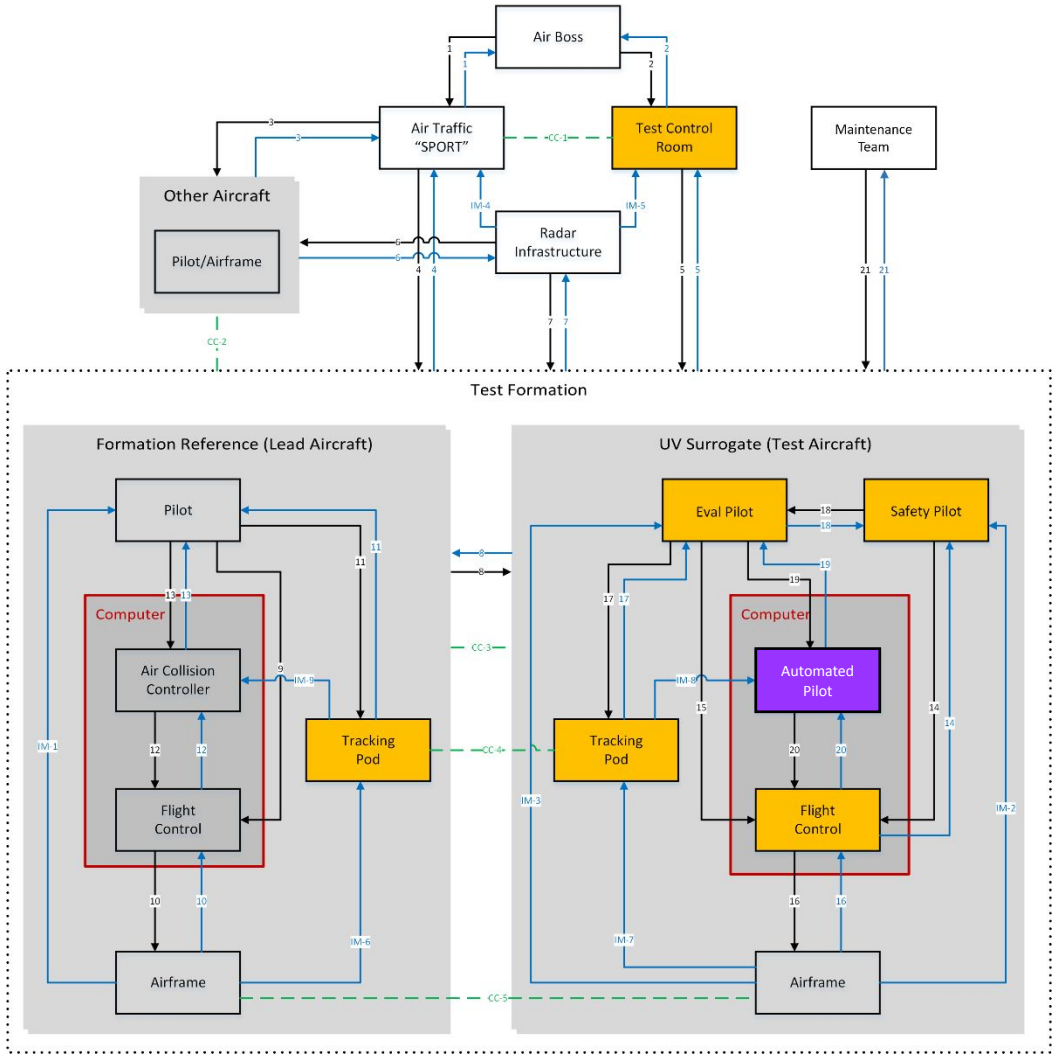
Network/Cognitive

Not covered well by traditional hazard analysis!

Sample architectures



Next Gen ATC – Transoceanic In Trail Procedure



DT of Loyal Wingman Algorithm

Air Defense

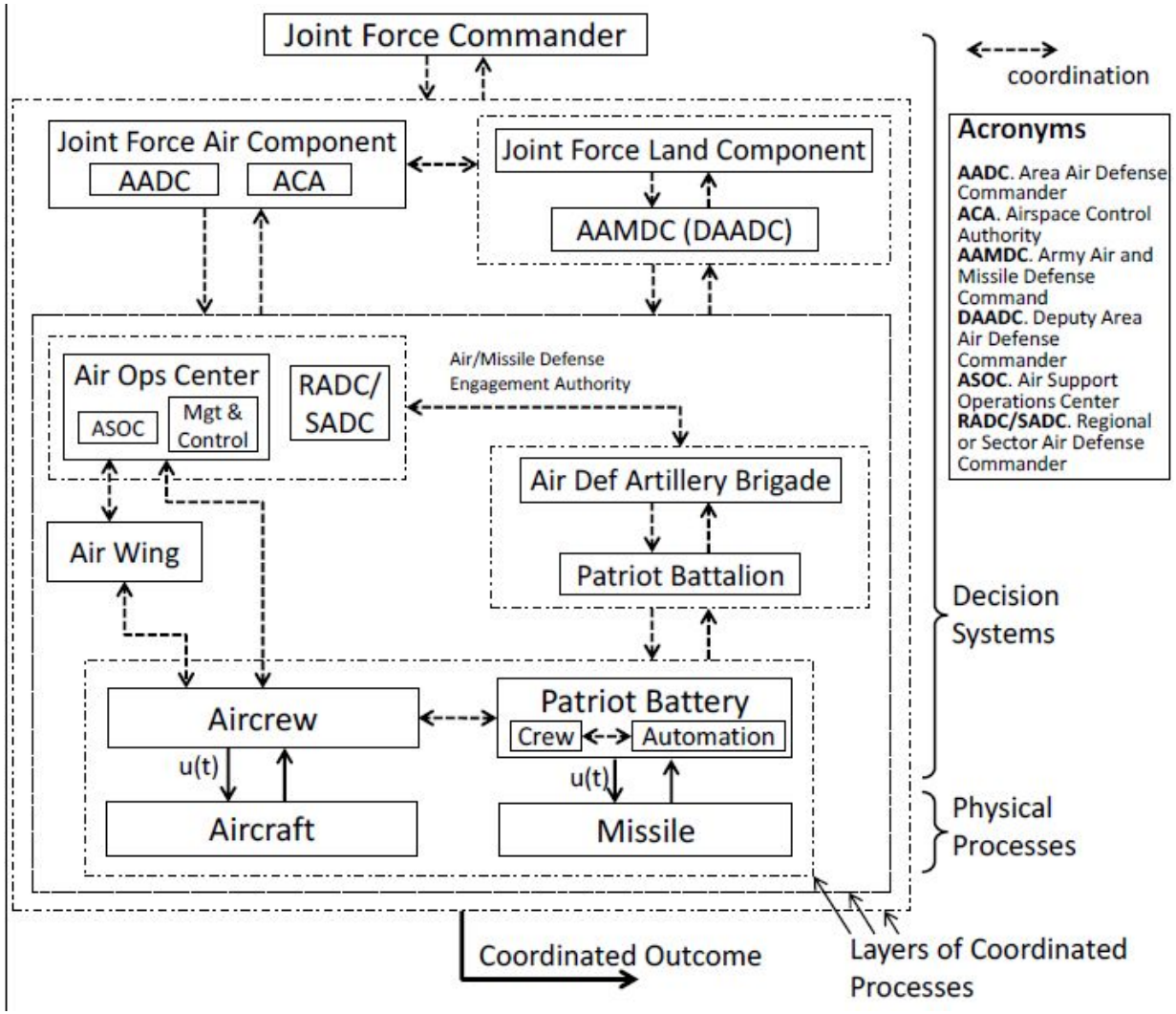
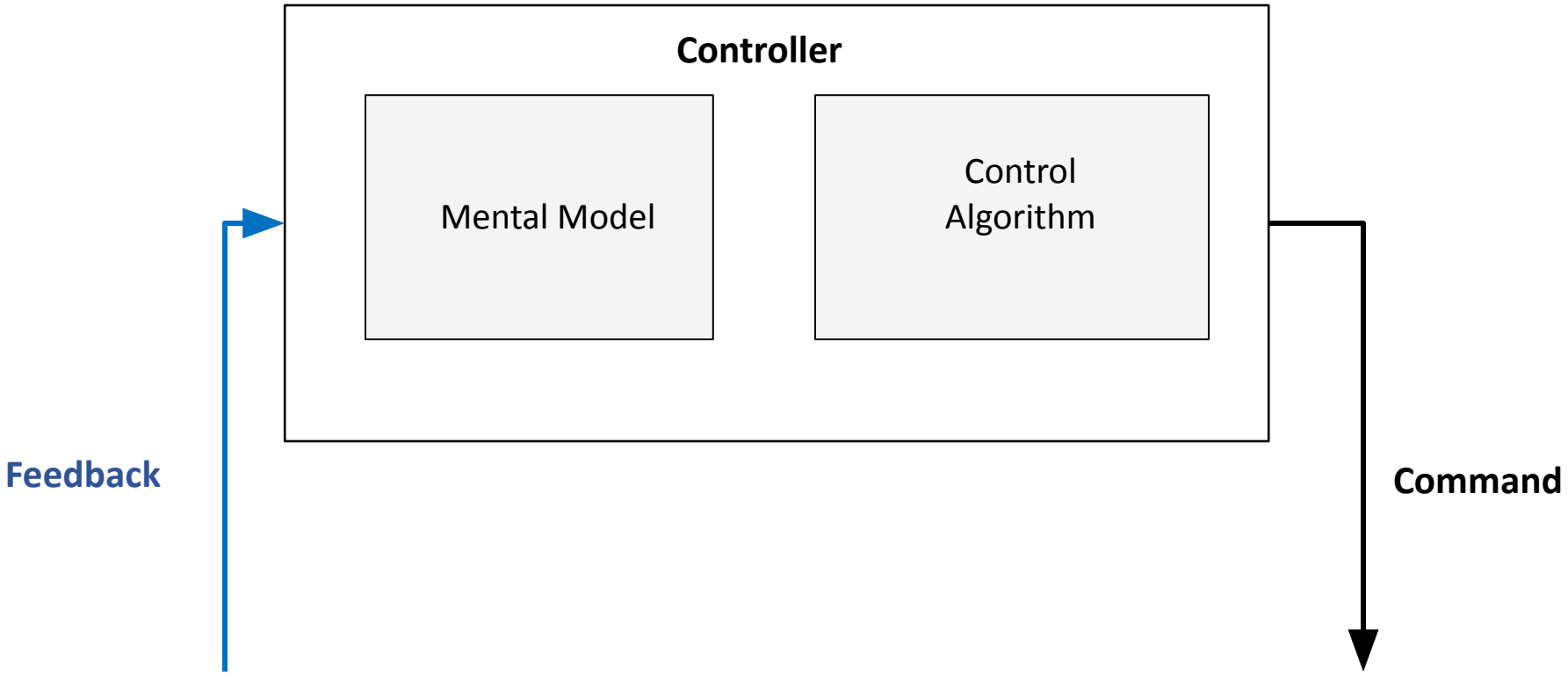


Fig. from Johnson, K.E. (2017)

Inside the Controller

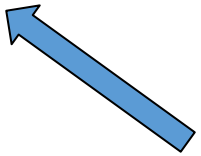


Mental Model

Controller's understanding (or 'belief') of the
real-time state(s) of the process(es) it is
monitoring and/or controlling

Human driving a car, example:

Speed, fuel quantity, obstacle distance, obstacle direction, obstacle closure, engine health…



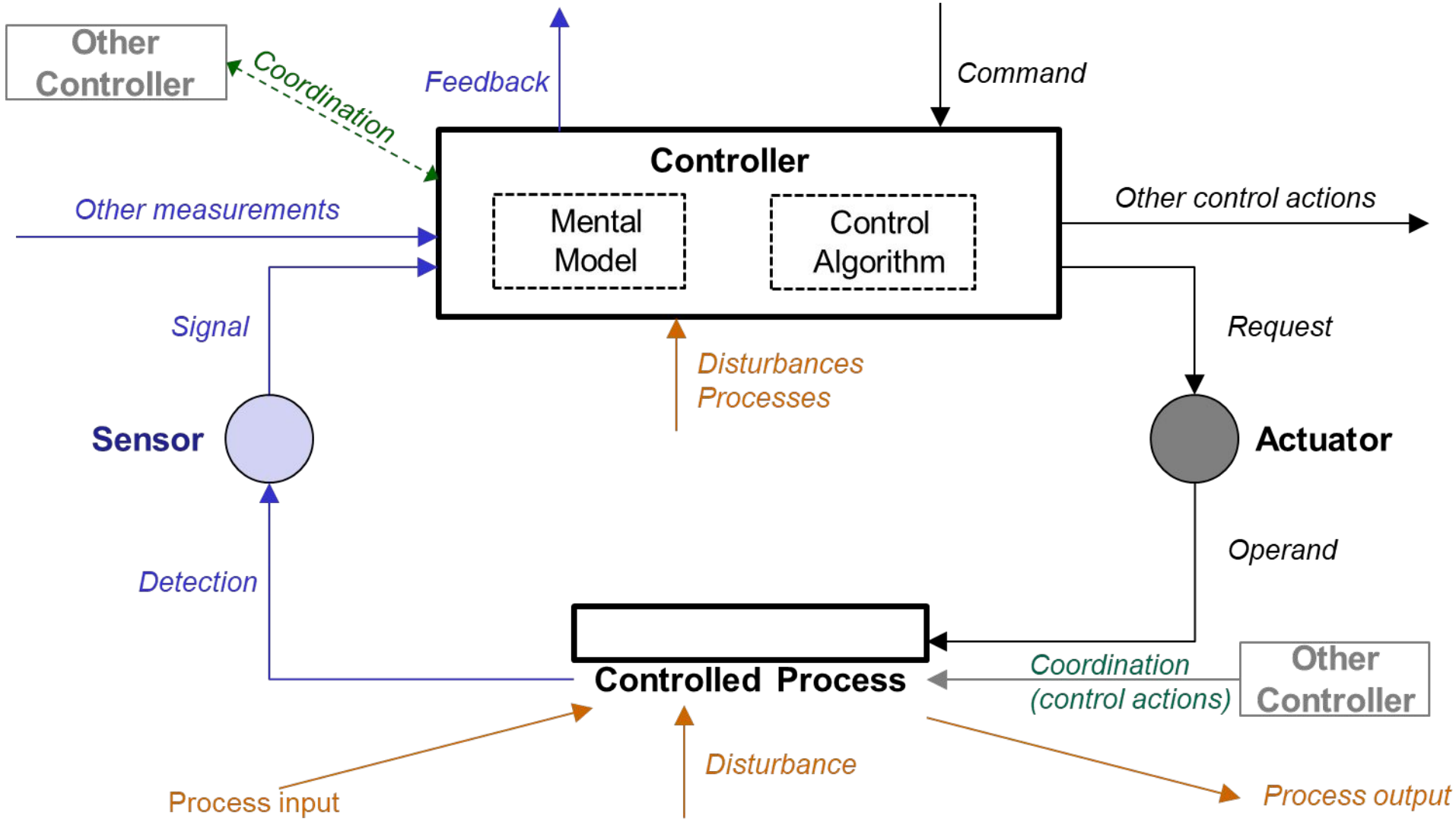
Mental Model's "Variables"

Control Algorithm

The set of rules and functions that enable a controller to make decisions about what actions to perform

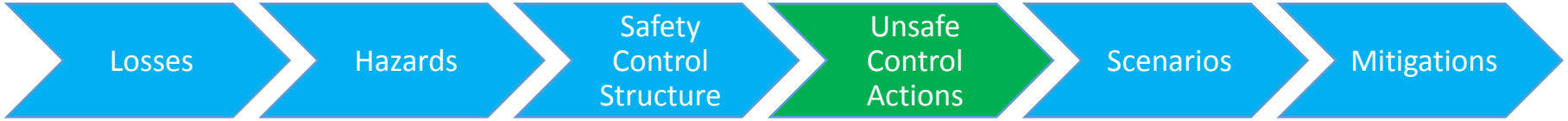
What is the 'input' to the control algorithm?

Additional Considerations



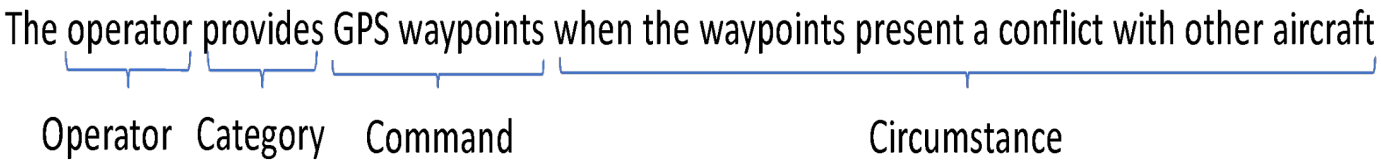
Unsafe Control Actions

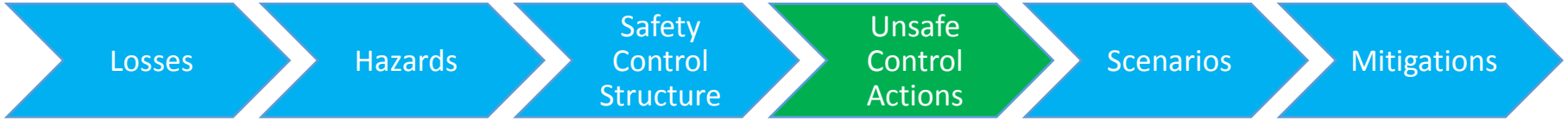
- When would the control actions in the FCD be unsafe?
- Another way to say this is what system property makes it unsafe?
- Example:
  - When is applying brakes while driving safe?
  - When is it unsafe?
- The properties that may make it unsafe are:
  - Location of vehicles relative to your vehicle
  - Road conditions (ice/water)
  - Others?



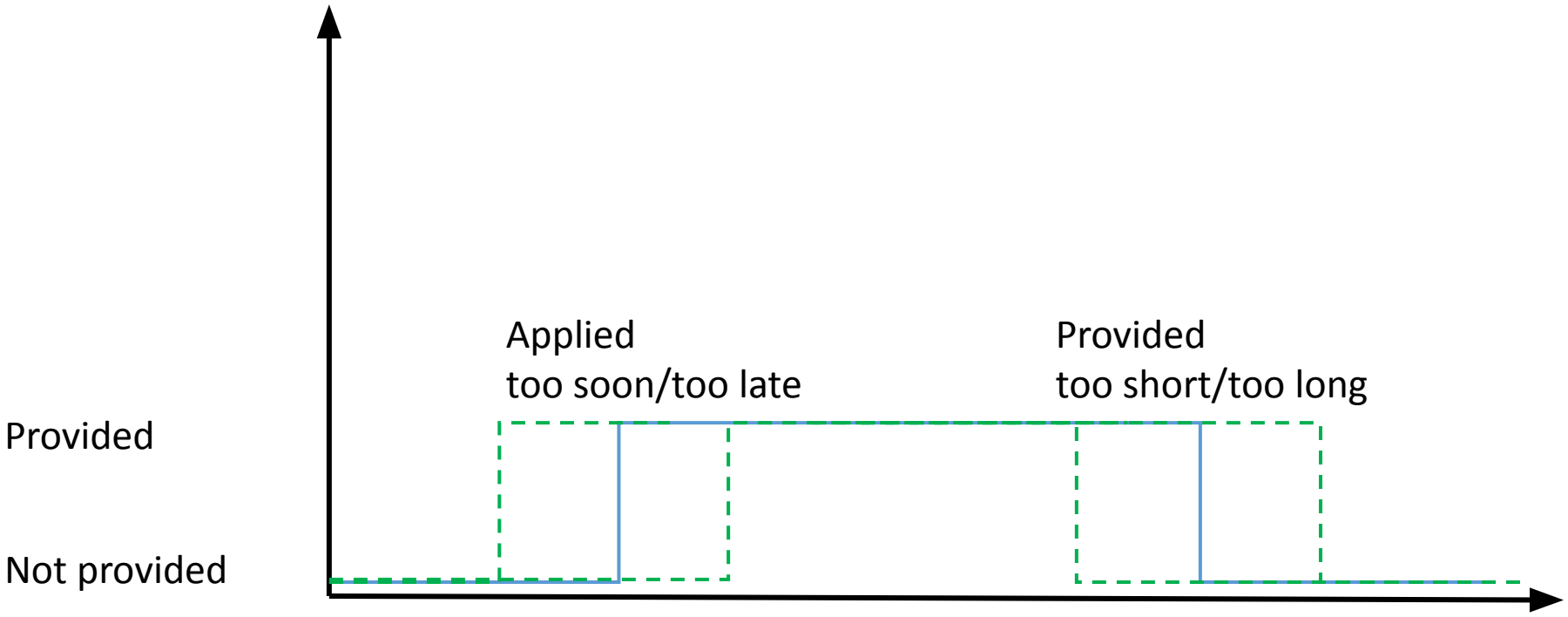
UCA Categories and Structure

- UCAs all fall into one of four categories:
  - Not Provided
  - Provided
  - Provided too soon/too late or out of order
  - Provided for too short or too long (non-discrete commands only)
- UCA has a specific structure



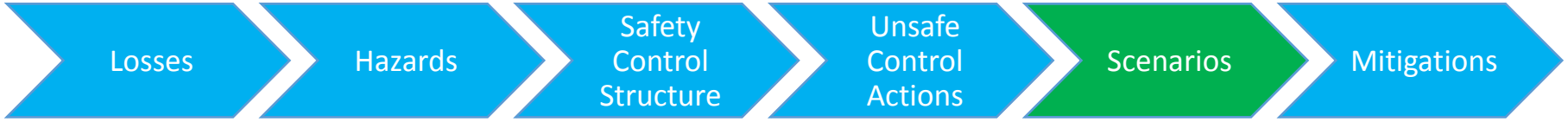


UCA Categories Cont.

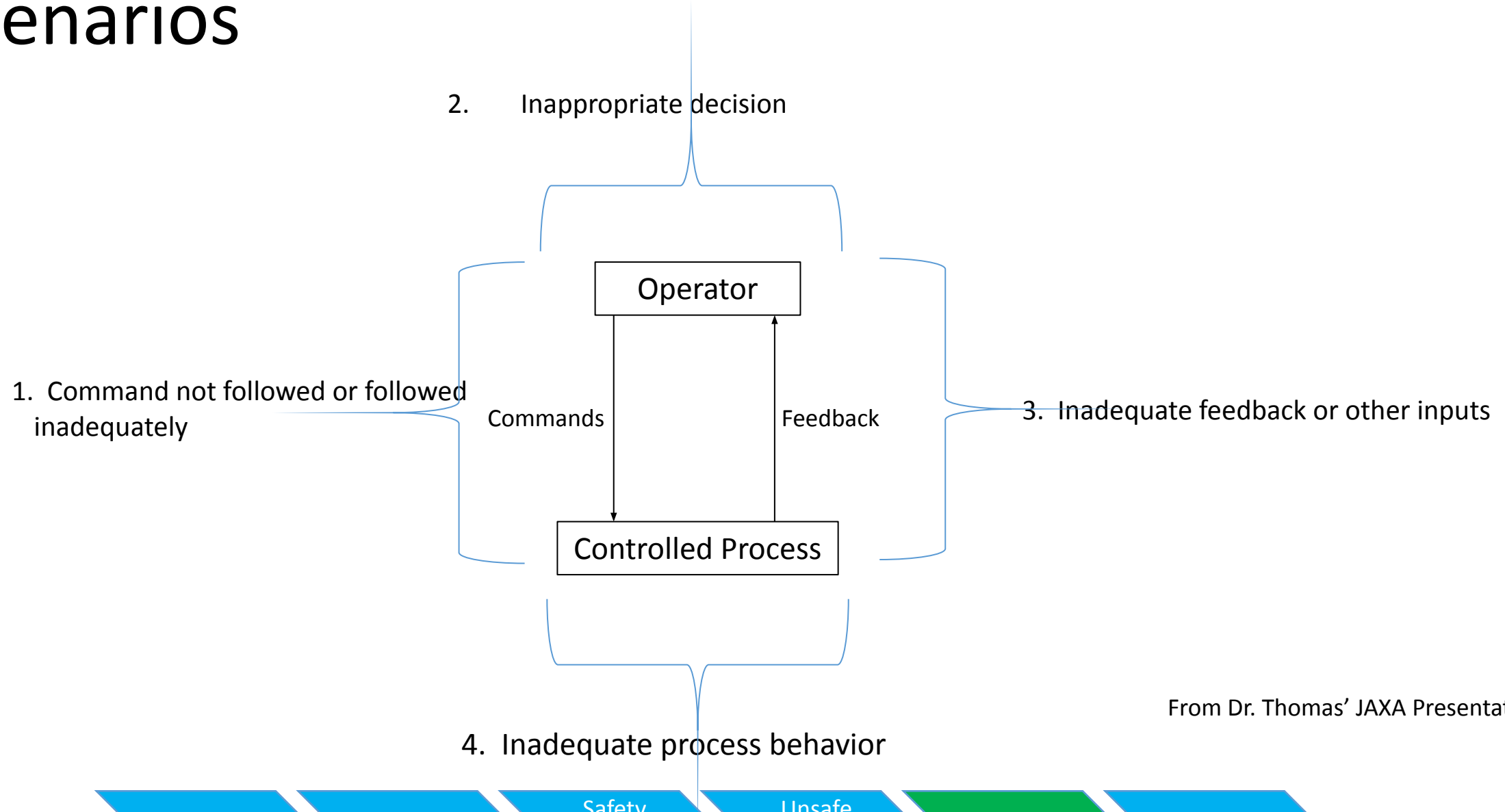


Scenarios

- Scenarios describe how the system got into the state that led to the hazard
- It's best to develop scenarios with multidisciplinary teams
  - Ops engineers, discipline engineers, operators, etc all have different experiences
  - Experiences drive how you will think about the UCA and associated scenarios
- Ex: Why did I stop applying brakes before the car in front began moving?
  - The car didn't have working brake lights, so I couldn't tell it was still stopped (inadequate feedback)
  - The light turned green, so I assumed the car in front would go (control algorithm)
  - My driving instructor told me it was safe to do (mental model, control algorithm)
- How would these scenarios change if the car was automated?



Scenarios

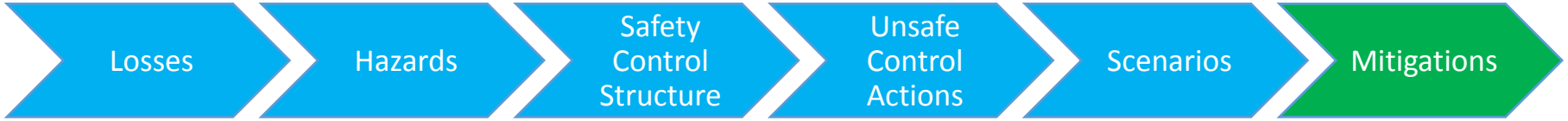


From Dr. Thomas' JAXA Presentation

Losses → Hazards → Safety Control Structure → Unsafe Control Actions → Scenarios → Mitigations

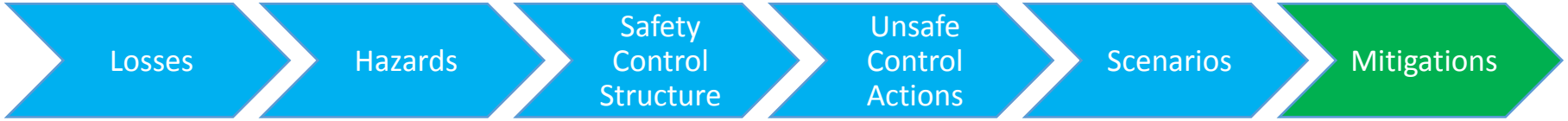
Minimizing Procedures (or Mitigations)

- Each scenario will have at least one minimizing procedure
- Minimizing procedures should be written such that they are actionable
- What are some minimizing procedures for the braking example?
  - The car didn't have working brake lights, so I couldn't tell it was still stopped (inadequate feedback)
    - Mandate working brake lights
    - Design backup system to notify drivers a car is stopped
    - Develop a sensor that detects an unsafe closure rate & alerts the driver
  - The light turned green, so I assumed the car in front would go (control algorithm)
    - Train drivers to take their cue from both other cars & the light
  - My driving instructor told me it was safe to do (mental model, control algorithm)
    - Evaluate and update training program
    - Fire the instructor (not really! Hindsight bias leads us to blame –focus on fixing the system)

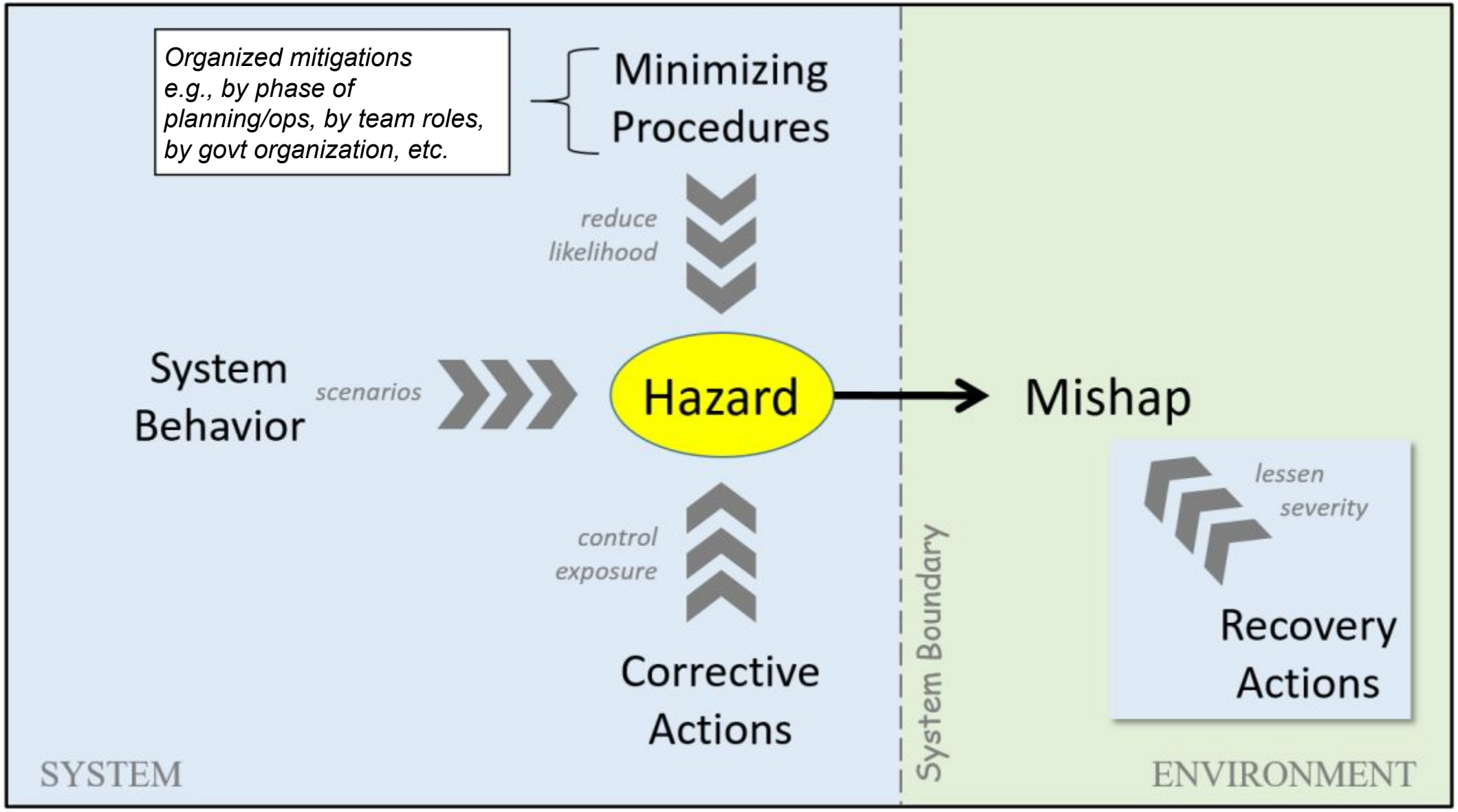


Mitigations, cont

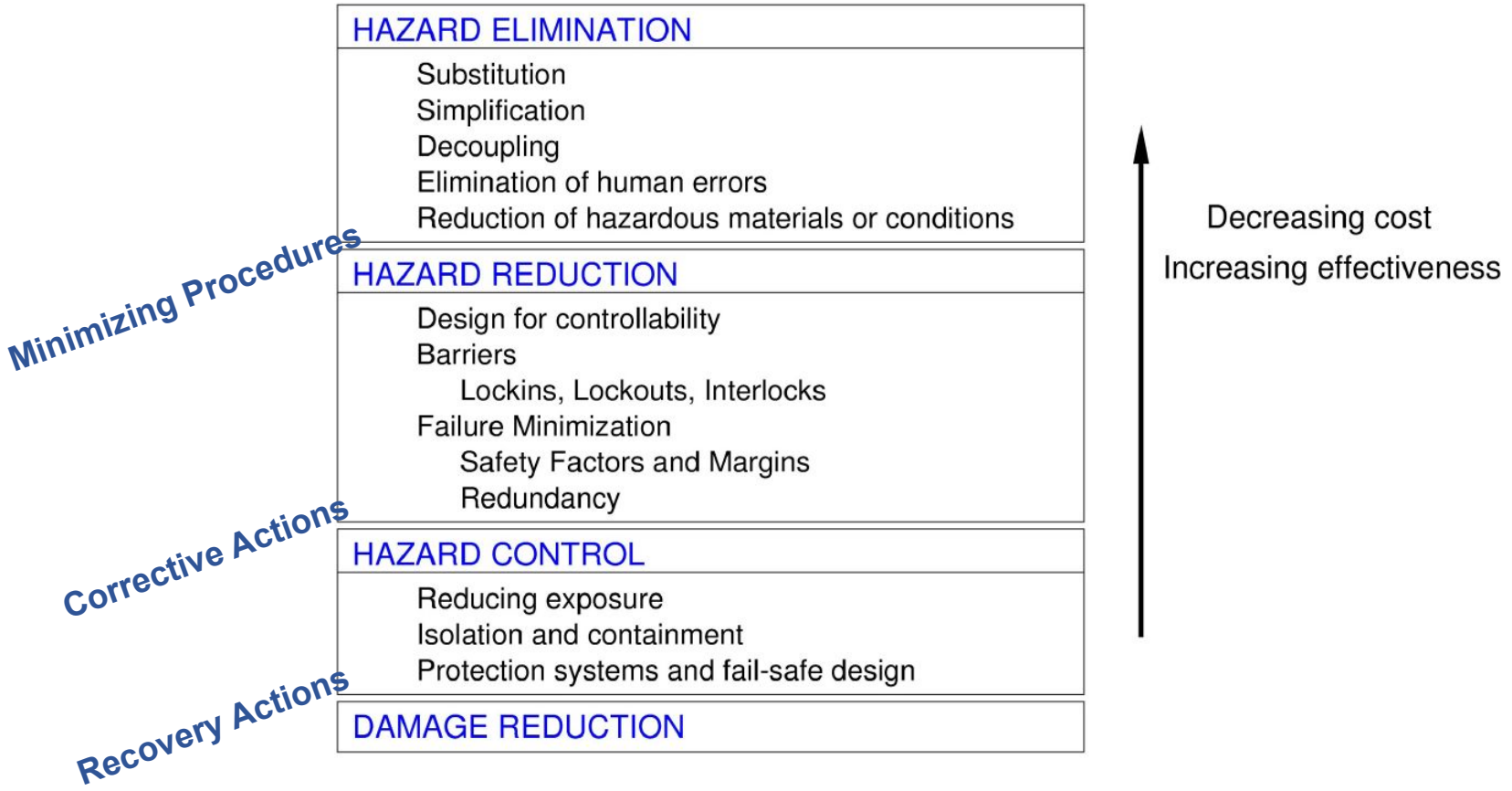
- Up to you whether you want to include mitigations you can't enforce, such as design mitigations
- Many scenarios may have more than 1 mitigation – try to go with the most effective mitigation first!
- Recommend organizing mitigations by categories:
  - Design, Test, Maintenance, Operations
  - Developing Influences, Settings and Configurations, Operational Procedures
  - Etc! Figure out what works best for your program



Putting it all together



Mitigation Order of Precedence



- From MIL-STD-882
  - Eliminate Hazard
  - Safety Devices
  - Warning Devices
  - Procedures and training

- Use higher precedence constraints as much as possible

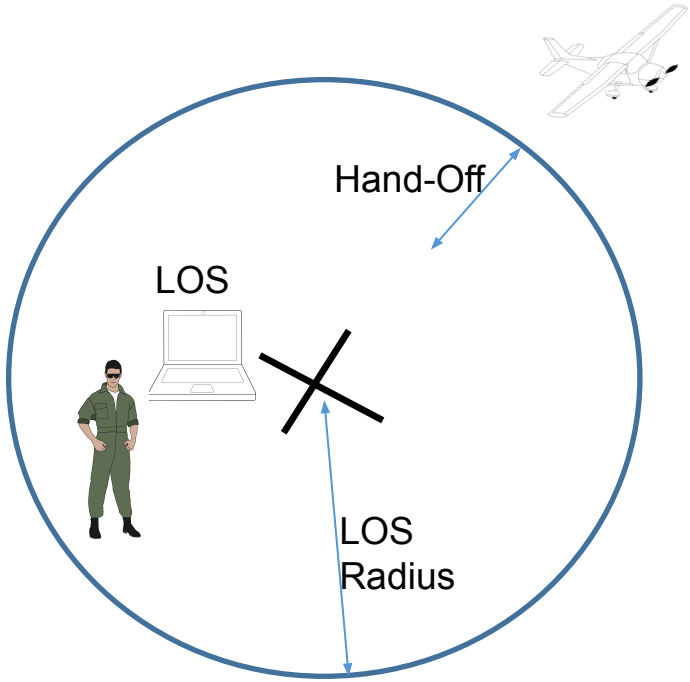
UAV Example

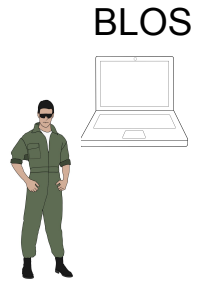
System Description

- General aviation (GA) aircraft that has been converted to a UAV
  - Controlled by ground stations
    - Line Of Sight (LOS) at airfield
    - Beyond LOS (BLOS) located elsewhere
  - Autopilot in Vehicle Management System (VMS) controls actuators connected to elevator, ailerons, rudder and engine throttle
  - Engine adapted with alternators to power VMS, actuators, and payload
  - Modified fuel tanks for longer endurance
  - Camera attached above instrument panel looking straight ahead

Typical Operational Sequence

- Preflight
- Tow to run-up area
- Engine Run-up
- Tow to runway
- Takeoff
- Climb
- Cruise
- Hand-off (LOS-BLOS)
- Cruise – conduct msn
- Hand-off (BLOS-LOS)
- Land
- Tow





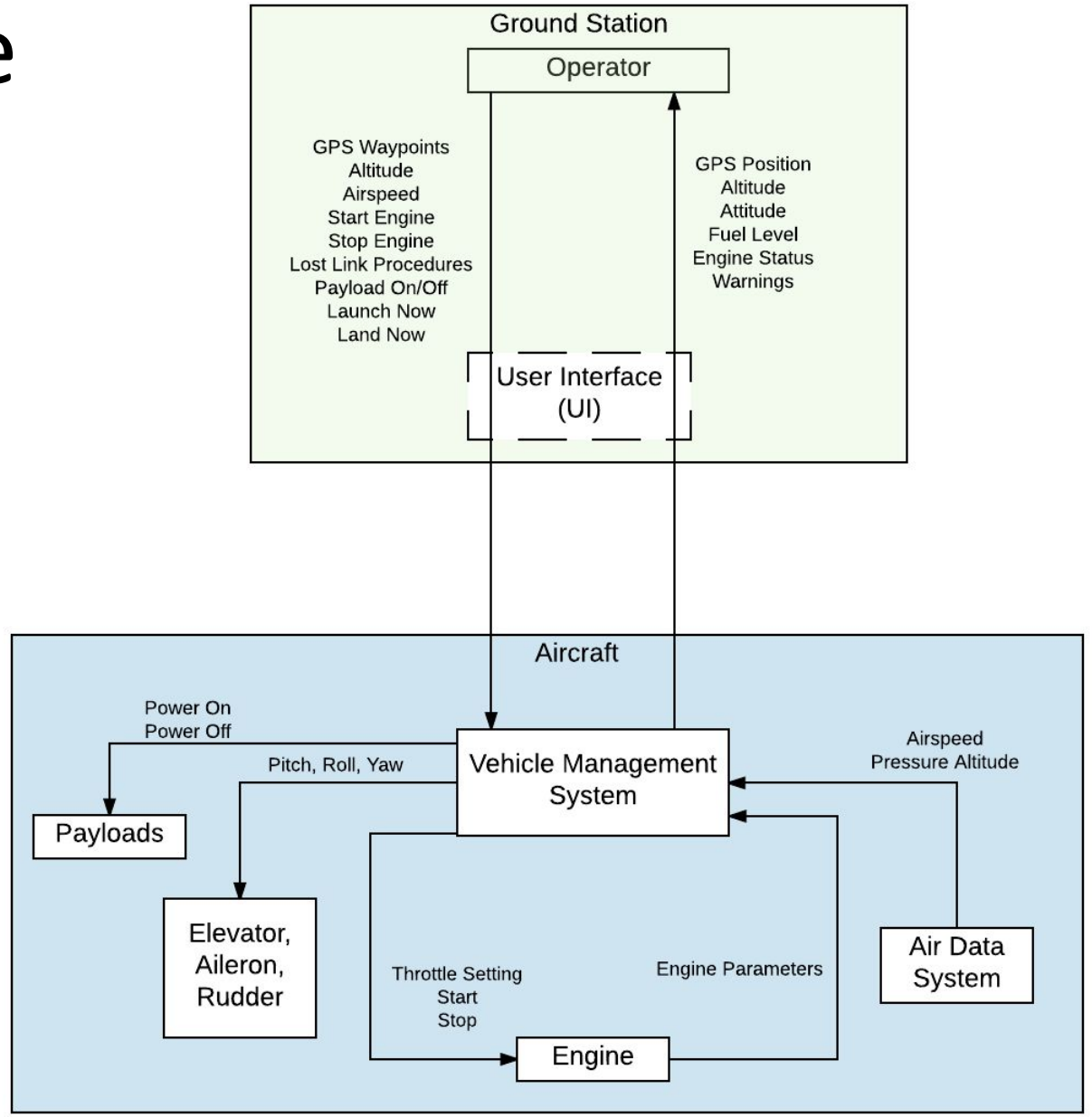
Accidents and Hazards

Designator	Accident Description
A1	Loss of life/injury
A2	Loss of or damage to UAV aircraft
A3	Loss of mission

Hazard	Assoc. Accident	Description of Hazard
H1	A1, A2	UAV too close to ground/building/person
H2	A1, A2	UAV violates minimum separation requirements
H3	A3	UAV does not complete mission
H4	A1, A2	UAV departs controlled flight
H5	A1, A2	UAV departs apron, taxiway, or runway during ground operations
H6	A1, A2	Loss of UAV airframe integrity

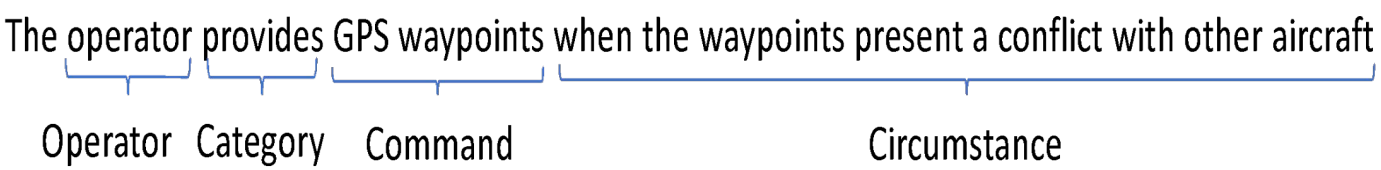
Safety Control Structure

- Ground station contains a laptop with a user interface (UI) and radios to link with the UAV
- VMS includes autopilot, and power distribution



UCA: GPS Waypoints

Operator	Not Providing Causes Hazard	Providing Causes Hazard	Incorrect Timing/Order	Stopped Too Soon/Applied Too Long
GPS Waypoints				



UCAs

Operator	Not Providing Causes Hazard	Providing Causes Hazard	Incorrect Timing/Order	Stopped Too Soon/Applied Too Long
GPS Waypoints	The operator does not provide GPS waypoints during prelaunch operations (H3)	The operator provides GPS waypoints when They do not align with the mission (H3)	The operator provides GPS waypoints after LOS is lost, but before BLOS radio link is established (H3, H4)	The operator provides GPS waypoints when the number of waypoints exceed the storage capacity of the autopilot (H3)
	The operator does not provide GPS waypoints when mission changes (H3)	The operator provides GPS waypoints when the waypoints present a conflict with other aircraft (H2)	The operator provides GPS waypoints after the UAV reaches bingo fuel (H4)	The operator provides GPS waypoints when the list of waypoints is not complete for the entire mission (H3)
		The operator provides GPS waypoints when the route length exceeds the fuel on board (H4)		
		The operator when the route is outside of LOS radius and BLOS is not being used (H3, H4)		

Scenarios

What are some scenarios for:

The operator provides GPS waypoints when the waypoints present a conflict with other aircraft (H2)

And what are possible mitigations for those scenarios?

Remember Order of Precedence

Scenarios

Scenario	Mitigation
The operator provides GPS waypoints that do not conflict with other traffic, but there is interference along the route. The waypoints are not received by the UAV, and autopilot uses waypoints from the previous sortie, which conflict with present traffic	UAV operators must be aware of EM usage in operating area and deconflict operations to avoid interference.
The operator provides waypoints to the UAV. The operator or mission planners used an old flight plan as a template for the current mission, but did not copy over all the data., The waypoints do not match the approved route from the airspace traffic operator (ATC).	The operator must verify the mission plan with the customer request and approve ATC route
The operator provides waypoints which do not conflict with air traffic, but the waypoints are far apart from each other, and travel between the waypoints do present a conflict with other aircraft	Waypoints must be sufficiently close together to control the behavior of the UAV and prevent it from conflicting with other traffic
The operator provides waypoints which do not conflict with the air traffic as reported by the air traffic operator, however the air traffic changes after planning or during the sortie. The operator does not receive the updated information in order to provide a different set of waypoints	The operator must be provided with and air traffic changes to ensure the UAV is properly deconflicting from other traffic
The operator sends the GPS waypoints to the UAV. They were not saved by the autopilot, and older waypoints already loaded were not overwritten.	The autopilot must save the GPS waypoints received by the UAV

Advice

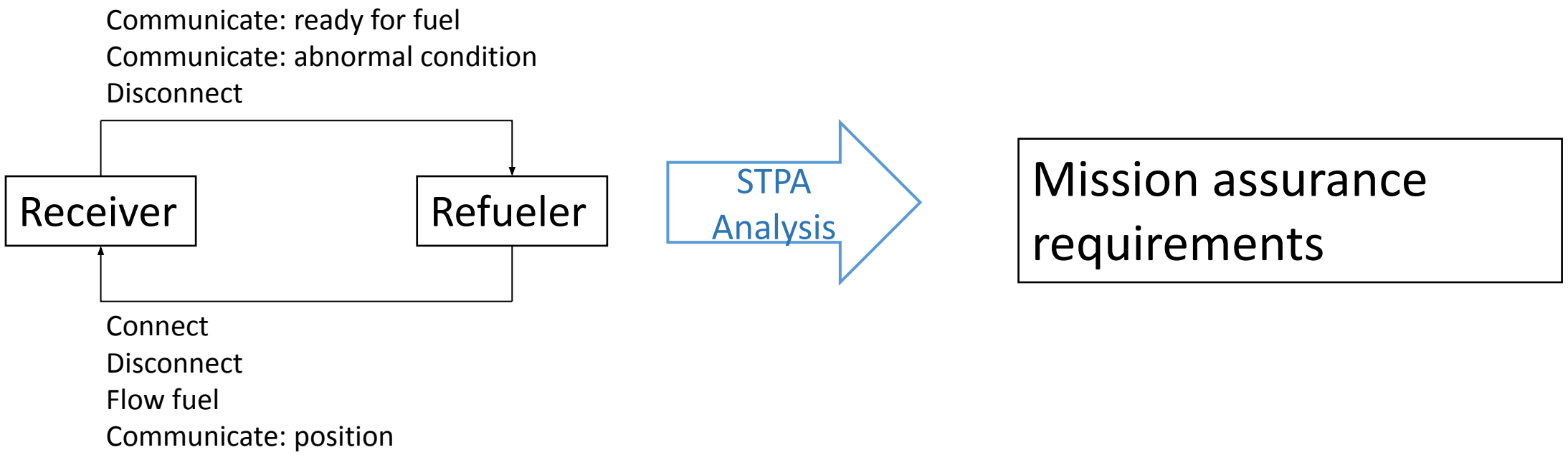
- Don’t dive into the details too early – stay high level and follow the high to low construct of this analysis
- Don’t make an assumption that the design is sufficient to prevent a hazard – the point of this exercise is to find the potential holes in the design solution
- Document any assumptions you make
- STPA is iterative due to the traceability – it forces you to revisit previous steps
- If a hazard or UCA isn’t traceable to the previous step: you missed a loss or hazard OR it’s outside the scope of the analysis

Test Example

Takeaways and Risk Discussion

Focus on System Behavior

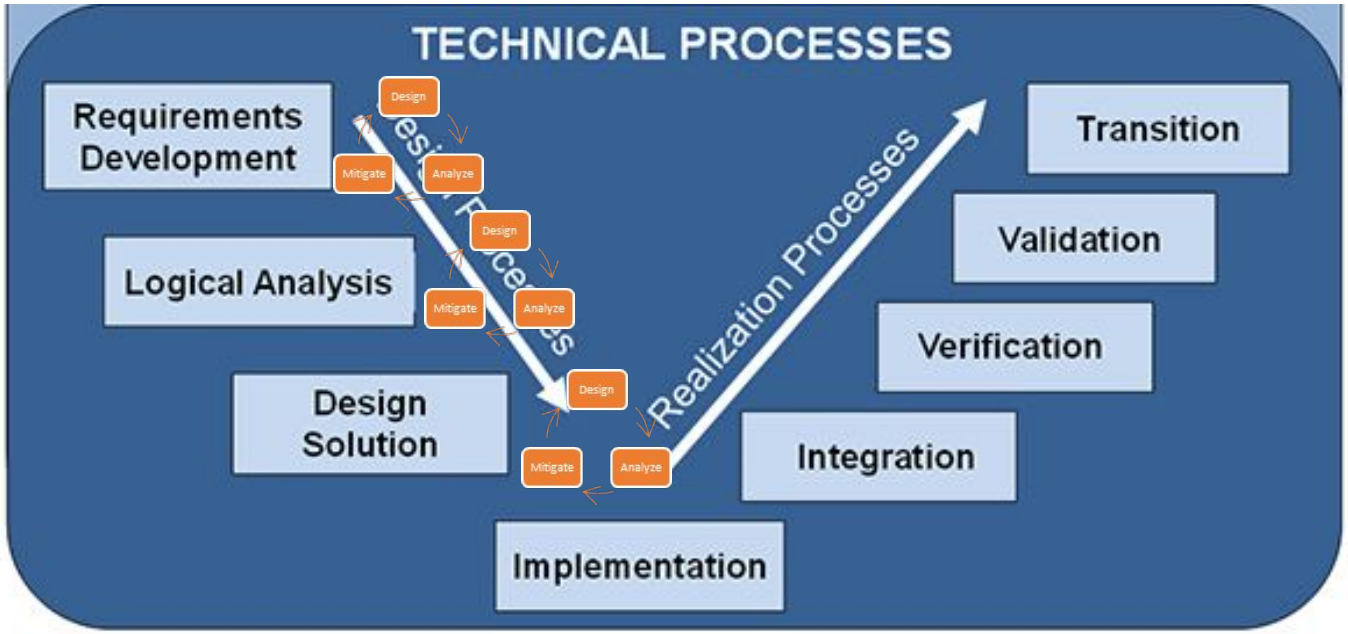
STPA can be used early in the design phase based on intended system functionality



Better requirements definition based on desired capabilities and interoperability

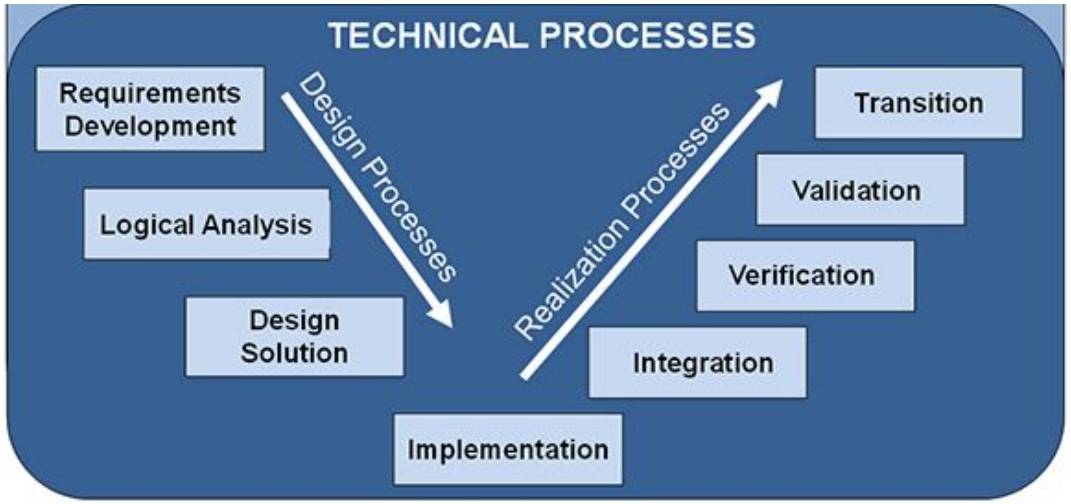
Left Side of V

- The tighter the decision loop is coupled, the faster constraints will be identified, reducing design rework
- Reduces surprises when you get to the right side of the V



Realization Processes - Right Side of V

- STPA requirements can be tested just as any other technical requirement
- If system behavior is not as expected a safety requirement is not being enforced:
  - Design was flawed, or
  - Operation of system was not within expected bounds
- V&V results fed back into STPA to resolve deficiencies



Preventing Mishaps: Leading Indicators

- Sociotechnical systems often trend towards an unsafe/unsecure scenario:
  - Manning changes
  - System/software modifications
  - Training updates
  - Operational utilization changes
  - Maintenance processes
- Leading indicators can be derived in two ways:
  - Documented assumptions
  - Safety constraints/mitigations
- Incidents often precede mishaps - a safety constraint was violated

In what other ways do systems trend towards unsafe/unsecure scenarios?

KC-135/KC-46 Example

KC-135	KC-46
Boom operator has direct view of receiver through window Non-visual cues: bow wave & tactile feedback through boom	Boom operator uses cameras & 3D glasses Non-visual cues gone

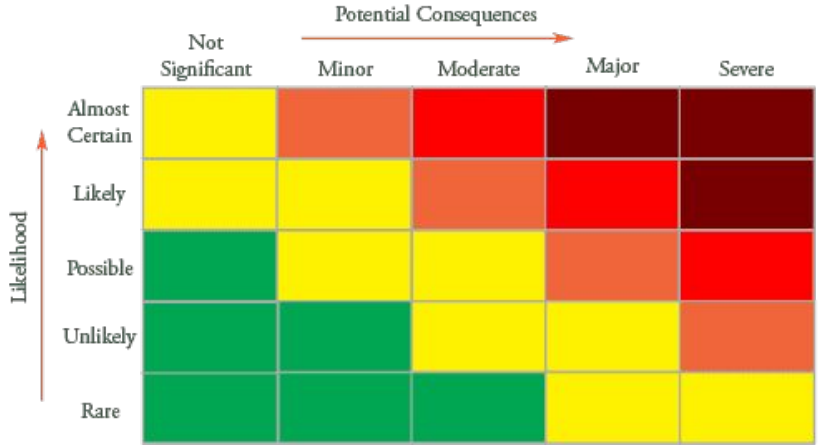
- Losses (test safety only):
  - Damage to or loss of receiver or tanker
- Hazards:
  - Midair collision
  - Boom strike
  - Fuel system incompatibility

- Will underlying causes of hazards (scenarios) be the same?
- Will the probability of each hazard be greater or smaller?

What About Risk?

- STPA is a hazard-scenario identification tool – it does not calculate risk
- Mechanical/zonal analyses and factors of safety are mature engineering determinations via traditional hazard analysis methods
- STPA does two things really well:
  - Identifies the ‘unknowns’ (the things that hurt most complex test programs!)
  - Frames everything (systemic and mechanical) into a traceable suite of model-based mitigations/recommendations

Risk Matrix



- MIL-STD-882 □ been around a long time
- Vertical axis: likelihood (of the loss)
  - Emergent properties (loss of mission) are the most important but most difficult to put a number on
  - Deterministic and quantitative probabilities rarely suitable for interdependent systems with coupling and agency
  - More recent MILSTD offers software control categories □ influence rubric
- Horizontal axis: severity/consequence (of the loss)
  - Does not scale linearly
  - In space, what is a loss? Mission, force package, asset, segment, damage vs degradation/denial, collateral/environmental effects, security/PP, publicity, temporary vs permanent, short vs long term, do we include adversary effects?

So what do we do?

- STPA identifies unsafe scenarios/actions that could lead to a loss – we choose how to act on that info
    - A systemic analysis methodology gives power to explain the reasoning and justifications to qualify risk/uncertainty and provide tradeoffs with other program pressures (cost, schedule, etc.)
- Determining probability is often not possible! What's the likelihood that:
    - The test team missed a critical safety of flight test parameter during safety planning?
    - The system doesn't function as designed?
    - A young engineer defers to the test director and doesn't call a knock it off even though he/she thinks there's an issue?
- Good substantiating info out of STPA for test approval authorities:
    - Frequency - number of hazards mitigated with a single requirement/mitigation
    - Tradeoff priorities – Compare different mishap severities to target traced hazards
- If a risk matrix is still required, the test team will still need to qualitatively (or quantitatively if possible) 'determine' the likelihood – our AF legacy test safety process does this a lot…

Takeaways

- Modern (human/software intensive) systems are tightly coupled
- Growing information flows/decisions cause emergent behavior
- Probabilistic risk assessments are not sufficient for safety
- Strive to understand your system and ask tough questions!
- Focus on system functionality then dive into the details
- If a mishap happens don’t blame – dig into systemic cause & fix it

Additional Resources

- http://psas.scripts.mit.edu/home/
- Using STPA to Inform Developmental Product Test
  - http://sunnyday.mit.edu/papers/Montes-Thesis-final.pdf
- Systems Theoretic Process Analysis Applied to Air Force Acquisition Technical Requirements Development
  - http://sunnyday.mit.edu/summers-thesis.pdf
- Annual STPA workshop
  - http://psas.scripts.mit.edu/home/stamp-workshops/
- STPA Handbook
  - https://psas.scripts.mit.edu/home/get_file.php?name=STPA_handbook.pdf

Questions?