Accident Causes

My company has had a safety program for 150 years. The program was instituted as a result of a French law requiring an explosives manufacturer to live on the premises with his family.

Crawford Greenwalt
(former president of Dupont)

Most accidents are not the result of unknown scientific principles, but rather of a failure to apply well−known, standard engineering practices.

Trevor Kletz

Causality

- Accident causes are often oversimplified:

  The vessel Baltic Star, registered in Panama, ran aground at full speed on the shore of an island in the Stockholm waters on account of thick fog. One of the boilers had broken down, the steering system reacted only slowly, the compass was maladjusted, the captain had gone down into the ship to telephone, the lookout man on the prow took a coffee break, and the pilot had given an erroneous order in English to the sailor who was tending the rudder. The latter was hard of hearing and understood only Greek.

  LeMonde

- Larger organizational and economic factors?

Root Causes of Accidents

- Flaws in the Safety Culture

  Safety Culture: The general attitude and approach to safety reflected by those who participate in an industry or organization, including management, workers, and government regulators.

  – Overconfidence and complacency

    Discounting risk
    Overrelying on redundancy
    Unrealistic risk assessment
    Ignoring high−consequence, low probability events
    Assuming risk decreases over time
    Underestimating software−related risks
    Ignoring warning signs

Risk Measurement

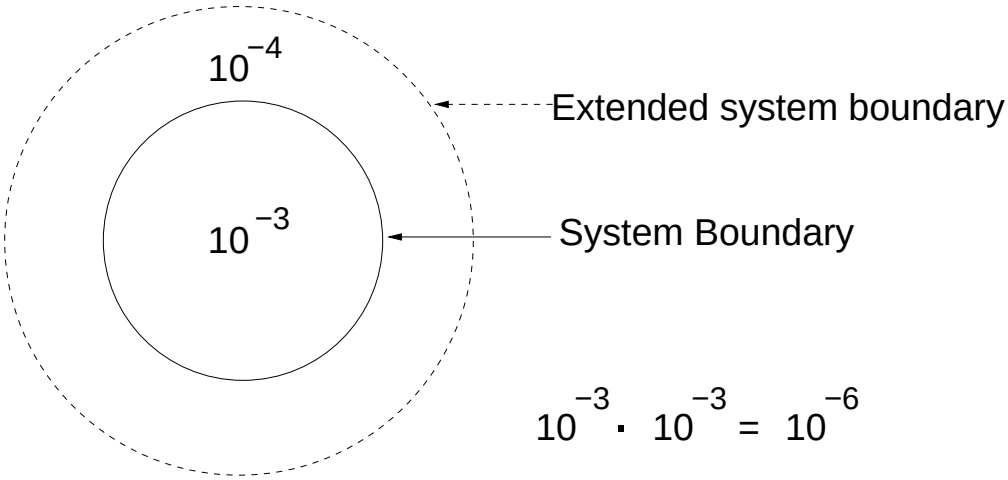
- Risk = f (likelihood, severity)
- Impossible to measure risk accurately.
- Instead, use risk assessment:
  – Accuracy of such assessments is controversial.

    ''To avoid paralysis resulting from waiting for definitive data, we assume we have greater knowledge than scientists actually possess and make decisions based on those assumptions.''
    William Ruckleshaus

  – Cannot evaluate probability of very rare events directly.
  – So use models of the interaction of events that can lead

Misinterpreting Risk

Risk assessments can easily be misinterpreted:



Risk Modeling

- In practice, models only include events that can be measured.
- Most causal factors involved in major accidents are unmeasurable.
- Unmeasurable factors tend to be ignored or forgotten.
- Can we measure software? (what does it mean to measure design?)

Risk assessment data can be like the captured spy;
if you torture it long enough, it will tell you anything
you want to know.

William Ruckelshaus
Risk in a Free Society

Root Causes of Accidents (con't.)

- Low priority assigned to safety
- Flawed resolution of conflicting goals
  - Downstream vs. upstream efforts

- Ineffective Organizational Structure
  - Diffusion of responsibility and authority
  - Lack of independence and low−level status of safety personnel.
  - Limited communication channels and poor information flow.

Root Causes of Accidents (con't.)

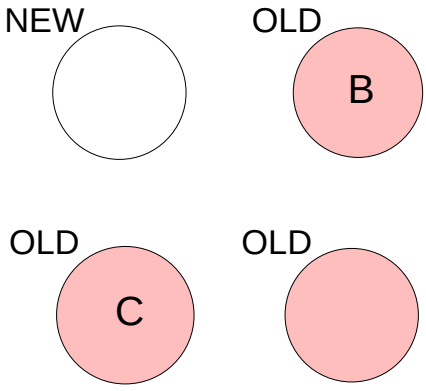
- Ineffective Technical Activities
  - Superficial safety efforts
  - Ineffective risk control
    - Failing to eliminate basic design flaws
    - Basing safeguards on false assumptions
    - Complexity
    - Using risk control devices to reduce safety margins
  - Failure to evaluate changes
  - Information deficiencies

Do Operators Cause Most Accidents?

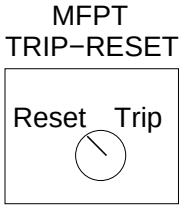
- Data may be biased and incomplete
- Positive actions usually not recorded
- Blame may be based on premise that operators can overcome every emergency
- Operators often have to intervene at the limits.
- Hindsight is always 20/20
- Separating operator error from design error is difficult and perhaps impossible.

Example accidents from chemical plants:

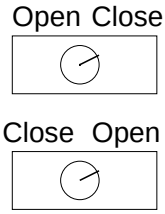
Operator told to fix pump 7.



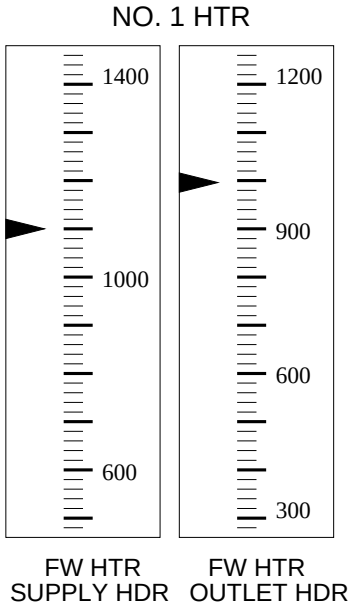
Operator told to replace crystallizer A



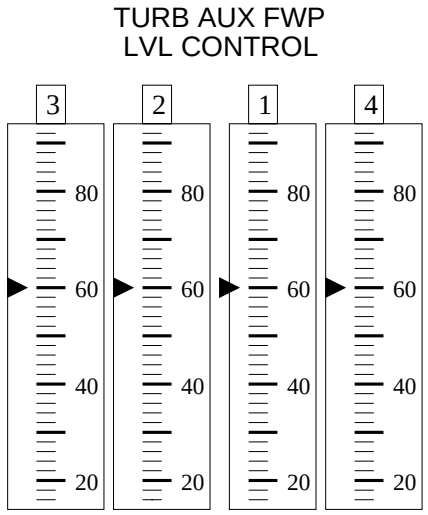
a. Note reversal of trip−reset positions



b. Another Inconsistency



. Heater pressure gauges. A hurried operator under stress might believe the outlet pressure is higher than the supply, even though it is lower.



d. A strange way to count.

−320 accident while landing at Warsaw:

Blamed on pilots for landing too fast.

as it that simple?

- Pilots told to expect windshear. In response, landed faster than normal to give aircraft extra stability and lift.
  - Meteorological information out of date −− no windshear by time pilots landed.
  - Polish government's meteorologist supposedly in toilet at time of landing.
- Thin film of water on runway that had not been cleared.
  - Wheels aquaplaned, skimming surface, without gaining enough rotary speed to tell computer braking systems that aircraft was landing.
  - Computers refused to allow pilots to use aircraft's braking systems. So did not work until too late.
- Still would not have been catastrophic if had not built a high bank at end of runway.
  - Aircraft crashed into bank and broke up.

Blaming pilots turns attention away from:

- Why pilots were given out−of−date weather information
- Design of computer−based braking system
  - Ignored pilots commands
  - Pilots not able to apply braking systems manually
  - Who has final authority?
- Why allowed to land with water on runway
- Why decision made to build a bank at end of runway

Cited probable causes of Cali American Airlines crash:

- Flightcrew's failure to adequately plan and execute the approach to runway 19 at Cali and their inadequate use of automation.
- Failure of flight crew to discontinue the approach into Cali, despite numerous cues alerting them of the inadvisability of continuing the approach.
- Lack of situational awareness of the flightcrew regarding vertical navigation, proximity to terrain, and the relative location of critical radio aids.
- Failure of the flightcrew 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.

Issues in Causality

- Filtering and subjectivity in accident reports
- Root cause seduction
  - Idea of a singular cause is satisfying to our desire for certainty and control.
  - Leads to fixing symptoms
- The "fixing" orientation
  - Well understood causes given more attention
    - Component failure
    - Operator error
  - Tend to look for linear cause−effect relationships
  - Makes it easier to select corrective actions (a "fix")

Limitations of Event Chain Models

- Selecting events
  - Subjective except for physical events immediately preceding or directly involved in accident
  - Root cause dependent on stopping rule
    (difficult to go "through" operators)

Possible Bhopal event chain:

E1: Worker washes pipes without inserting slip blind
E2: Water leaks into MIC tank
E3: Explosion occurs
E4: Relief valve opens
E5: MIC vented into air
E6: Wind carries MIC into populated area around plant

Limitations of Event Chain Models (2)

- Selecting conditions
  Links between events, chosen to explain them, are subjective

Cali AA B−757 accident:

E1: Pilot asks for clearance to take ROZO approach
E2: Pilot types R into the FMS

What is the link between these two events?

Pilot Error?
Crew Procedure Error?
Approach Chart and FMS inconsistencies?
American Airlines training deficiency?
Manufacturer deficiency?
International standards deficiency?

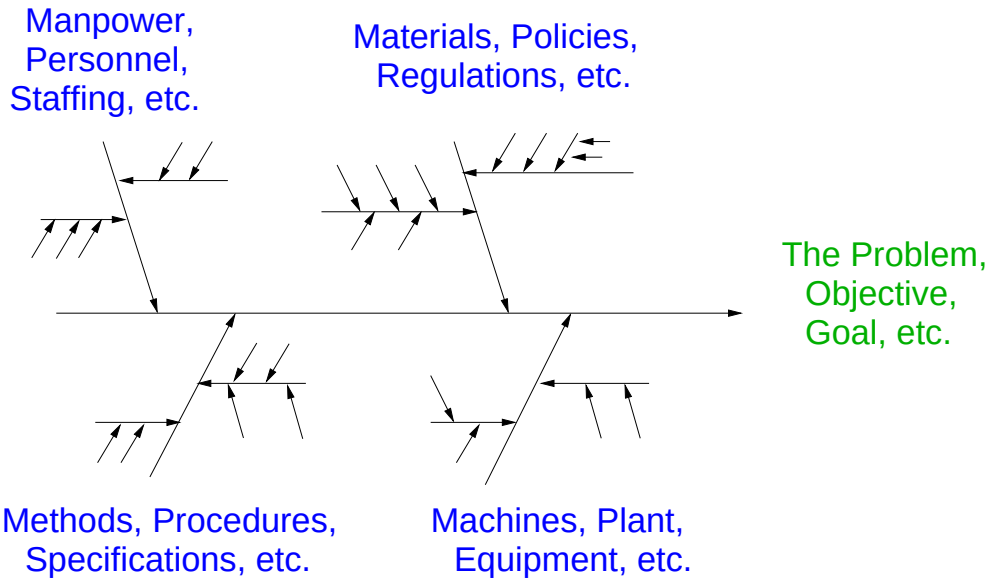
Selection of linking condition will greatly influence accident cause identified

Limitations of Event Chain Models (3)

- Selecting countermeasures
  - Leads to overreliance on redundancy
- Risk assessment
  - Usually assumes independence between events
  - Events chosen will affect accuracy but subjective
  - Usually concentrates on failure events
- Treating events and conditions as causes
  - Can miss systemic causes
  - Root cause analysis limited if use event chain models
    (fault trees, fishbone diagrams, barrier analysis, etc.)

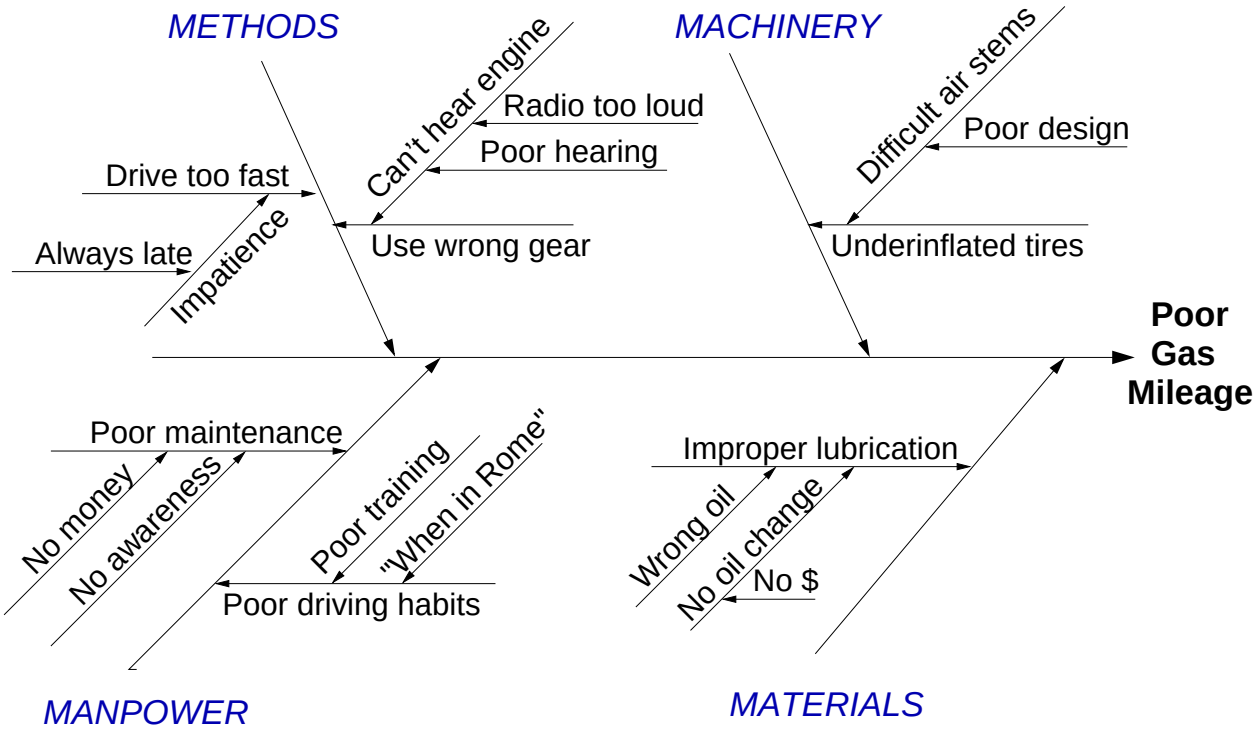
Fishbone Diagram

- Reinvention of fault trees by a management professor
  (fault tree drawn differently and with some guidance on content)

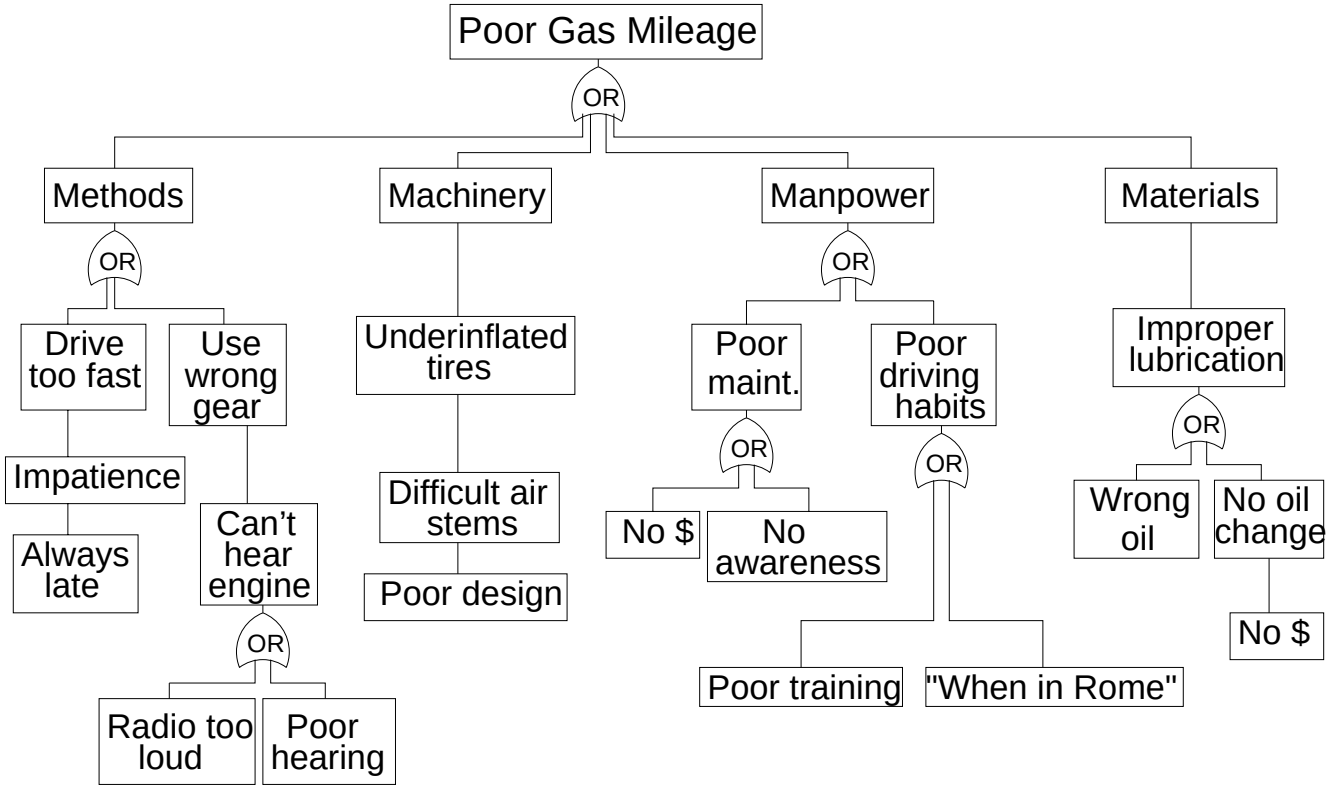


Fishbone Diagram Example

(taken off the web)



Fishbone Example Redrawn in Form of Fault Tree



Limitations of Event Chain models (4)

- Social and organizational factors in accidents.

> *Underlying every technology is at least one basic science, although the technology may be well developed long before the science emerges. Overlying every technical or civil system is a social system that provides purpose, goals, and decision criteria.*
>
> *Ralph Miles Jr.*

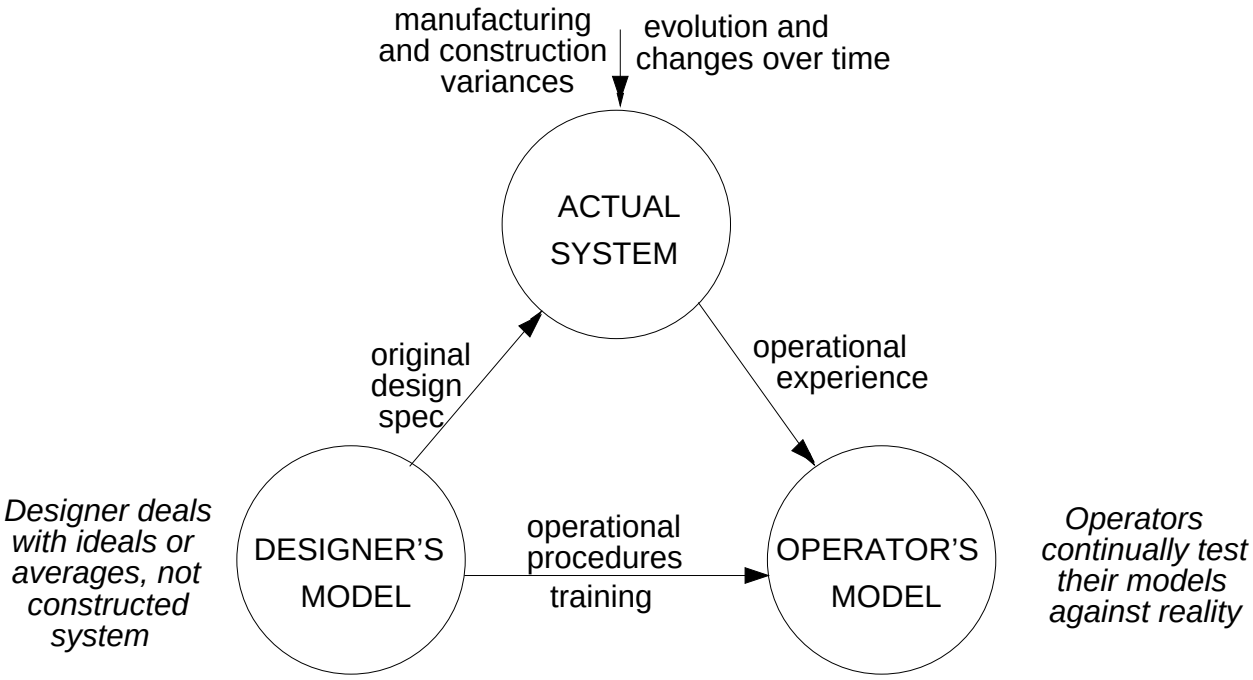
  - Models need to include the social system as well as the technology and its underlying science.

- System accidents
- Software

Limitations of Event Chain Models (5)

- Human error
  - Define as deviation from normative procedure, but operators always deviate from standard procedures.
    - normative procedures vs. effective procedures
    - sometimes violation of rules has prevented accidents
  - Cannot effectively model human behavior by decomposing it into individual decisions and acts and studying it in isolation from the
    - physical and social context
    - value system in which takes place
    - dynamic work process

Mental Models

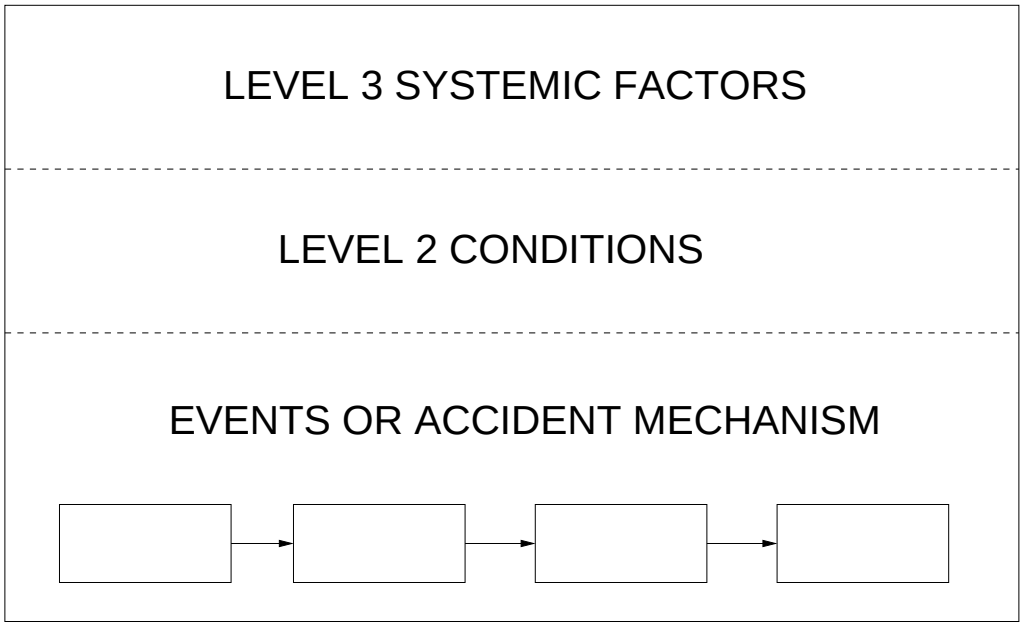


System changes and so must operator's model

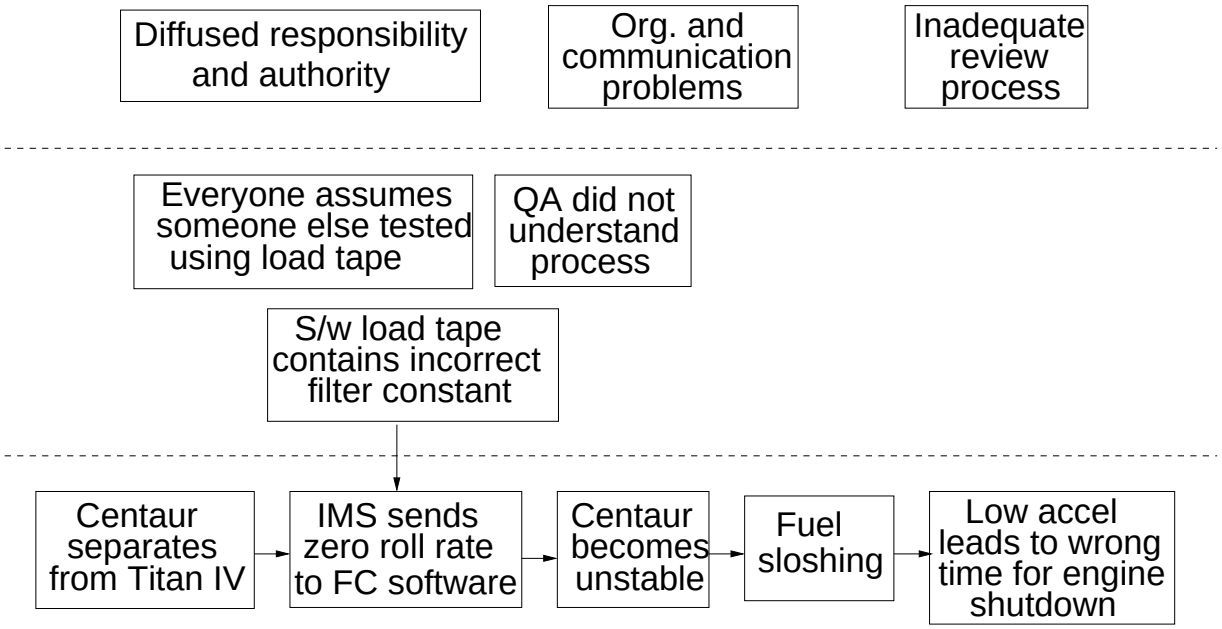
Limitations of Event Chain Models (6)

- Human error (con't.)
  - Less successful actions are natural part of search by operator for optimal performance.
- Adaptation
  - Systems are continually changing
  - Systems and organizations migrate toward accidents

Hierarchical Models



Hierarchical Analysis Example



A New Accident Model

A Systems Theory Model of Accidents

- Accidents arise from interactions among humans, machines, and the environment.
  - Not simply chains of events or linear causality, but more complex types of causal connections.
- Safety is an emergent property that arises when components of system interact with each other within a larger environment.
  - A set of constraints related to behavior of components in system enforces that property.
  - Accidents when interactions violate those constraints (a lack of appropriate constraints on the interactions).
  - Software as a controller embodies or enforces those constraints.

STAMP (Systems−Theoretic Accident Model and Processes)

- Based on systems and control theory
- Systems not treated as a static design
  - A socio−technical system is a dynamic process continually adapting to achieve its ends and to react to changes in itself and its environment
  - Preventing accidents requires designing a control structure to enforce constraints on system behavior and adaptation.

STAMP (2)

- Views accidents as a control problem

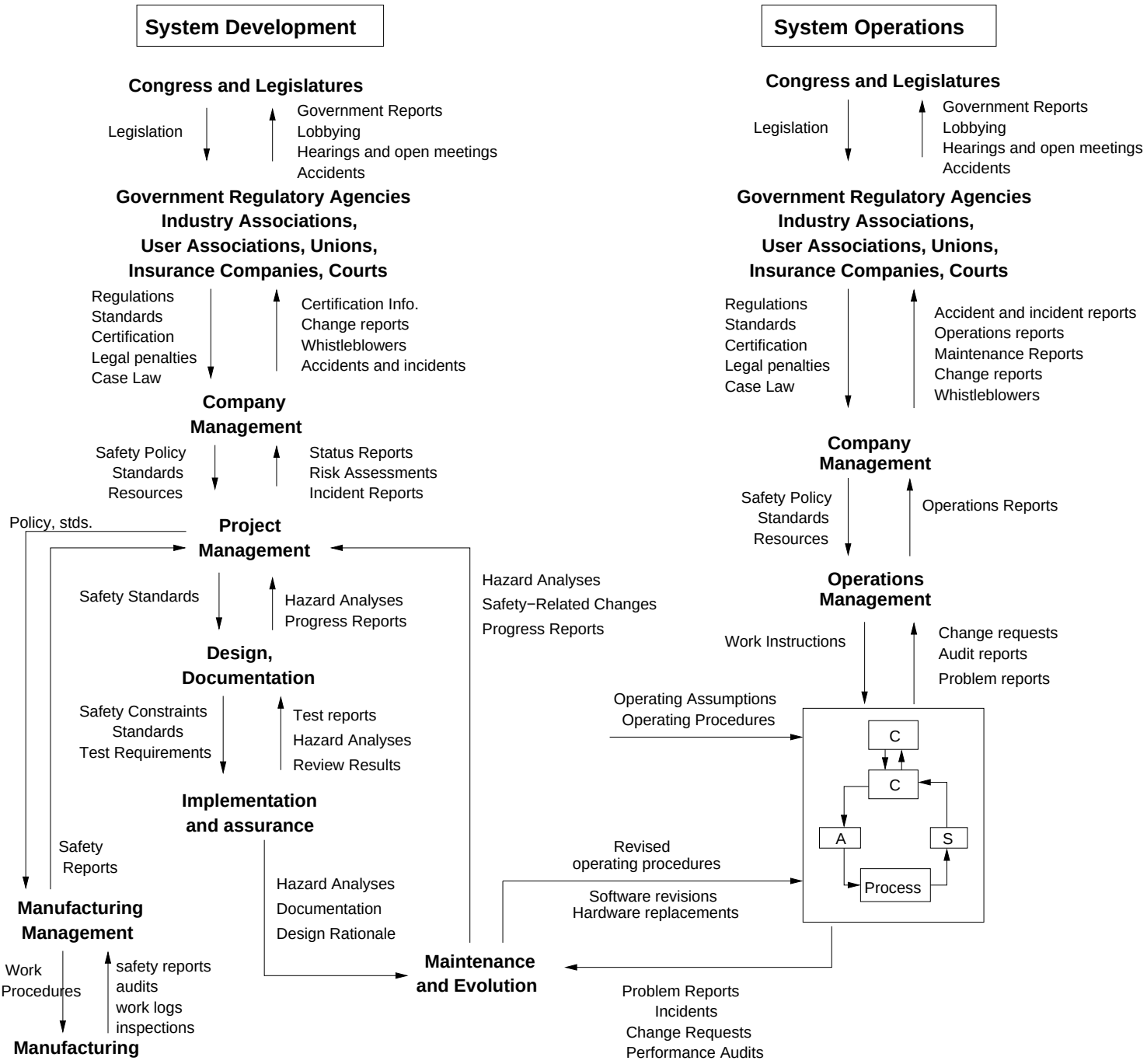
  e.g., O−ring did not control propellant gas release by sealing gap in field joint

  Software did not adequately control descent speed of Mars Polar Lander.

- Events are the result of the inadequate control

  Result from lack of enforcement of safety constraints

- To understand accidents, need to examine control structure itself to determine why inadequate to maintain safety constraints and why events occurred.


System Development
System Operations
Congress and Legislatures
Legislation
Government Reports
Lobbying
Hearings and open meetings
Accidents
Government Regulatory Agencies
Industry Associations,
User Associations, Unions,
Insurance Companies, Courts
Regulations
Standards
Certification
Legal penalties
Case Law
Certification Info.
Change reports
Whistleblowers
Accidents and incidents
Accident and incident reports
Operations reports
Maintenance Reports
Change reports
Whistleblowers
Company
Management
Safety Policy
Standards
Resources
Status Reports
Risk Assessments
Incident Reports
Operations Reports
Policy, stds.
Project
Management
Safety Standards
Hazard Analyses
Progress Reports
Hazard Analyses
Safety−Related Changes
Progress Reports
Design,
Documentation
Safety Constraints
Standards
Test Requirements
Test reports
Hazard Analyses
Review Results
Implementation
and assurance
Safety
Reports
Hazard Analyses
Documentation
Design Rationale
Manufacturing
Management
Work
Procedures
safety reports
audits
work logs
inspections
Manufacturing
Maintenance
and Evolution
Operations
Management
Work Instructions
Change requests
Audit reports
Problem reports
Operating Assumptions
Operating Procedures
C
C
A
S
Process
Revised
operating procedures
Software revisions
Hardware replacements
Problem Reports
Incidents
Change Requests
Performance Audits

Note:

- Does not imply need for a "controller"

  Component failures may be controlled through design

  e.g., redundancy, interlocks, fail−safe design

  or through process

  manufacturing processes and procedures

  maintenance procedures

- But does imply the need to enforce the safety constraints in some way.

- New model includes what do now and more

Accidents occur when:

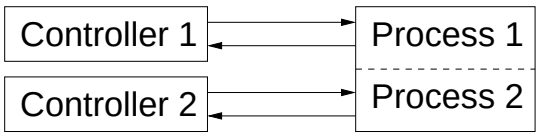
- Design does not enforce safety constraints

  unhandled disturbances, failures, dysfunctional interactions

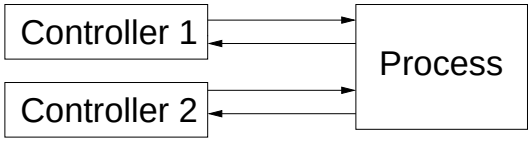
- Inadequate control actions

- Control structure degrades over time, asynchronous evolution

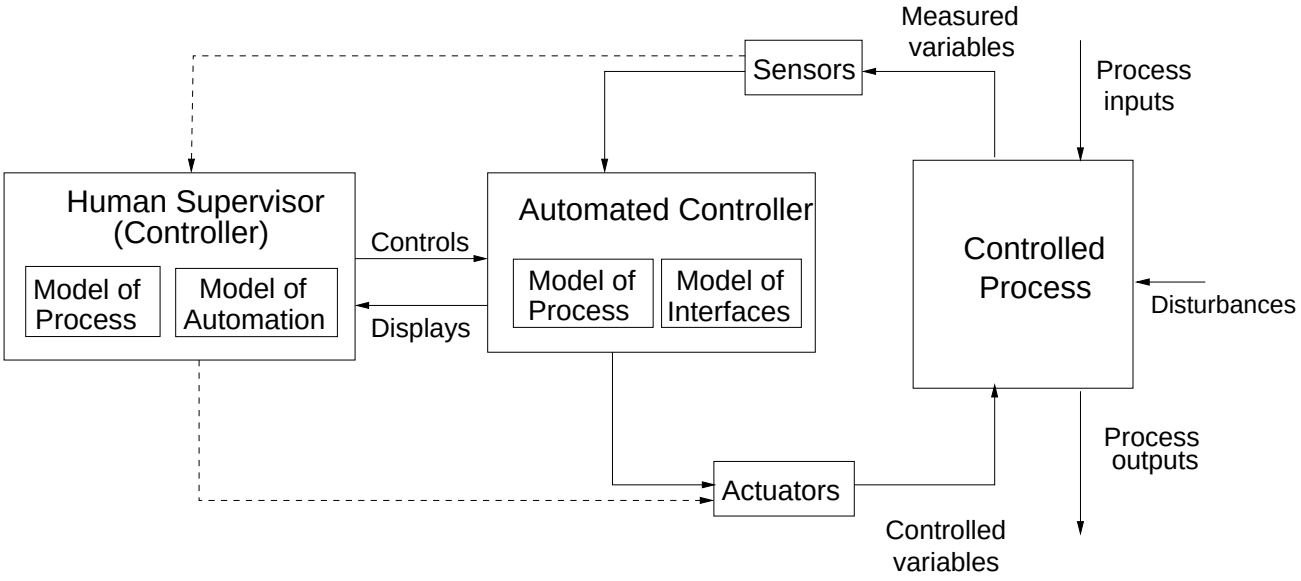
- Control actions inadequately coordinated among multiple controllers.
  - Boundary areas



  - Overlap areas (side effects of decisions and control actions)



Process Models



Process models must contain:

Required relationship among process variables
Current state (values of process variables)
The ways the process can change state

Relationship between Safety and Process Model

- Accidents occur when the models do not match the process and incorrect control commands are given (or correct ones not given)
- How do they become inconsistent?
  - Wrong from beginning
    e.g. uncontrolled disturbances
    unhandled process states
    inadvertently commanding system into a hazardous state
    unhandled or incorrectly handled system component failures
    [Note these are related to what we called system accidents]
  - Missing or incorrect feedback and not updated correctly
  - Time lags not accounted for
- Explains most software−related accidents

Safety and Human Mental Models

- Explains developer errors
  - May have incorrect model of
    - required system or software behavior
    - development process
    - physical laws
    - etc.
- Also explains most human/computer interaction problems
  - Pilots and others are not understanding the automation

    What did it just do?
    Why did it do that?
    What will it do next?
    How did it get us into this state?
    How do I get it to do what I want?
    Why won't it let us do that?
    What caused the failure?
    What can we do so it does not happen again?
  - Or don't get feedback to update mental models or disbelieve it

---

Validating and Using the Model

- Can it explain (model) accidents that have already occurred?
- Is it useful?
  - In accident and mishap investigation
  - In preventing accidents

    Hazard analysis
    Designing for safety
- Is it better for these purposes than the chain−of−events model?

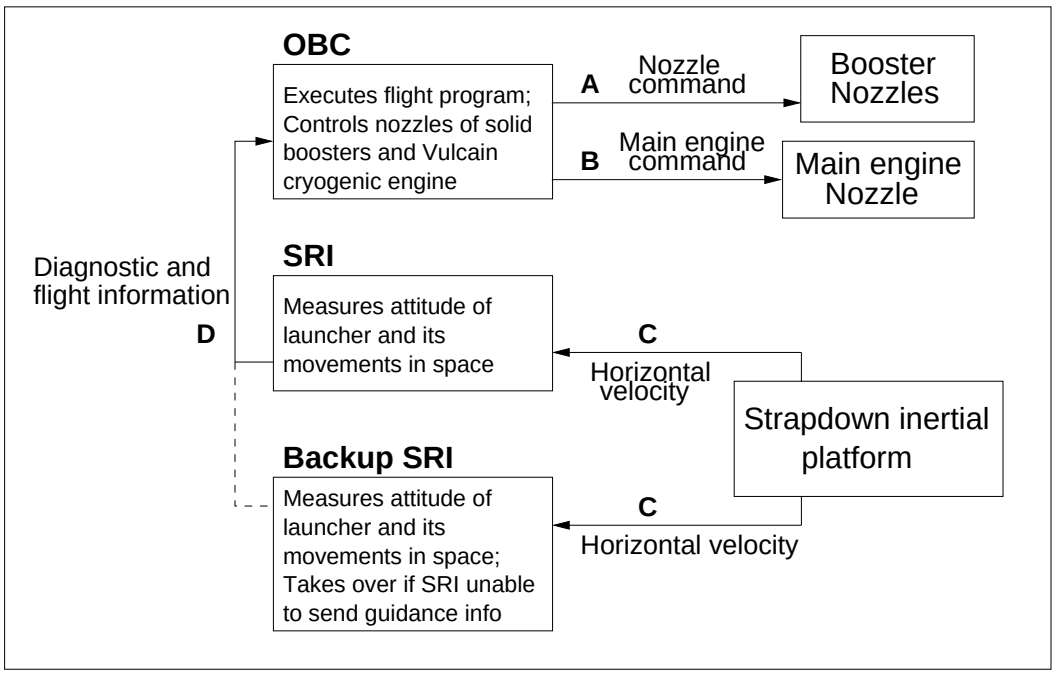
Using STAMP in Accident and Mishap Investigation and Root Cause Analysis

Modeling Accidents Using STAMP

Three types of models are needed:

1. Static safety control structure
2. Dynamic structure
   - Shows how the safety control structure changed over time
3. Behavioral dynamics
   - Dynamic processes behind the changes, i.e., why the system changes

ARIANE 5 LAUNCHER



Ariane 5: A rapid change in attitude and high aerodynamic loads stemming from a high angle of attack create aerodynamic forces that cause the launcher to disintegrate at 39 seconds after command for main engine ignition (H0).

Nozzles: Full nozzle deflections of solid boosters and main engine lead to angle of attack of more than 20 degrees.

Self−Destruct System: Triggered (as designed) by boosters separating from main stage at altitude of 4 km and 1 km from launch pad.

OBC (On−Board Computer)

OBC Safety Constraint Violated: Commands from the OBC to the nozzles must not result in the launcher operating outside its safe envelope.

Unsafe Behavior: Control command sent to booster nozzles and later to main engine nozzle to make a large correction for an attitude deviation that had not occurred.

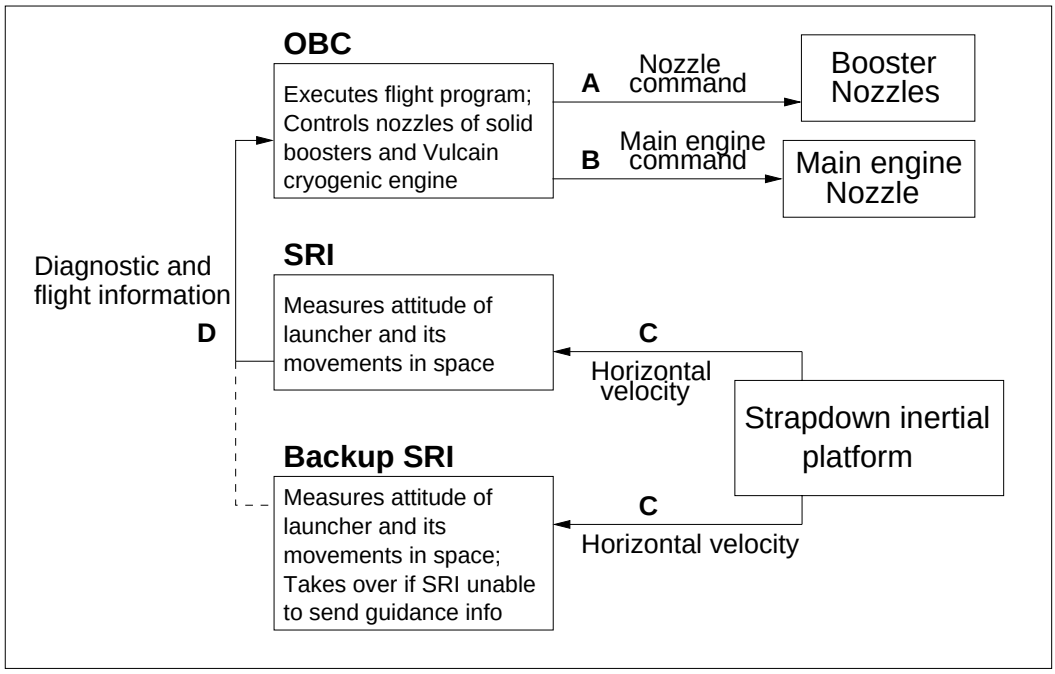
Process Model: Model of the current launch attitude is incorrect, i.e., it contains an attitude deviation that had not occurred. Results in incorrect commands being sent to nozzles.

Feedback: Diagnostic information received from SRI

Interface Model: Incomplete or incorrect (not enough information in accident report to determine which) − does not include the diagnostic information from the SRI that is available on the databus.

Control Algorithm Flaw: Interprets diagnostic information from SRI as flight data and uses it for flight control calculations. With both SRI and backup SRI shut down and therefore no possibility of getting correct guidance and attitude information, loss was inevitable.

ARIANE 5 LAUNCHER



SRI (Inertial Reference System):

SRI Safety Constraint Violated: The SRI must continue to send guidance information as long as it can get the necessary information from the strapdown inertial platform.

Unsafe Behavior: At 36.75 seconds after H0, SRI detects an internal error and turns itself off (as it was designed to do) after putting diagnostic information on the bus (D).

Control Algorithm: Calculates the Horizontal Bias (an internal alignment variable used as an indicator of alignment precision over time) using the horizontal velocity input from the strapdown inertial platform (C). Conversion from a 64−bit floating point value to a 16−bit signed integer leads to an unhandled overflow exception while calculating the horizontal bias. Algorithm reused from Ariane 4 where horizontal bias variable does not get large enough to cause an overflow.

Process Model: Does not match Ariane 5 (based on Ariane 4 trajectory data); Assumes smaller horizontal velocity values than possible on Ariane 5.

Backup SRI (Inertial Reference System):

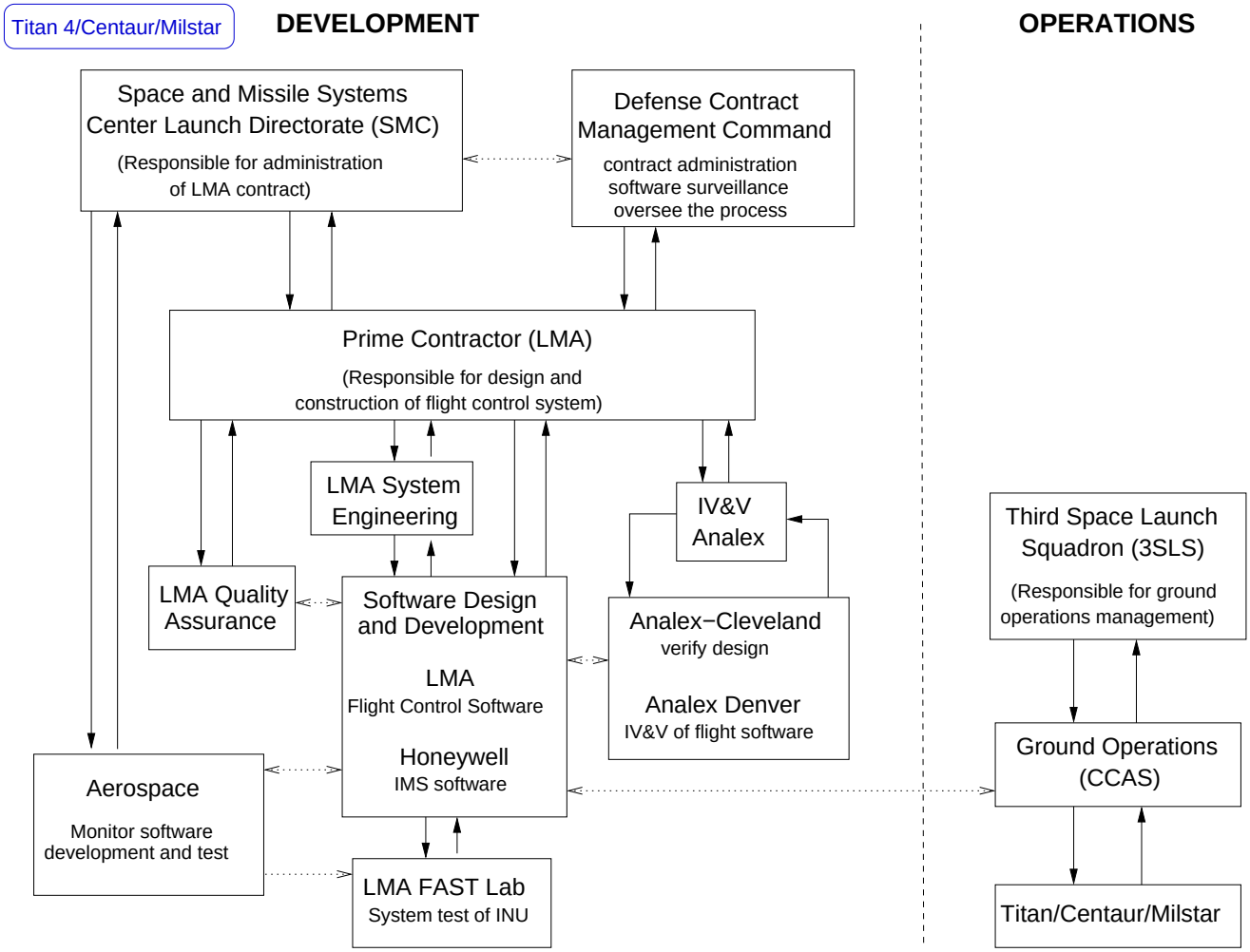
SRI Safety Constraint Violated: The backup SRI must continue to send guidance information as long as it can get the necessary information from the strapdown inertial platform.

Unsafe Behavior: At 36.75 seconds after H0, backup SRI detects an internal error and turns itself off (as it was designed to do).

Control Algorithm: Calculates the Horizontal Bias (an internal alignment variable used as an indicator of alignment precision over time) using the horizontal velocity input from the strapdown inertial platform (C). Conversion from a 64−bit floating point value to a 16−bit signed integer leads to an unhandled overflow exception while calculating the horizontal bias. Algorithm reused from Ariane 4 where horizontal bias variable does not get large enough to cause an overflow. Because the algorithm was the same in both SRI computers, the overflow results in the same behavior, i.e., shutting itself off.

Process Model: Does not match Ariane 5 (based on Ariane 4 trajectory data); Assumes smaller horizontal velocity values than possible on Ariane 5.

Titan 4/Centaur/Milstar



Analex IV&V

Safety Constraint:

- IV&V must be performed on the as−flown system
- All safety−crtiical data and software must be included

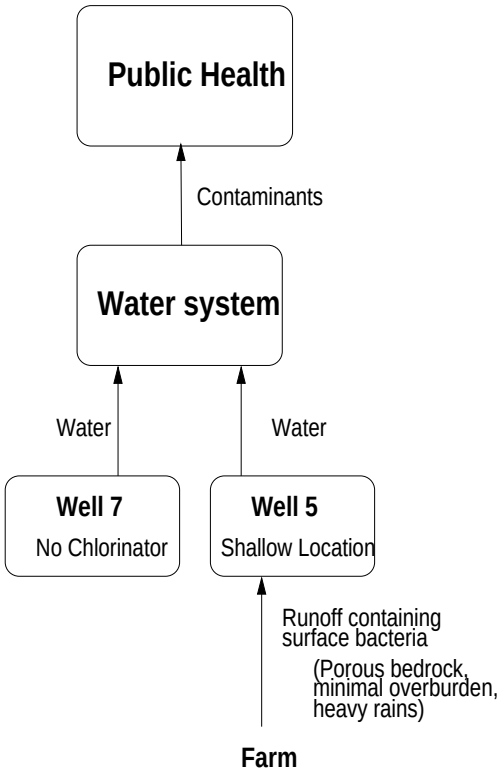
Control Flaws:

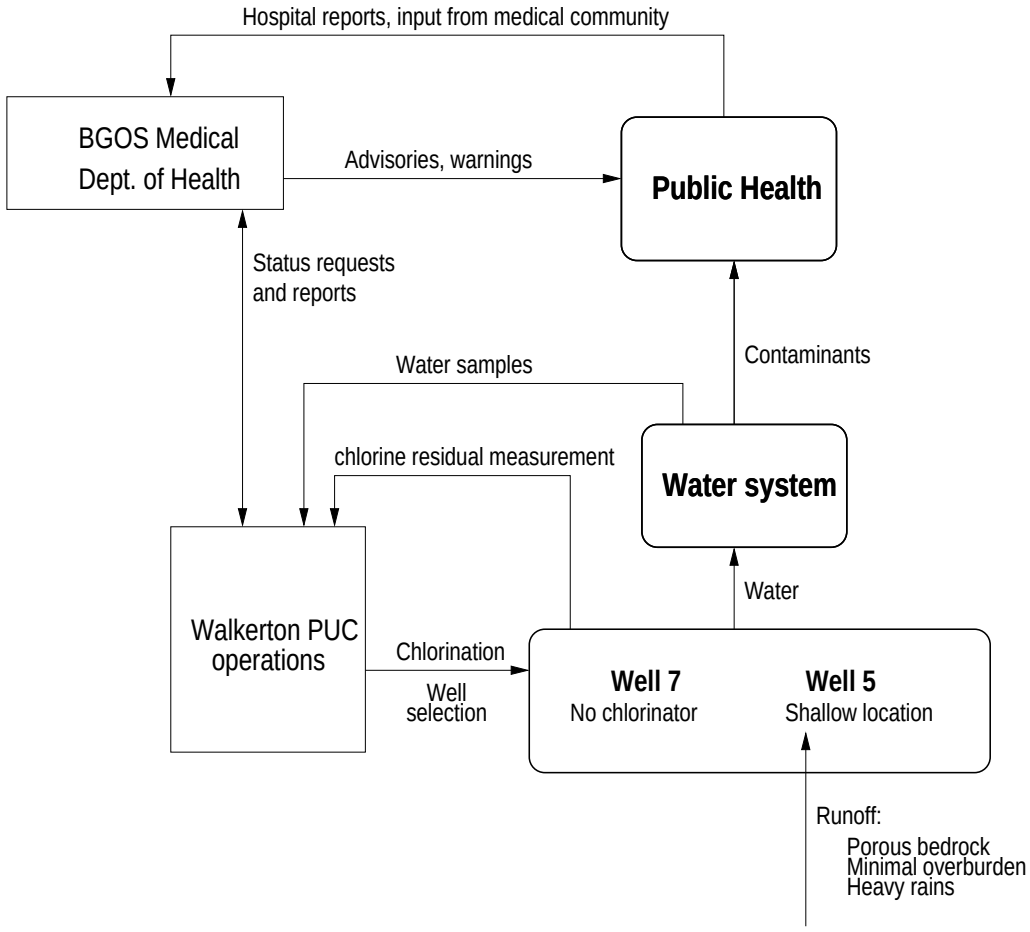
- Designed an IV&V process that did not include load tape
- Used default values for testing software implementation
- Validated design constant but not actual constant

Mental Model Flaws:

- Misunderstanding about what could be tested
- Misunderstainding of load tape creation process

Walkerton Physical Process



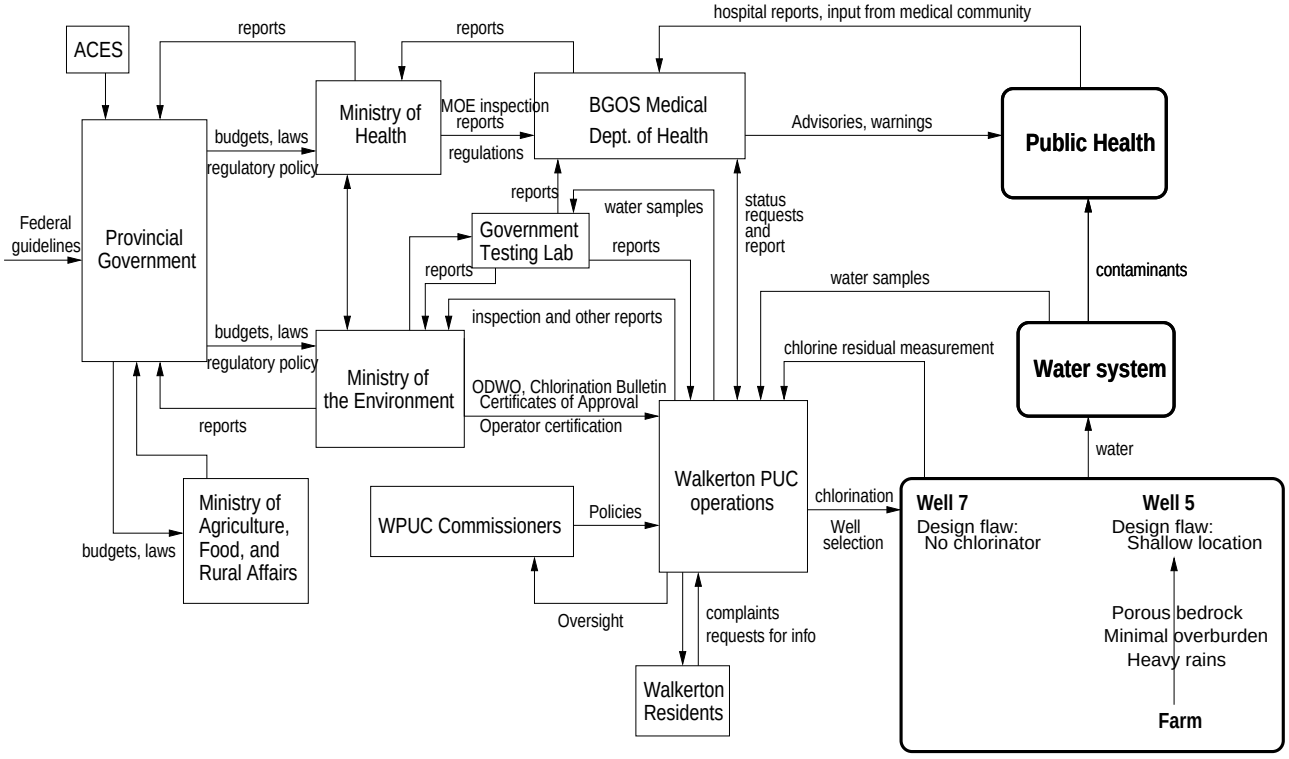


System Hazard: Public is exposed to e. coli or other health−related contaminants through drinking water.

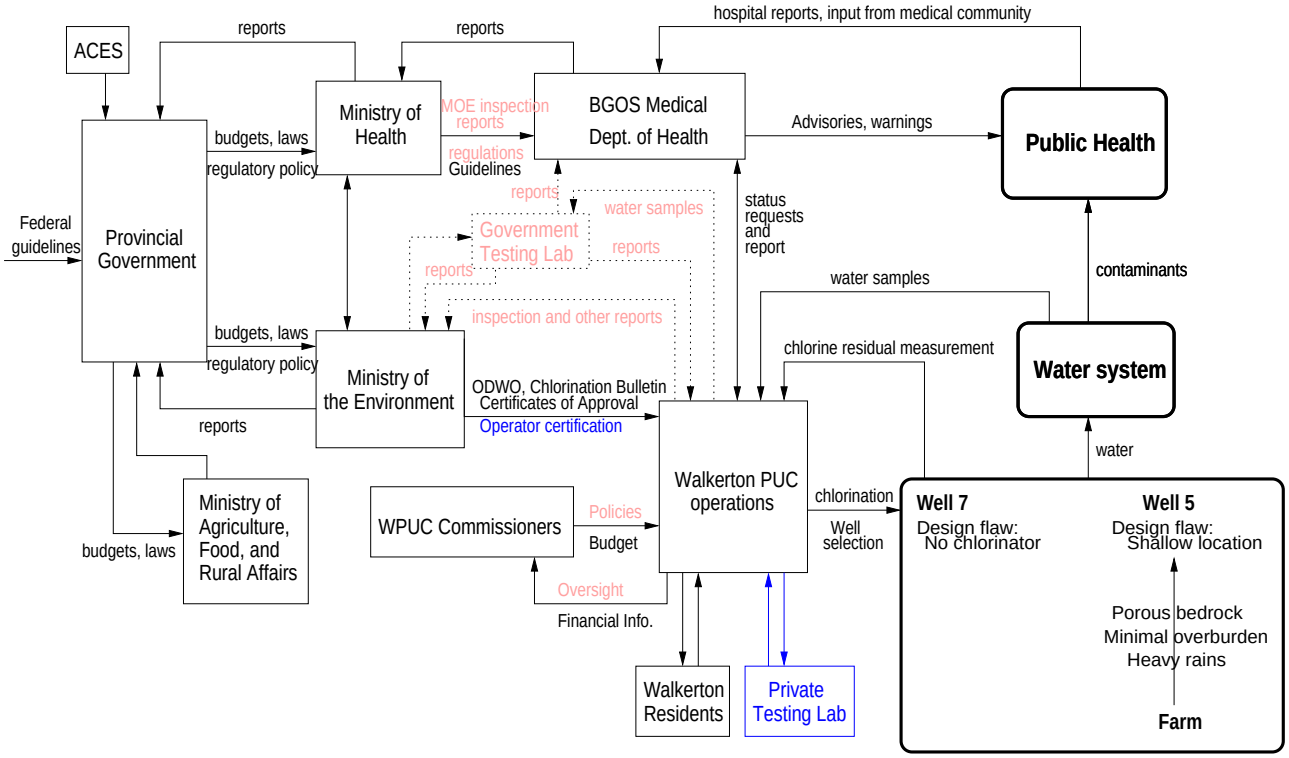
System Safety Constraints: The safety control structure must prevent exposure of the public to contaminated water.

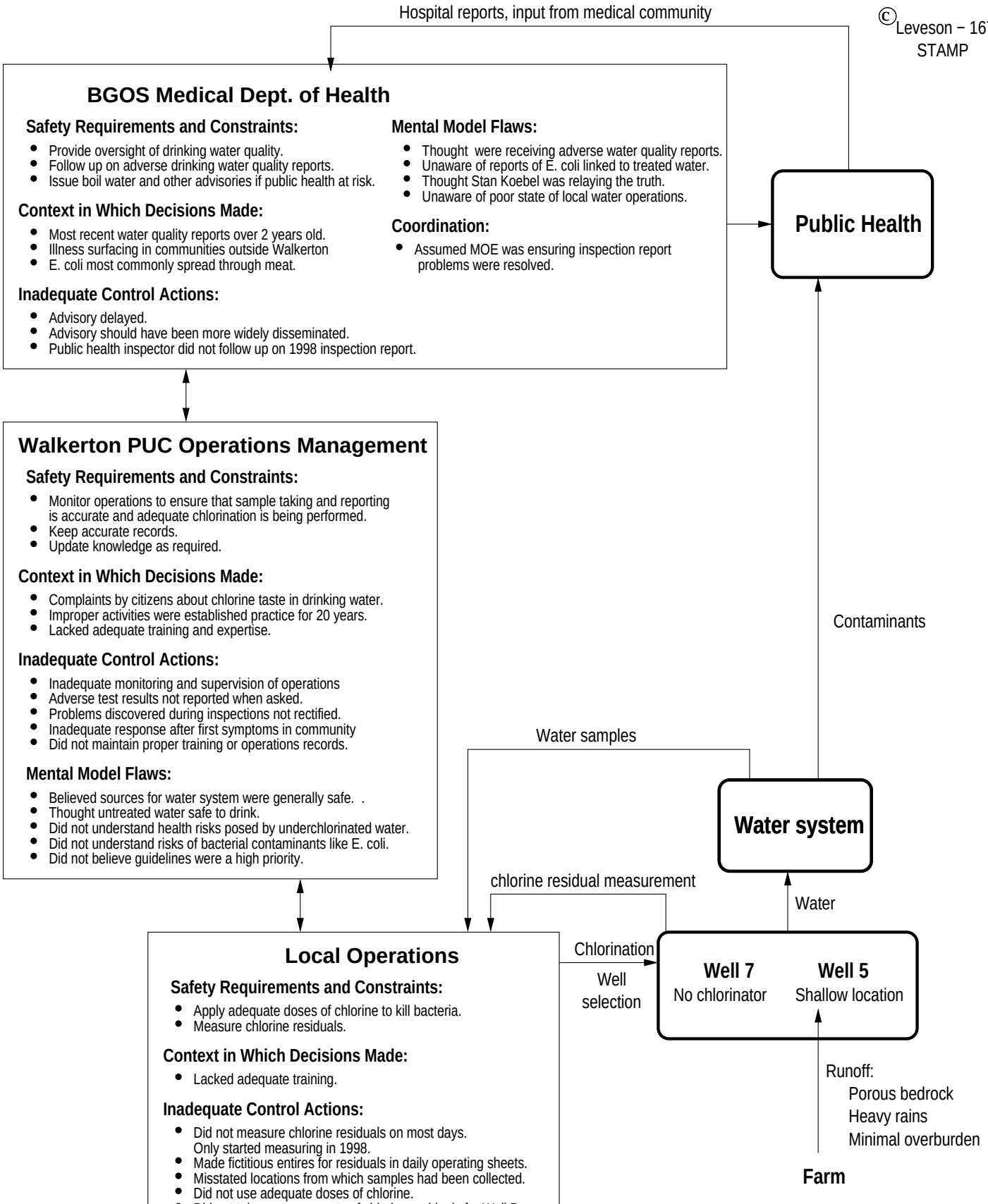
(1) Water quality must not be compromised.

(2) Public health measures must reduce risk of exposure if water quality is compromised (e.g., notification and procedures to follow)

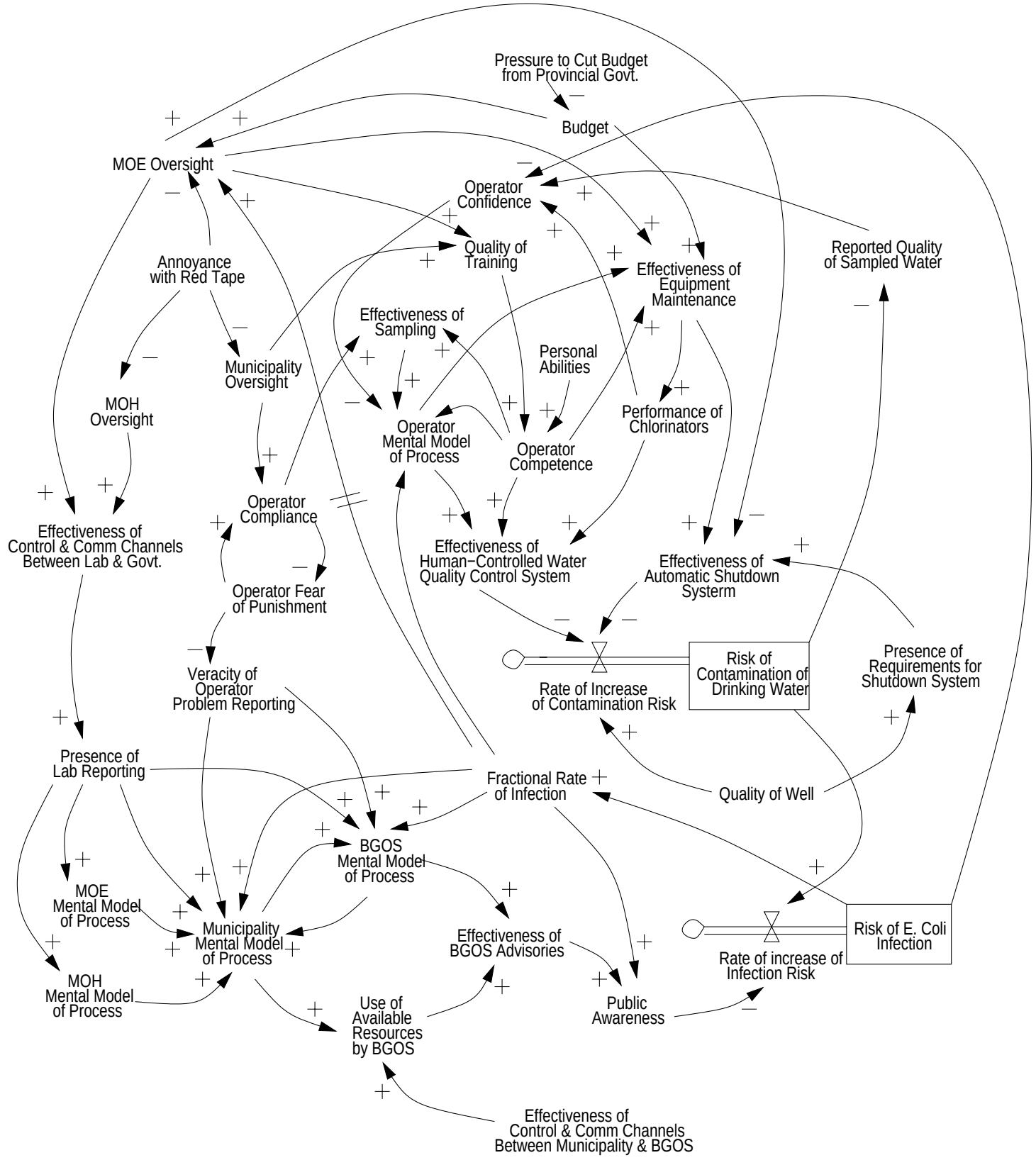


Dynamic Structure

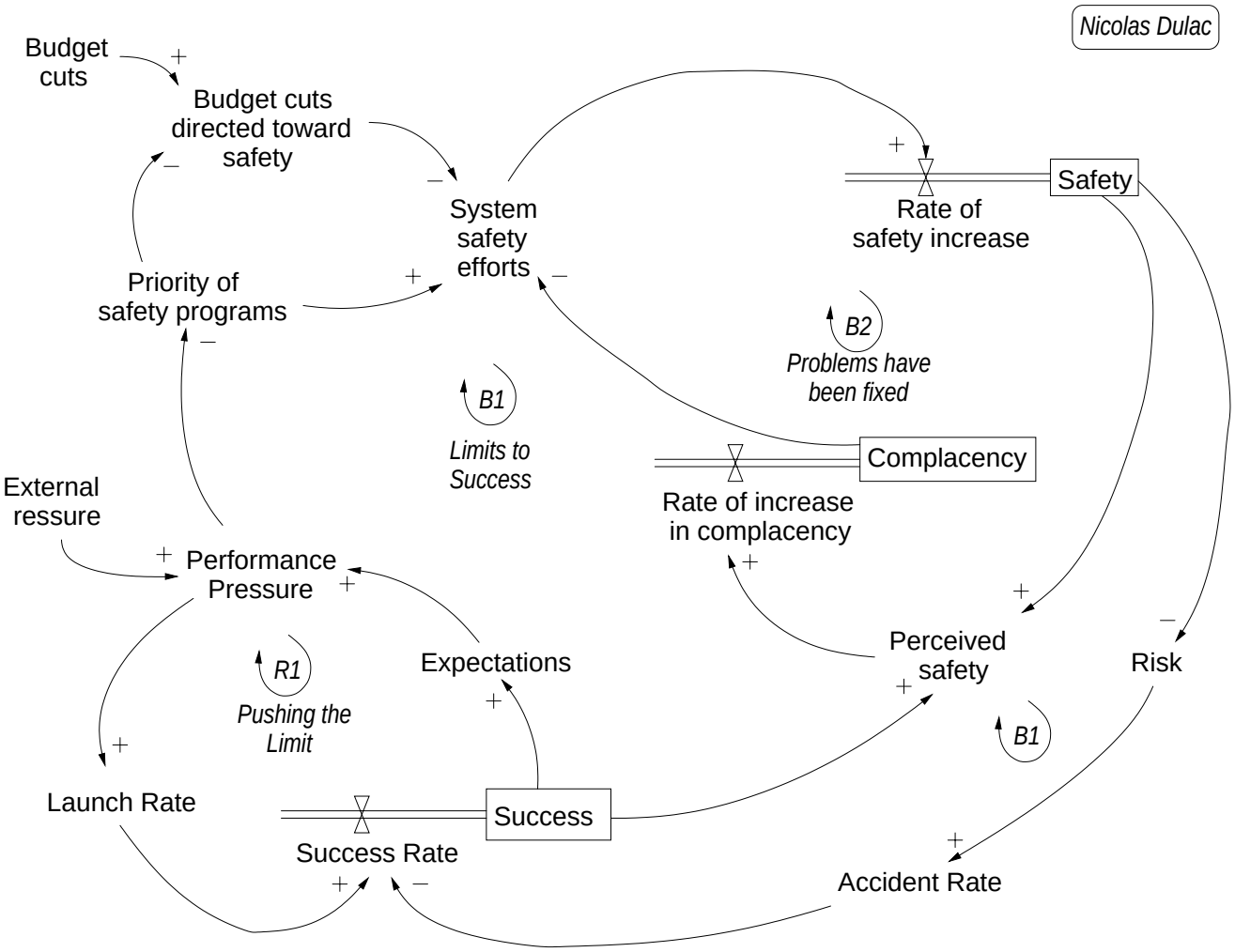




Modeling Behavioral Dynamics



A (Partial) System Dynamics Model of the Columbia Accident



Steps in a STAMP analysis:

1. Identify
   - System hazards
   - System safety constraints and requirements
   - Control structure in place to enforce constraints
2. Model dynamic aspects of accident:
   - Changes to static safety control structure over time
   - Dynamic processes in effect that led to changes
3. Create the overall explanation for the accident
   - Inadequate control actions and decisions
   - Context in which decisions made
   - Mental model flaws
   - Control flaws (e.g., missing feedback loops)
   - Coordination flaws

STAMP vs. Traditional Accident Models

- Examines interrelationships rather than linear cause−effect chains
- Looks at the processes behind the events
- Includes entire socio−economic system
- Includes behavioral dynamics (changes over time)
  - Want to not just react to accidents and impose controls for a while, but understand why controls drift toward ineffectiveness over time and
    - Change those factors if possible
    - Detect the drift before accidents occur

Using STAMP to Prevent Accidents

Hazard Analysis

Safety Metrics and Performance Auditing

Risk Assessment

STAMP−Based Hazard Analysis (STPA)

- Provides information about how safety constraints could be violated.

  Used to eliminate, reduce, and control hazards in system design, development, manufacturing, and operations

- Assists in designing safety into system from the beginning

  Not just after−the−fact analysis

- Includes software, operators, system accidents, management, regulatory authorities

- Can use a concrete model of control (SpecTRM−RL) that is executable and analyzable

STPA − Step1: Identify hazards and translate into high−level requirements and constraints on behavior

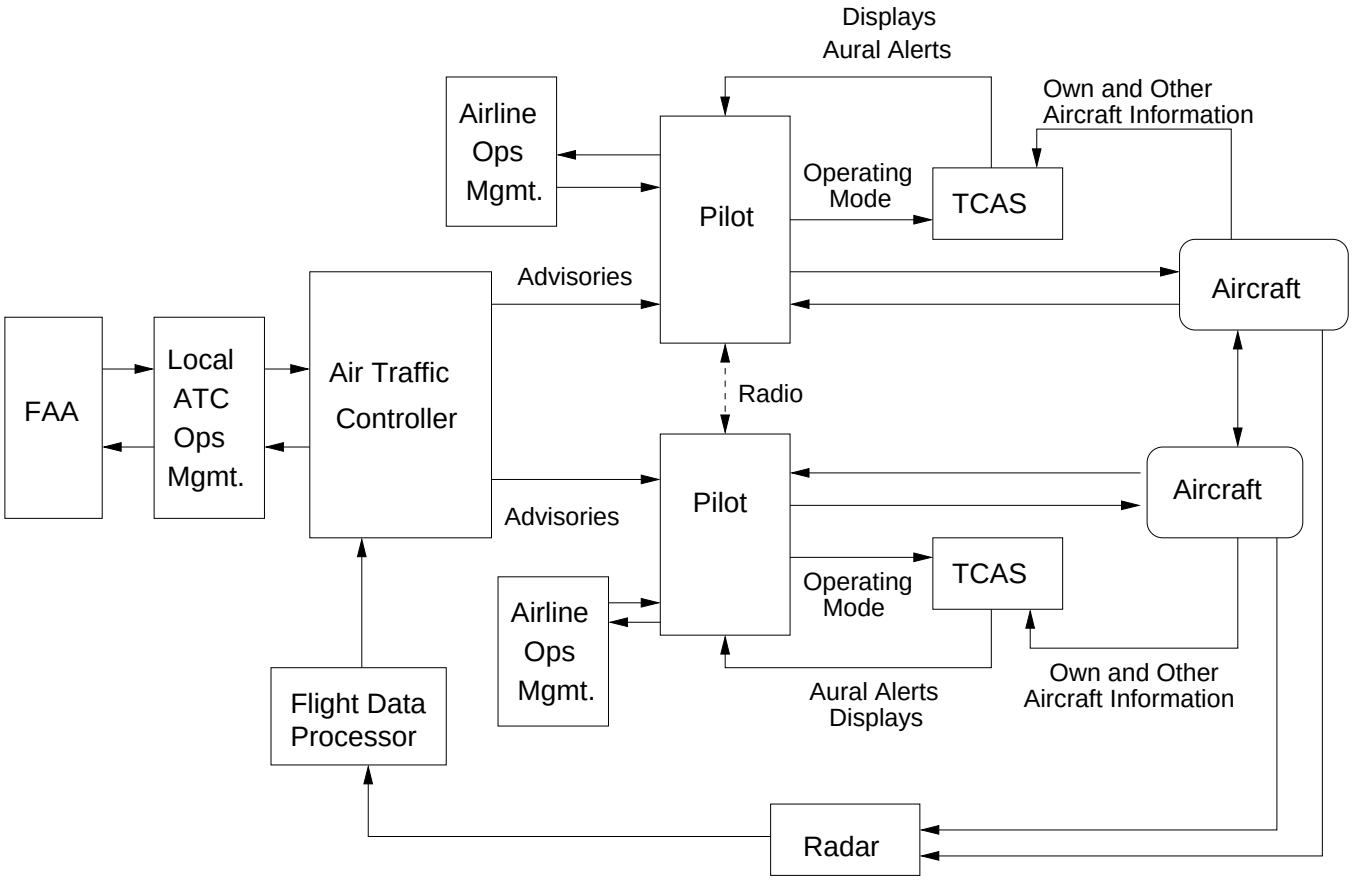
TCAS Hazards

1. A near mid−air collision (NMAC)

   (a pair of controlled aircraft violate minimum separation standards)

2. A controlled maneuver into the ground
3. Loss of control of aircraft
4. Interference with other safety−related aircraft systems
5. Interference with ground−based ATC system
6. Interference with ATC safety−related advisory

STPA − Step 2: Define basic control structure



STPA − Step 3: Identify potential inadequate control actions that could lead to hazardous process state

In general:

1. A required control action is not provided
2. An incorrect or unsafe control action is provided.
3. A potentially correct or inadequate control action is provided too late (at the wrong time)
4. A correct control action is stopped too soon

For the NMAC hazard:

TCAS:

1. The aircraft are on a near collision course and TCAS does not provide an RA
2. The aircraft are in close proximity and TCAS provides an RA that degrades vertical separation
3. The aircraft are on a near collision course and TCAS provides an RA too late to avoid an NMAC
4. TCAS removes an RA too soon.

Pilot:

1. The pilot does not follow the resolution advisory provided by TCAS (does not respond to the RA)
2. The pilot incorrectly executes the TCAS resolution advisory.
3. The pilot applies the RA but too late to avoid the NMAC
4. The pilot stops the RA maneuver too soon.

STPA − Step 4: Determine how potentially hazardous control actions could occur.

Eliminate from design or control or mitigate in design or operations

In general:

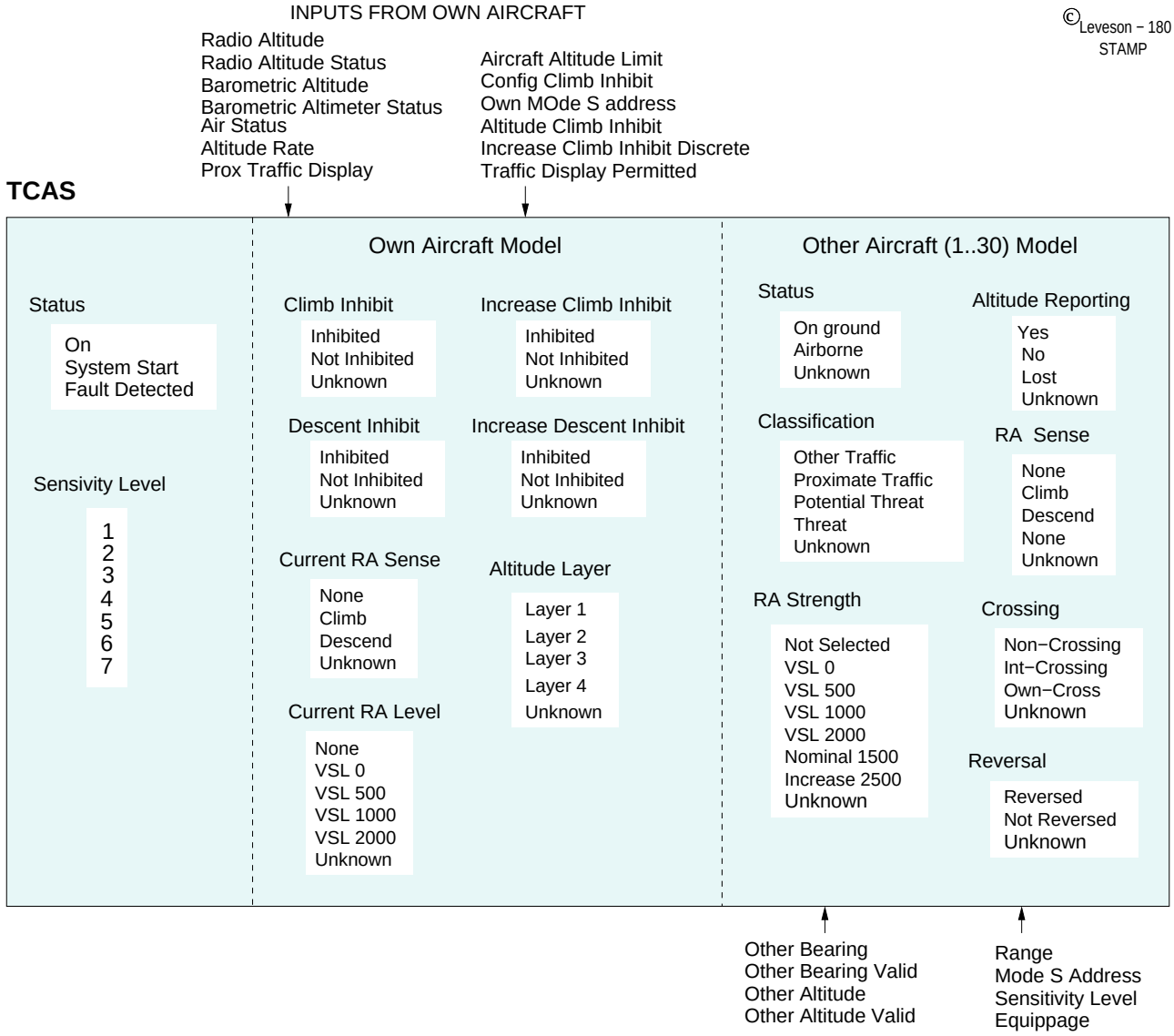
- Can use a concrete model in SpecTRM−RL
  - Assists with communication and completeness of analysis
  - Provides a continuous simulation and analysis environment to evaluate impact of faults and effectiveness of mitigation features.

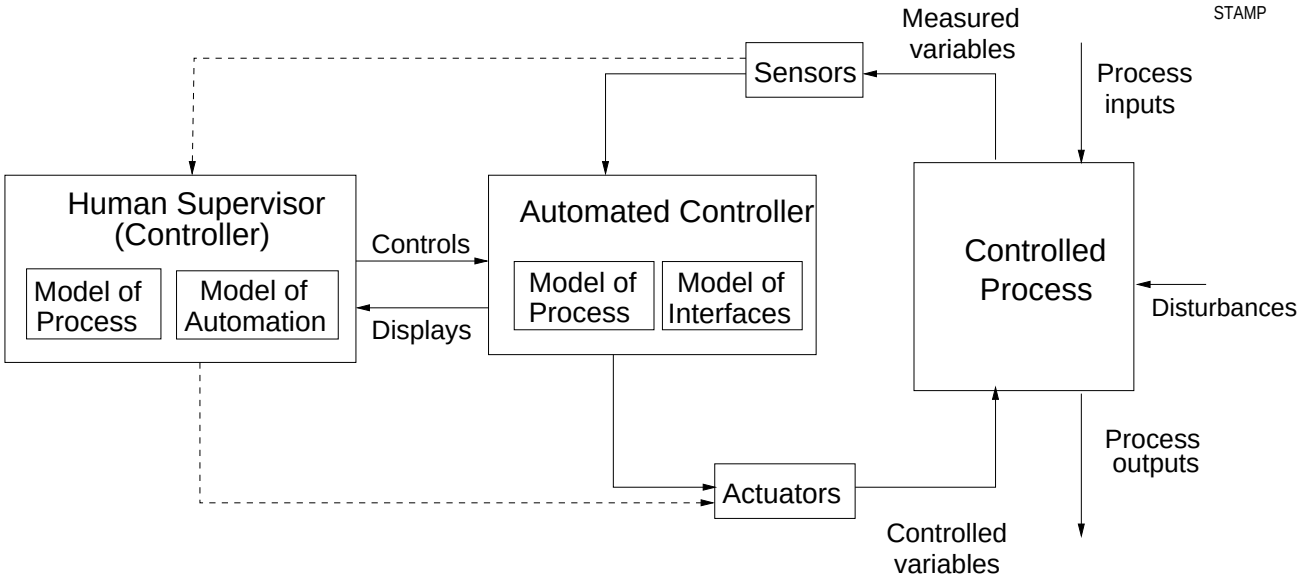
Step 4a: Augment control structure with process models for each control component

Step 4b: For each of inadequate control actions, examine parts of control loop to see if could cause it.

- Guided by set of generic control loop flaws
- Where human or organization involved must evaluate:
  - Context in which decisions made
  - Behavior−shaping mechanisms (influences)

Step 4c: Consider how designed controls could degrade over time


INPUTS FROM OWN AIRCRAFT
Radio Altitude
Radio Altitude Status
Barometric Altitude
Barometric Altimeter Status
Air Status
Altitude Rate
Prox Traffic Display
Aircraft Altitude Limit
Config Climb Inhibit
Own MOde S address
Altitude Climb Inhibit
Increase Climb Inhibit Discrete
Traffic Display Permitted
TCAS
Own Aircraft Model
Other Aircraft (1..30) Model
Status
On
System Start
Fault Detected
Sensivity Level
1
2
3
4
5
6
7
Climb Inhibit
Inhibited
Not Inhibited
Unknown
Increase Climb Inhibit
Inhibited
Not Inhibited
Unknown
Descent Inhibit
Inhibited
Not Inhibited
Unknown
Increase Descent Inhibit
Inhibited
Not Inhibited
Unknown
Current RA Sense
None
Climb
Descend
Unknown
Altitude Layer
Layer 1
Layer 2
Layer 3
Layer 4
Unknown
Current RA Level
None
VSL 0
VSL 500
VSL 1000
VSL 2000
Unknown
Status
On ground
Airborne
Unknown
Altitude Reporting
Yes
No
Lost
Unknown
Classification
Other Traffic
Proximate Traffic
Potential Threat
Threat
Unknown
RA Sense
None
Climb
Descend
None
Unknown
RA Strength
Not Selected
VSL 0
VSL 500
VSL 1000
VSL 2000
Nominal 1500
Increase 2500
Unknown
Crossing
Non−Crossing
Int−Crossing
Own−Cross
Unknown
Reversal
Reversed
Not Reversed
Unknown
Other Bearing
Other Bearing Valid
Other Altitude
Other Altitude Valid
Range
Mode S Address
Sensitivity Level
Equippage
INPUTS FROM OTHER AIRCRAFT


Measured variables
Sensors
Process inputs
Human Supervisor (Controller)
Model of Process
Model of Automation
Controls
Displays
Automated Controller
Model of Process
Model of Interfaces
Controlled Process
Disturbances
Process outputs
Actuators
Controlled variables

STPA − Step 4b: Examine control loop for potential to cause inadequate control actions

- **Inadequate Control Actions (enforcement of constraints)**
  - Design of control algorithm (process) does not enforce constraints
  - Process models inconsistent, incomplete, or incorrect (lack of linkup)
    - Flaw(s) in creation or updating process
    - Inadequate or missing feedback
      - Not provided in system design
      - Communication flaw
      - Inadequate sensor operation (incorrect or no information provided)
    - Time lags and measurement inaccuracies not accounted for
  - Inadequate coordination among controllers and decision−makers (boundary and overlap areas)
- **Inadequate Execution of Control Action**
  - Communication flaw
  - Inadequate "actuator" operation
  - Time lag

STPA − Step4c: Consider how designed controls could degrade over time.

- E.g., specified procedures ==> effective procedures
- Use system dynamics models?

• Use information to design protection against changes:

e.g. operational procedures

controls over changes and maintenance activities

auditing procedures and performance metrics

management feedback channels to detect unsafe changes

Comparisons with Traditional HA Techniques

- Top−down (vs. bottom−up like FMECA)
- Considers more than just component failures and failure events
- Guidance in doing analysis (vs. FTA)
- Handles dysfunctional interactions, software, management, etc.
- Concrete model (not just in head)
  - Not physical structure (HAZOP) but control (functional) structure
  - General model of inadequate control
    - HAZOP guidewords based on model of accidents being caused by deviations in system variables
    - Includes HAZOP model but more general
- Compared with TCAS II Fault Tree (MITRE)

  STPA results more comprehensive