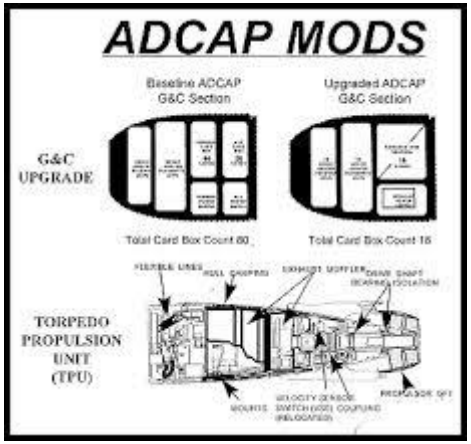


Introduction to STAMP: Part 1

Prof. Nancy G. Leveson

Aeronautics and Astronautics
MIT



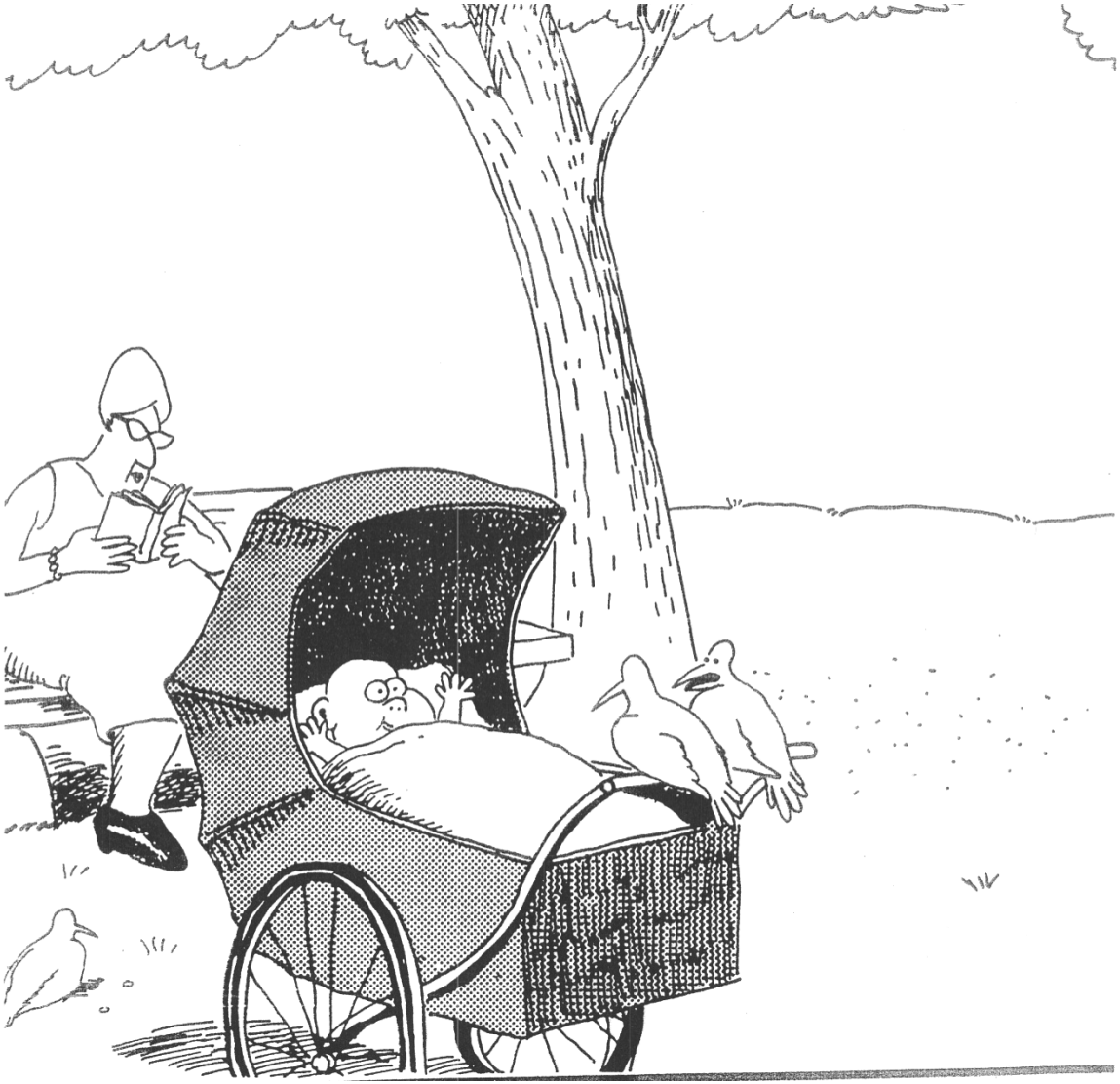

ADCAP MODS
Baseline ADCAP
G&C Section
Upgraded ADCAP
G&C Section
G&C
UPGRADE
Total Card Box Count 60
Total Card Box Count 16
TORPEDO
PROPULSION
UNIT
(TPU)

Some Current Uses

Aviation	Maritime
Automobiles	Dams
Defense	Manufacturing
Transportation	Communication
Healthcare/Hospitals	Rail
Workplace Safety	Robotics
Petrochemicals	Unmanned vehicles (air, ground, water)
Space	Process Control
Power and Energy	Mining
Nuclear	…
Agriculture	

General Definition of “Safety”

- Accident = Mishap = Loss: Any undesired and unplanned event that results in a loss
  - Including loss of human life or injury, property damage, environmental pollution, mission loss, negative business impact (damage to reputation, etc.), product launch delay, legal entanglements, etc.
  - Includes inadvertent and intentional losses (security)
- System goals vs. constraints (limits on how can achieve the goals)
- Safety: Absence of losses



It’s still hungry … and I’ve been stuffing worms into it all day.

Agenda

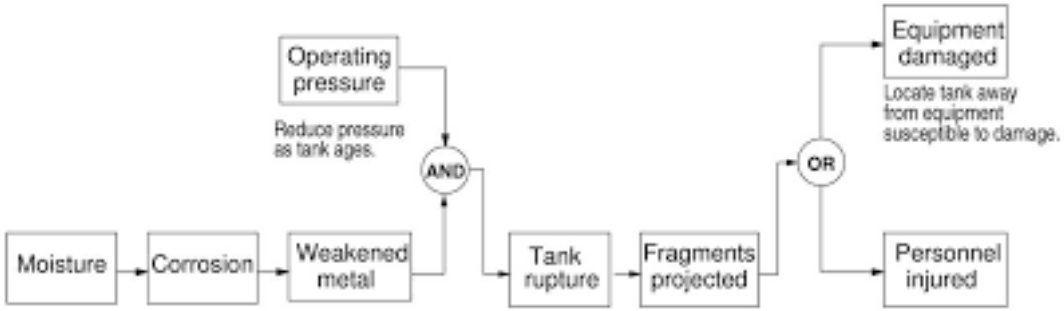
Part 1: Introduction to Causality and Accident Causality Models in today's world: LESSONS LEARNED

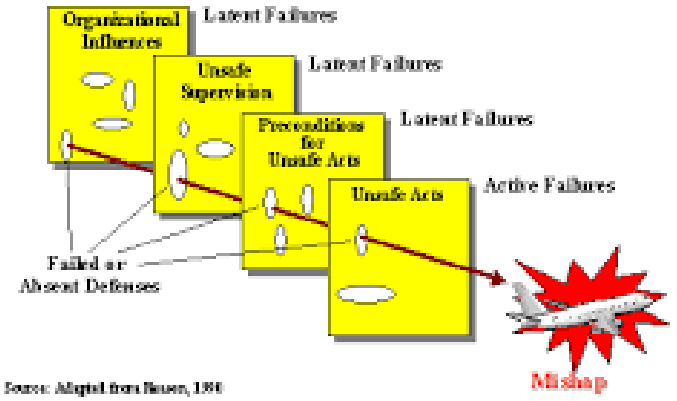
- Why do accidents occur today?
- An exercise in identifying events, conditions, systemic factors
- Limitations of the traditional chain-of-failure events model
- Role of software in accidents
- Accounting for human contributions to accidents

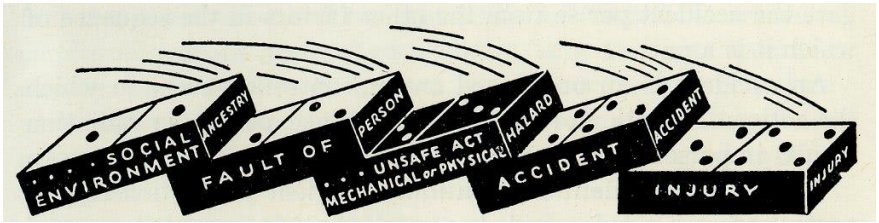
Part 2: STAMP: A Systems-Theoretic Model

- Informal introduction to systems theory
- What is STAMP?
- What are the STAMP-based analysis methods?
- Evaluations and comparisons

Introduction to Causality and Accident Causality Models

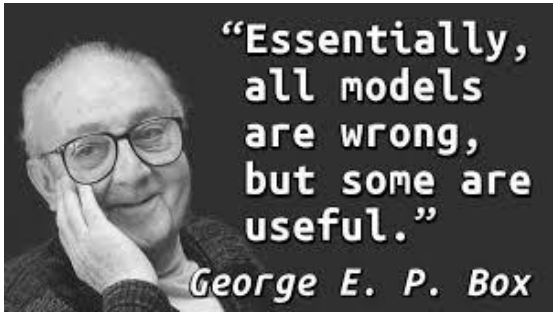






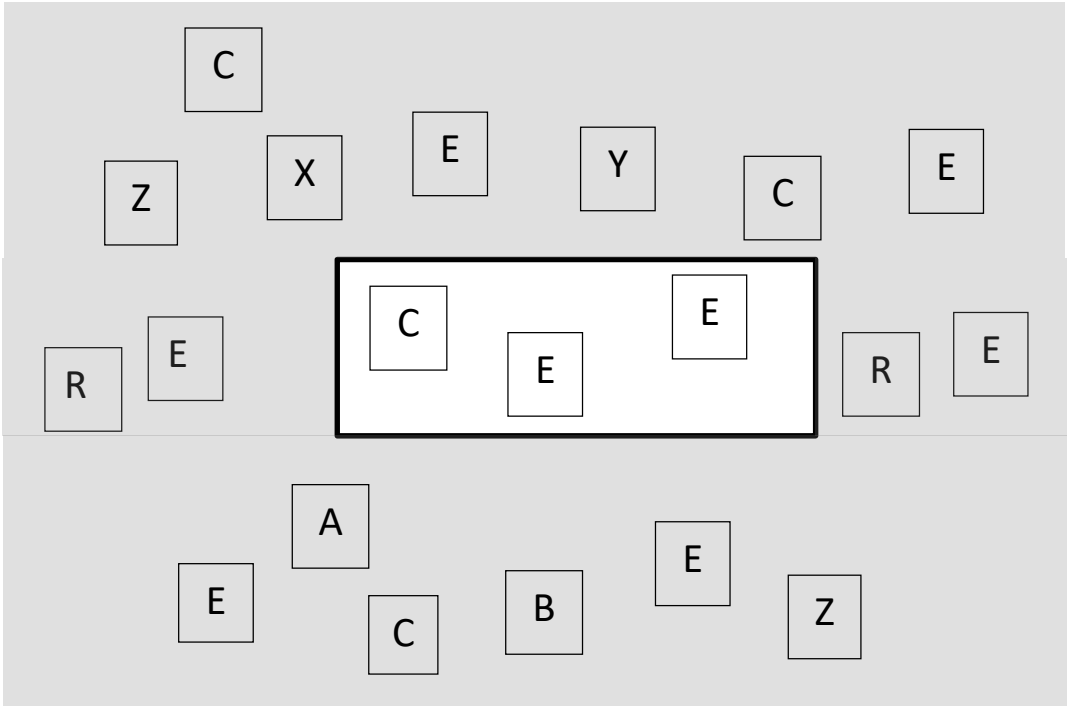
What is a Causality Model?

- Explain how things work and help predict how they will behave in the future
- No right or wrong model, only comparative effectiveness and usefulness

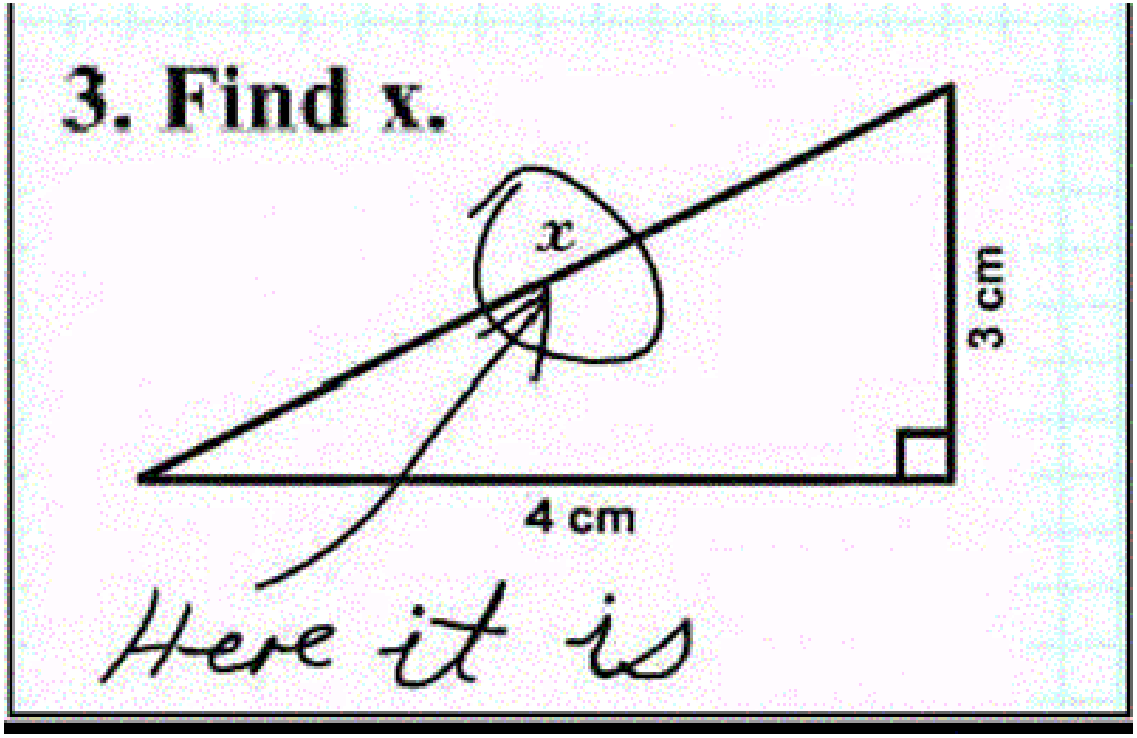


- Models help us deal with a messy world

Models Filter Out “Irrelevant” Information (for problem being solved)



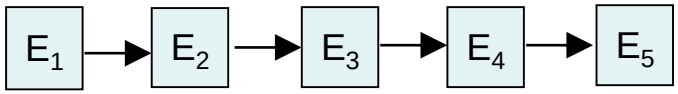
We want simple answers to complex questions.



So we get simple (but not useful) answers

Chain of (Failure) Events (COE) Model

- Assumes linear causality is enough to understand the world



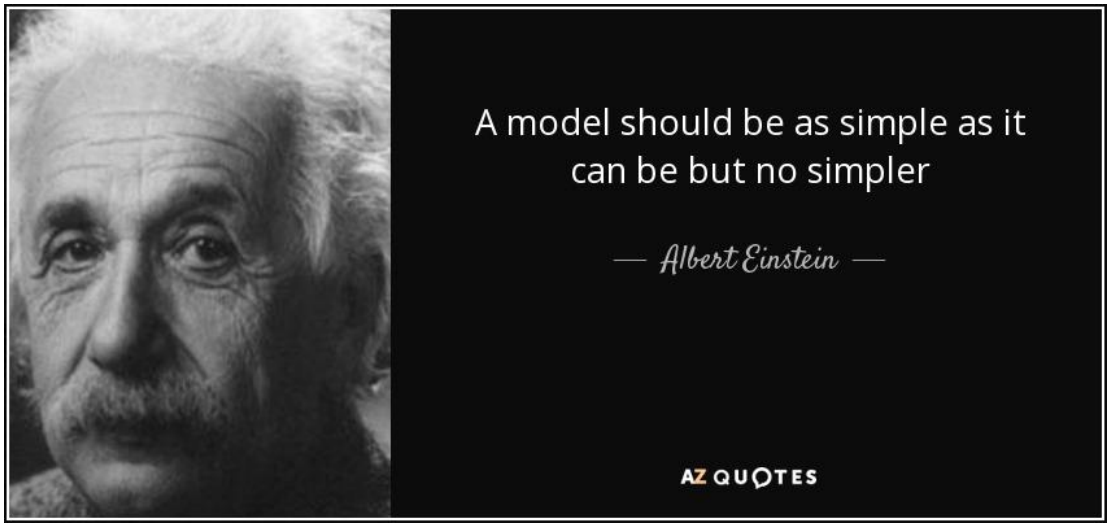
→ means “is necessary and sufficient for”

F_2 happens if and only if F_1 did

- One event is root or probable cause of final loss event
- Root and contributory causes are assumed to be in event chain

Chain of Events Model of Causality

- The chain of events model is very simple. But is it still useful?



- Does it leave out important causal factors in today's world?

Bhopal December 1984

Bhopal

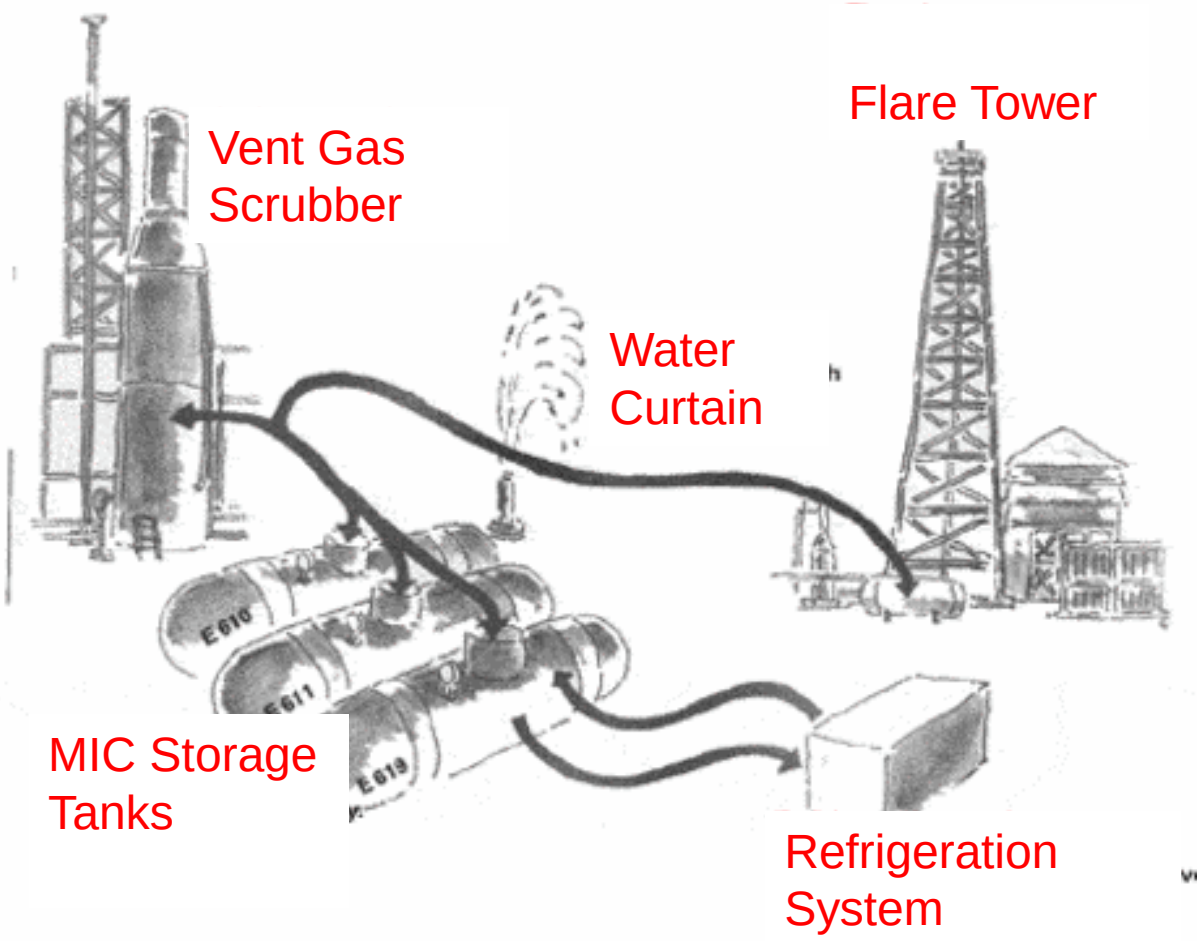
- Worst industrial accident in history
  - Blamed by management on operator error
  - Union Carbide blamed on sabotage
- MIC (methyl isocyanate) used in production of pesticides and polyurathanes (plastics, varnishes, and foams)
  - A major hazard is contact with water, which results in large amounts of heat.
  - Gas burns any moist part of body (throat, eyes, lungs)

Safety Features

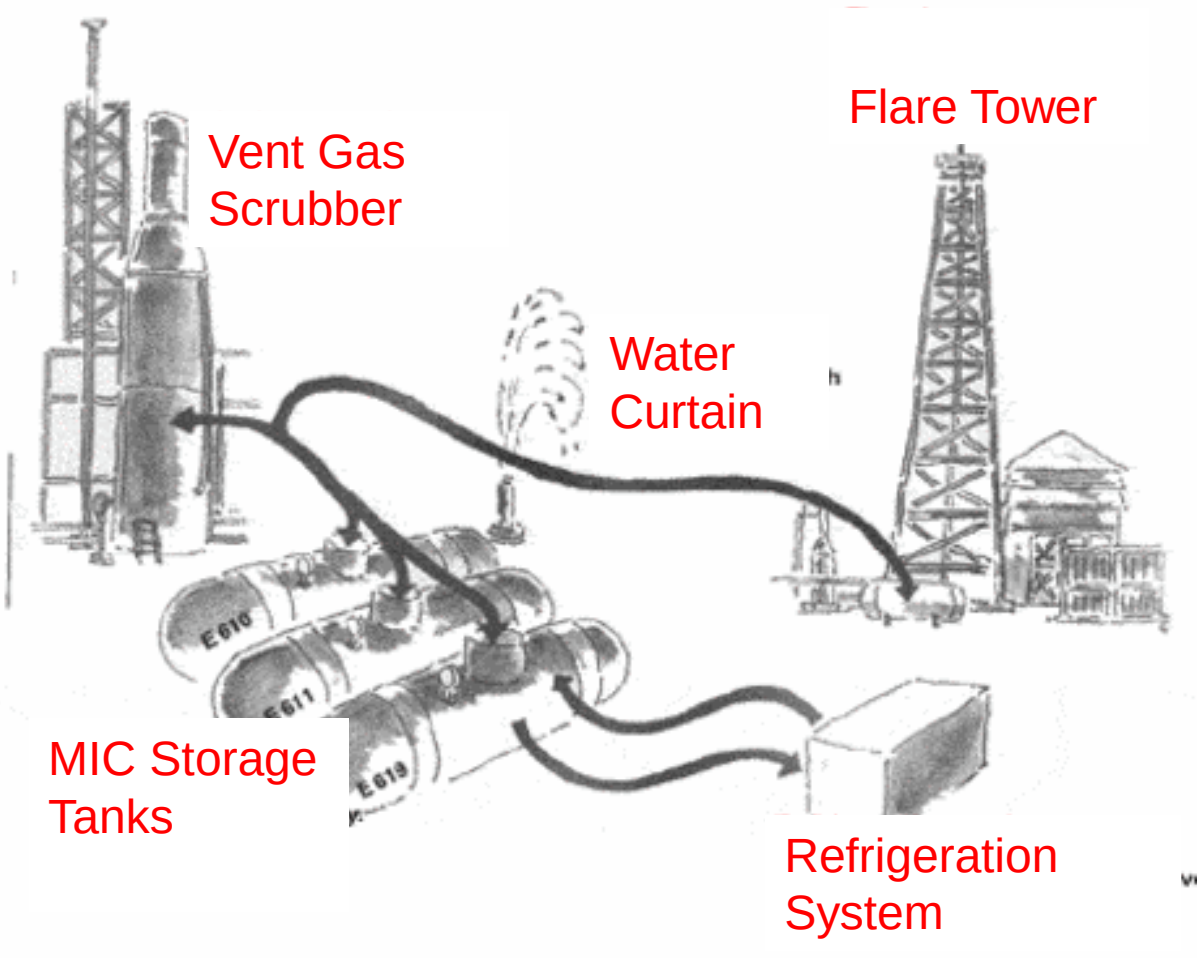
- UC specified requirements to reduce hazards
- Several backup protection systems ("defense in depth")
  - If one fails, the other will protect against an accident
  - Basically multiply probabilities together to get a probability for a loss.
- Standard design for safety approach in the process industries (nuclear, chemical, oil and gas, …)

– Bhopal used three double-walled, stainless steel tanks, each with a capacity of 60 tons.

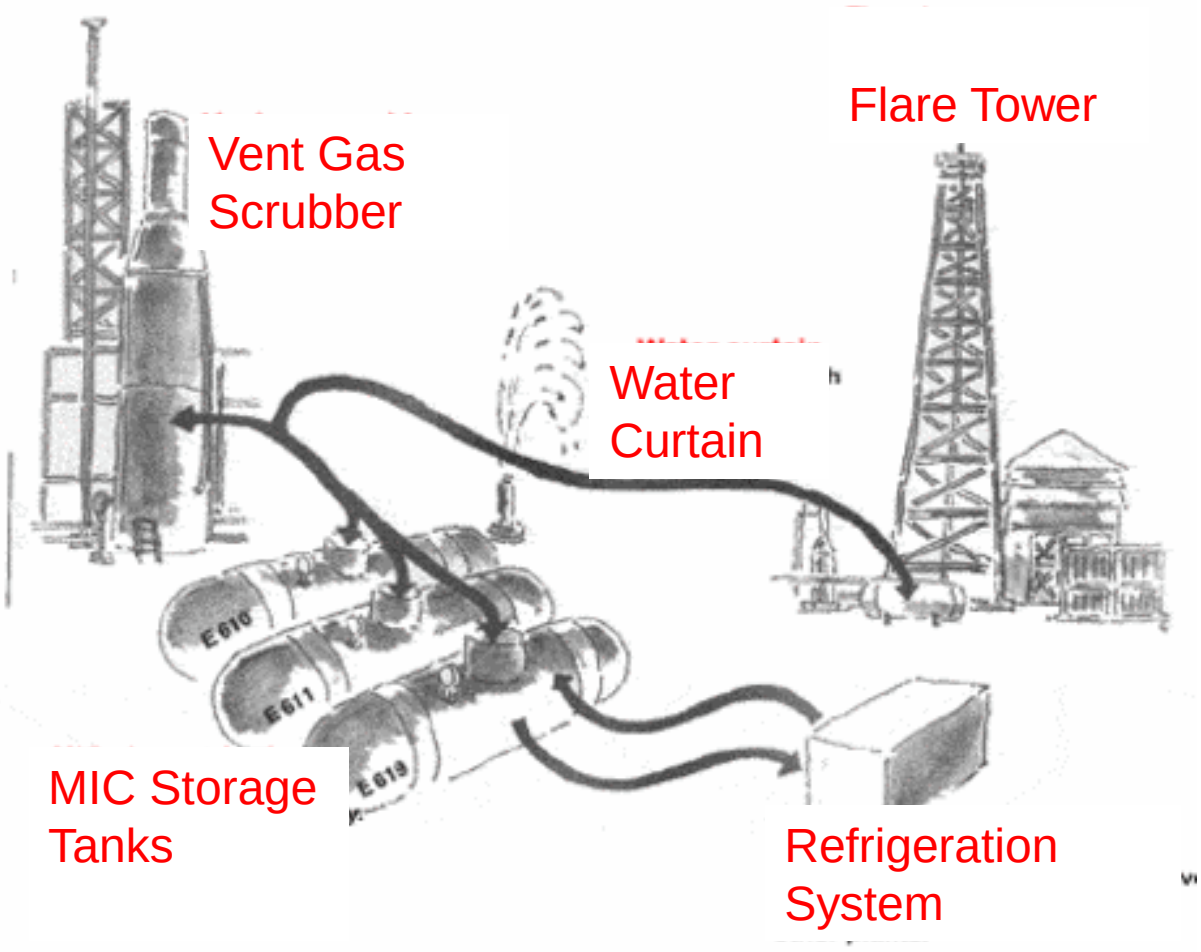
– Operating manual specified that tanks were never to contain more than half their maximum volume or a standby tank was to be available to which some of chemical could be transferred in case of trouble



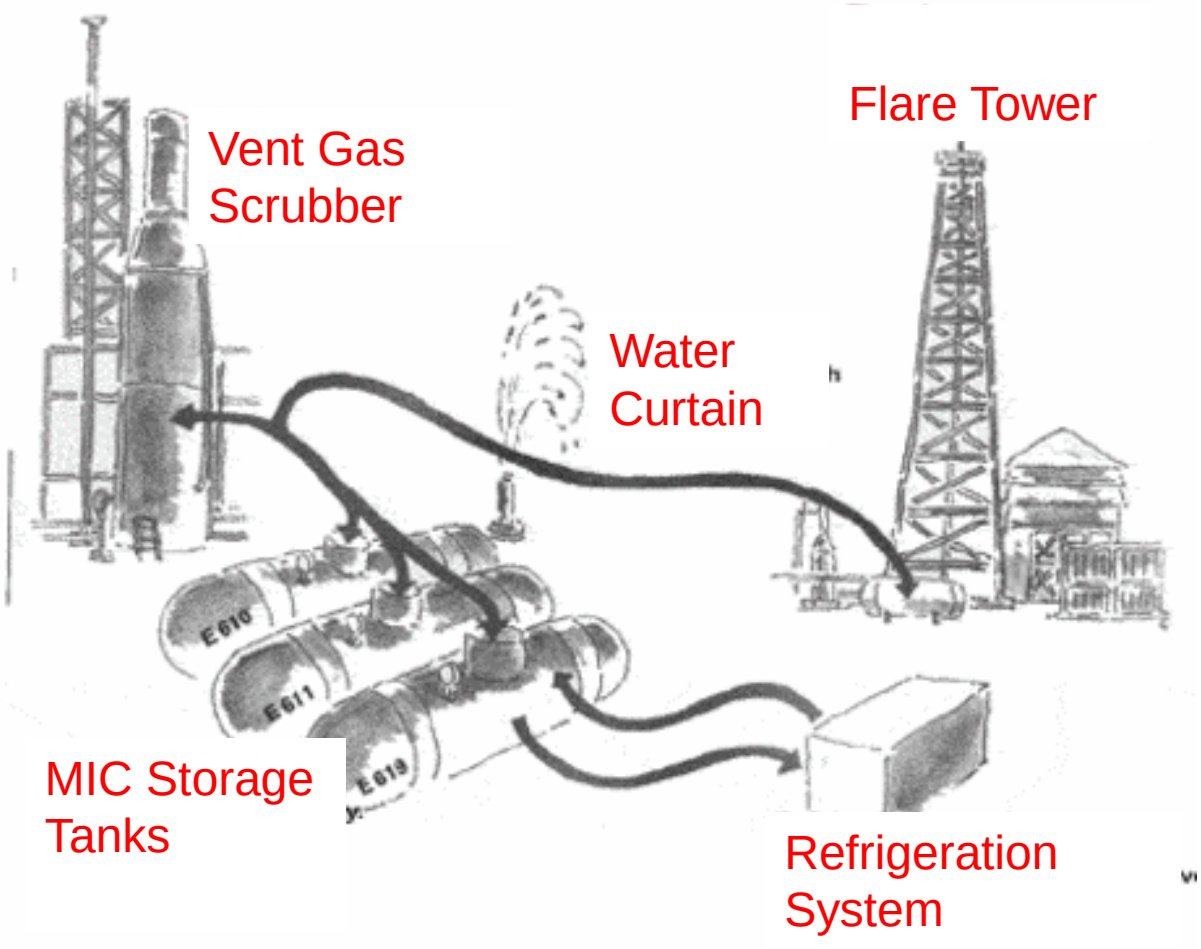
– Vent gas scrubber designed to neutralize any escaping gas with caustic soda. Scrubber was capable of neutralizing about 8 tons of MIC per hour at full capacity



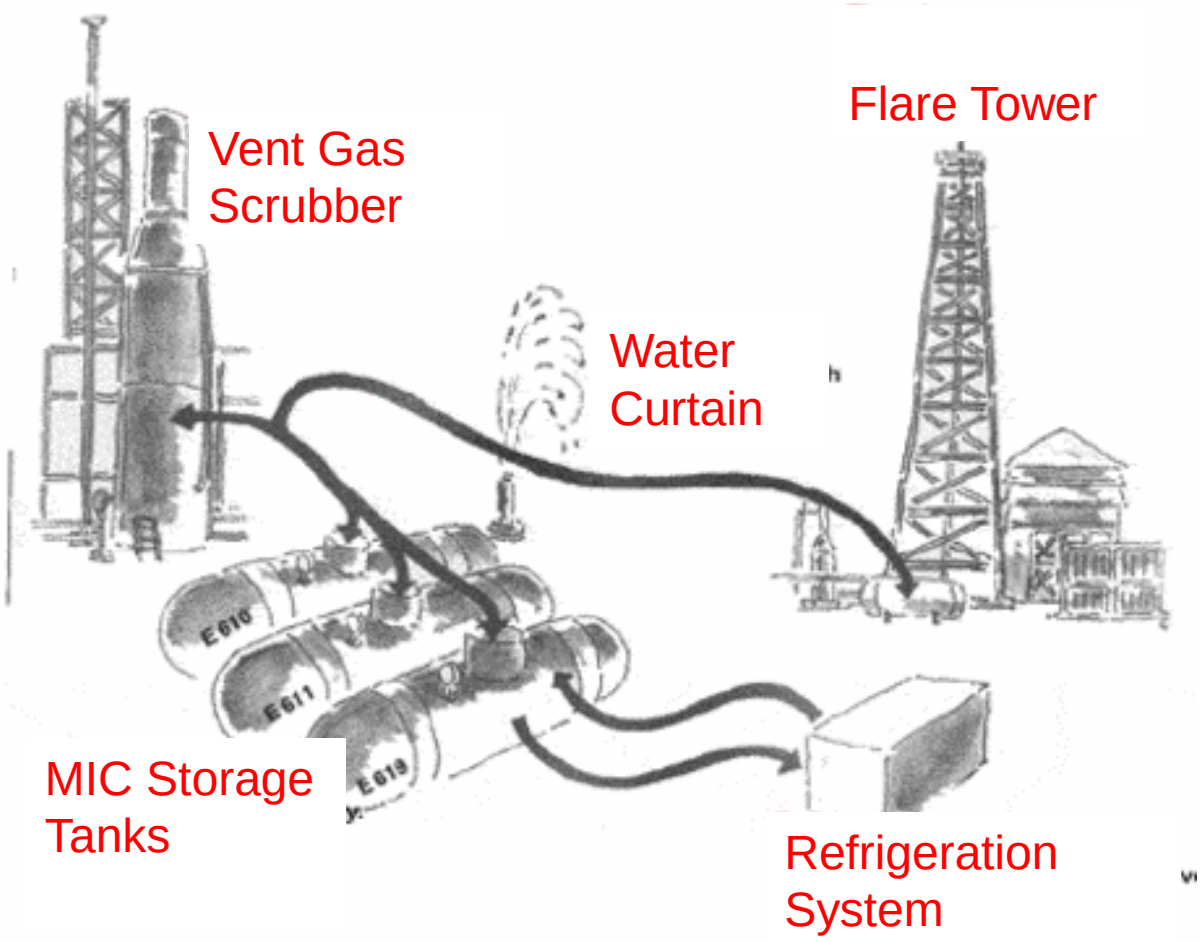
– Flare tower to burn off any escaping gas missed by scrubber; toxic gases would be burned high in the air, making them harmless



– Small amounts of gas missed by scrubber and flare tower were to be knocked down by a water curtain that reached 40 to 50 feet above ground. Water jets could reach as high as 115 feet, but only if operated individually.



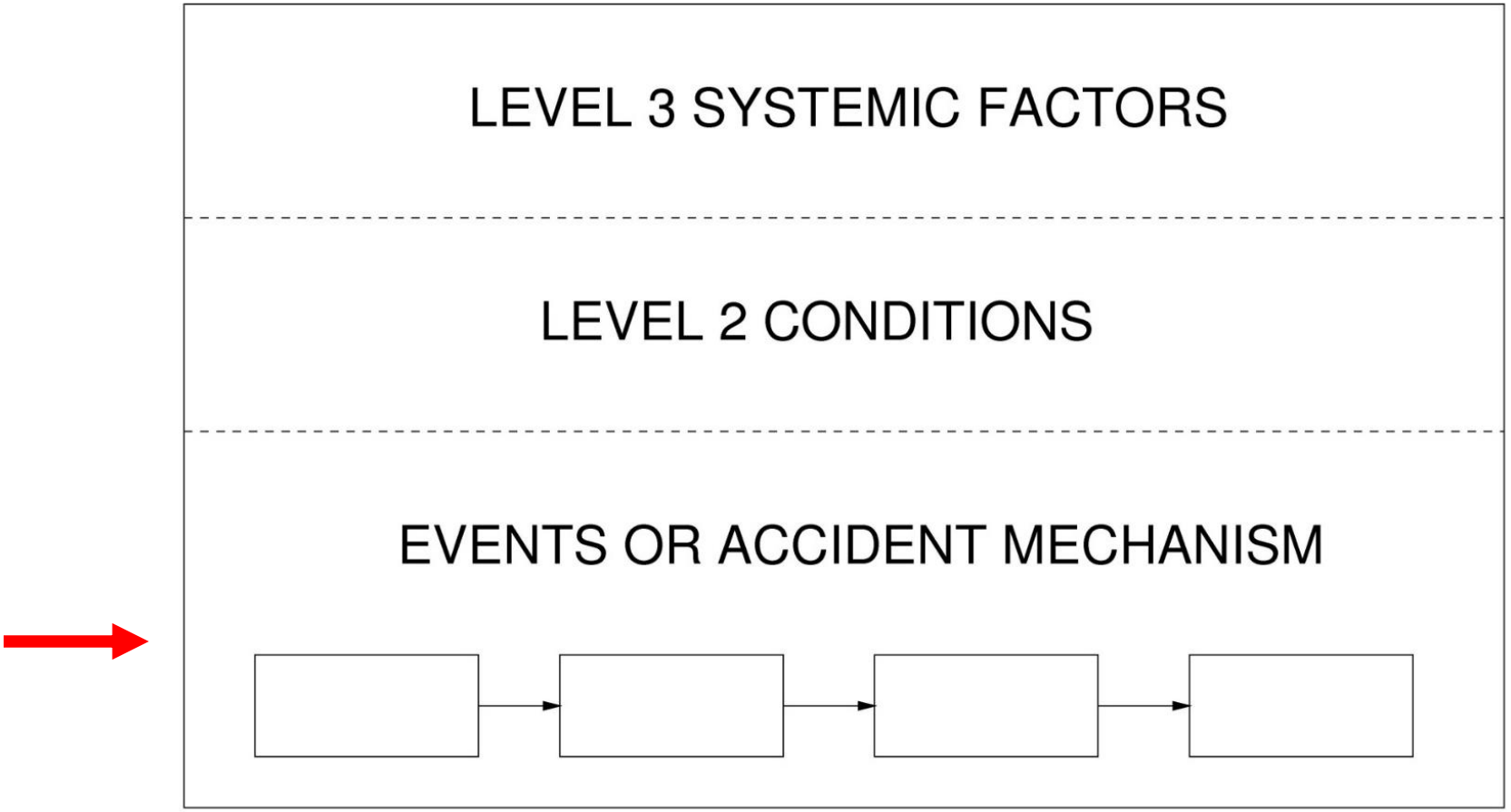
- To limit its reactivity, MIC was to be maintained at a temperature near 0°C
  - Refrigeration unit provided for this purpose
  - High temperature alarm if MIC reached 11°C



Safety Features (con't)

- MIC was to be stored in an inert atmosphere of nitrogen gas at 2 to 10 psi over atmospheric pressure.
- Regularly scheduled inspection and cleaning of valves specified as imperative
- Storage limited to 12 months maximum.
- If staff were doing sampling, testing, or maintenance at a time when there was a possibility of a leak or spill, operating manual specified they were to use protective rubber suits and air-breathing equipment.
- In case of an uncontrolled leak, a siren was installed to warn workers and surrounding community.

Levels of Causality



Exercise

1. Write down the causal factors of this accident as I go through the events (don’t guess beyond events).
2. What was the “root cause”?
3. What questions are raised in your mind by these events?

Events at Bhopal

- Dec. 2, 1984, relatively new worker assigned to wash out some pipes and filters, which were clogged.
- Pipes being cleaned were connected to the MIC tanks by a relief valve vent header, normally closed
- Worker closed valve to isolate tanks but nobody inserted required safety disk (slip blind) to back up valves in case they leaked
- Night shift came on duty at 11 pm.
- Pressure gauge indicated pressure was rising (10 psi instead of recommended 2 to 3 psi). But at upper end of normal range.

- Temperature in tank about 20 C.
- Temperature and pressure gauges were ignored because believed to be inaccurate. Operators told instead to use eye irritation as first sign of exposure.
- 11:30 pm: detected leak of liquid from an overhead line after some workers noticed slight eye irritation.
  - Leaky valves were common and were not considered significant
- Workers looked for leak and saw a continuous drip on outside of MIC unit.
  - Reported it to the MIC supervisor
  - Shift supervisor did not consider it urgent and postponed an investigation until after the tea break.

- 12:40 am on Dec. 3: Control room operator noticed tank 610 pressure gauge was approaching 40 psi and temperature was at top of scale (25 C)
- 12:45 am: Loud rumbling noises heard from tank. Concrete around tank cracked.
- Temperature in tank rose to 400 C, causing an increase in pressure that ruptured relief valve.
- Pressurized gas escaped in a fountain from top of vent stack and continued to escape until 2:30 am.
- MIC vented from stack 108 feet above ground. 50,000 pounds of MIC gas would escape.

- Operator turned off water-washing line when first heard loud noises at 12:45 am and turned on vent scrubber system, but flow meter showed no circulation of caustic soda.
  - He was unsure whether meter was working
  - To verify flow had started, he would have to check pump visually.
  - He refused to do so unless accompanied by supervisor
  - Supervisor declined to go with him.
- Operator never opened valve connecting tank 610 to the spare tank 619 because level gauge showed it to be partially full.

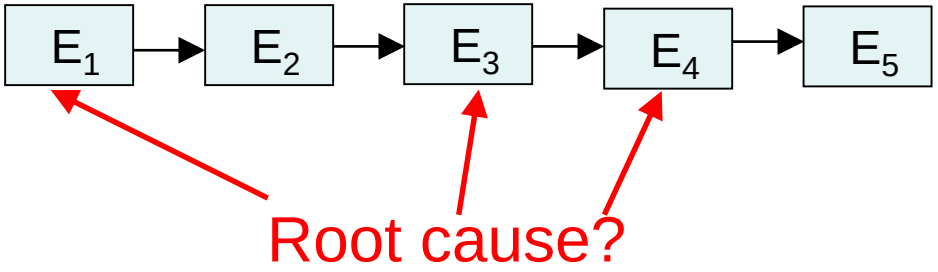
- MIC supervisor could not find his oxygen mask and ran to boundary fence, where he broke his leg attempting to climb over it.
- Control room supervisor stayed in control room until the next afternoon, when he emerged unharmed.
- Toxic gas warning siren not activated until 12:50 am when MIC seen escaping from vent stack.
  - Turned off after only 5 minutes, which was Union Carbide policy.
  - Remained off until turned on again at 2:30 am.
  - Police were not notified and when they called between 1 and 2, were given no useful information.

What were the causes of this accident given what you know so far? (poll 1)

What was the “root cause”? (poll 2)

What questions were raised by the events that you think need to be answered?

Selection of a "Root Cause" is Arbitrary



- We like the concept of a "root cause"
  - Usually focus on the operator or on physical failures
  - Ignore system-related, management factors (not in the events)
  - What "event" is involved in design of aircraft, design of pilot-vehicle interface, competitive or productivity pressures?
- "Root Cause Seduction" (John Carroll)
  - We want a root cause so we make up a convenient one. Why?
  - Provides an illusion of control
  - So fix symptoms but not process that led to those symptoms

Root Cause

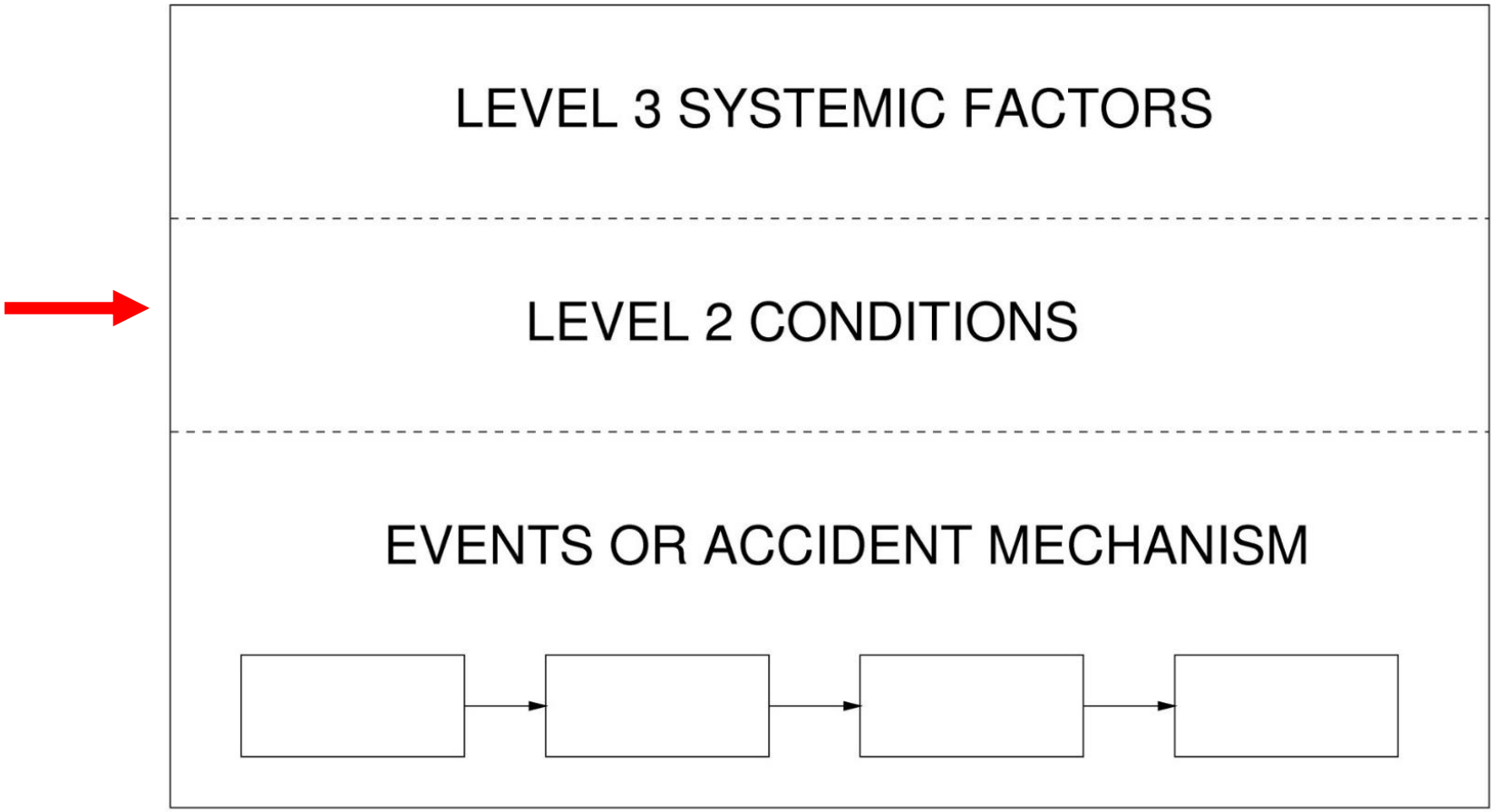
- Accident report cause:
  - Water entry into MIC tank
- Root cause: Not putting in slip blind
- Identified by Bhopal management: Pipe washer
  - Union Carbide said had to be sabotage because no worker could be so stupid.
  - Pipe washer was put in jail

What Questions are Raised by the Events?

- Why didn't the pipe washer or maintenance insert the slip disk?
- Why didn't the operators investigate before the tea break?
- Why did all the safety devices fail at once? Wouldn't that have a probability of essentially zero?
- Why did the operator not open the valve to bleed the MIC into the extra tank?
- Why was warning alarm not sounded until late and then turned off after 5 minutes?
- Why didn't the workers follow their training and evacuate properly when the alarm sounded?
- Why didn't the refrigeration unit keep the temperature low?
- etc.

Not possible to understand the events without knowing **why** they occurred (chain of events is not enough)

Hierarchical models



What about all the safety devices and procedures?

- How could the vent scrubber, flare tower, water spouts, refrigeration unit, alarms, and monitoring instruments (all the defenses in depth) fail simultanously?
- Isn’t the probability of that virtually zero?

- Flare tower was totally inadequate to deal with estimated 40 tons of MIC that escaped during accident.
  - Could not be used anyway because pipe was corroded and had not been replaced.
- Vent scrubber (had it worked) was designed to neutralize only small quantities of gas at fairly low pressures and temperatures.
  - Pressure of escaping gas during accident exceeded scrubber's design by nearly 2 ½ times
  - Temperature of escaping gas at least 80 degrees more than scrubber could handle.
  - Shut down for maintenance
- Water curtain designed to reach height of 40 to 50 feet. MIC vapor vented over 100 feet above ground.

These are design or maintenance errors. Are failure probabilities relevant?

- Operating manual said refrigeration unit must be operating whenever MIC was in the system
  - Chemical has to be maintained at a temp no higher than 5 C. to avoid uncontrolled reactions.
  - High temperature alarm to sound if MIC reached 11 C.
  - Refrigeration unit turned off and MIC usually stored at nearly 20 C.
  - Plant management adjusted threshold of alarm, accordingly, from 11 C to 20 C., thus eliminating possibility of an early warning of rising temperatures.

- Not uncommon for a company to turn off passive safety devices to save money; gauges are frequently out of service.
  - At Bhopal, few alarms, interlocks, or automatic shutoff systems in critical locations that might have warned operators of abnormal conditions or stopped the gas leak before it spread.
  - Thresholds established for production of MIC routinely exceeded. e.g., workers said it was common to leave MIC in the spare tank.
- Practice alerts did not seem to be effective in preparing for an emergency (ran from contaminated areas and ignored buses sitting idle and ready to evacuate them)

- Why didn’t anyone insert the slip blind?
  - Maintenance sheet contained no instruction to insert disk
  - Worker assigned task to wash out pipes did not check to see whether pipe properly isolated because said it was not his job to do so.
  - He knew valves leaked, but safety disks were job of maintenance department.
- Pipe-washing operation should have been supervised by second shift operator, but that position had been eliminated due to cost cutting.

- Tank 610 contained 40 to 50 tons of MIC out of total capacity of 60 tons, which violated safety requirements.
  - Tanks were not to be more than half filled
  - Spare tank was to be available to take excess
  - Adjacent tank thought to contain 15 tons according to shipping records, but contained nearer to 21 tons
  - Spare tank (619) contained less than 1 ton, but level gauge showed it was 20 percent full
  - Many of gauges not working properly or were improperly set.

- Alarms sounded so many times a week (20 to 30) that no way to know what the siren signified
  - Emergency signal was identical to that used for other purposes, including practice drills.
  - Not turned on until 2 hours after MIC leak started and then turned off after 5 minutes (company policy)

- Plant workers had only bare minimum of emergency equipment, e.g., shortage of oxygen masks discovered after accident started.
  - They had almost no knowledge or training about how to handle non-routine events.

- Police were not notified when chemical release began
  - When called by police and reporters, plant spokesmen first denied accident and then claimed MIC was not dangerous.

Has your view of this accident changed with this additional information? (poll 3)

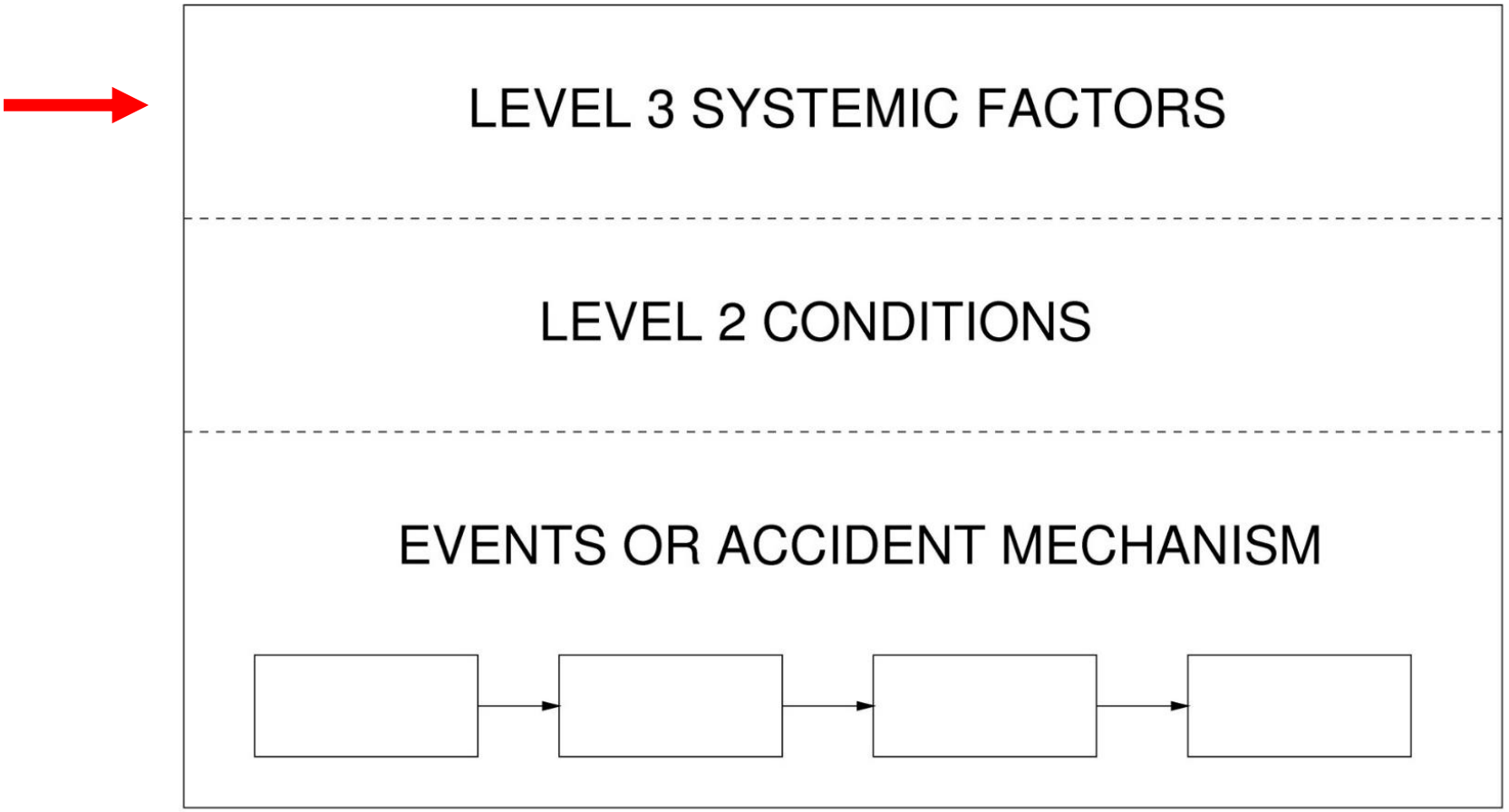
What additional causal factors would you now include?

What additional questions would you want answered?

Some Additional Questions

- Why was the refrigeration turned off?
- Why did nobody supervise the low-level pipe washer?
- Did anyone know about all the design errors? Why were they made?
- Why so many design flaws?
- Maintenance seems very poor. Why?
- Why did nobody know that the written procedures were not being followed? Why were they not being followed?
- Where was the operational safety group?

Hierarchical models



Systemic Factors

- Conditions can often be linked directly to events or even conditions;
  - No simple box and arrow diagrams
  - Usually involve indirect causality
- May involve things like “safety culture”
- Often left out of accident reports
  - Probably the most important to fix if want to prevent future accidents
  - Involve generic lessons learned that apply widely

Some Systemic Factors at Bhopal

- Demand for MIC dropped sharply after 1981, leading to reductions in production and pressure on company to cut costs.
  - Plant operated at less than half capacity when accident occurred.
  - UC put pressure on Indian subsidiary to reduce losses, but gave no specific details about how this was to be done.
- In response, maintenance and operating personnel cut in half.
  - Top management justified cuts as merely reducing avoidable and wasteful expenditures without affecting overall safety.
- As plant lost money, many of skilled workers left for more secure jobs. They either were not replaced or replaced by unskilled workers.

- Maintenance procedures severely cut back and shift relieving system suspended (if no replacement showed up at end of shift, following shift went unmanned).
- Indian government required plant to be operated completely by Indians
  - At first, UC flew plant personnel to West Virginia for intensive training and had teams of U.S. engineers make regular on-site safety inspections.
  - By 1982, financial pressures led UC to give up direct supervision of safety at the plant, even though it retained general financial and technical control.
  - No American advisors resident at Bhopal after 1982.
- Minimal training of many of workers in how to handle non-routine emergencies.

- Several Indian staff who were trained in U.S. resigned and were replaced by less experienced technicians.
  - When plant first built, operators and technicians had equivalent of two years of college education in chemistry or chemical engineering.
  - In addition, UC provided them with 6 months training.
  - When plant began to lose money, educational standards and staffing levels were reportedly reduced.
- In 1983, chemical engineer managing MIC plant resigned because he disapproved of falling safety standards. He was replaced by an electrical engineer.

- Morale at the plant was low. Management and labor problems followed the financial losses.

  "There was widespread belief among employees that the management had taken drastic and imprudent measures to cut costs and that attention to the details that ensure safe operation were absent."

- Five months before accident, local UC India management decided to shut down refrigeration system.
  - Most common reason given was cost cutting.
  - Local management claimed unit was too small and never worked satisfactorily.
  - Disagreement about whether UC in U.S. approved this measure.
  - High temperature alert reset and logging of tank temperatures discontinued.

- Lots of other examples of unsafe conditions that were permitted to exist:
  - At time of accident, chloroform contamination of MIC was 4 to 5 times higher than specified in operating manual, but no corrective action taken.
  - MIC tanks were not leak-tight to a required pressure test.
  - Workers regularly did not wear safety equipment, such as gloves or masks because of high temperatures in plant. There was no air conditioning.
  - Inspections and safety audits at the plant were few and superficial.

- A review and audit of Bhopal plant in 1982 noted many of deficiencies involved in accident
  - No follow-up to ensure deficiencies were corrected.
  - A number of hazardous conditions were known and allowed to persist for considerable amounts of time or inadequate precautions were taken against them.
  - Report noted such things as filter-cleaning operations without using slip blinds, leaking valves, possibility of contaminating the tank with material from the vent gas scrubber, bad pressure gauges.
  - Report recommended raising capacity of water curtain. Pointed out that alarm at flare tower was non-operational and thus any leakage could go unnoticed for a long time.
  - According to Bhopal manager, all improvements called for in the report had been taken care of, but obviously not true.

- Prior warnings and events presaging the accident were ignored:
  - 6 serious incidents between 1981 and 1984, several of which involved MIC
  - One worker killed in 1981, but official inquiries required by law were shelved or tended to minimize government's or company's role.
  - A leak similar to one involved in the big one had occurred the year before.
  - Journalists and others tried to warn of dangers
  - At least one person within government tried to bring up hazards of plant. He was forced to resign.
  - Local authorities and plant managers did nothing in response.

Are these systemic factors unique to the Bhopal accident?

Focus on Pilot/Operator Error ("Failures")

- Pilots/operators almost always in COE for an accident
  - So can always select something they did as the root cause
  - After a while, becomes established that they cause most accidents
  - But human behavior always affected by the context in which it occurs
    - We are designing systems in which human error inevitable
  - Need to understand
    - WHY pilots/operators behaved the way they did
    - Reasons behind why the events occurred

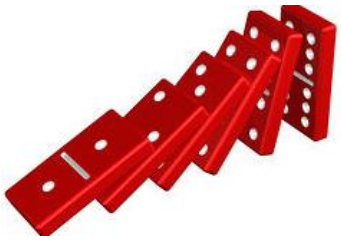
Focus on Identifying a Root or Probable Cause

- May be used to deflect attention from powerful interests.
  - What is declared to be root cause is arbitrary so want to direct attention to someone else.

  - Easy to accomplish when only direct or simple relationships included in chain
  - Sometimes argue that because not everyone made a mistake when presented with same circumstances, those circumstances cannot be the cause.
    - Other pilots flew 737 MAX before crashes and they overcame design flaws so design flaws cannot be "cause" of the accident

Various Incarnations of COE Model

- All use different real world analogies for same thing
  - Bow ties[1], dominoes, cheese slices, etc.
  - Different names and graphical notations for same thing



- Easily understood but is the COE model too simple for today's increasingly complex world (technical and social)?
  - Question is <u>not</u> whether the COE model is right or wrong
  - Question <u>is</u> whether it provides the most useful explanation for the goals of accident causal analysis and prevention.

[1]Nancy Leveson, *Shortcomings of the Bow Tie and Other Safety Tools Based on Linear Causality*, July 2019

Lesson Learned

- Need to look beyond events to prevent accidents
  - Why did events occur?
  - To learn, we need to look at:
    - Conditions that lead to the events
    - Systemic factors that influence almost everything but not necessarily directly related (cannot just draw an arrow or assume a “failure”)

Dealing with Complexity

Our current tools are all 50-75 years old but our technology is very different today

1940 1950 1960 1970 1980 1990 2000 2010 2020

FMEA

FTA HAZOP

ETA

Bow Tie (CCA) FTA + ETA

- Introduction of computer control
- Exponential increases in complexity
- New technology
- Changes in human roles

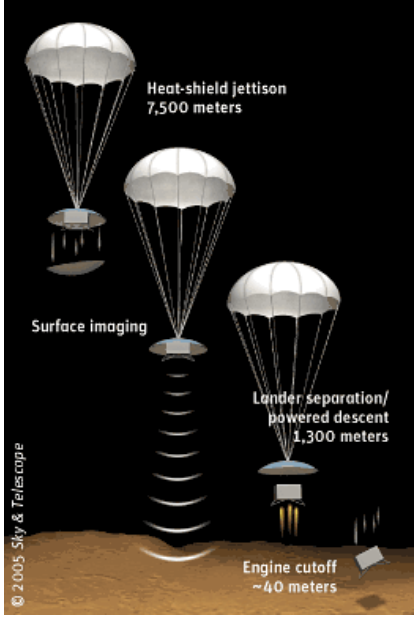
Assumes accidents caused by component failures


It's only a random failure, sir! It will never happen again.

What Failed Here?

- Navy aircraft were ferrying missiles from one location to another.
- One pilot executed a planned test by aiming at aircraft in front and firing a dummy missile.
- Nobody involved knew that the software was designed to substitute a different missile if the one that was commanded to be fired was not in a good position.
- In this case, there was an antenna between the dummy missile and the target so the software decided to fire a live missile located in a different (better) position instead.

Accident with No Component Failures



- Mars Polar Lander
  - Have to slow down spacecraft to land safely
  - Use Martian atmosphere, parachute, descent engines (controlled by software)
  - Software knows landed because of sensitive sensors on landing legs. Cut off engines when determine have landed.
  - But “noise” (false signals) by sensors generated when landing legs extended. Not in software requirements.
  - Software not supposed to be operating at that time but software engineers decided to start early to even out the load on processor.
  - Software thought spacecraft had landed and shut down descent engines while still 40 meters above surface

Warsaw A320 Accident

- Software protects against activating thrust reversers when airborne
- Hydroplaning and other factors made the software think the plane had not landed
- Pilots could not activate the thrust reversers and ran off end of runway into a small hill.

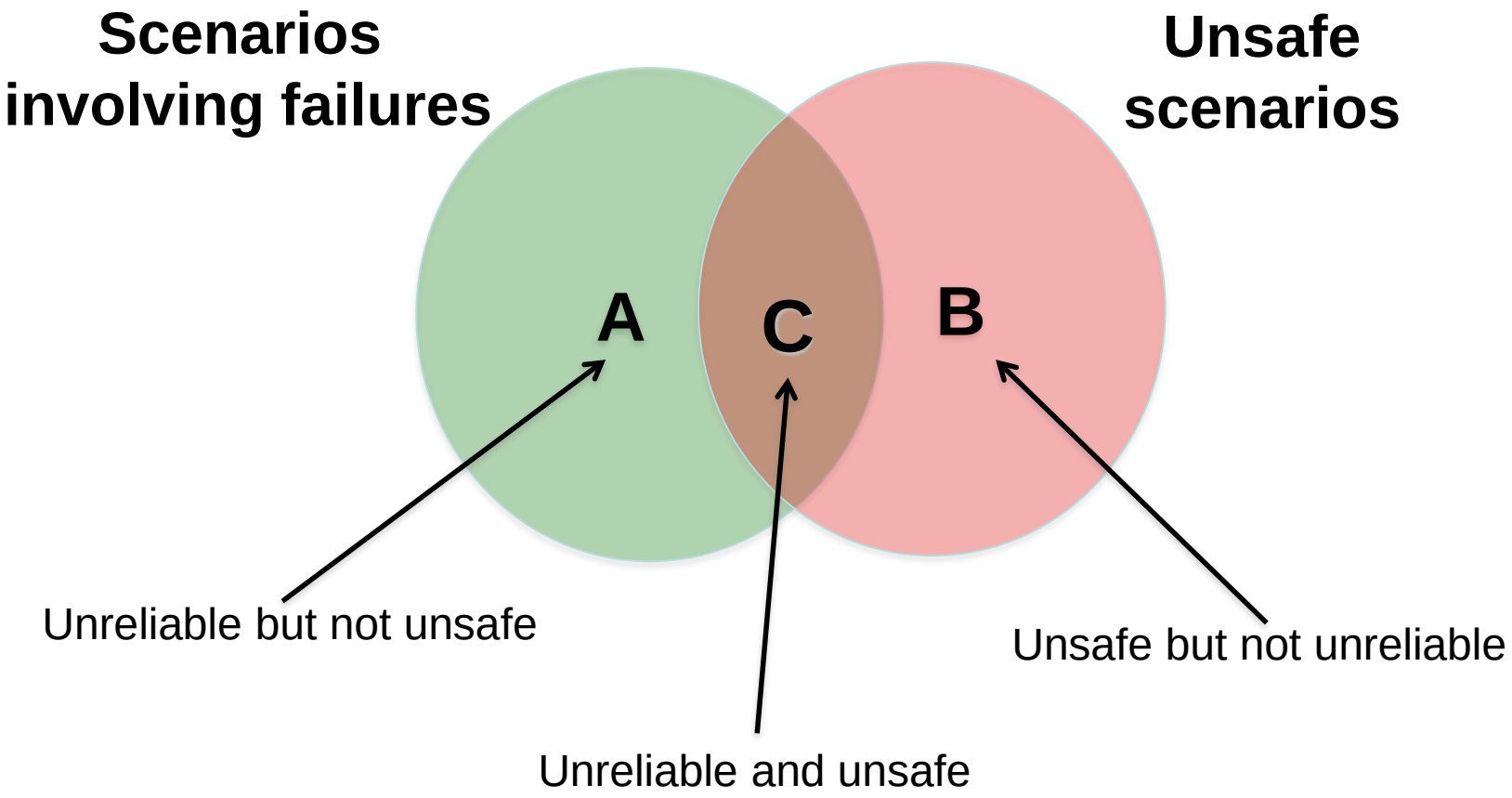
Washington State Ferry Problem

- Local rental car company installed a security device to prevent theft by disabling cars if car moved when engine stopped
- When ferry moved and cars not running, disabled them.
- Rental (and other) cars could not be driven off ferries when got to port

Two Types of Accidents

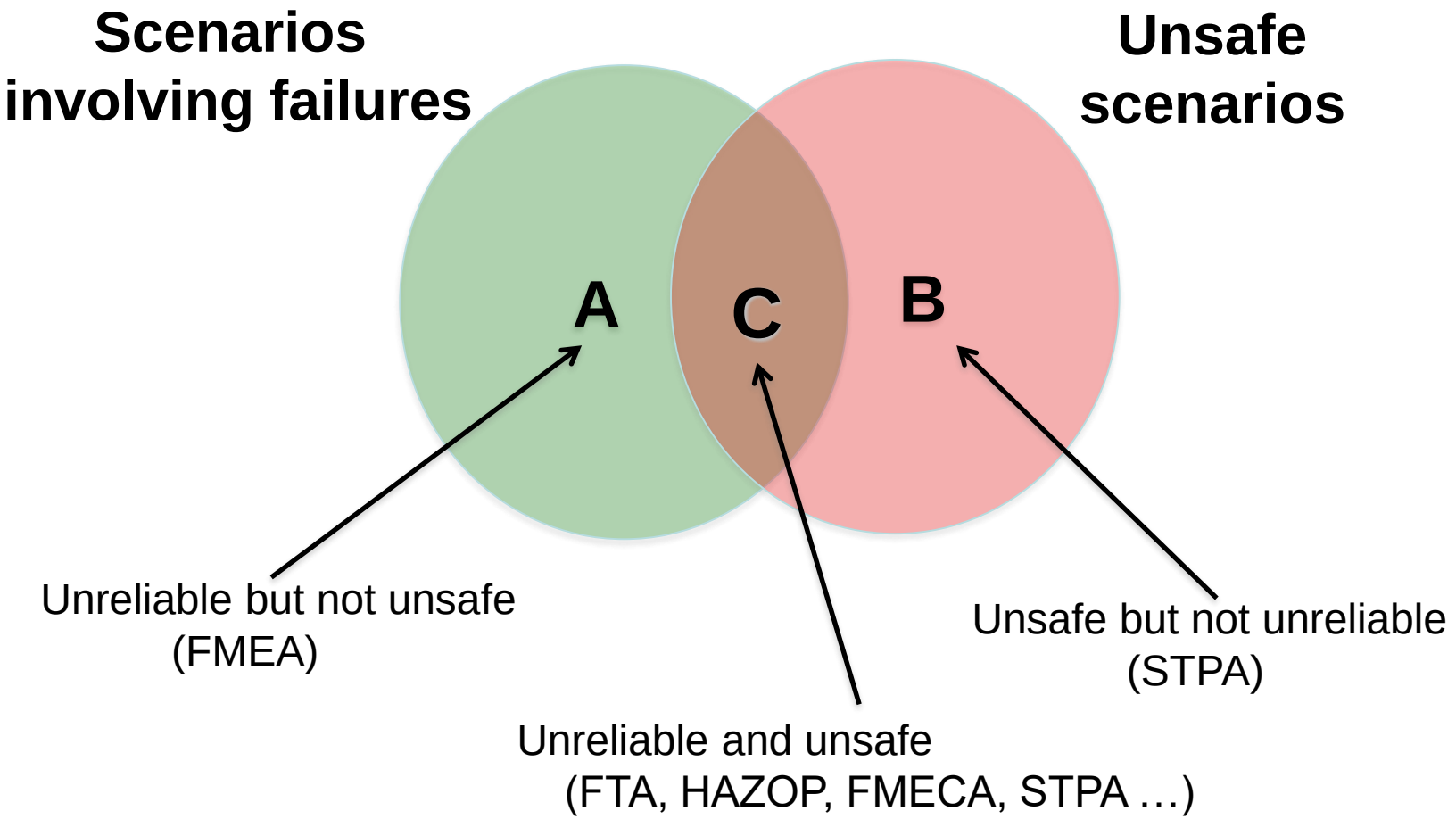
- **Component Failure Accidents**
  - Single or multiple component failures
  - Usually assume random failure
- **Component Interaction Accidents**
  - Arise in interactions among components
  - Related to complexity (coupling) in our system designs, which leads to design and system engineering errors
  - No components may have “failed”
  - Exacerbated by introduction of computers and software but the problem is system design errors
    - Software allows almost unlimited complexity in our designs

Confusing Safety and Reliability



Preventing Component or Functional Failures is Not Enough

Confusing Safety and Reliability



Preventing Component or Functional Failures is Not Enough

Software has revolutionized engineering

1. Software does not “fail”

General Purpose Machine + Software = Special Purpose Machine

Software is simply the design of a machine abstracted from its physical realization

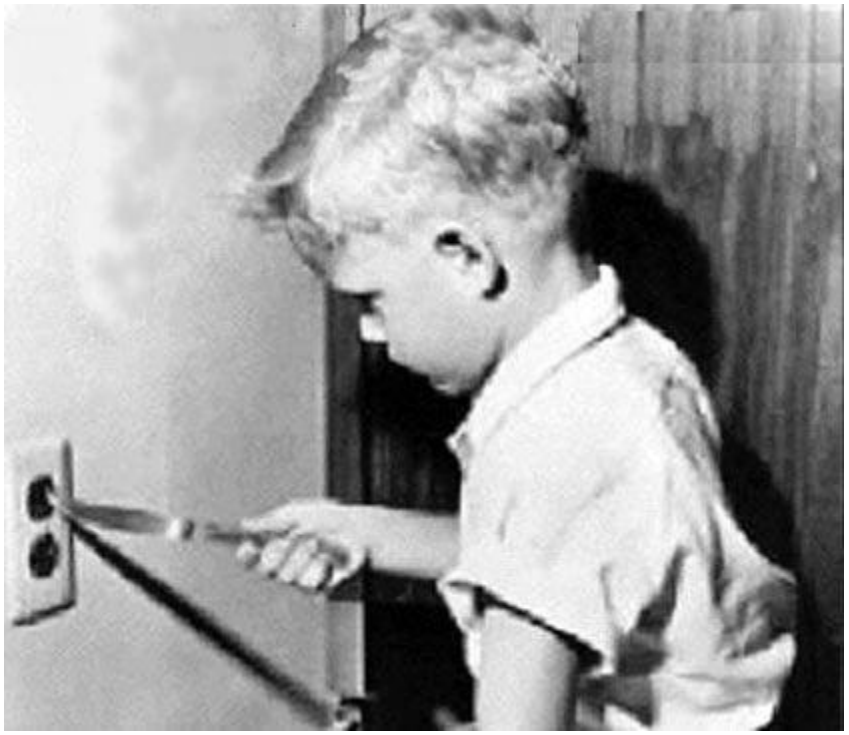
Software is pure design and designs do not “fail”

2. Software allows almost unlimited system complexity

- Can no longer
  - Plan, understand, anticipate, and guard against all undesired system behavior
  - Exhaustively test to get out all design errors
- **Context** determines whether software is safe
  - Ariane 4 software was safe but when reused in Ariane 5, the spacecraft exploded
  - “SIL” (safety integrity level) concept is technically meaningless
  - Not possible to look at software alone and determine “safety”

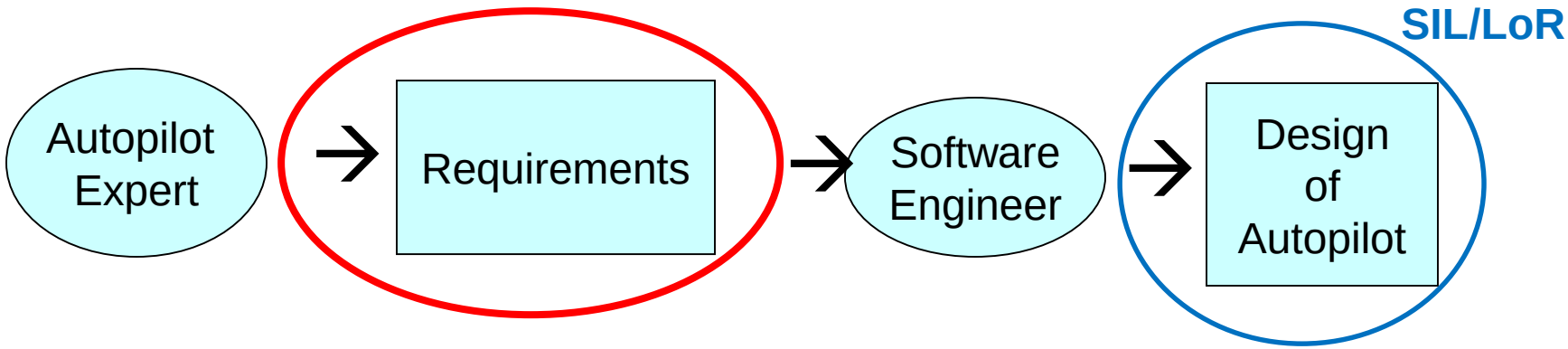
Safe or Unsafe?

Safety Depends on Context



3. The role of software in accidents almost always involves flawed requirements

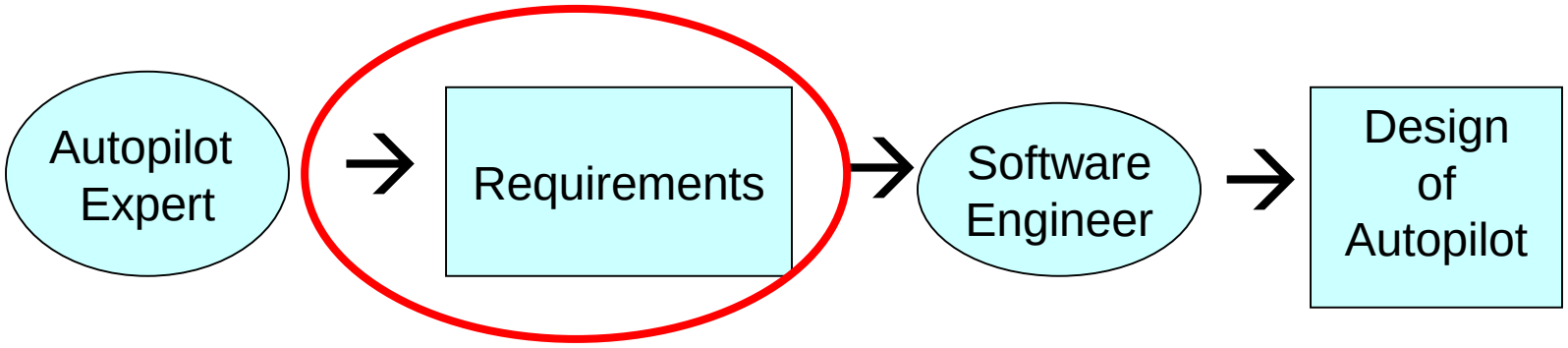
- Incomplete or wrong assumptions about operation of controlled system or required operation of computer
- Unhandled controlled-system states and environmental conditions



- Level of rigor in producing the software design or DAL (design assurance level) has almost nothing to do with system safety.

3. The role of software in accidents almost always involves flawed requirements

- Incomplete or wrong assumptions about operation of controlled system or required operation of computer
- Unhandled controlled-system states and environmental conditions



- Software assurance shows that the software implements the specified requirements
- In losses related to software, it is usually doing exactly what the software engineers intended it to do

Lesson Learned

- Accidents today do not just result from component failures. Need to consider design errors
- Software
  - Contributes differently to accidents than hardware
    - Does not “fail” but can contribute to unsafe system behavior (including unsafe human behavior)
  - Adds almost unlimited complexity but
    - Cannot exhaustively test
    - Is not by itself safe or unsafe

Software changes the role of humans in systems

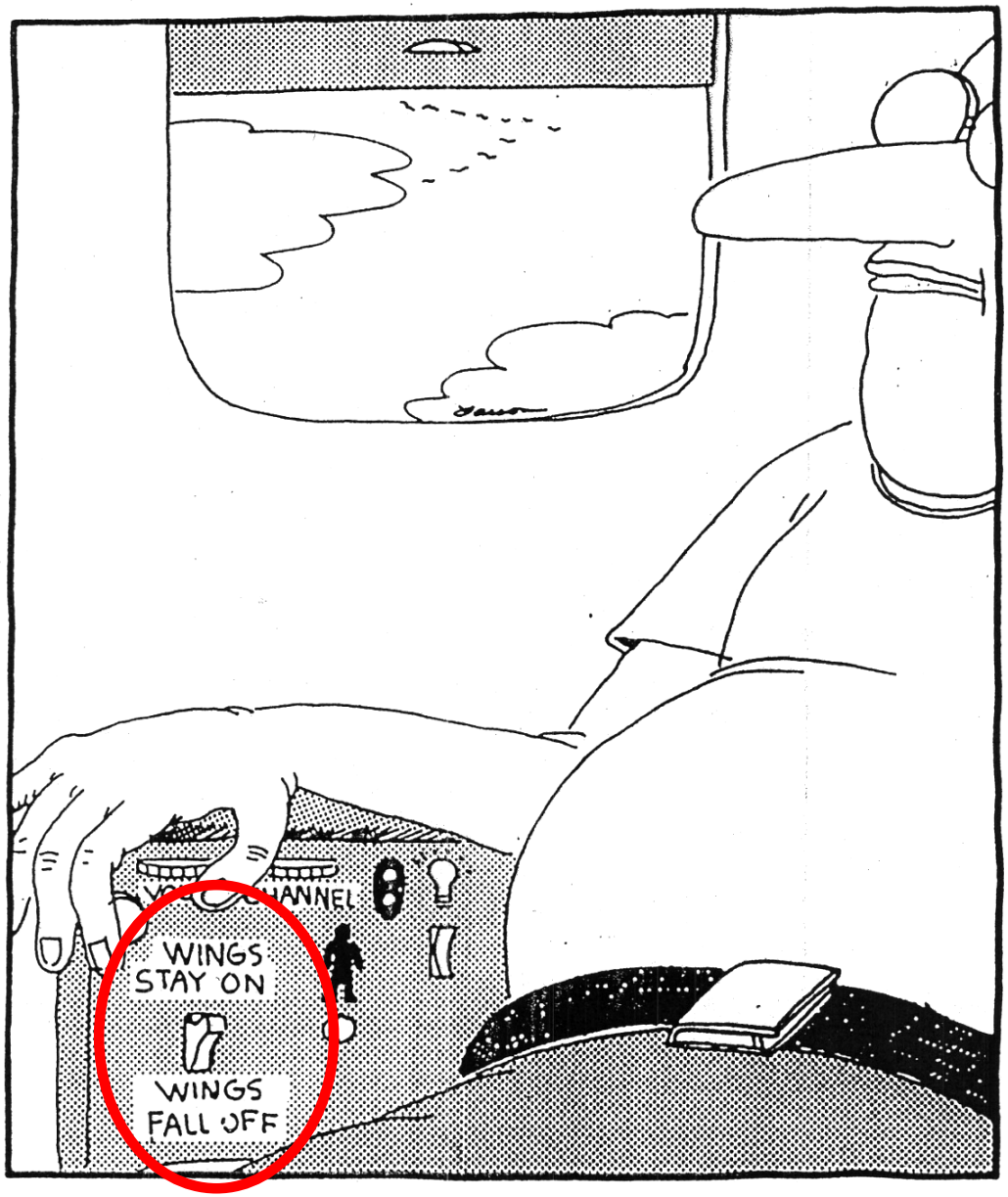
Traditional Approach

Typical assumption is that operator error is cause of most incidents and accidents

- So do something about operator involved (admonish, fire, retrain them)
- Or do something about operators in general
  - Marginalize them by putting in more automation
  - Rigidify their work by creating more rules and procedures

"Cause" from the American Airlines B-757 accident report (in Cali, Columbia):

> "Failure of the flight crew to revert to basic radio navigation at the time when the FMS-assisted navigation became confusing and demanded an excessive workload in a critical phase of flight."



Fumbling for his recline button Ted unwittingly instigates a disaster

Another Accident Involving Thrust Reversers

- Tu-204, Moscow, 2012
- Red Wings Airlines Flight 9268
- The soft 1.12g touchdown made runway contact a little later than usual.
- With the crosswind, this meant weight-on-wheels switches did not activate and the thrust-reverse system would not deploy.

Another Accident Involving Thrust Reversers

- Pilots believe the thrust reversers are deploying like they always do. With the limited runway space, they quickly engage high engine power to stop quicker. Instead this accelerated the Tu-204 forwards, eventually colliding with a highway embankment.

Another Accident Involving Thrust Reversers

- Pilots believe the thrust reversers are deploying like they always do. With the limited runway space, they quickly engage high engine power to stop quicker. Instead this accelerates the Tu-204 forwards, eventually colliding with a highway embankment.

In complex systems, human and technical considerations cannot be isolated

Human factors concentrates on the “screen out”

Hardware/Software engineering concentrates on the “screen in”

Not enough attention on integrated system as a whole

(e.g, mode confusion, situation awareness errors, inconsistent behavior, etc.

A New Systems View of Operator Error

- Operator error is a symptom, not a cause
- All behavior affected by context (system) in which occurs
  - Role of operators is changing in software-intensive systems as is the errors they make
  - Designing systems in which operator error inevitable and then blame accidents on operators rather than designers
- To do something about operator error, must look at system in which people work:
  - Design of equipment
  - Usefulness of procedures
  - Existence of goal conflicts and production pressures
- **Human error is a symptom of a system that needs to be redesigned**

Lesson Learned

- Cannot effectively reduce accidents without integrating human/software/hardware engineering

Part 2: STAMP and Systems Theory

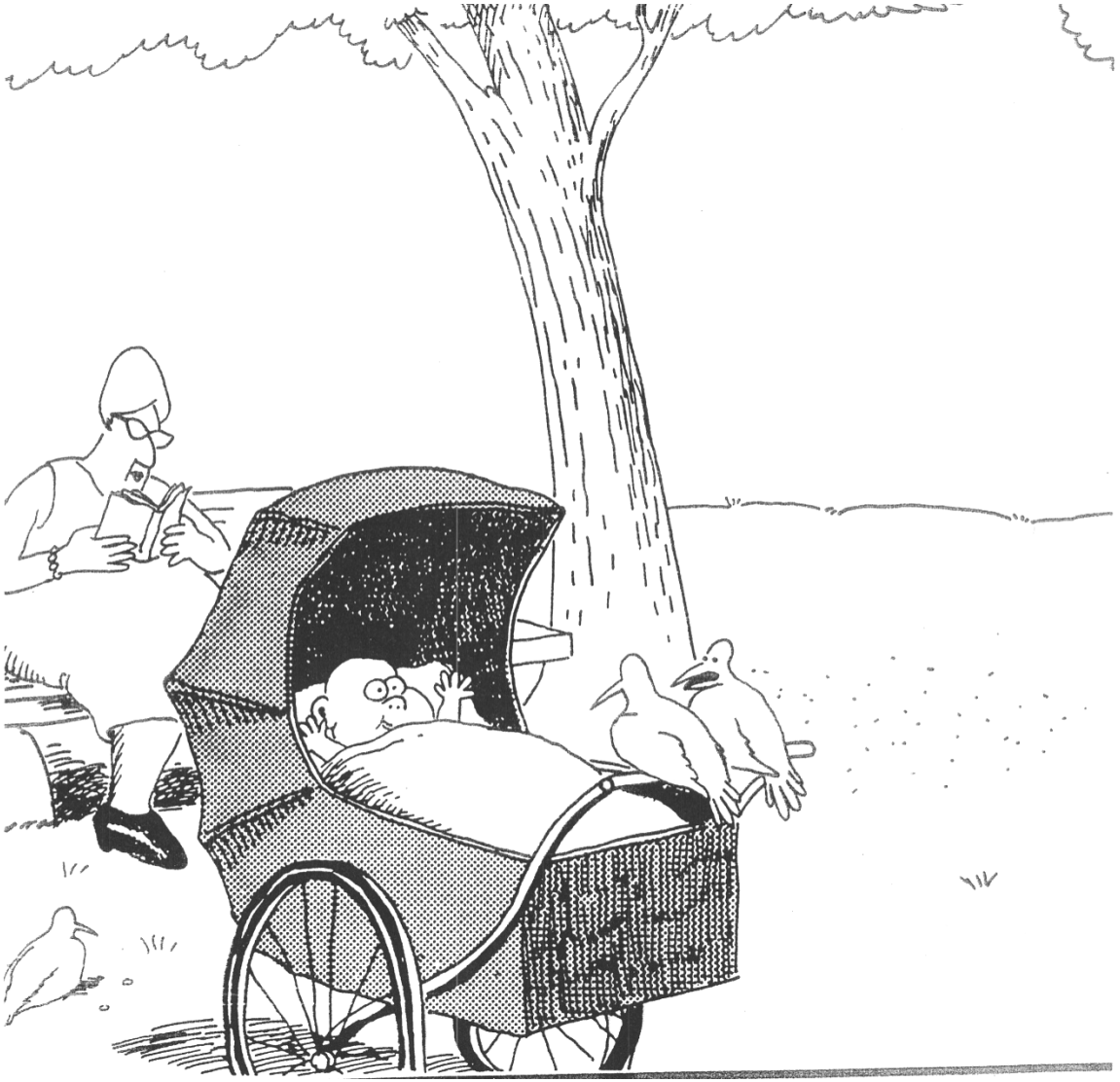
Summary of Lessons Learned

- Need to look beyond events to prevent accidents
- Accidents today do not just result from component failures. Need to consider design errors
- Software
  - Contributes differently to accidents than hardware
  - The problem is in identifying safety-related requirements
- Cannot effectively tackle system safety without integrating human/software/hardware engineering.

The Problem to be Solved:

- We need models and tools that handle:
  - Hardware and hardware failures
  - Software (particularly requirements)
  - Human factors
  - Interactions among system components
  - System design errors
  - Management, regulation, policy
  - Environmental factors
  - “Unknown unknowns”

And the interactions among all these things



It's still hungry … and I've been stuffing worms into it all day.

We Need New Tools for the New Problems

Paradigm Change

- Does not imply what previously done is wrong and new approach correct
- Einstein:

  "Progress in science (moving from one paradigm to another) is like climbing a mountain"

As move further up, can see farther than on lower points

Paradigm Change (2)

New perspective does not invalidate the old one, but extends and enriches our appreciation of the valleys below

Value of new paradigm often depends on ability to accommodate successes and empirical observations made in old paradigm.

New paradigms offer a broader, richer perspective for interpreting previous answers.

Bottom Line: We Need Something New

- Two approaches being taken now:

Pretend there is no problem

Shoehorn new technology and new levels of complexity into old methods

New levels of complexity are creating new problems that cannot be solved using traditional techniques.

What is STAMP and how does it differ from what people do now?

- A new causality model based on system theory
- Can build more powerful new tools based on it.

The Problem is Complexity

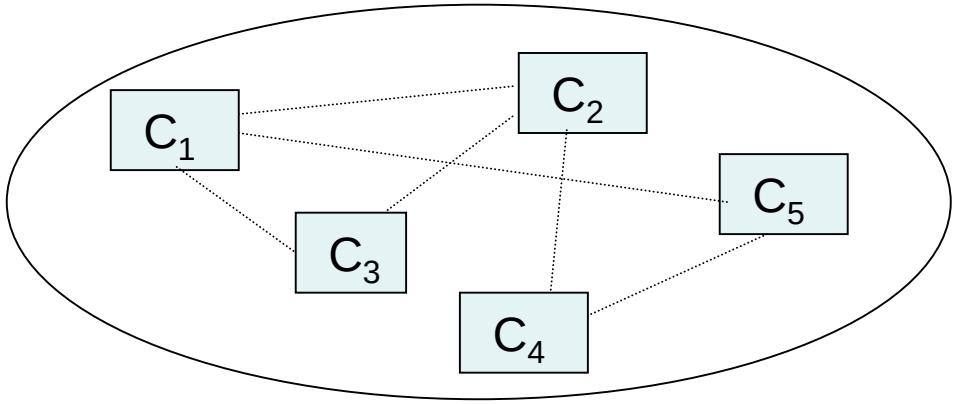
Ways to Cope with Complexity

- Analytic Decomposition
- Statistics
- Systems Theory

Analytic Decomposition ("Divide and Conquer")

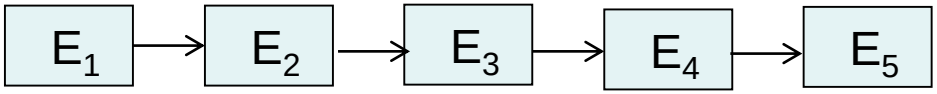
1. Divide system into separate parts

Physical/Functional: Separate into distinct components



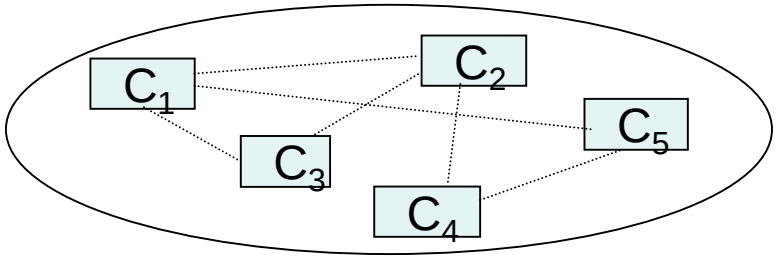
Components interact
In direct ways

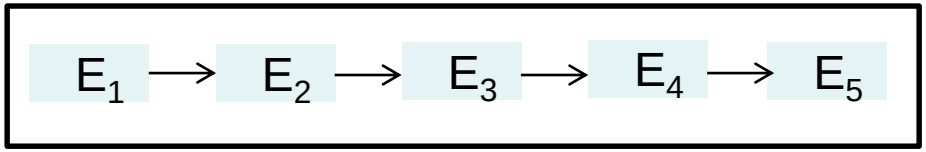
Behavior: Separate into events over time



Each event is the direct
result of the preceding event

Analytic Decomposition (2)

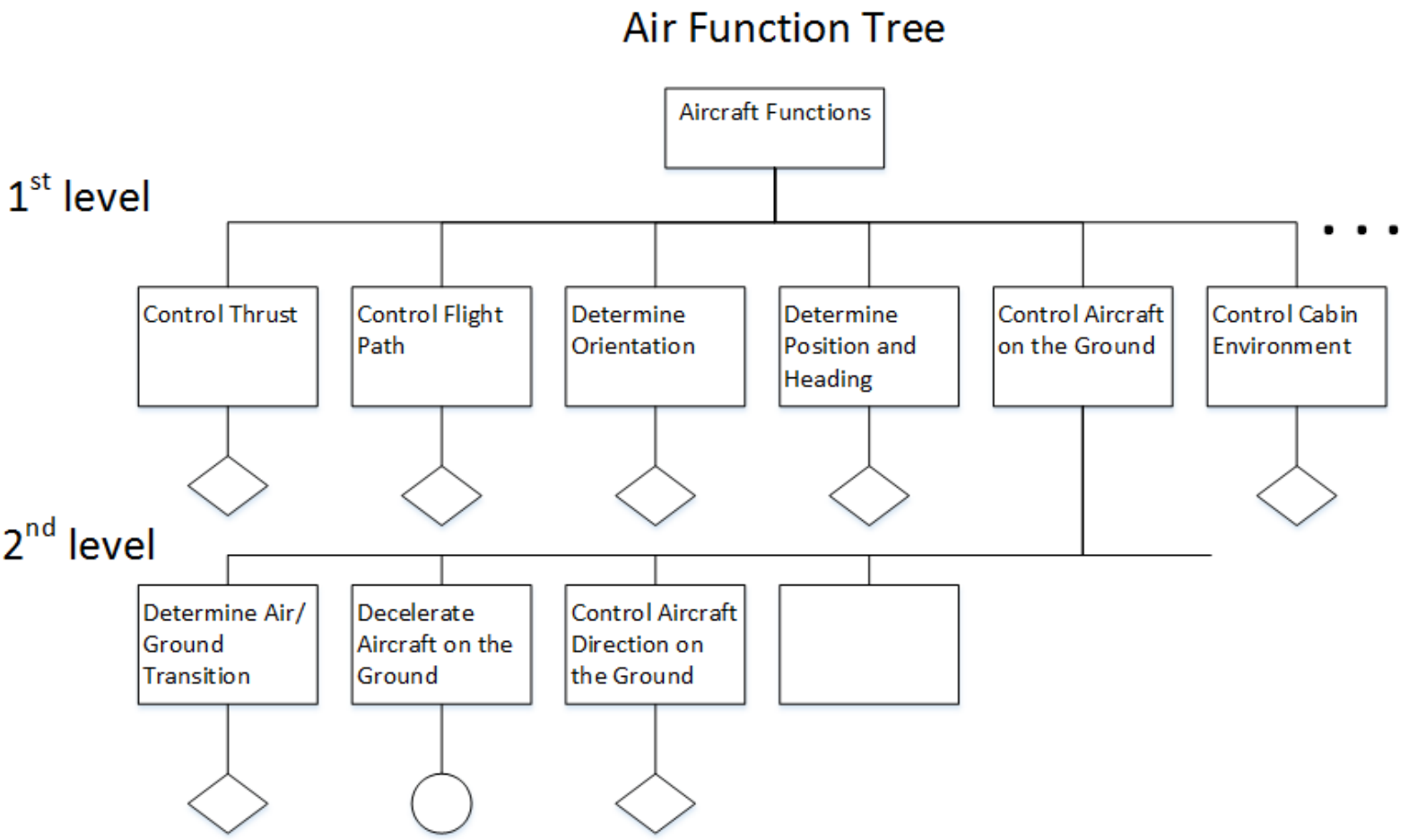




2. Analyze/examine pieces separately and combine results

- Assumes such separation does not distort phenomenon
  - ✓ Each component or subsystem operates independently
  - ✓ Components act the same when examined singly as when playing their part in the whole
  - ✓ Components/events not subject to feedback loops and non-linear interactions
  - ✓ Interactions can be examined pairwise

Typical Decomposition Approach (ARP 4761)

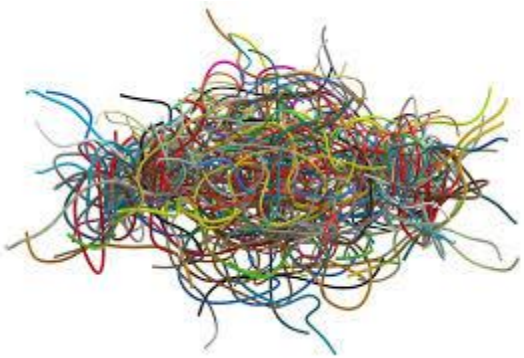


- ARP 4761A adding interactions among “failures” of functions but that is not the problem. Still bottom up.

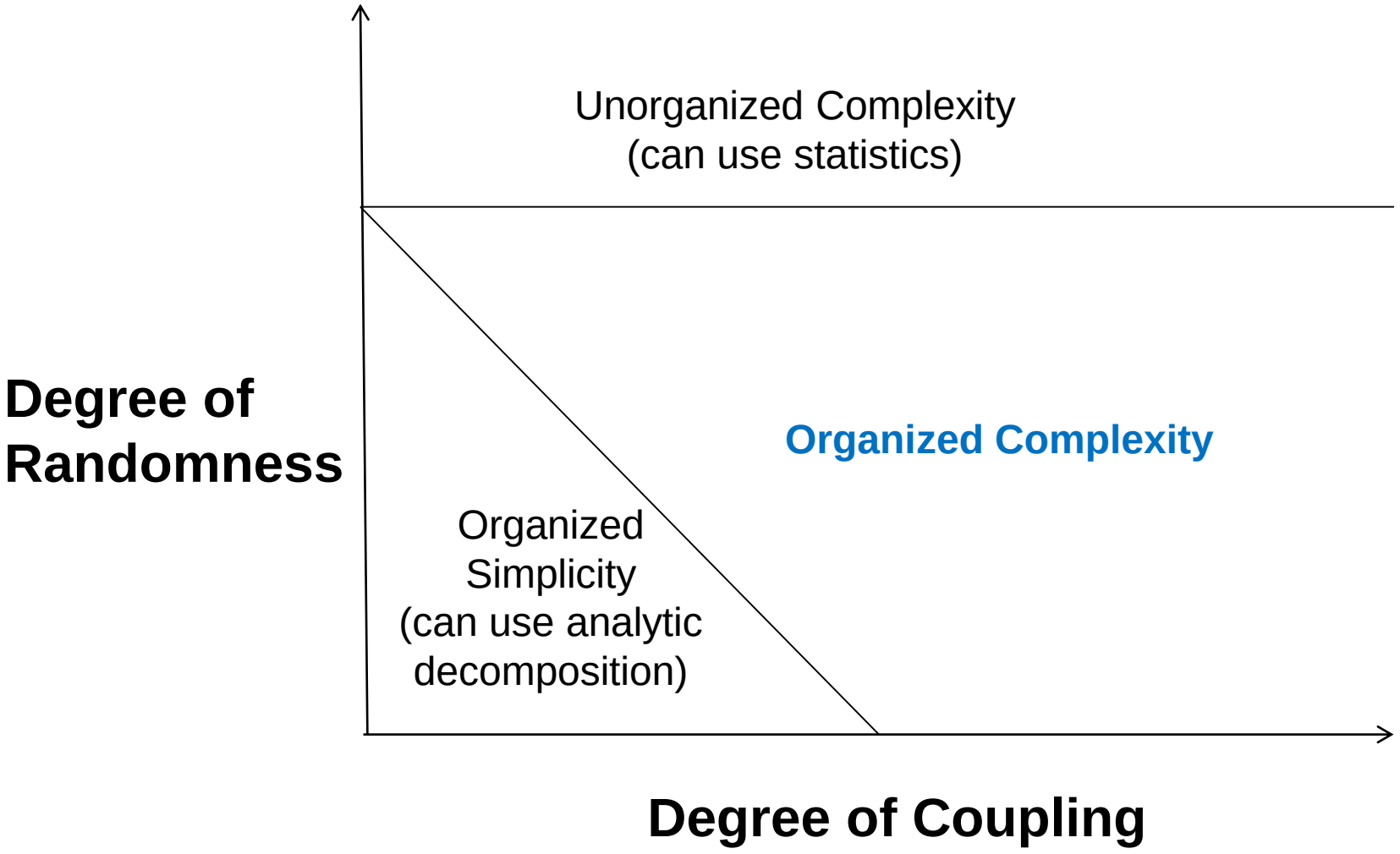
The Problem

- These assumptions are no longer true in our
  - Tightly coupled
  - Software intensive
  - Highly automated
  - Connected

  engineered systems



- Need a new theoretical basis
  - *System theory* can provide it



[Credit to Gerald Weinberg]

Here comes the paradigm change!

Systems Theory

- Developed for systems that are
  - Too complex for complete analysis
    - Separation into (interacting) subsystems distorts the results
    - The most important properties are emergent
  - Too organized for statistics
    - Too much underlying structure that distorts the statistics
    - New technology and designs have no historical information
- First used on ICBM systems of 1950s/1960s

System Theory was created to provide a more powerful way to deal with complexity

Systems Theory (2)

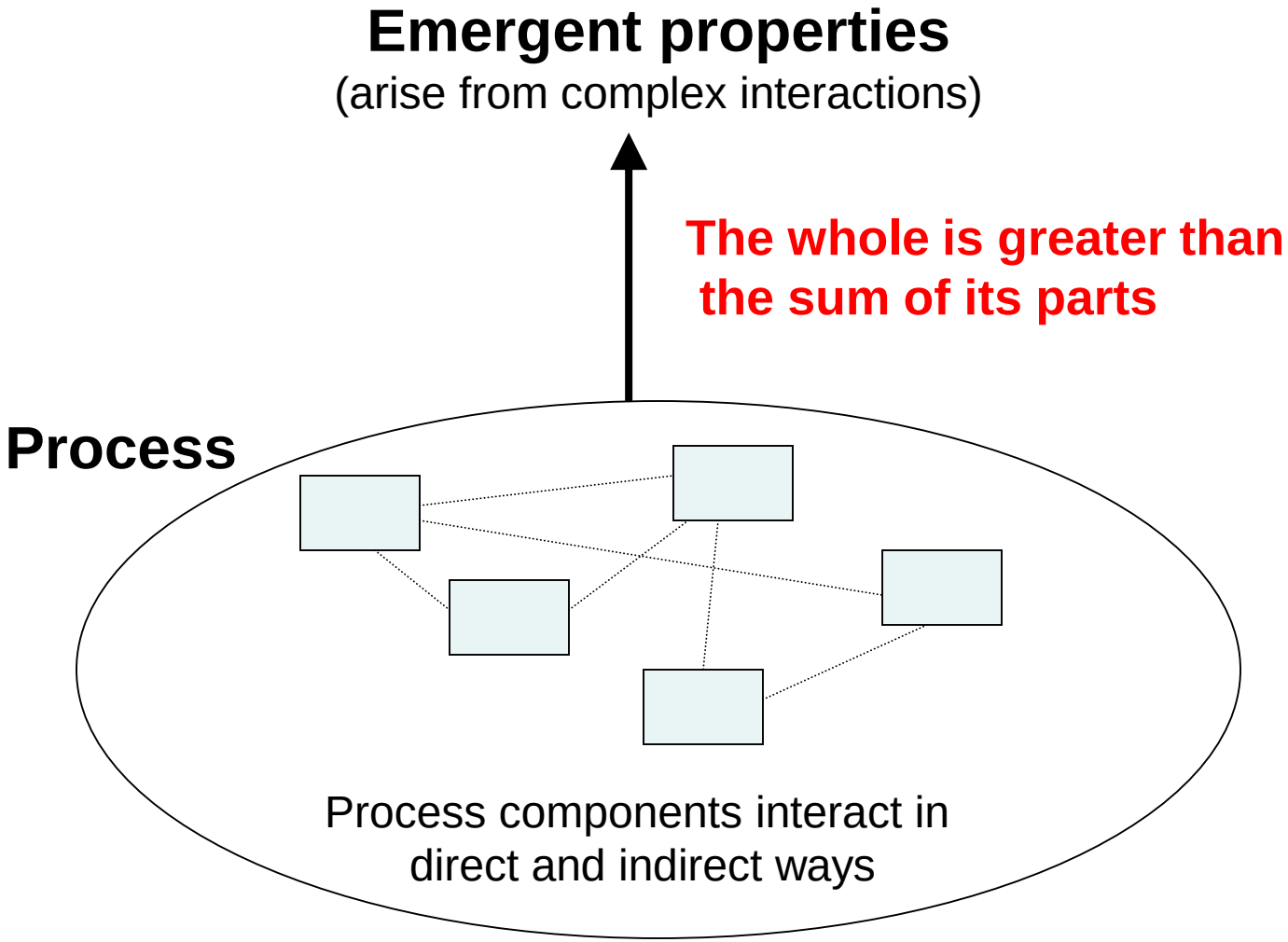
- Focuses on systems taken as a whole, not on parts taken separately
- Emergent properties
  - Some properties can only be treated adequately in their entirety, taking into account all social and technical aspects

    "The whole is greater than the sum of the parts"

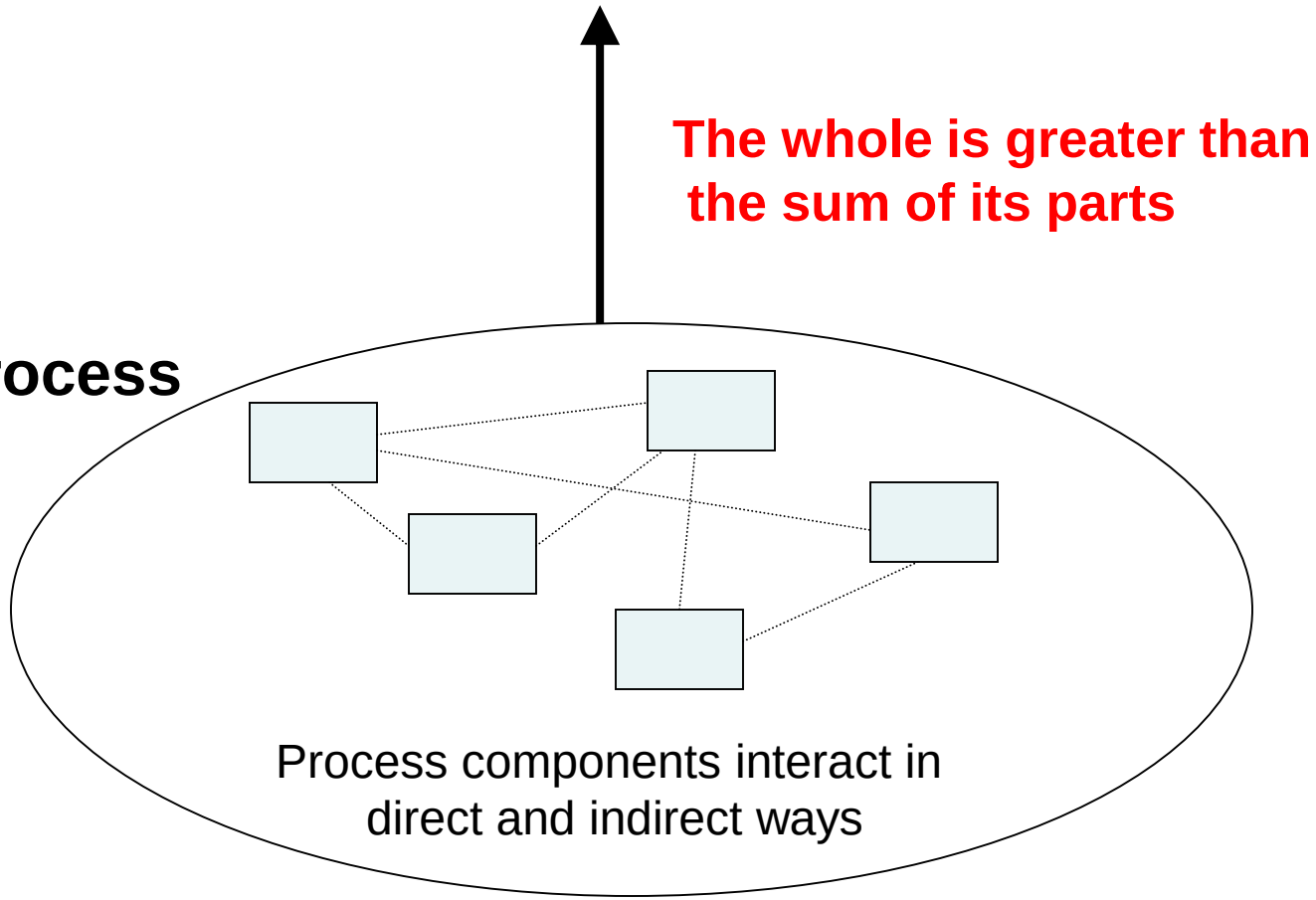
  - These properties arise from relationships among the parts of the system

    How they interact and fit together

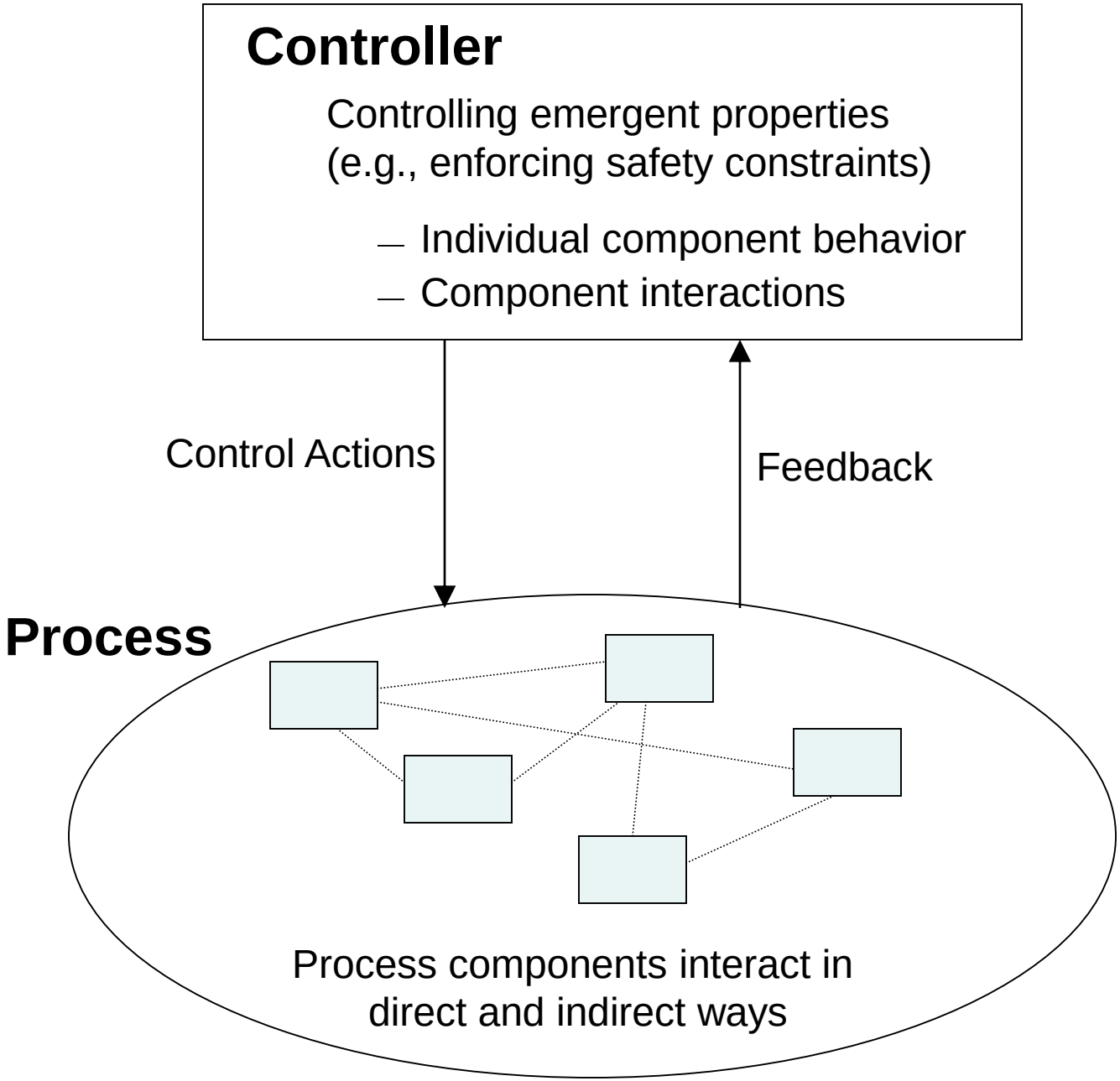
System Theory



Emergent properties
(arise from complex interactions)



Safety and security are emergent properties

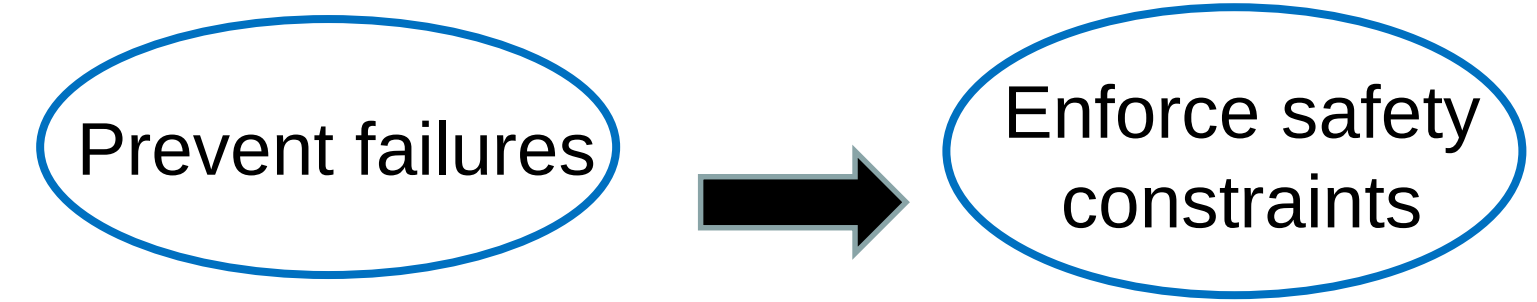

Controller
Controlling emergent properties
(e.g., enforcing safety constraints)
– Individual component behavior
– Component interactions
Control Actions
Feedback
Process
Process components interact in
direct and indirect ways

Controls/Controllers Enforce Safety Constraints

- Power must never be on when access door open (e.g, lockout/tagout)
- Two aircraft/automobiles must not violate minimum separation
- Aircraft must maintain sufficient lift to remain airborne
- Integrity of hull must be maintained on a submarine
- Toxic chemicals/radiation must not be released from plant
- Workers must not be exposed to workplace hazards
- Public health system must prevent exposure of public to contaminated water and food products
- Pressure in a offshore well must be controlled

These are the High-Level Functional Safety/Security Requirements to Address During Design

The paradigm change for effective safety and security engineering!



Treat Safety as a **Reliability** Problem

Treat Safety as a **Control** Problem

A Broad View of “Control”

Component failures and unsafe interactions may be “controlled” through design

(e.g., redundancy, interlocks, fail-safe design)

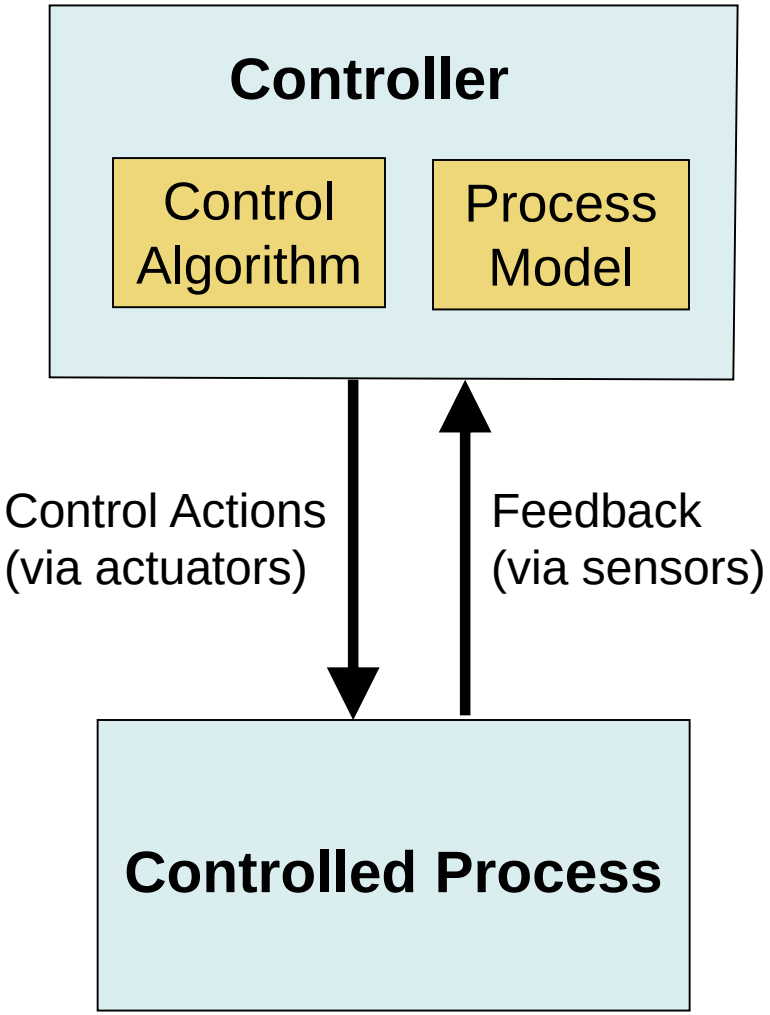
or through process

- Manufacturing processes and procedures
- Maintenance processes
- Operational processes

or through social controls

- Governmental or regulatory
- Culture
- Insurance
- Law and the courts
- Individual self-interest (incentive structure)

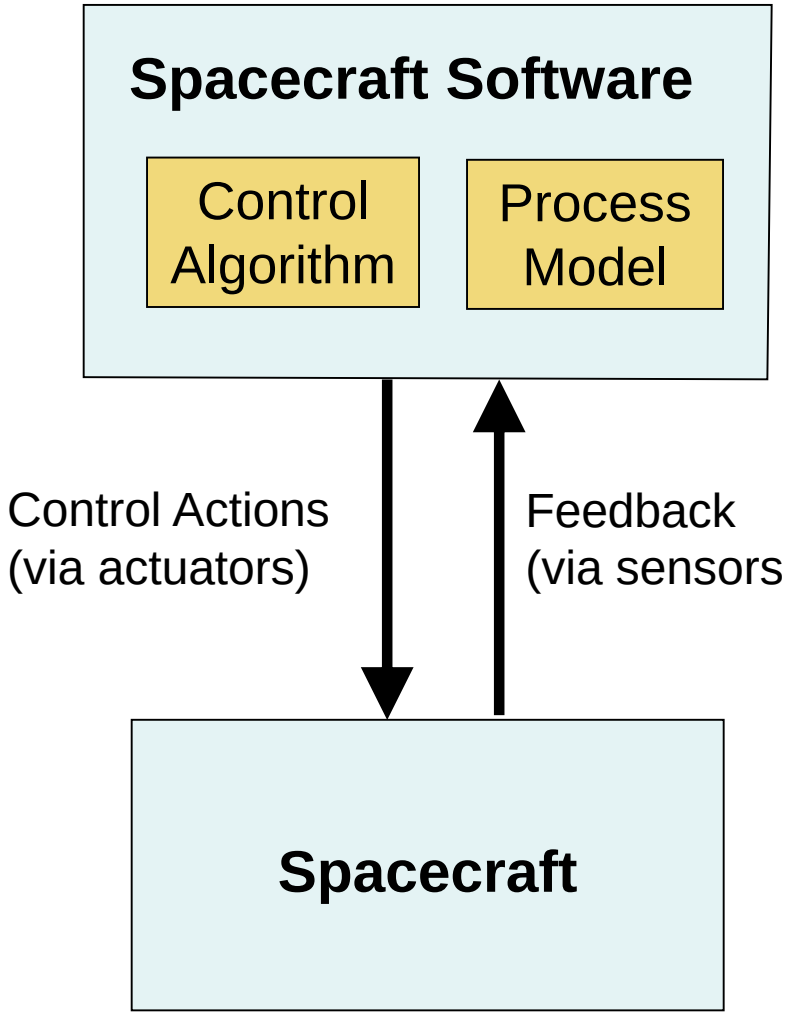
Treating Safety as a Control Problem



- Controllers use a **process model** to determine control actions
- Software/human related accidents often occur when the process model is incorrect
- Captures software errors, human errors, flawed requirements …

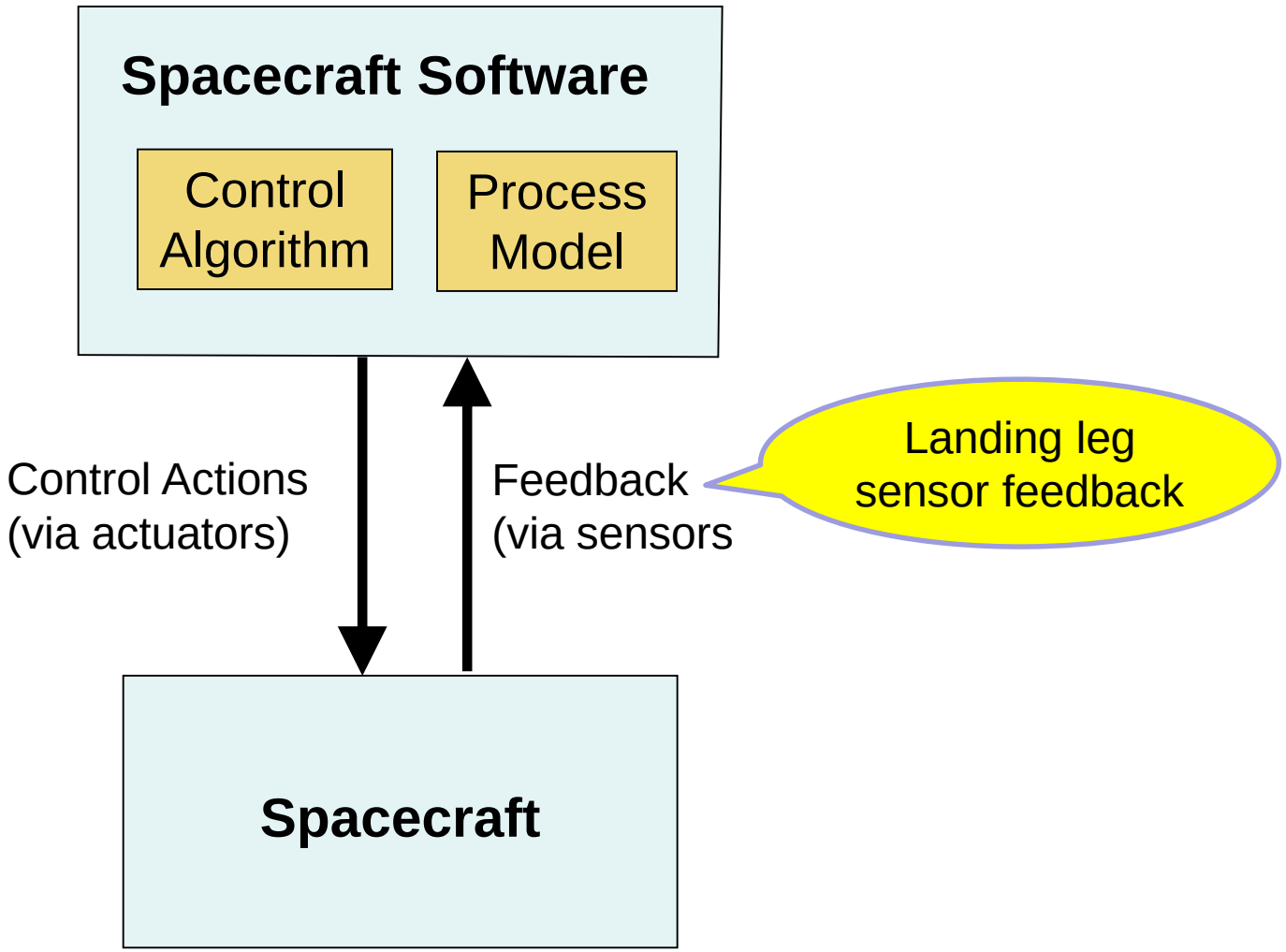
Mars Polar Lander

Hazard: landing on planet with too much force



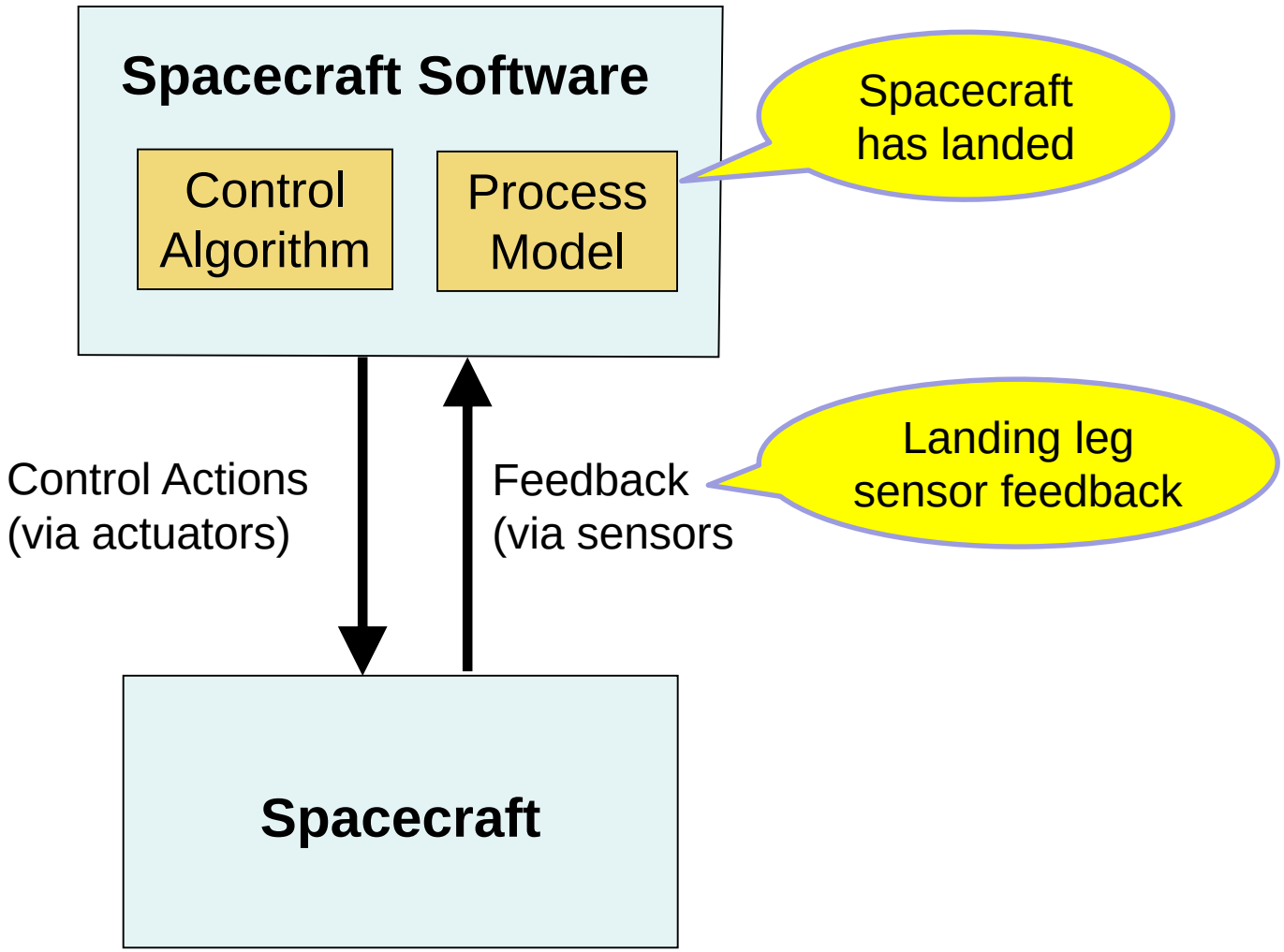
Mars Polar Lander

Hazard: landing on planet with too much force



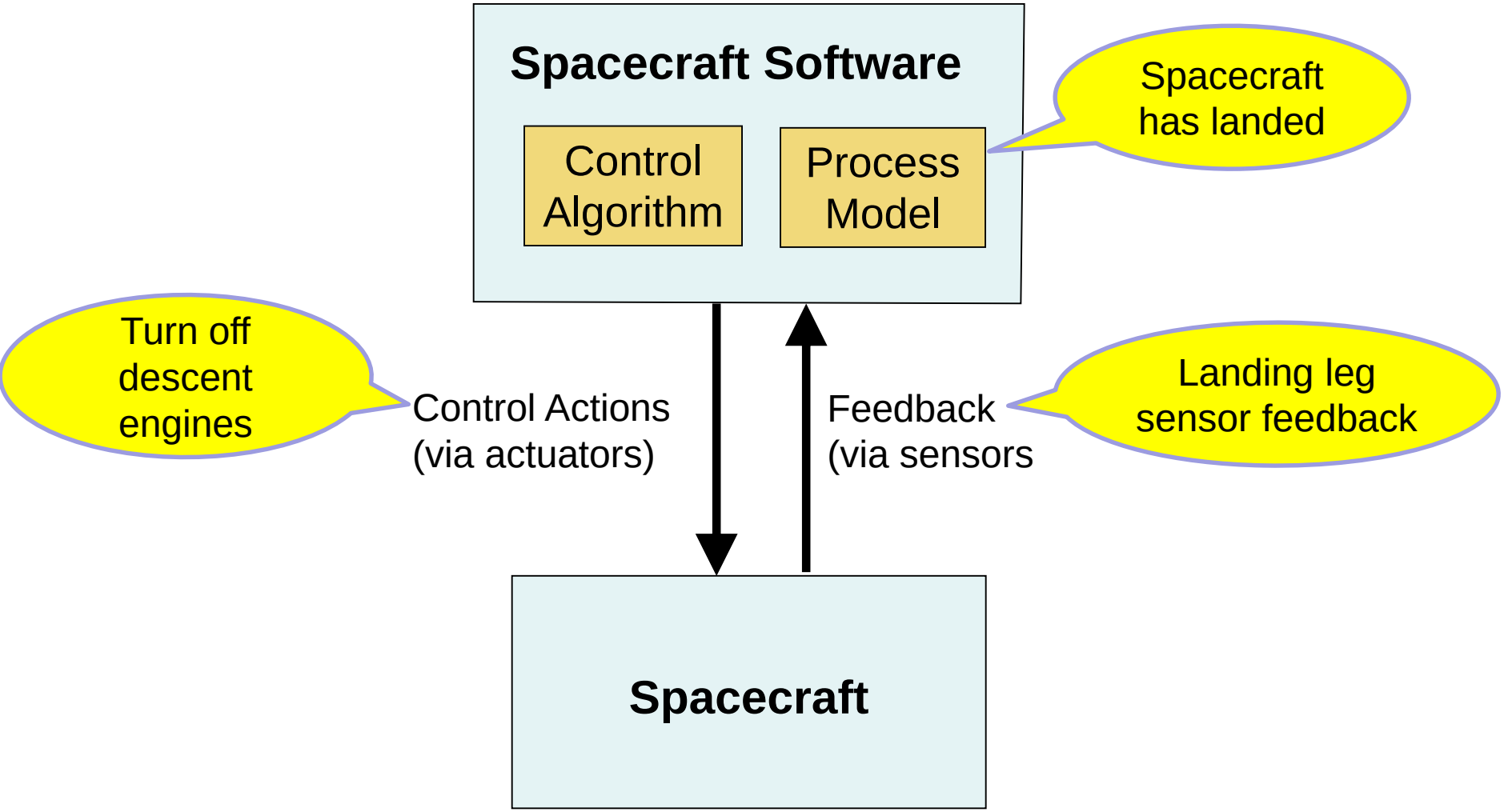
Mars Polar Lander

Hazard: landing on planet with too much force



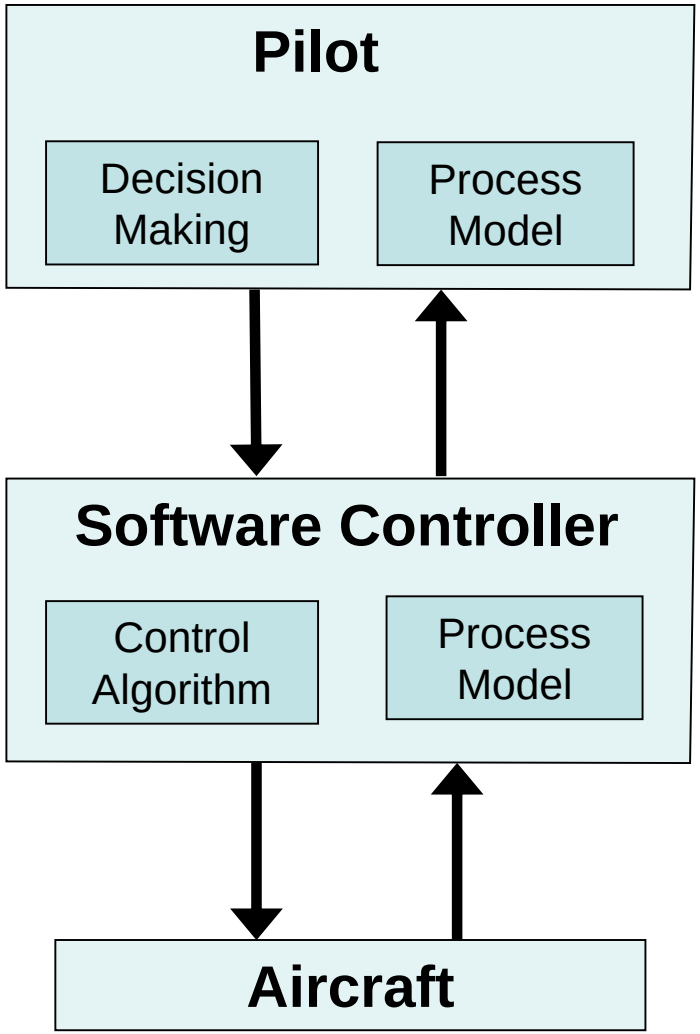
Mars Polar Lander

Hazard: landing on planet with too much force



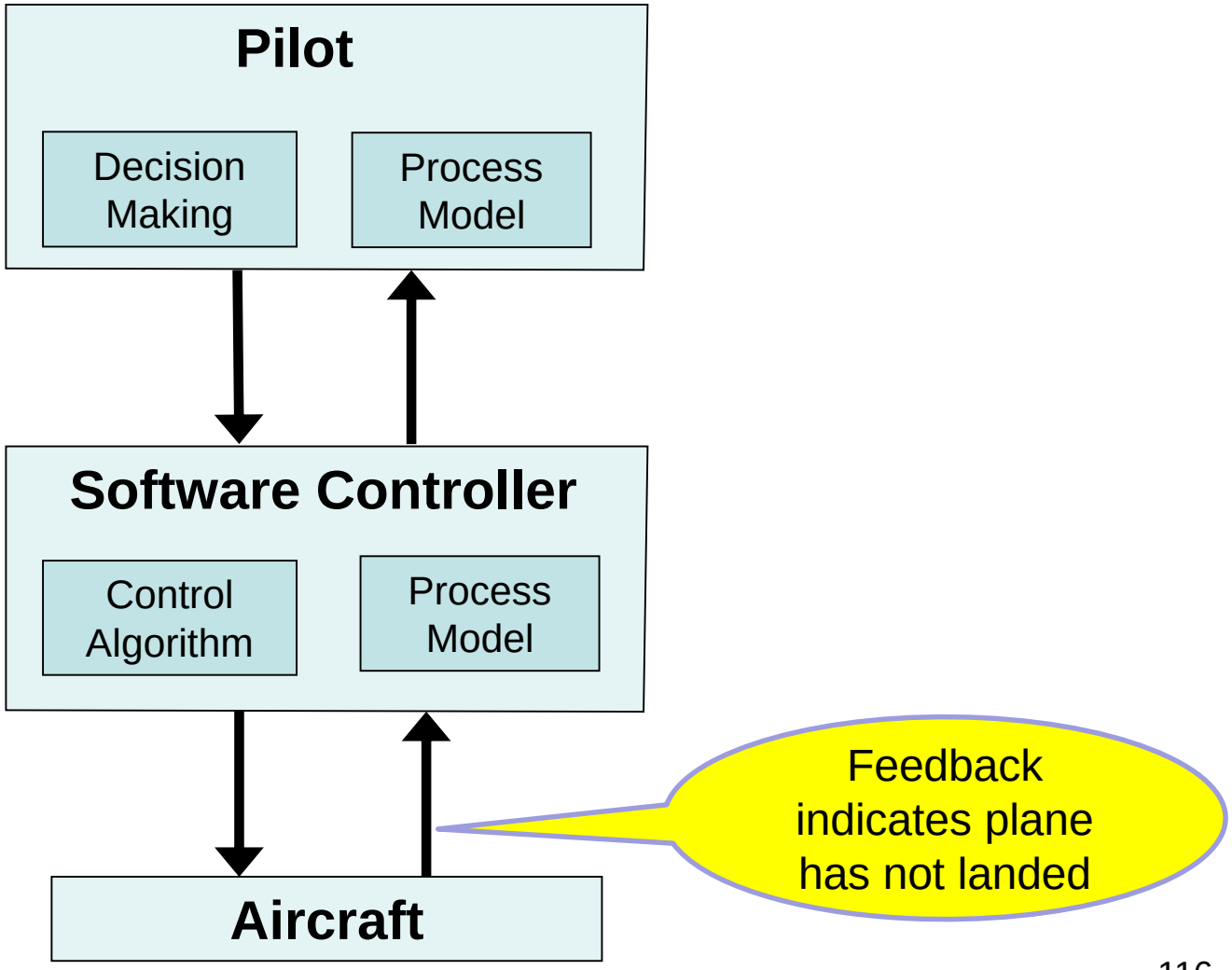
Warsaw (Reverse Thrusters)

Hazard: Inadequate aircraft deceleration after landing



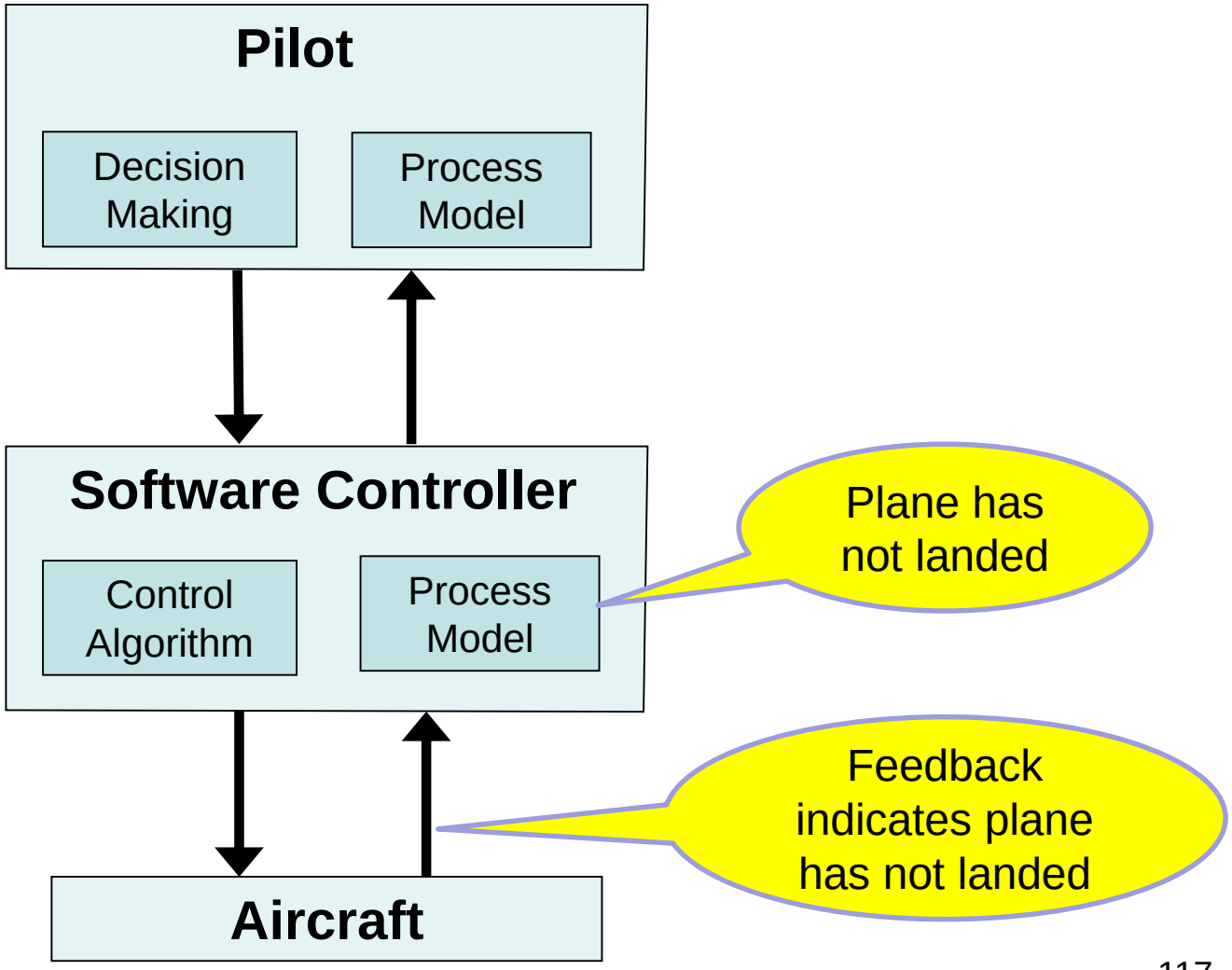
Warsaw (Reverse Thrusters)

Hazard: Inadequate aircraft deceleration after landing



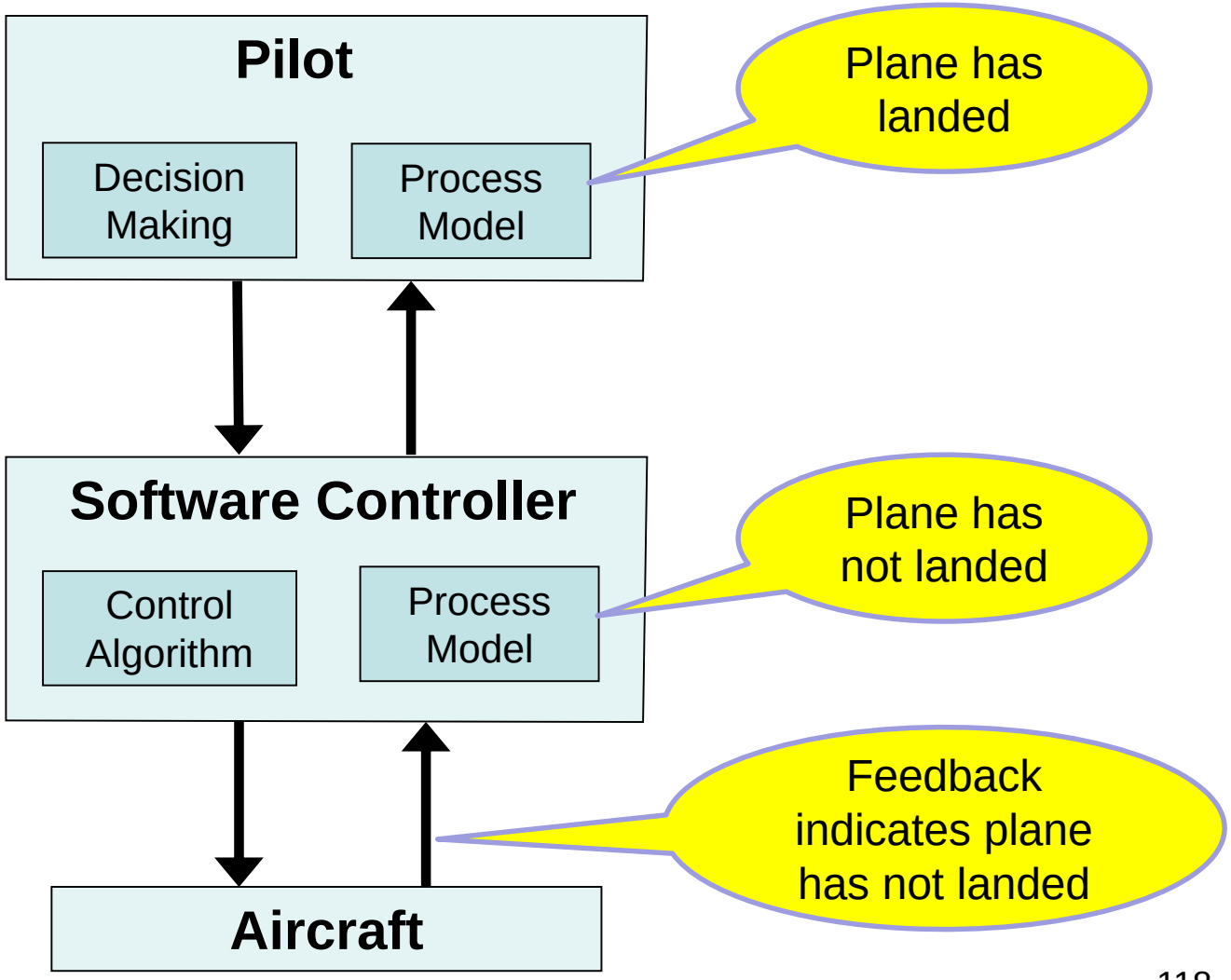
Warsaw (Reverse Thrusters)

Hazard: Inadequate aircraft deceleration after landing



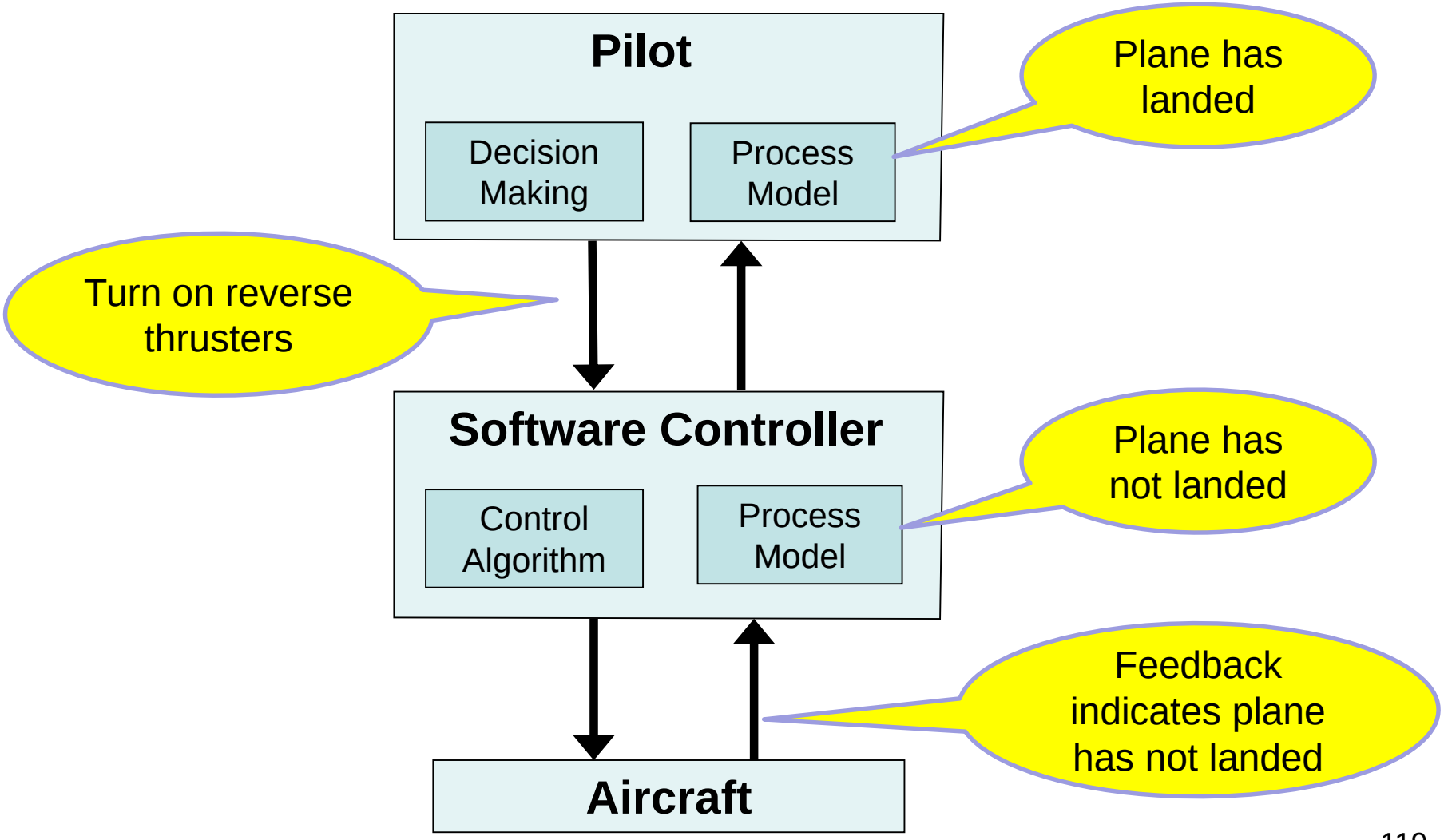
Warsaw (Reverse Thrusters)

Hazard: Inadequate aircraft deceleration after landing



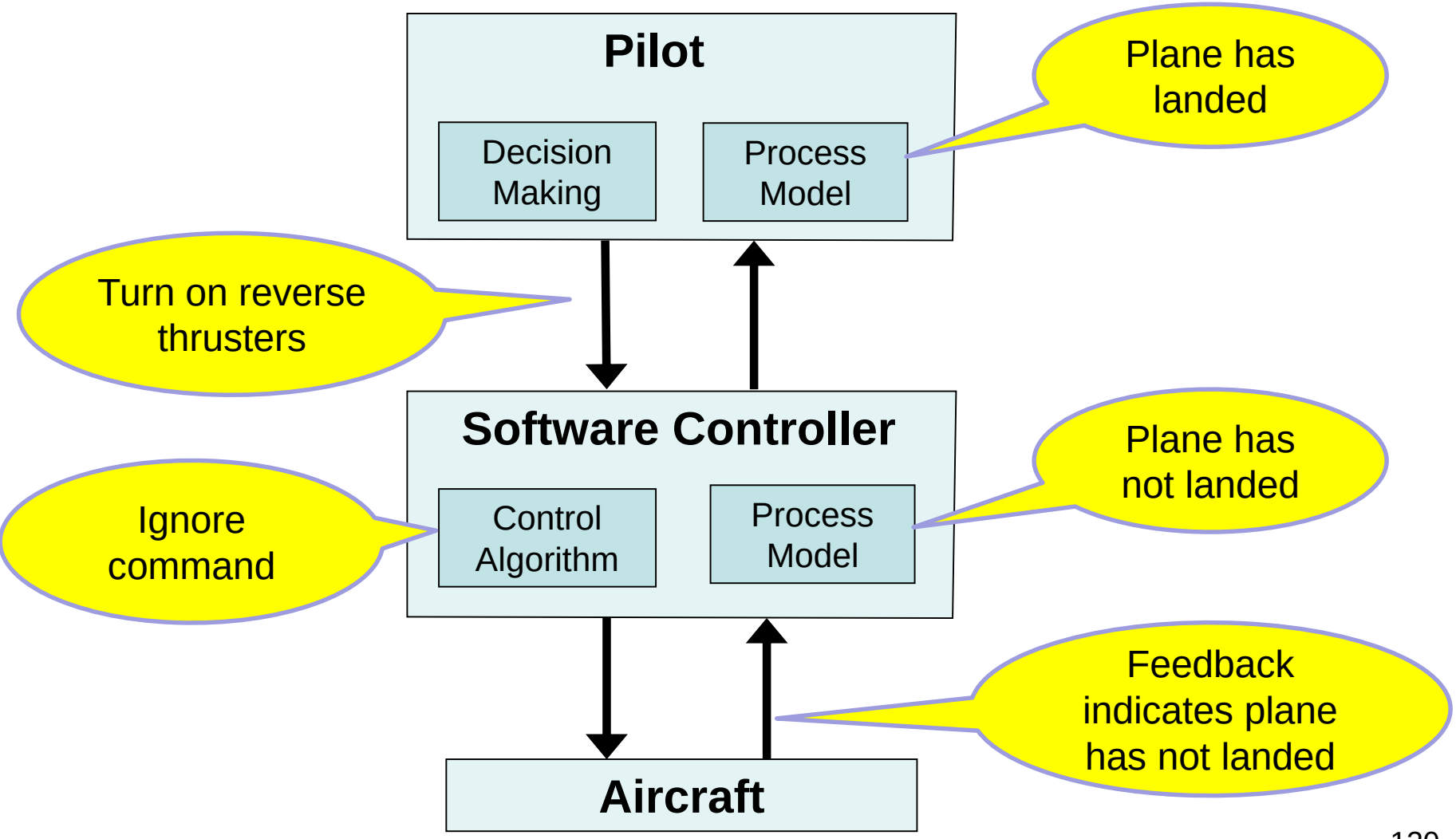
Warsaw (Reverse Thrusters)

Hazard: Inadequate aircraft deceleration after landing



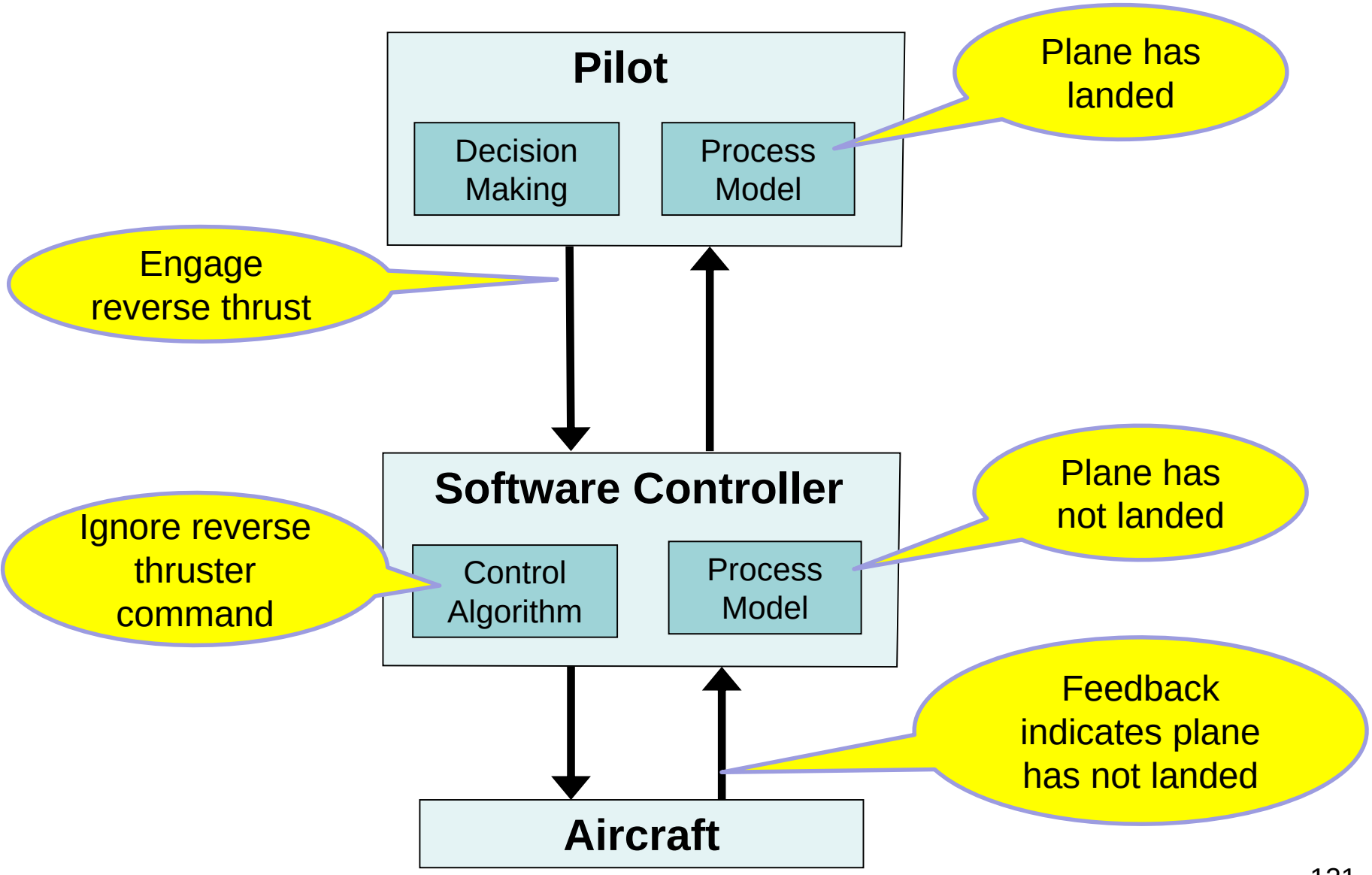
Warsaw (Reverse Thrusters)

Hazard: Inadequate aircraft deceleration after landing



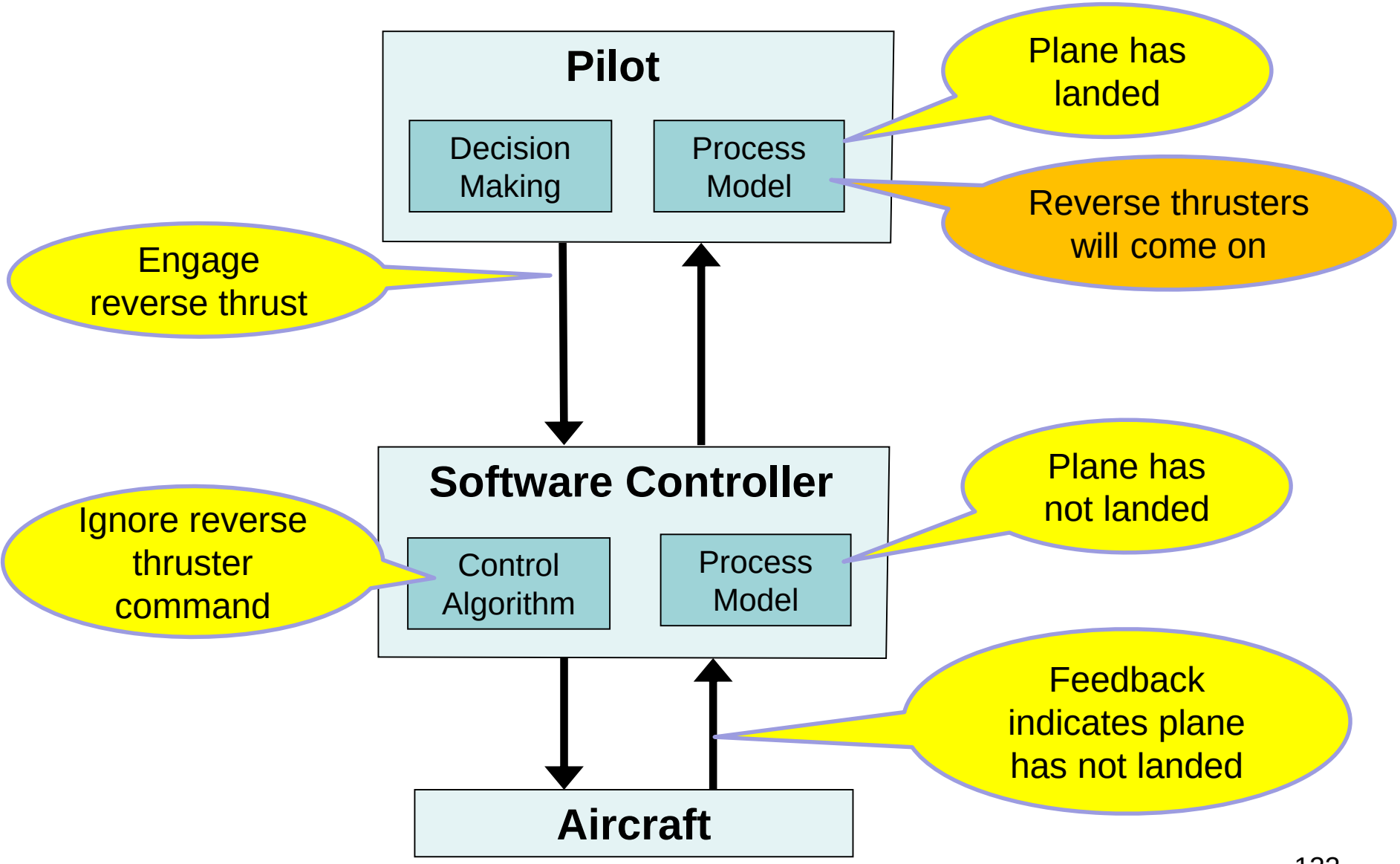
Moscow (Reverse Thrusters)

Hazard: Inadequate Deceleration after Landing



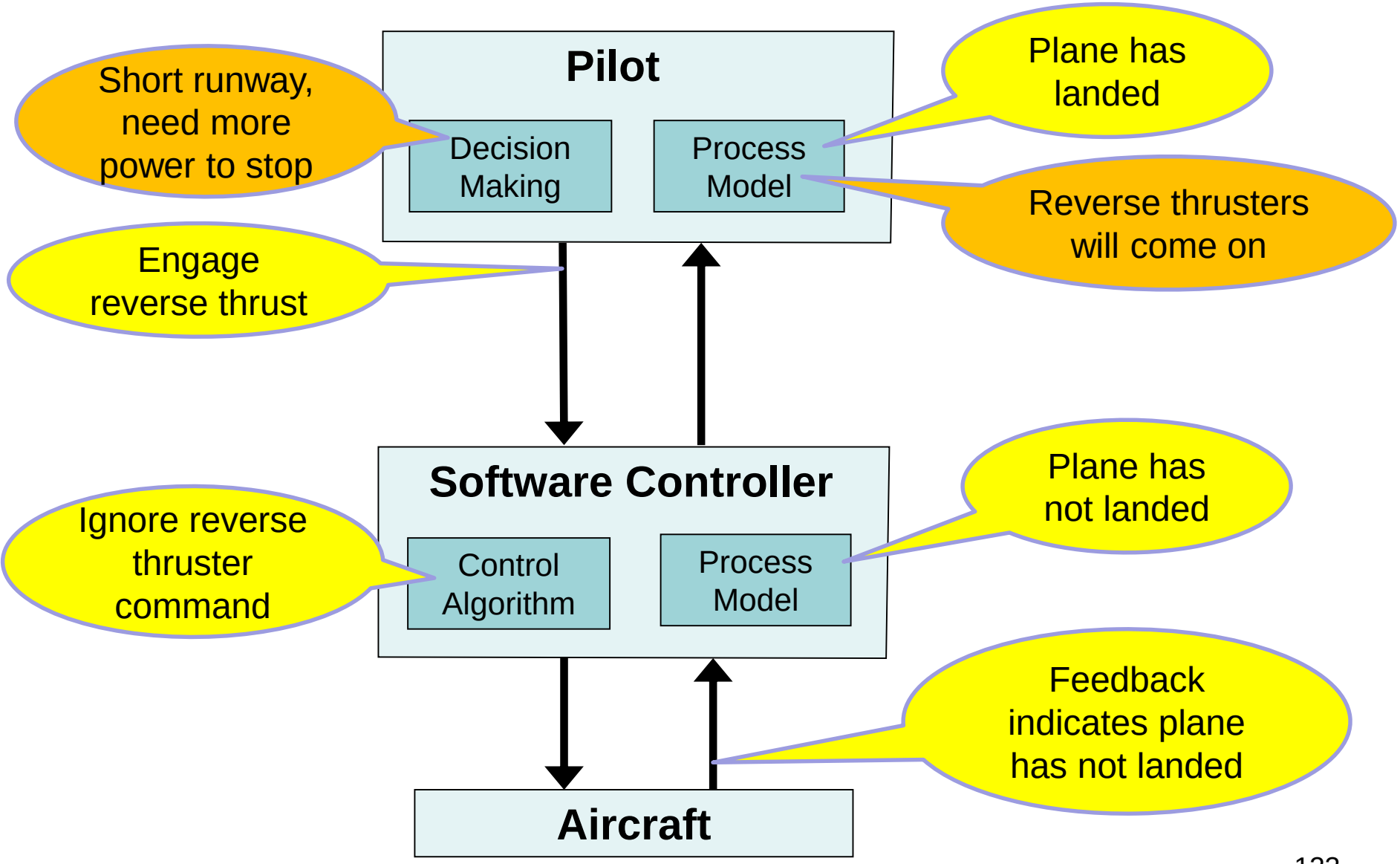
Moscow (Reverse Thrusters)

Hazard: Inadequate Deceleration after Landing



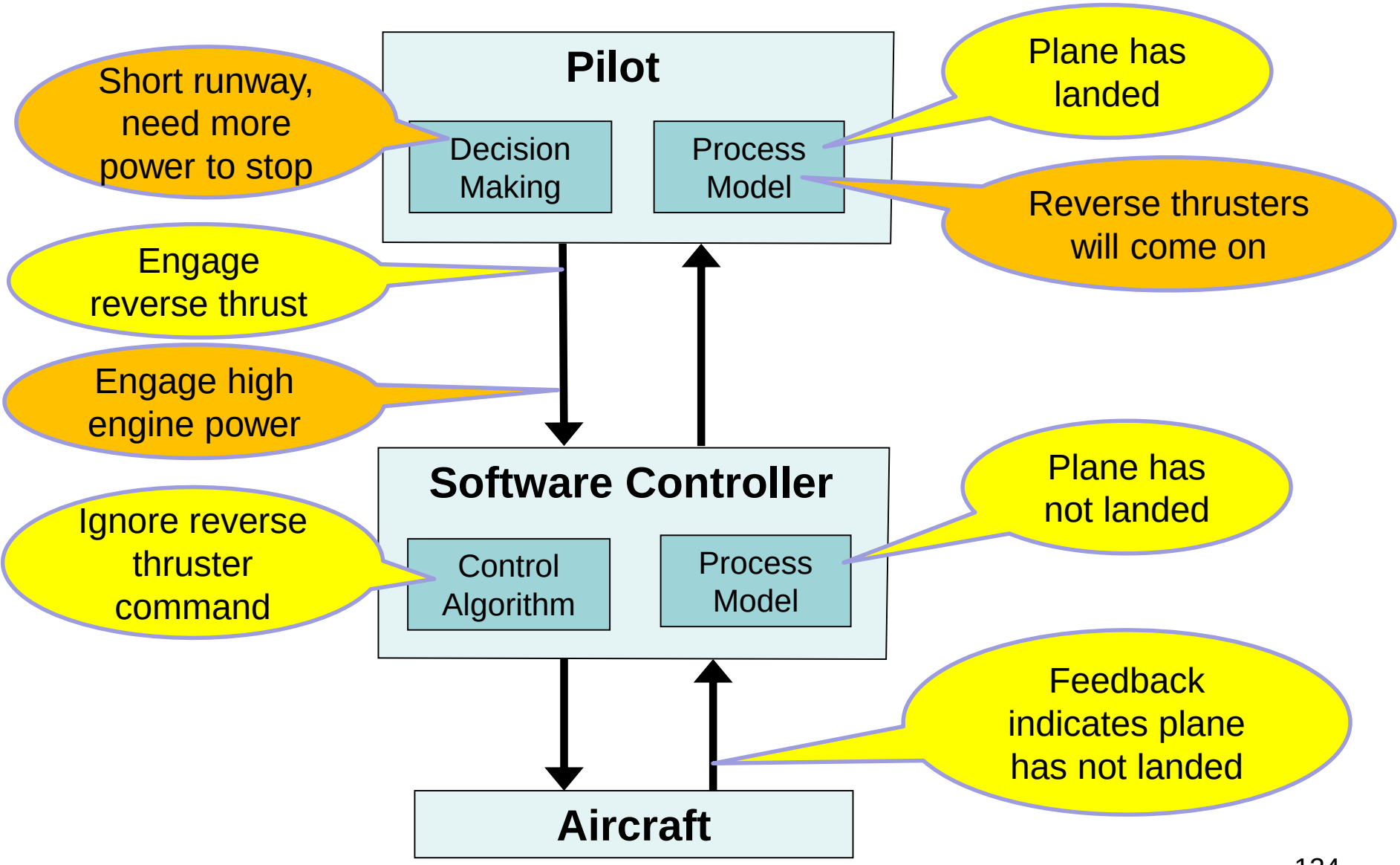
Moscow (Reverse Thrusters)

Hazard: Inadequate Deceleration after Landing



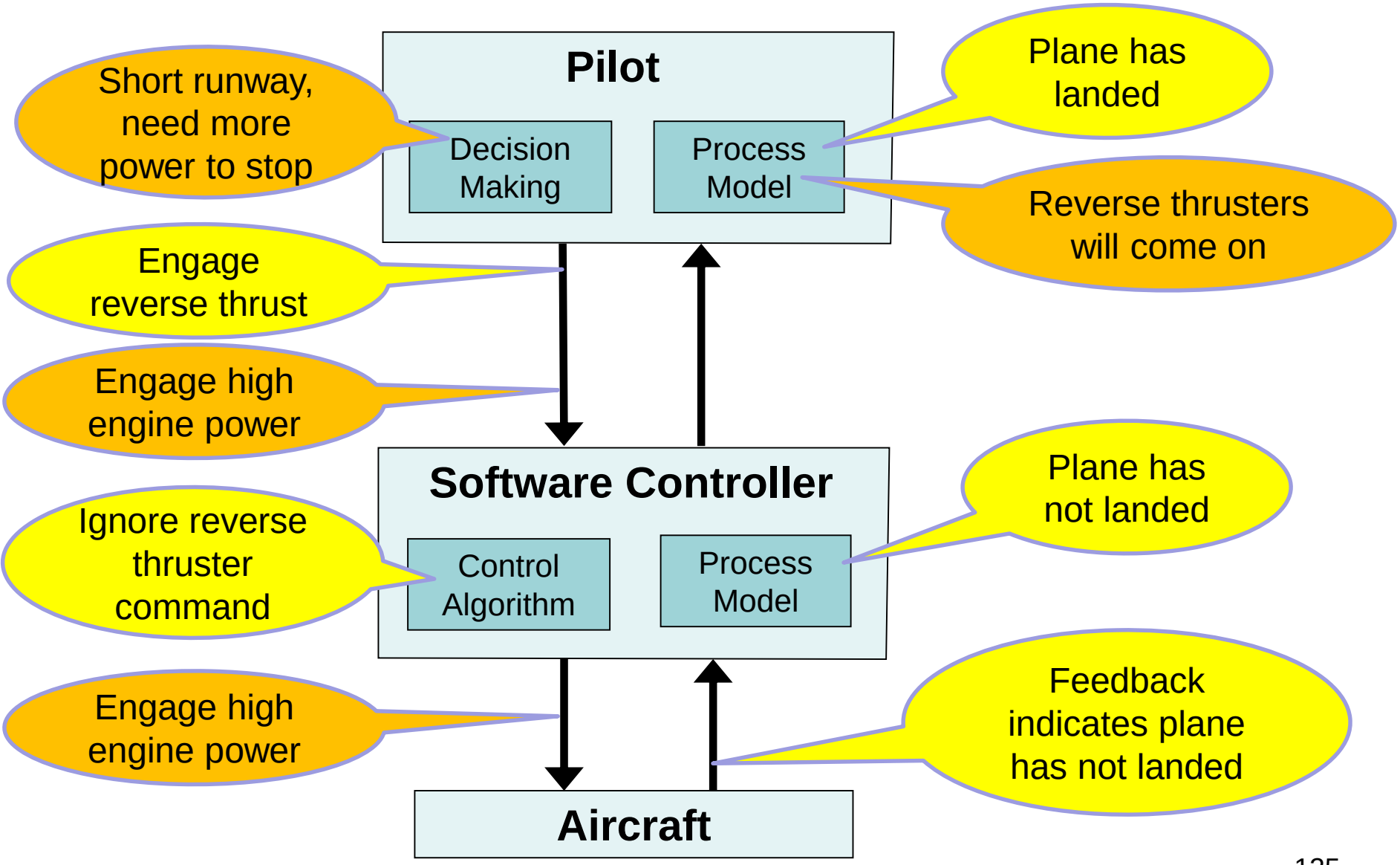
Moscow (Reverse Thrusters)

Hazard: Inadequate Deceleration after Landing



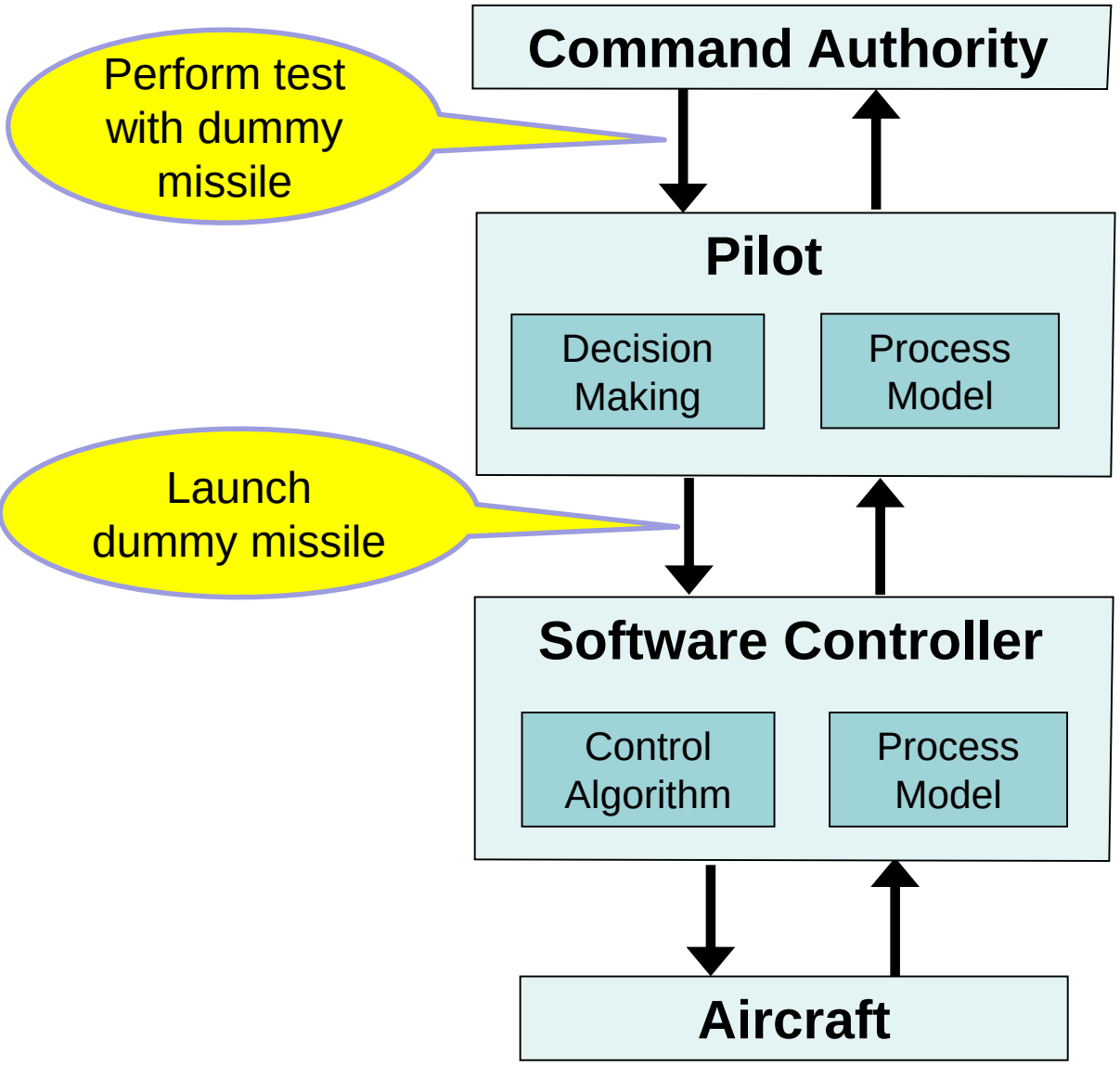
Moscow (Reverse Thrusters)

Hazard: Inadequate Deceleration after Landing



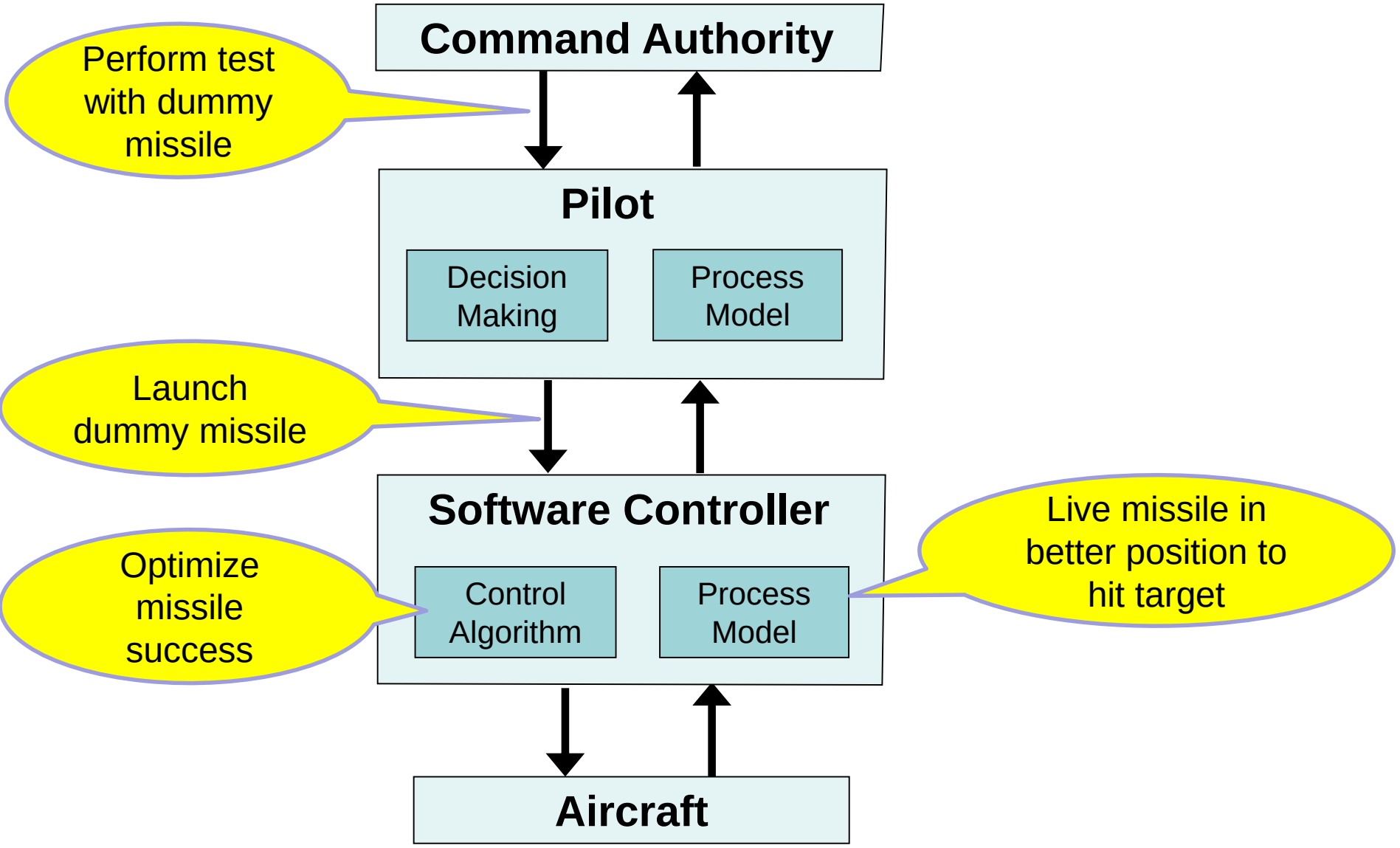
Missile Release Mishap

Hazard: Friendly Fire



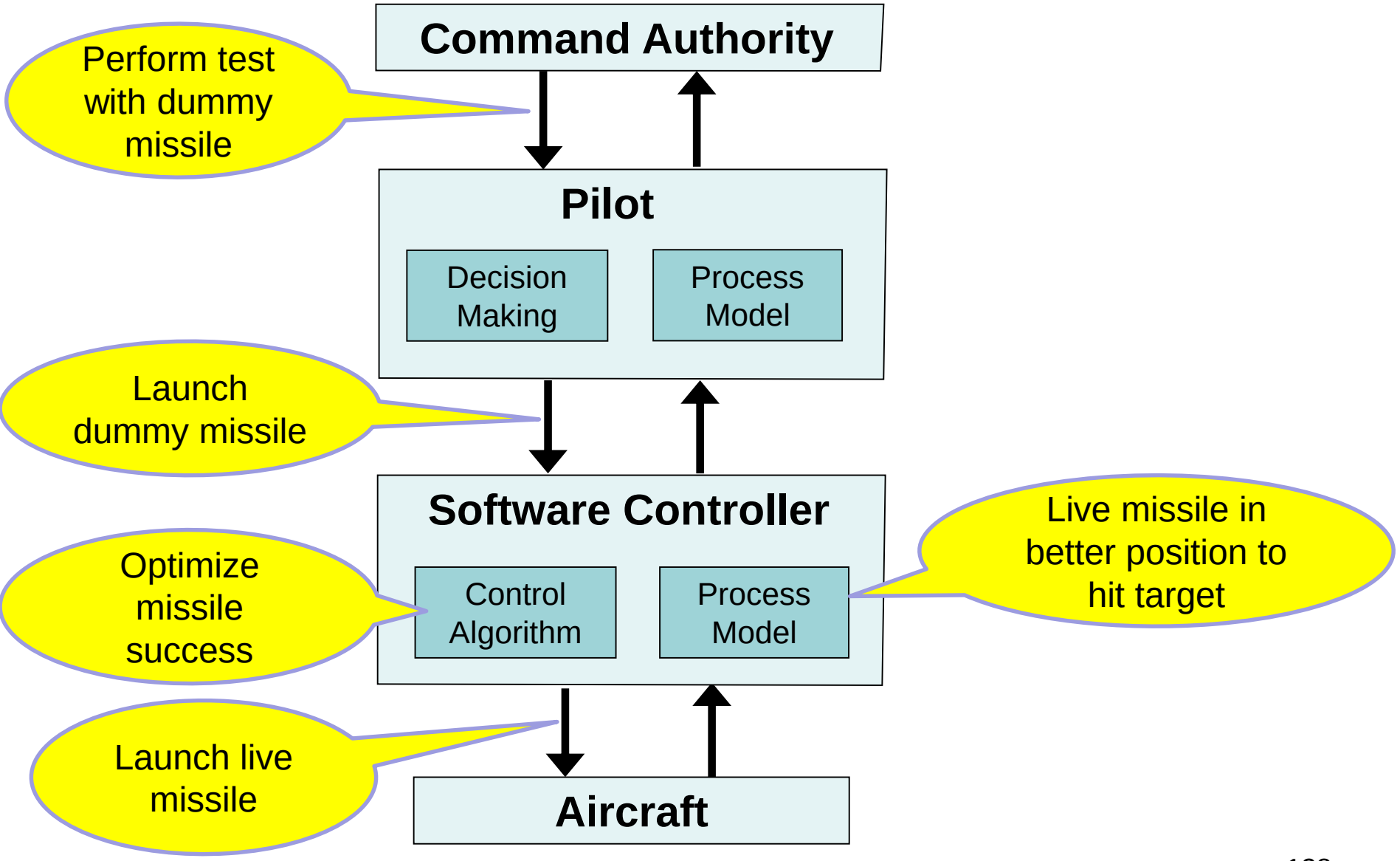
Missile Release Mishap

Hazard: Friendly Fire

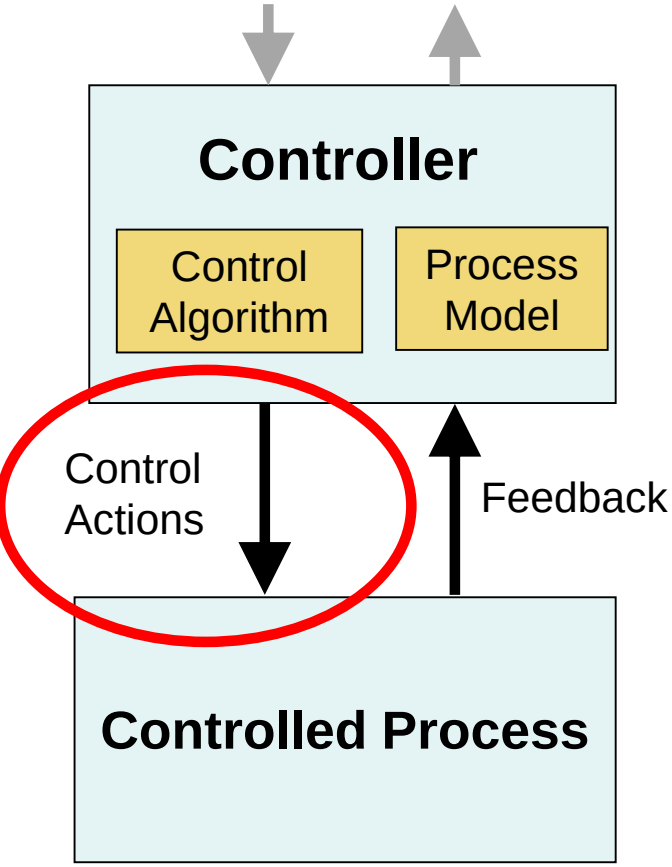


Missile Release Mishap

Hazard: Friendly Fire

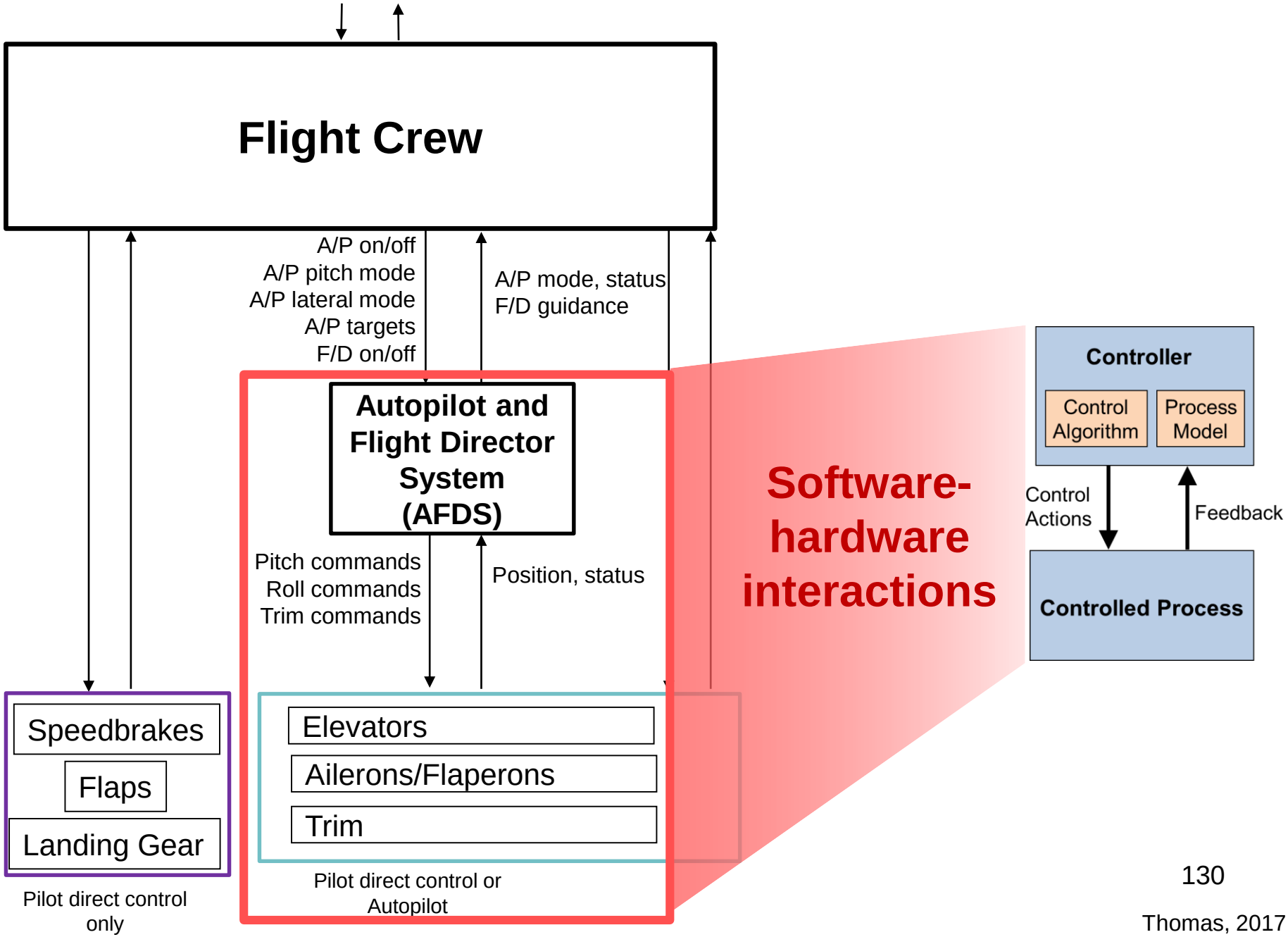


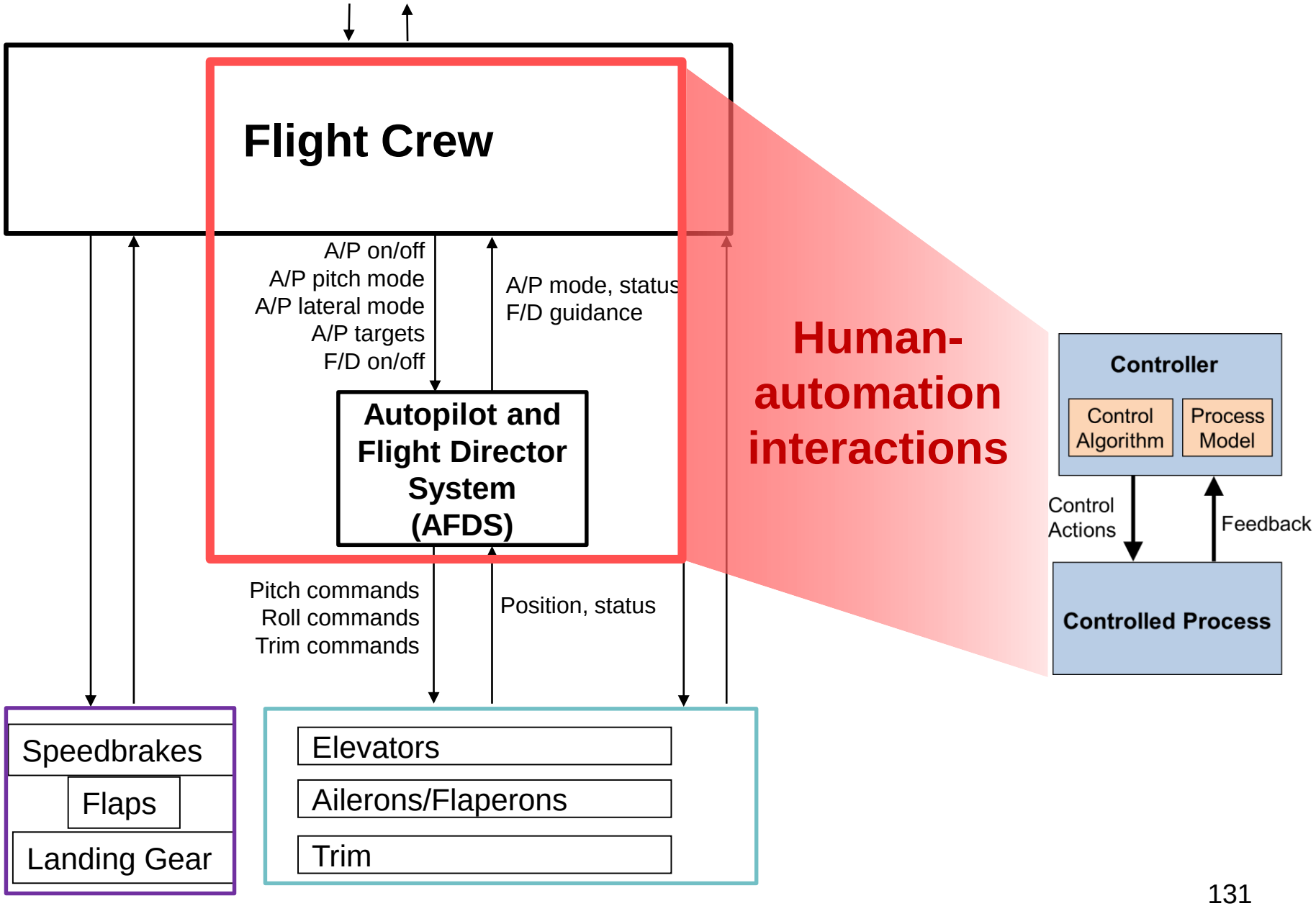
Hazard and Accident Analysis with STAMP

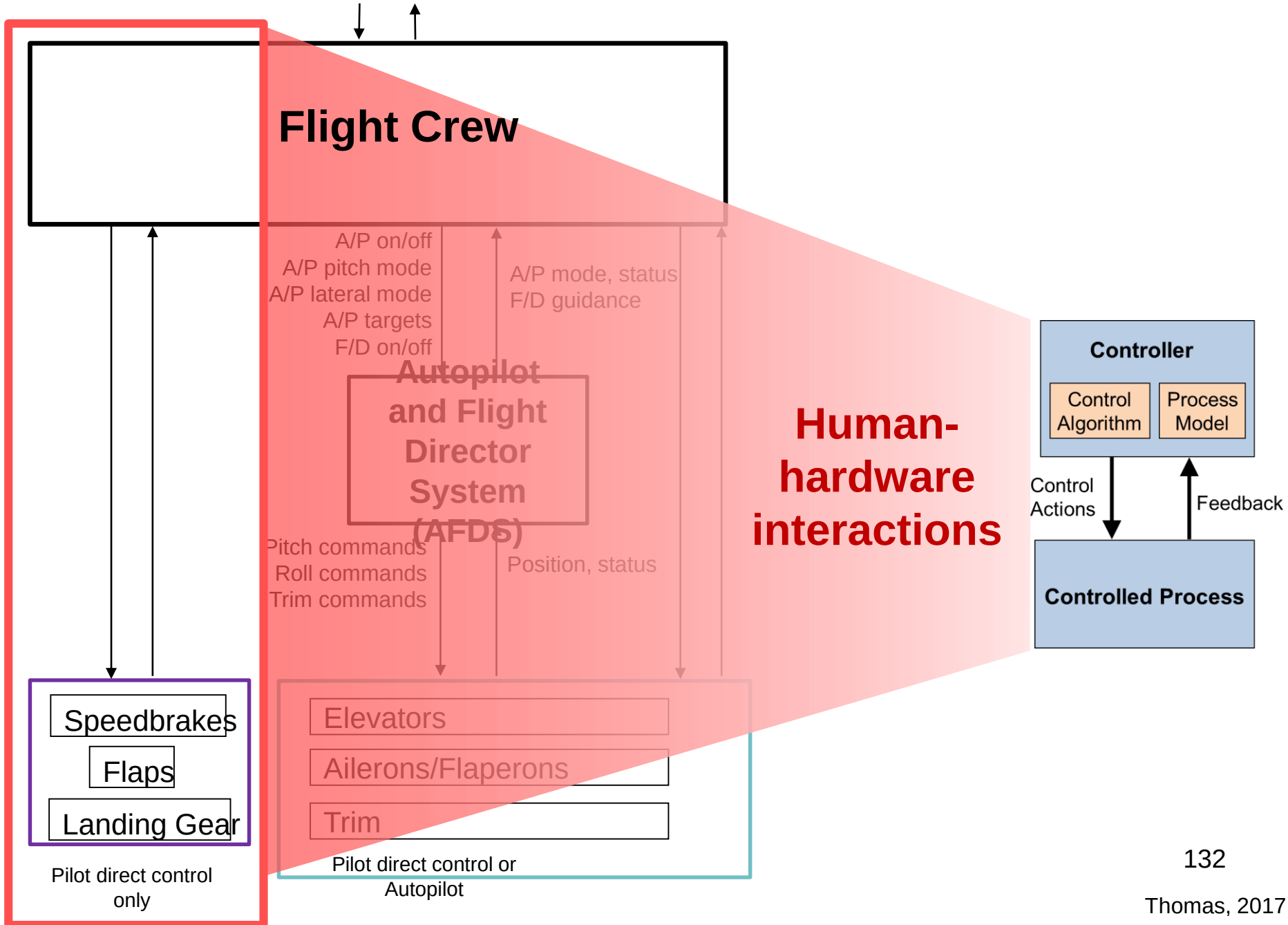


Analysis:

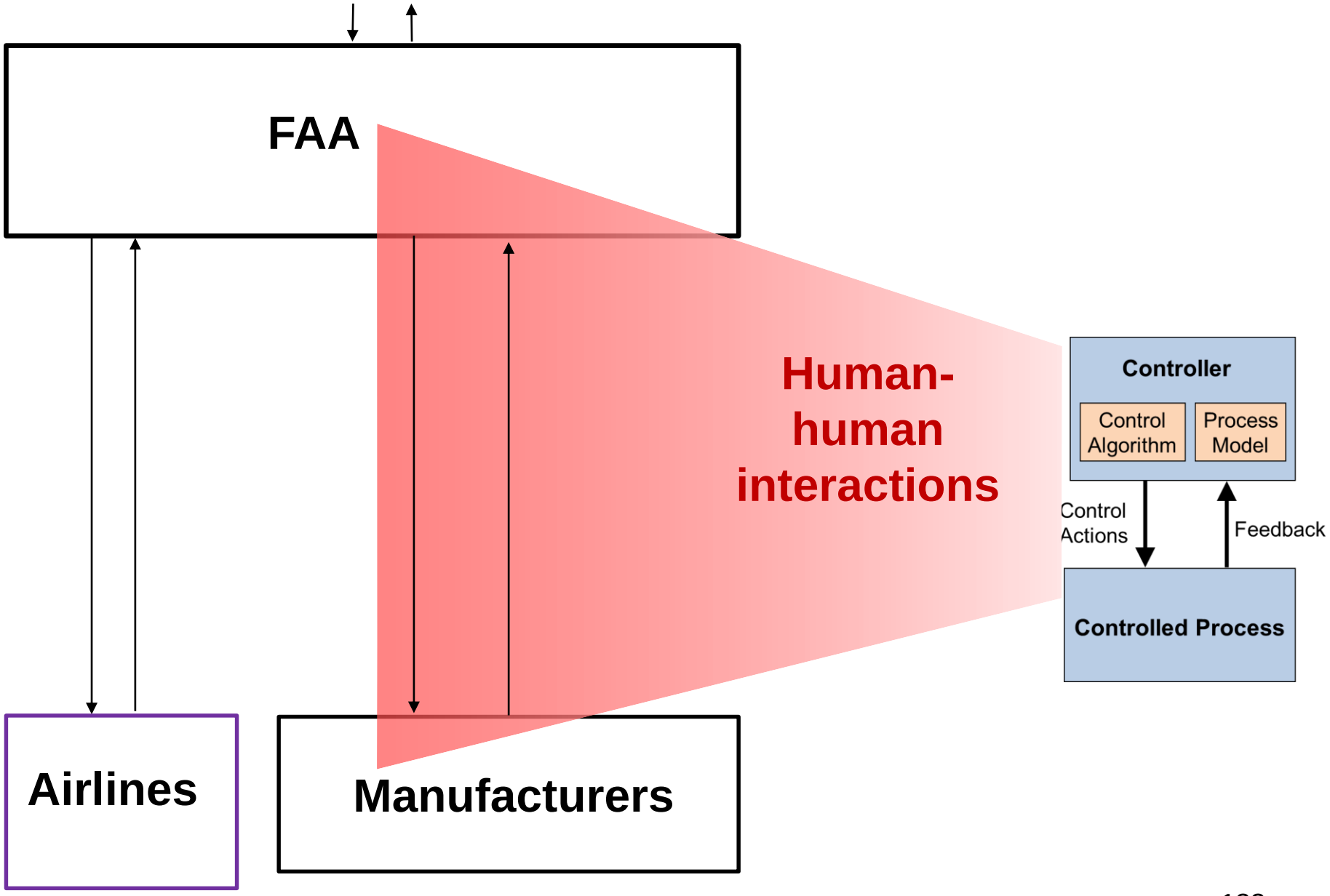
1. Identify potential unsafe control actions
2. Identify why they might be given (eliminate or mitigate)
3. If safe ones provided, then why not followed?


Flight Crew
A/P on/off
A/P pitch mode
A/P lateral mode
A/P targets
F/D on/off
A/P mode, status
F/D guidance
Autopilot and Flight Director System (AFDS)
Pitch commands
Roll commands
Trim commands
Position, status
Speedbrakes
Flaps
Landing Gear
Elevators
Ailerons/Flaperons
Trim
Pilot direct control only
Pilot direct control or Autopilot
Software-hardware interactions
Controller
Control Algorithm
Process Model
Control Actions
Feedback
Controlled Process


Flight Crew
A/P on/off
A/P pitch mode
A/P lateral mode
A/P targets
F/D on/off
A/P mode, status
F/D guidance
Autopilot and Flight Director System (AFDS)
Pitch commands
Roll commands
Trim commands
Position, status
Speedbrakes
Flaps
Landing Gear
Elevators
Ailerons/Flaperons
Trim
Pilot direct control only
Pilot direct control or Autopilot
Human-automation interactions
Controller
Control Algorithm
Process Model
Control Actions
Feedback
Controlled Process


Flight Crew
A/P on/off
A/P pitch mode
A/P lateral mode
A/P targets
F/D on/off
A/P mode, status
F/D guidance
Autopilot and Flight Director System (AFDS)
Pitch commands
Roll commands
Trim commands
Position, status
Human-hardware interactions
Controller
Control Algorithm
Process Model
Control Actions
Feedback
Controlled Process
Speedbrakes
Flaps
Landing Gear
Pilot direct control only
Elevators
Ailerons/Flaperons
Trim
Pilot direct control or Autopilot

Thomas, 2017


FAA
Human-
human
interactions
Controller
Control
Algorithm
Process
Model
Control
Actions
Feedback
Controlled Process
Airlines
Manufacturers

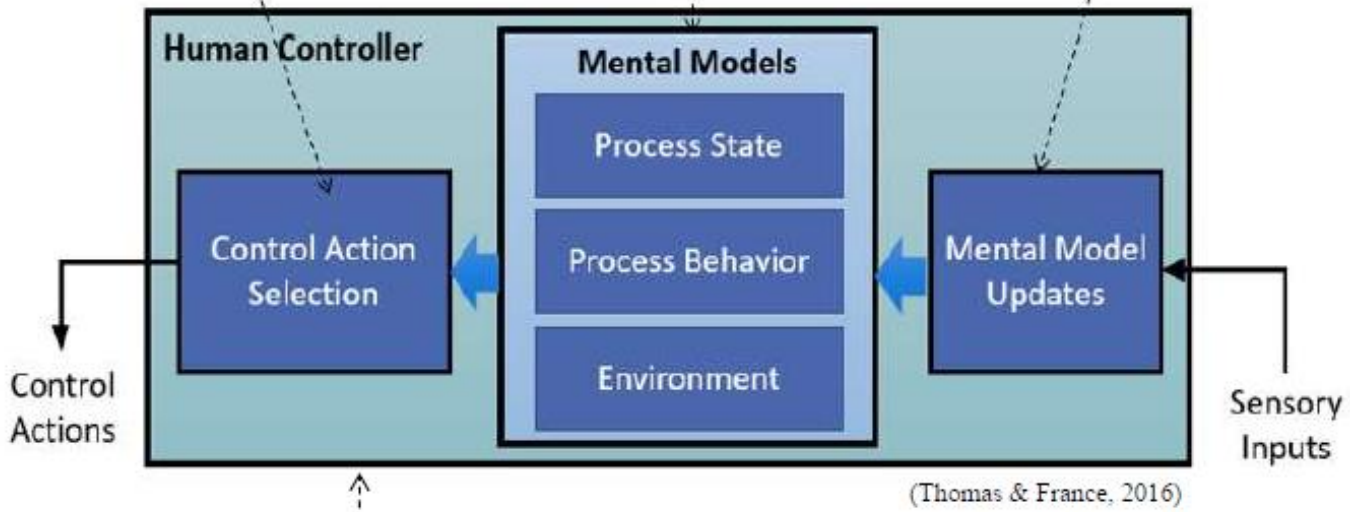
Thomas, 2017

A NEW MODEL FOR HUMAN CONTROLLERS

Captures the controller's goals and how decisions are made based on the mental models

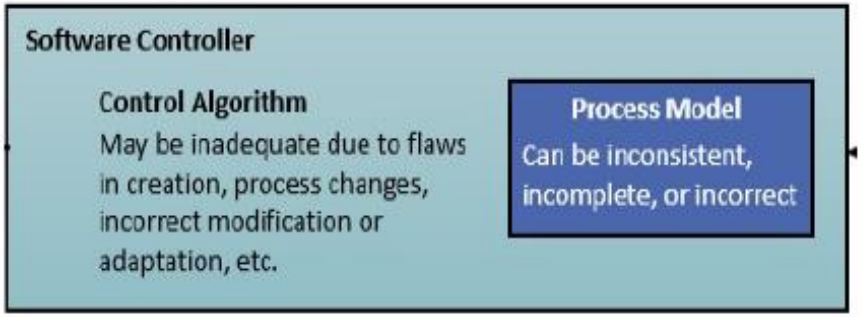
Captures specific types of flaws in the way the human controller conceptualizes the system and environment

Captures the influence of human experiences, and expectations on the processing of sensory input



(Thomas & France, 2016)

Provides an alternative to the existing controller model which is better suited for software controllers



Integrated Approach to Safety and Security

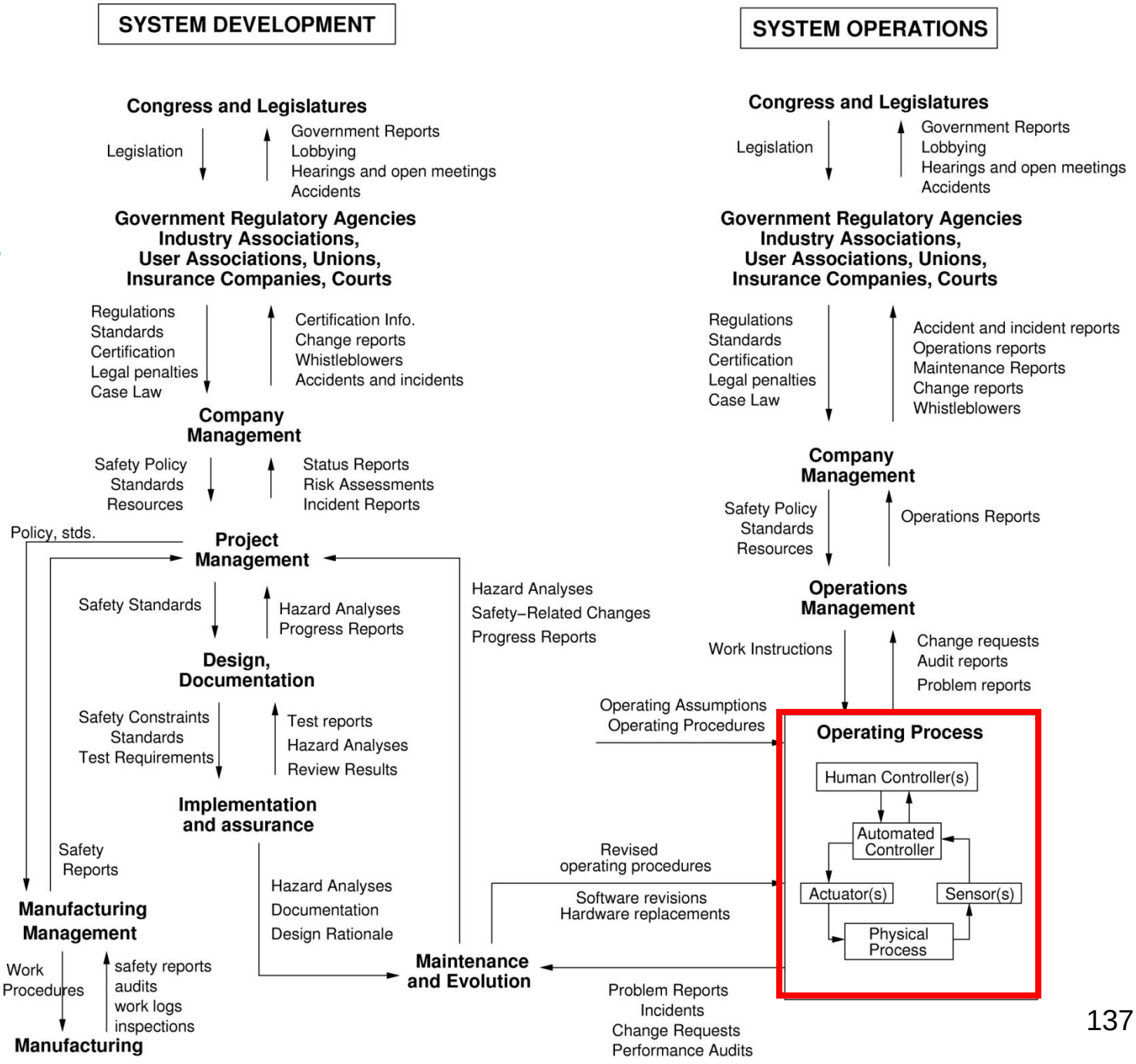
- Both concerned with losses (intentional or unintentional)
  - Mission assurance (vs. information protection)
  - Ensure that critical functions and services are maintained
  - New paradigm for safety will work for security too
    - May have to add new causes, but rest of process is the same
  - A top-down, system engineering approach to designing safety and security into systems

Example: Stuxnet

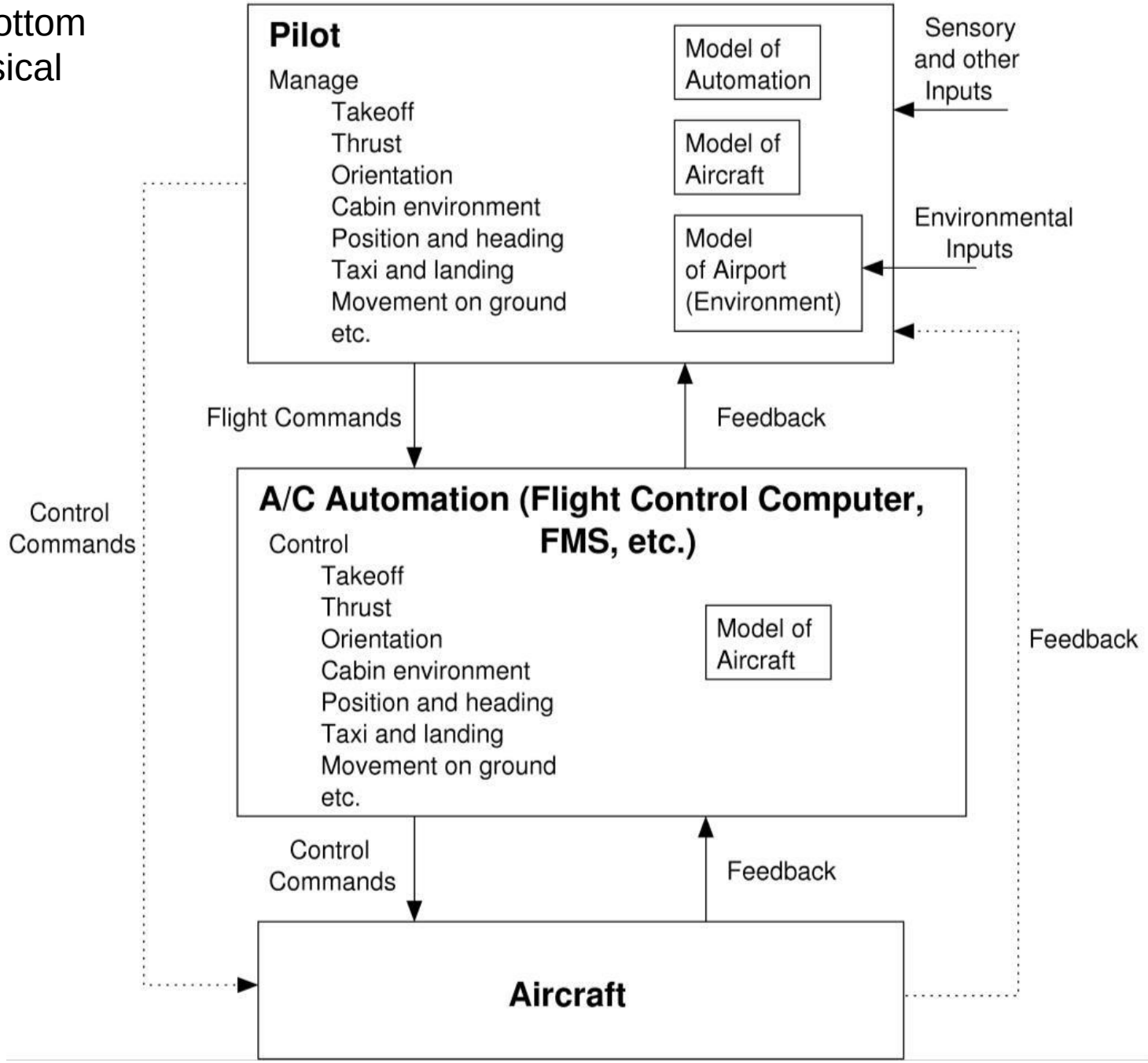
- Loss: Damage to reactor (in this case centrifuges)
- Hazard/Vulnerability: Centrifuges are damaged by spinning too fast
- Constraint to be Enforced: Centrifuges must never spin above maximum speed
- Hazardous control action: Issuing *increase speed* command when already spinning at maximum speed
- One potential causal scenario:
  - Incorrect process model: thinks spinning at less than maximum speed
    - Could be inadvertent or deliberate
- Potential controls:
  - Mechanical limiters (interlock), Analog RPM gauge

Focus on preventing hazardous state (not keeping intruders out)

Example Safety Control Structure (SMS)



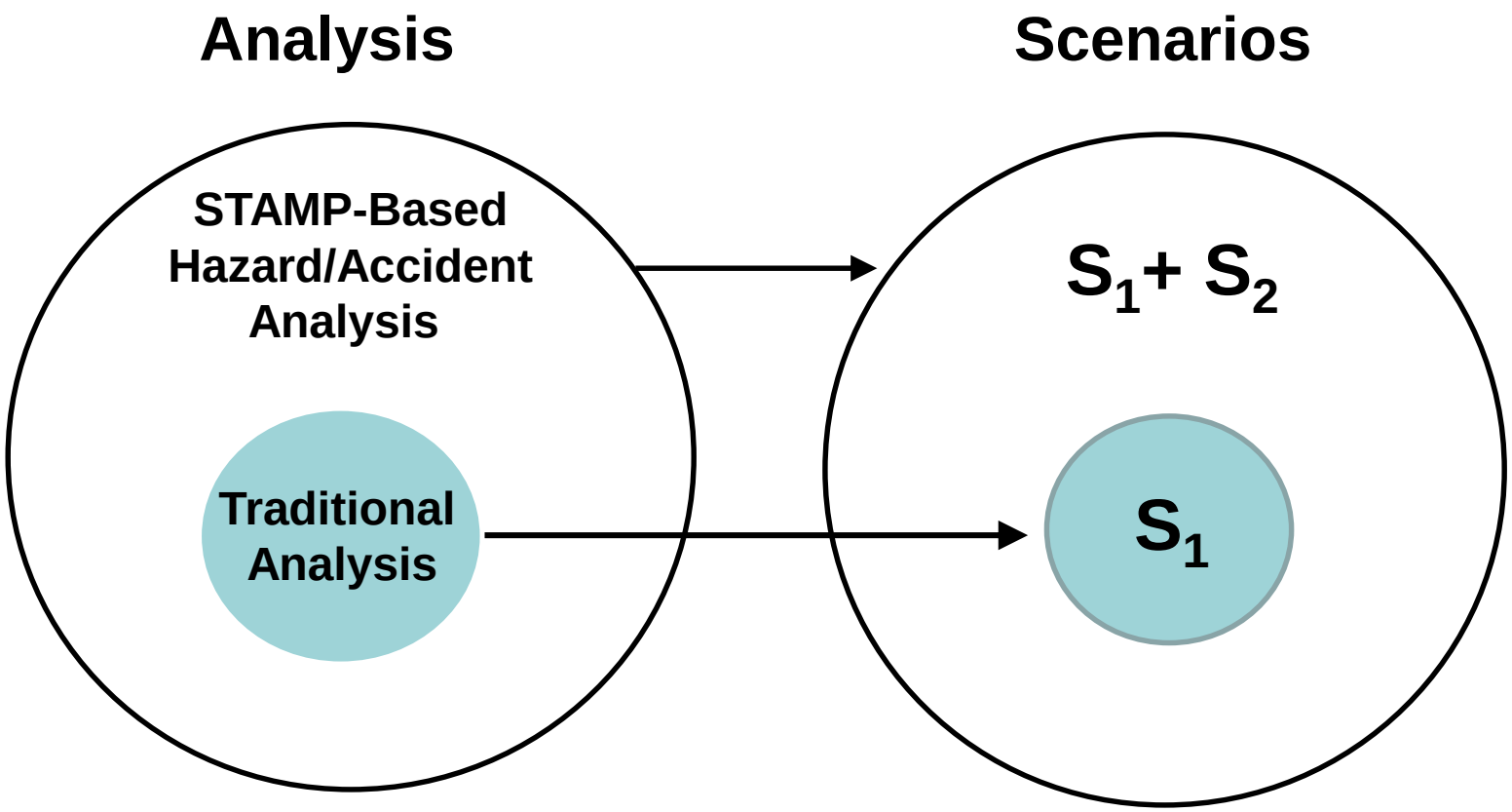
[Box on bottom right, physical process]



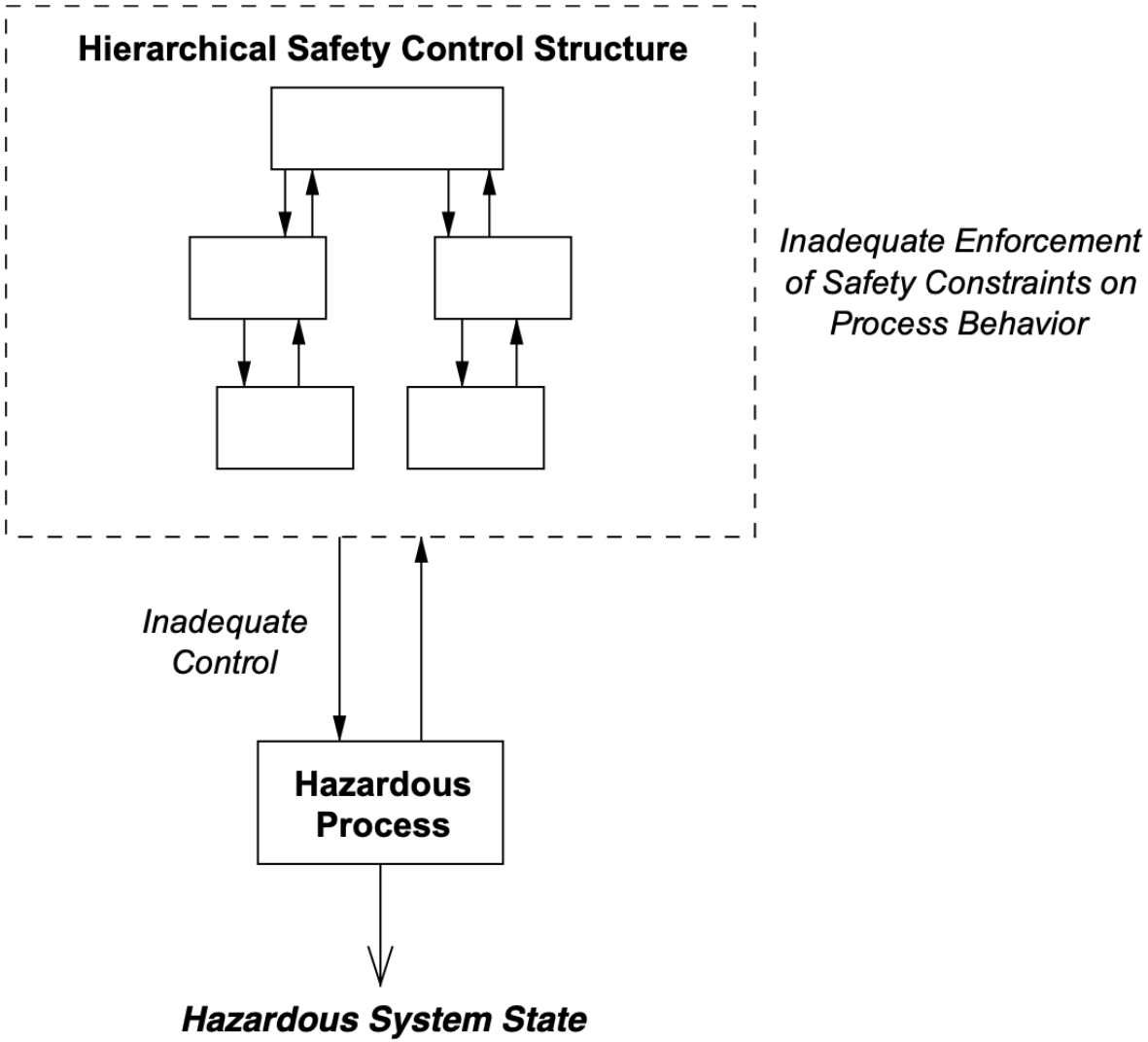
STAMP
(System-Theoretic Accident Model and Processes)

- A new, more powerful accident/loss causality model
- Based on systems theory, not reliability theory
- Defines accidents/losses as a dynamic control problem (vs. a failure problem)
- Applies to VERY complex systems
- Includes
  - Scenarios from traditional hazard analysis methods (failure events)
  - Component interaction accidents
  - Software and system design errors
  - Human errors
  - Entire socio-technical system (not just technical part)

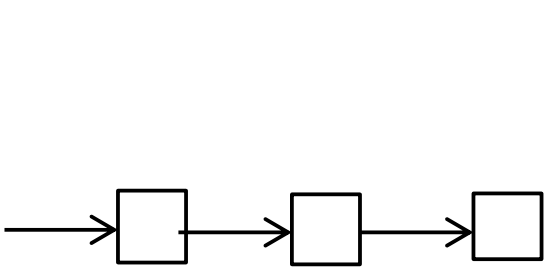
STAMP-Based vs. Traditional Analysis



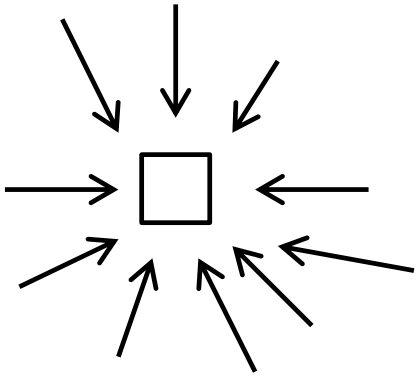
STAMP Causality Model



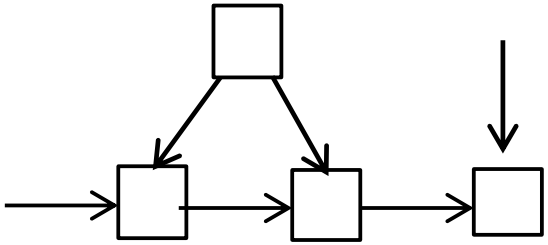
Captures More Types of Causality than Linear

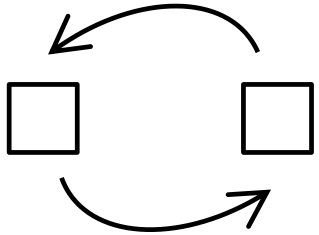


(a)



(b)





Herald of Free Enterprise

Deckhand overslept	Deckhand did not close doors	Captain in hurry to leave	Bosun did not check doors closed	Ferry capsizes

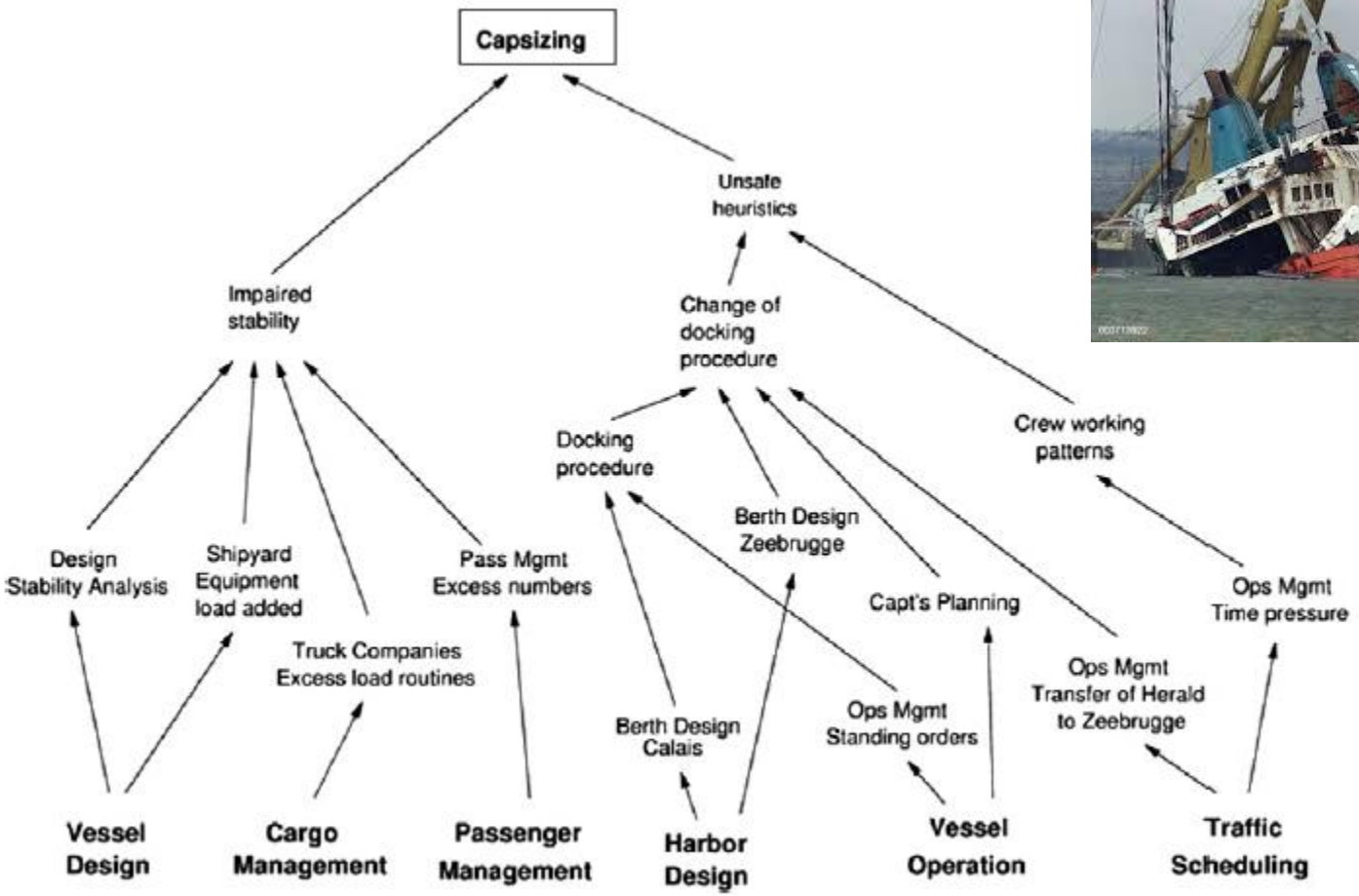


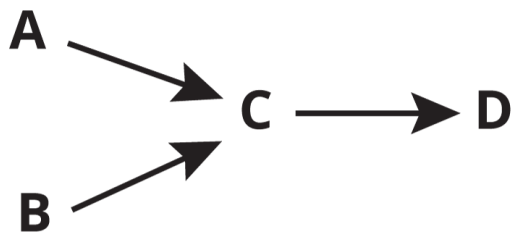
Fig. 1. The complex interactions in the Zeebrugge ferry accident (adapted from Rasmussen, (1997)).

“Reality is made up of circles, but we see straight lines”

Peter Senge, *Fifth Discipline* (p. 73)

Event Oriented Thinking

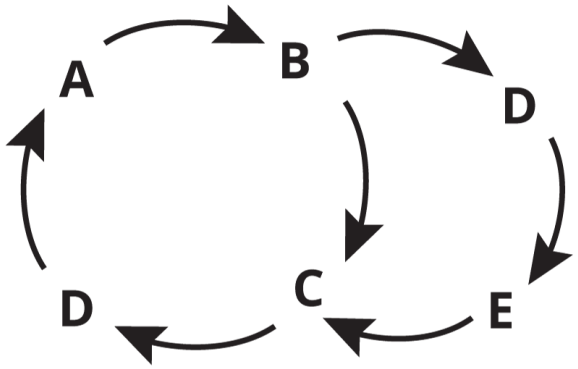
Thinks in straight lines



In event oriented thinking everything can be explained by causal chains of events. From this perspective the **root causes** are the events starting the chains of cause and effect, such as A and B.

Systems Thinking

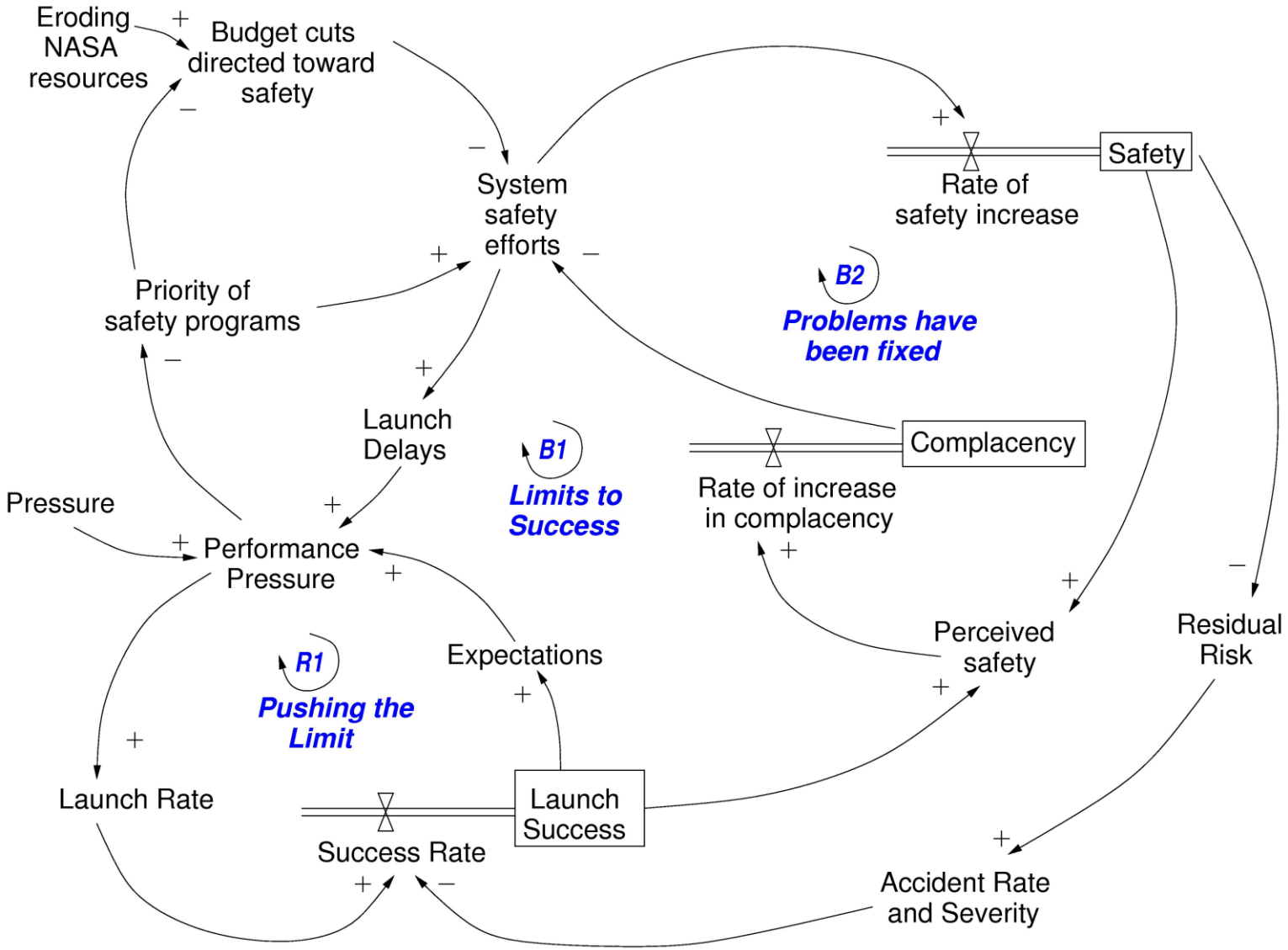
Thinks in loop structure



In systems thinking a system’s behavior emerges from the structure of its feedback loops. **Root causes** are not individual nodes. They are the forces emerging from particular feedback loops.

Created by Thwink.org

Some Factors in the Columbia Shuttle Loss



Safety as a Control Problem

Goal: Design an effective control structure that eliminates or reduces adverse events.

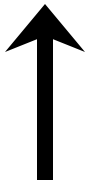
- Need clear definition of expectations, responsibilities, authority, and accountability at all levels of safety control structure
- Need appropriate feedback
- Entire control structure must together enforce the system safety property (constraints)
  - Physical design (inherent safety)
  - Operations
  - Management
  - Social interactions and culture

What kinds of tools are available?

Processes

System Engineering | Risk Management | Organizational Design (SMS)

Operations | Certification and Acquisition | Regulation



Tools

Accident Analysis
CAST

Hazard Analysis
STPA

Security Analysis
STPA-Sec

Organizational/Cultural Risk Analysis

Identifying Leading Indicators

MBSE
SpecTRM & …

STAMP: Theoretical Causality Model

Does it work?

Evaluations and Estimates of ROI

- Hundreds of evaluations and comparison with traditional approaches used now
  - Controlled scientific and empirical (in industry)
  - All show STPA is better (identifies more critical requirements or design flaws)
  - Identified real accidents that other methods missed
  - All (that measured) show STPA requires orders of magnitude fewer resources than traditional techniques
- ROI estimates only beginning but one large defense industry contractor claims they are seeing 15-20% return on investment when using STPA

Ballistic Missile Defense System (MDA)

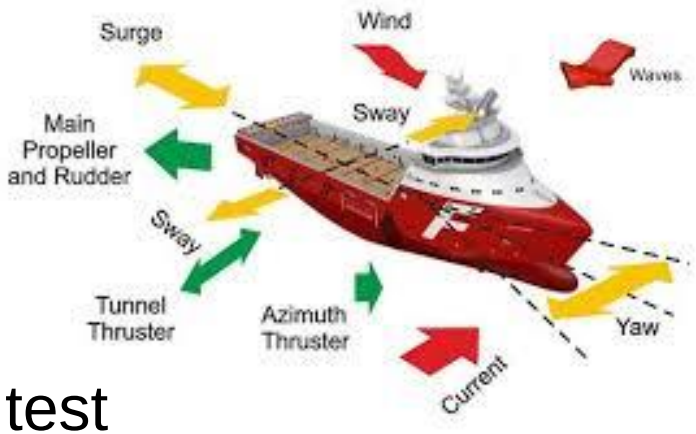
- Hazard was inadvertent launch
- Analyzed right before deployment and field testing (so done late)
  - 2 people, 5 months (unfamiliar with system)
  - Found so many paths to inadvertent launch that deployment delayed six months
- One of first uses of STPA on a real defense system (2005)

Sea-based sensors on the Aegis platform, upgraded early warning radars (UEWR), the Cobra Dane Upgrade (CDU), Ground-based Midcourse Defense (GMD) Fire Control and Communications (GFC/C), a Command and Control Battle Management and Communications (C2BMC) Element, and Ground-based interceptors (GBI). Future block upgrades were originally planned to introduce additional Elements into the BMDS, including Airborne Laser (ABL) and Terminal High Altitude Area Defense (THAAD).

UH-60MU (Blackhawk)

- Analyzed Warning, Caution, and Advisory (WCA) system
- STPA results were compared with an independently conducted hazard analysis of the UH-60MU using traditional safety processes described in SAE ARP 4761 and MIL-STD-882E.
  - STPA found the same hazard causes as the traditional techniques and
  - Also identified things not found using traditional methods, including design flaws, human behavior, and component integration and interactions

Navy Escort Vessels (Lt. Blake Abrecht)

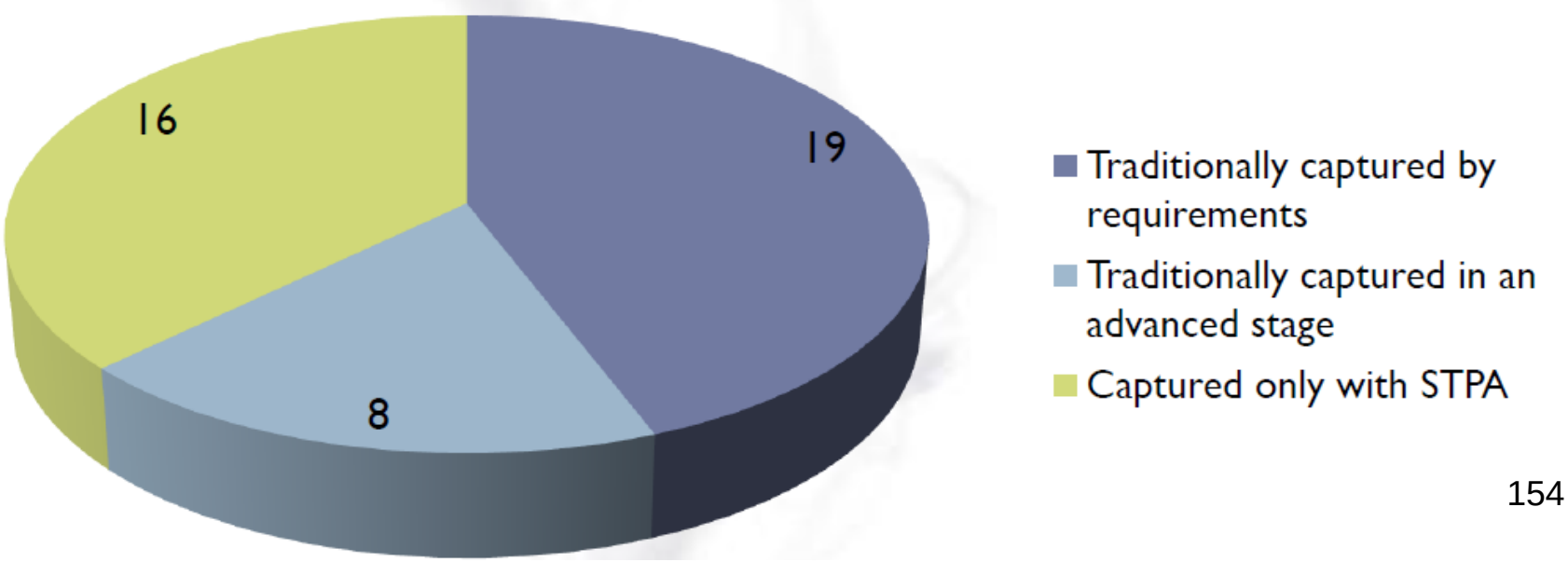


- Dynamic positioning system
- Ran into each other twice during test
- Performed a CAST analysis (on two incidents) and STPA on system as a whole
- STPA found scenarios not found by MIL-STD-882 analysis (fault trees and FMEA)
- Navy admiral rejected our findings saying “We’ve used PRA for 40 years and it works just fine”
- Put into operation and within 2 months ran into a submarine
- Scenario was one we had found

EPRI Nuclear Power Plant Controlled Experiment

- Compared FTA, FMEA, ETA, HAZOP and STPA
- Two graduate students spent 2 weeks on this
- Only STPA found accident that had occurred in plant but analysts did not know about

Embraer Aircraft Smoke Control System requirements captured by STPA

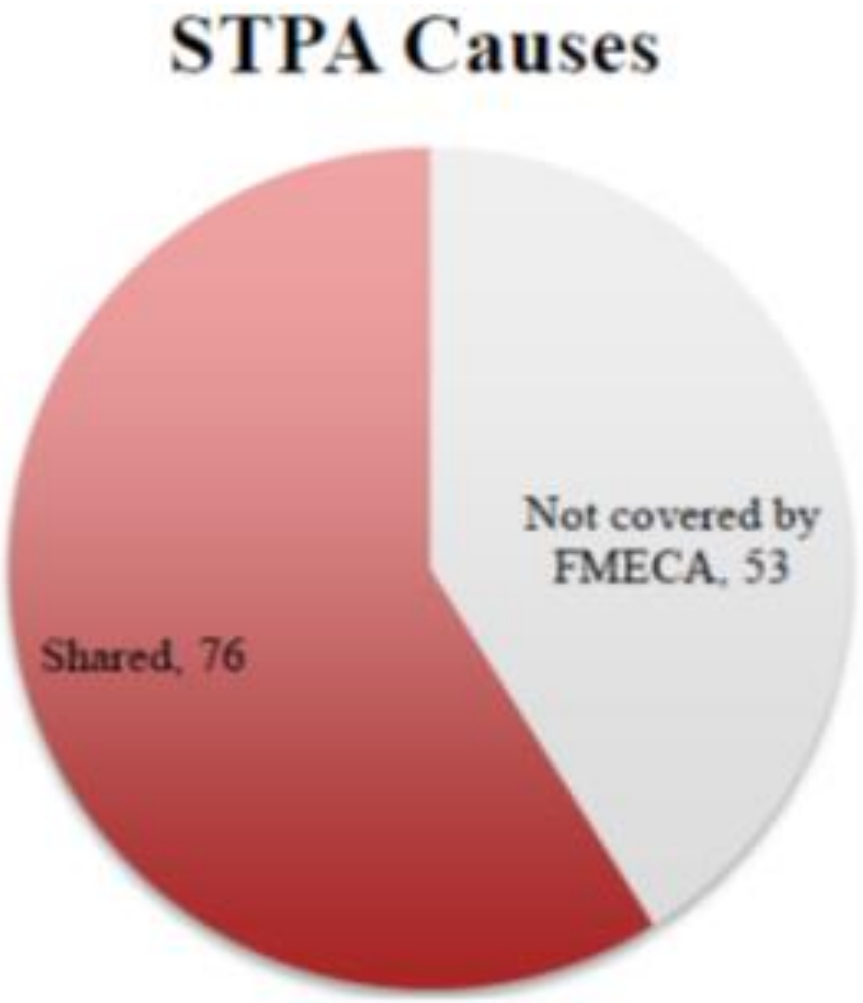


- Range Extender System for Electric Vehicles (Valeo)
  - FTA/CPA took 3 times effort of STPA, found less

- Medical Device (Class A recall)

FMECA	STPA
70+ causes of accidents	175+ causes accidents (9 related to adverse event)
Team of experts	Single semi-expert
Time dedication: months/years)	Time: weeks/month
Identified only single fault causes	Identified complex causes of accidents

- Automotive Electric Power Steering System



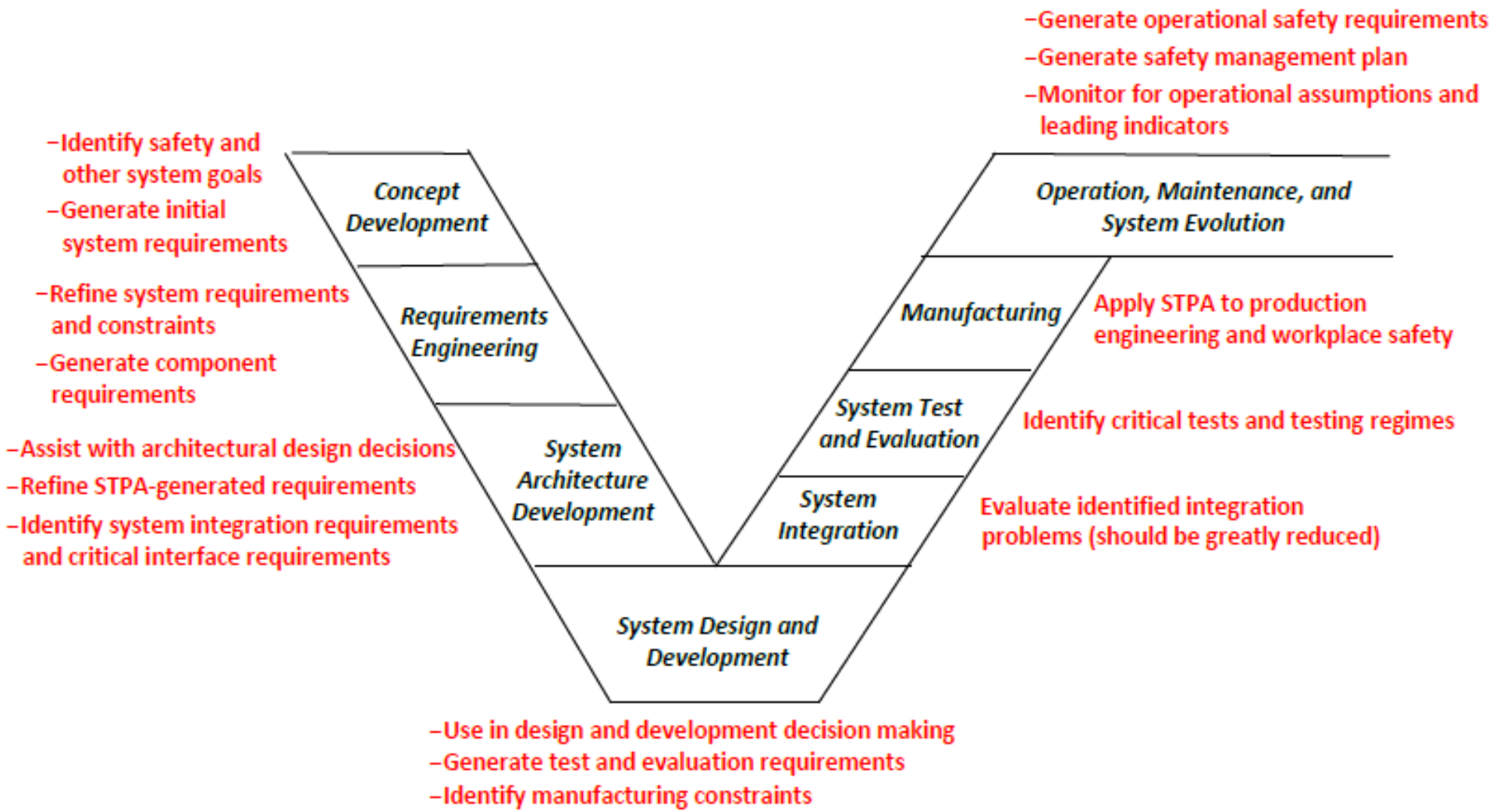
Some Other Uses

- Workplace safety
- Hospital safety
- Design for Safe Manufacturing/Assembly
- Production Engineering
- Serviceability and Diagnostics (farm equipment)
- Organizational Analysis (e.g., system engineering process)
- Supply chain analysis

A Systems Approach to Safety and Security

- Emphasizes building in safety rather than measuring it or adding it on to a nearly completed design
- Looks at system as a whole, not just components (a top-down holistic approach)
- Takes a larger view of causes than just failures
  - Accidents today are not just caused by component failures
  - Includes software and requirements flaws, human behavior, design flaws, management flaw, etc.
- Goal is to use modeling and analysis to design and operate the system to be safe/secure, not to predict the likelihood of a loss.

STAMP tools can be used throughout product development and operations



System Engineering Benefits

- Finds faulty underlying assumptions in concept development before flow downstream as anomalies (where more costly to change)
  - 70-80% of safety-critical decisions made during concept development
- Finds incomplete information, basis for further discussion with customer
- Both intended and unintended functionality are handled
- Includes software and operators in the analysis
  - Provides deeper insight into system vulnerabilities, particularly for cyber and human operator behavior.

System Engineering Benefits (2)

- Can analyze very complex systems.
  - "Unknown unknowns" usually only found during ops can be identified early in development process
- Can be started early in concept analysis
  - Assists in identifying safety/security requirements before architecture or design exists
  - Then used to design safety and security into system, eliminating costly rework when design flaws found later.
  - As design is refined and more detailed design decisions are made, STPA analysis is refined to help make those decisions
- Complete traceability from requirements to system artifacts
  - Enhances maintainability and evolution

System Engineering Benefits (3)

- Models developed for the analysis provide documentation of system functionality (vs. physical or logical design)
  - Often missing or difficult to find in documentation for large, complex systems
- Augments system engineering process and model based system engineering.
  - Models are functional models rather than simply physical or logical models.

To Make Progress We Need To:

- Develop and use different approaches that match the world of engineering today
- Consider the entire sociotechnical system
- Focus on building safety/security in rather than assuring/measuring it after the design is completed

"The best way to predict the future is to create it."

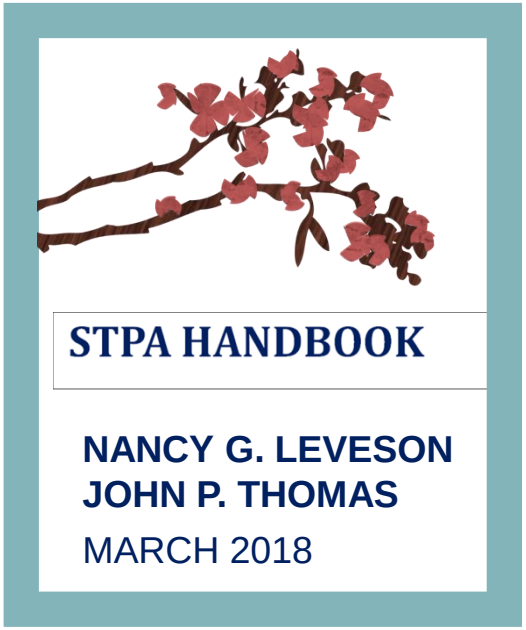
Abraham Lincoln

- Develop and use new approaches to certification, regulation, risk management, and risk assessment

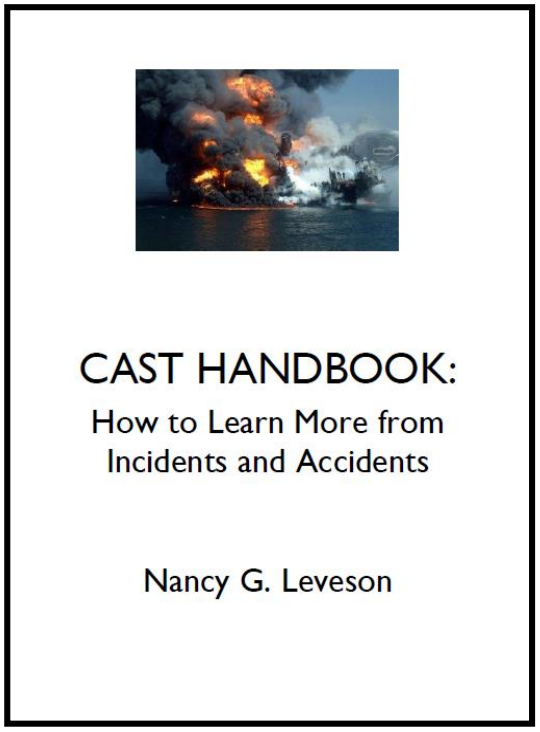
More Information

- http://psas.scripts.mit.edu (papers, presentations from conferences, tutorial slides, examples, etc.)

Free download:
http://mitpress.mit.edu/books/engineering-safer-world



http://psas.scripts.mit.edu
(**65,000+** downloads in 24 mos. Japanese, Chinese, and Korean versions)



Free download:
http://sunnyday.mit.edu/CAST-Handbook.pdf