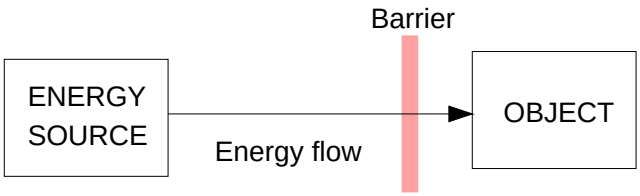
Accident models provide the basis for

- Investigating and analyzing accidents
- Preventing accidents
  - Hazard analysis
  - Design for safety
- Assessing risk (determining whether systems are suitable for use)
- Performance modeling and defining safety metrics

Basic Energy Model

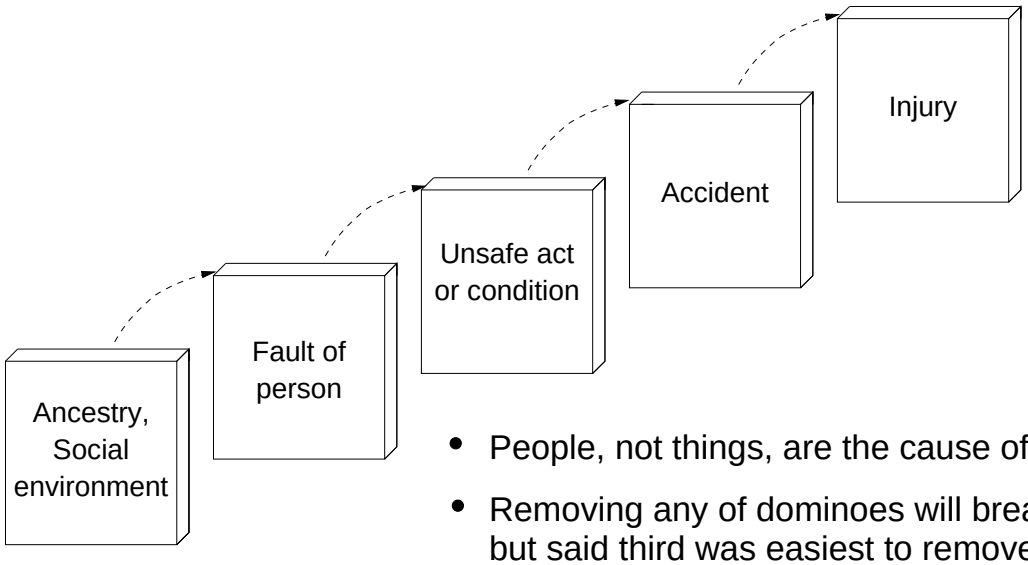
- Assumes accidents are the result of an uncontrolled and undesired release of energy.
- Use barriers or control energy flows to prevent them.



Variations:

- Both (1) application of energy and (2) interference in normal exchange of energy.
- Energy transformation vs. energy deficiency.
- Action systems (systems that produce energy) vs. nonaction systems (systems that constrain energy)

Heinrich's Domino Model of Accidents



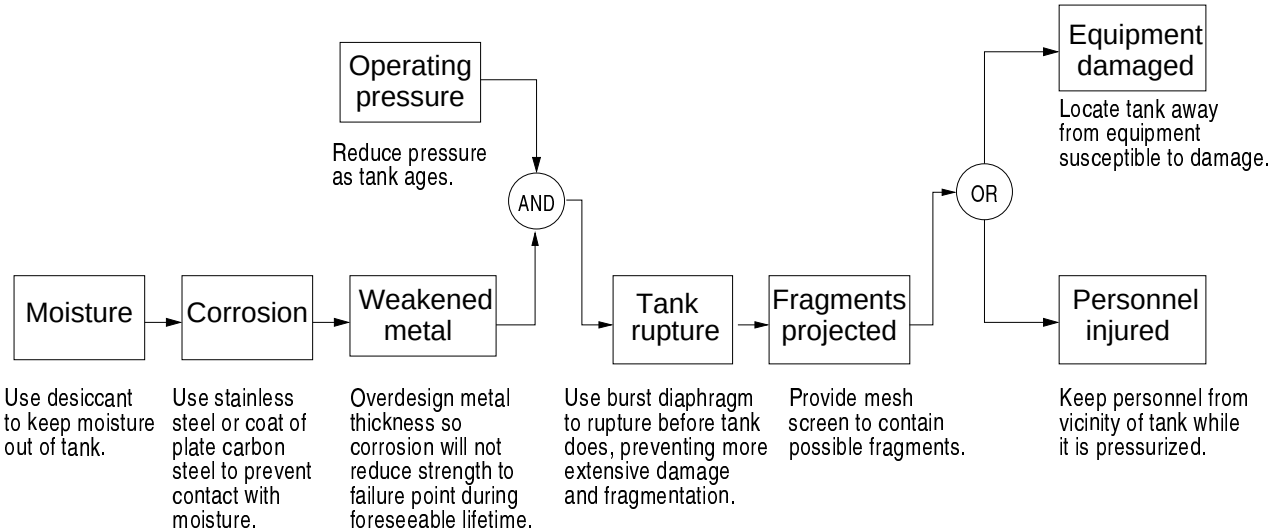
- People, not things, are the cause of accidents.
- Removing any of dominoes will break sequence, but said third was easiest to remove.
- Focus on single causes.

Chain−of−Events Models

- Explain accidents in terms of multiple events, sequenced as a forward chain over time.
- Events almost always involve component failure, human error, or energy−related event
- Form the basis of most safety−engineering and reliability engineering analysis:

  e.g., Fault Tree Analysis, Probabilistic Risk Assessment, FMEA, Event Trees

  and design: e.g., redundancy, overdesign, safety margins, ...



Chain−of−Events Example: Bhopal

E1: Worker washes pipes without inserting slip blind

E2: Water leaks into MIT 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:

- 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.

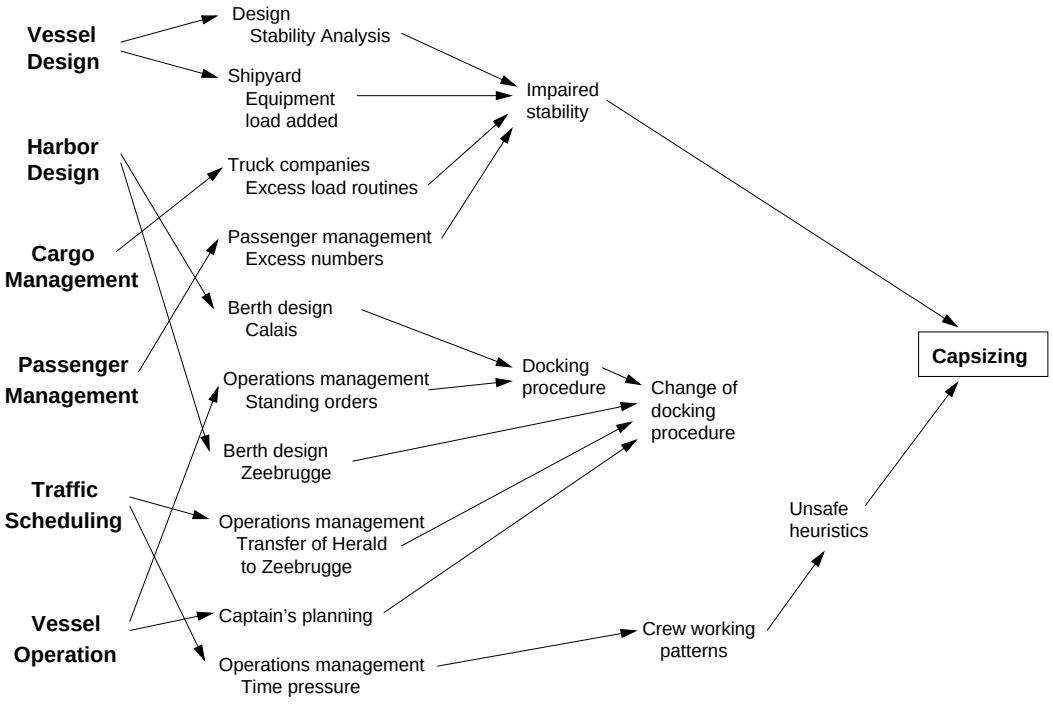
  Ralph Miles Jr.

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

- System accidents
- Software error

Limitations of Event Chain Models (2)

- Human error
  - Deviation from normative procedure vs. established practice
  - Cannot effectively model human behavior by decomposing it into individual decisions and actions and studying it in isolation from the
    - physical and social context
    - value system in which it takes place
    - dynamic work process
- Adaptation
  - Major accidents involve systematic migration of organizational behavior under pressure toward cost effectiveness in an aggressive, competitive environment.



Operational Decision Making:
Decision makers from separate departments in operational context very likely will not see the forest for the trees.

Accident Analysis:
Combinatorial structure of possible accidents can easily be identified.

STAMP

(Systems Theory Accident Modeling and Processes)

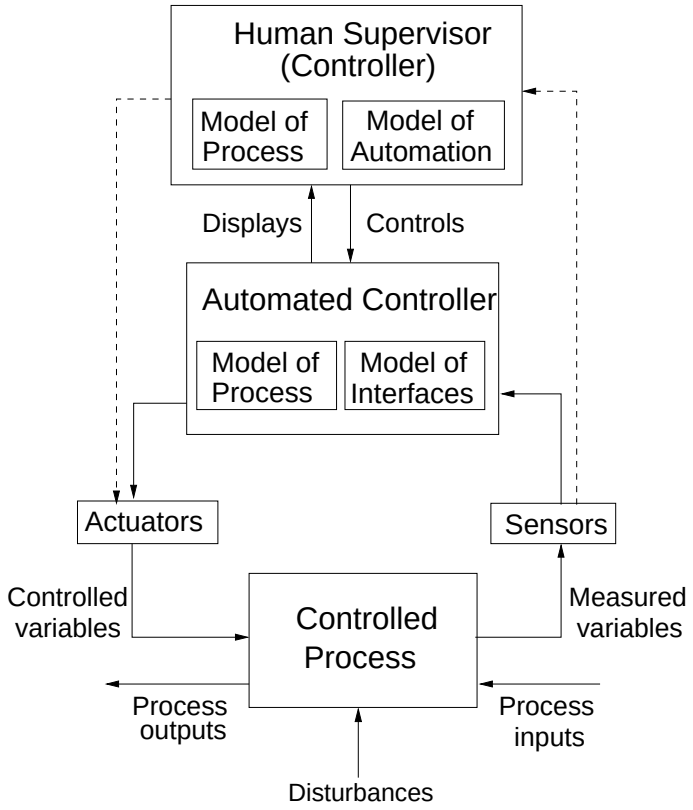
To effect control over a system requires four conditions:

Goal Condition: The controller must have a goal or goals (e.g., to maintain a setpoint)

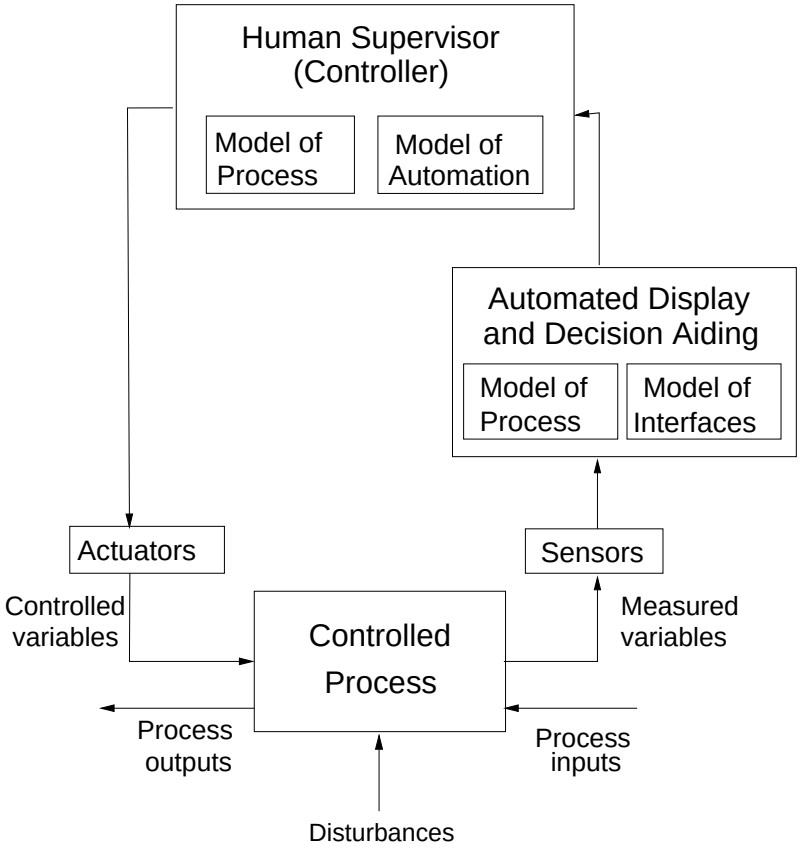
Action Condition: The controller must be able to affect the system state.

Model Condition: The controller must be (or contain) a model of the system

Observability Condition: The controller must be able to ascertain the state of the system.



Process models must contain:
- Required relationship among system vars
- Current state (values of system vars)
- The ways the process can change state



Safety and the Process Models

- Accidents occur when the models do not match the process
  - Wrong from beginning
  - Missing or incorrect feedback so not updated
- Must also account for time lags
- Explains human/machine 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?

  - Don't get feedback to update mental models or disbelieve it

A Systems Theory Model of Accidents

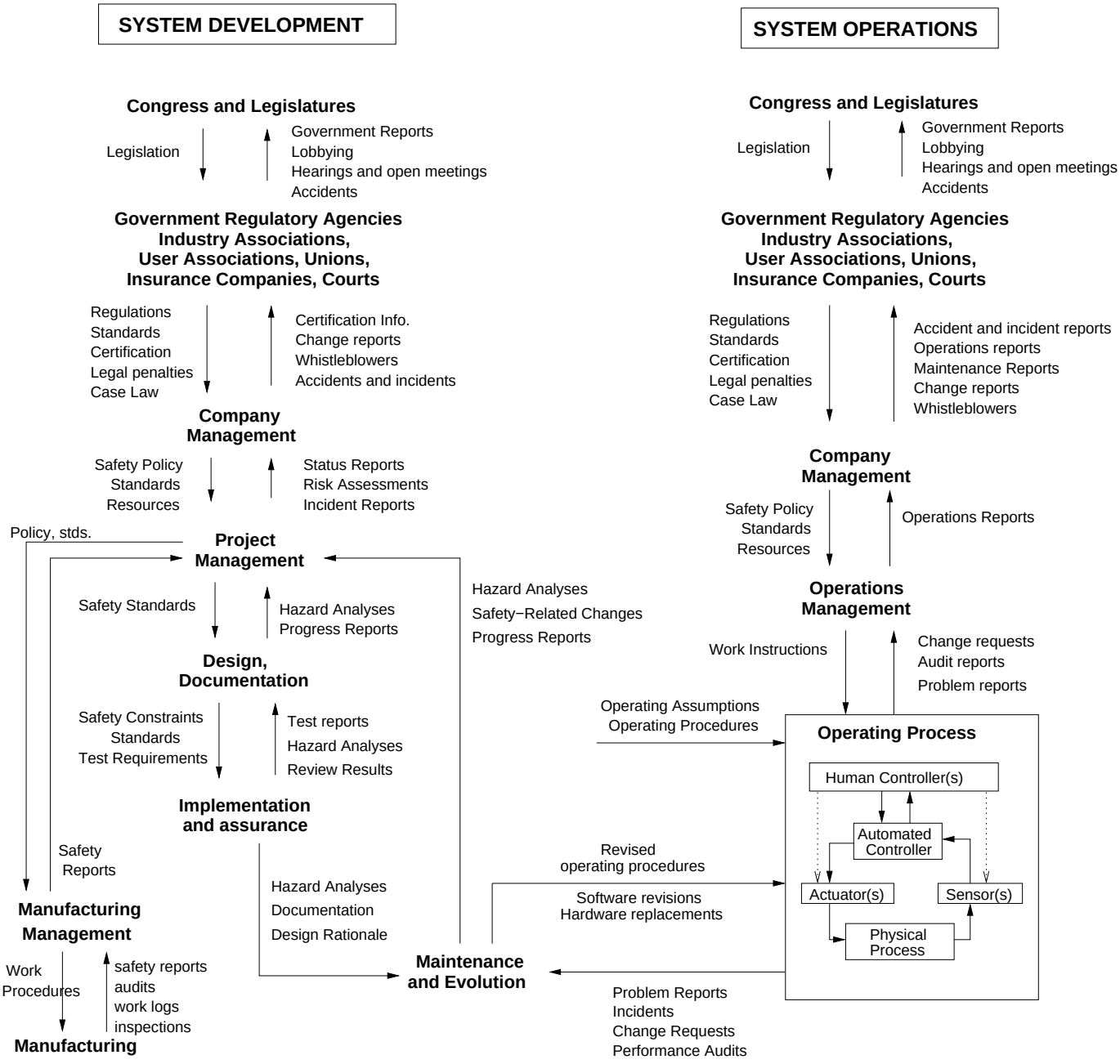
- Accidents arise from interactions among humans, machines, and the environment.
  - Not simply chains of events or linear causality, but more complex types of causal connections.
- Safety is an emergent property that arises when components of system interact with each other within a larger environment.
  - 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.

A Systems Theory Model of Accidents (2)

- Safety can be viewed as a control problem

  e.g. O−rings did not adequately control propellant gas release
  Software did not adequately control descent speed of MPL

- Safety management is a control structure embedded in an adaptive system.
  - Events indirectly reflect the effects of dysfunctional interactions and inadequate control
  - Need to examine control structure itself to understand accidents Result from:
    - Inadequate enforcement of constraints
    - At each level of socio−technical system controlling development and operations


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
Operating Process
Human Controller(s)
Automated Controller
Actuator(s)
Sensor(s)
Physical Process
Revised operating procedures
Software revisions
Hardware replacements
Problem Reports
Incidents
Change Requests
Performance Audits

GOAL: Provide a framework for classifying factors leading to accidents and a system engineering methodology for handling them.

Some causes of dysfunctional interactions:

- Asynchronous evolution
- Inconsistent models
  - inadequate or missing feedback
  - time lags
  - inadequate engineering design activities
  - etc.
- Inadequate coordination among controllers and decision makers
  - Boundary areas
  - Overlap areas

Control Flaws Leading to Hazards

- **Inadequate control actions (enforcement of constraints)**
  - Unidentified hazards
  - Inappropriate, ineffective, or missing control actions for identified hazards
    - Design of control algorithm (process) does not enforce constraints
    - Process models inconsistent, incomplete, or incorrect (lack of linkup)
      - Flaw(s) in creation process
      - Flaws(s) in updating process (asynchronous evolution)
      - 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
- **Inadequate or missing feedback**
  - Not provided in system design
  - Communication flaw
  - Time lag
  - Inadequate sensor operation (incorrect or no information provided)

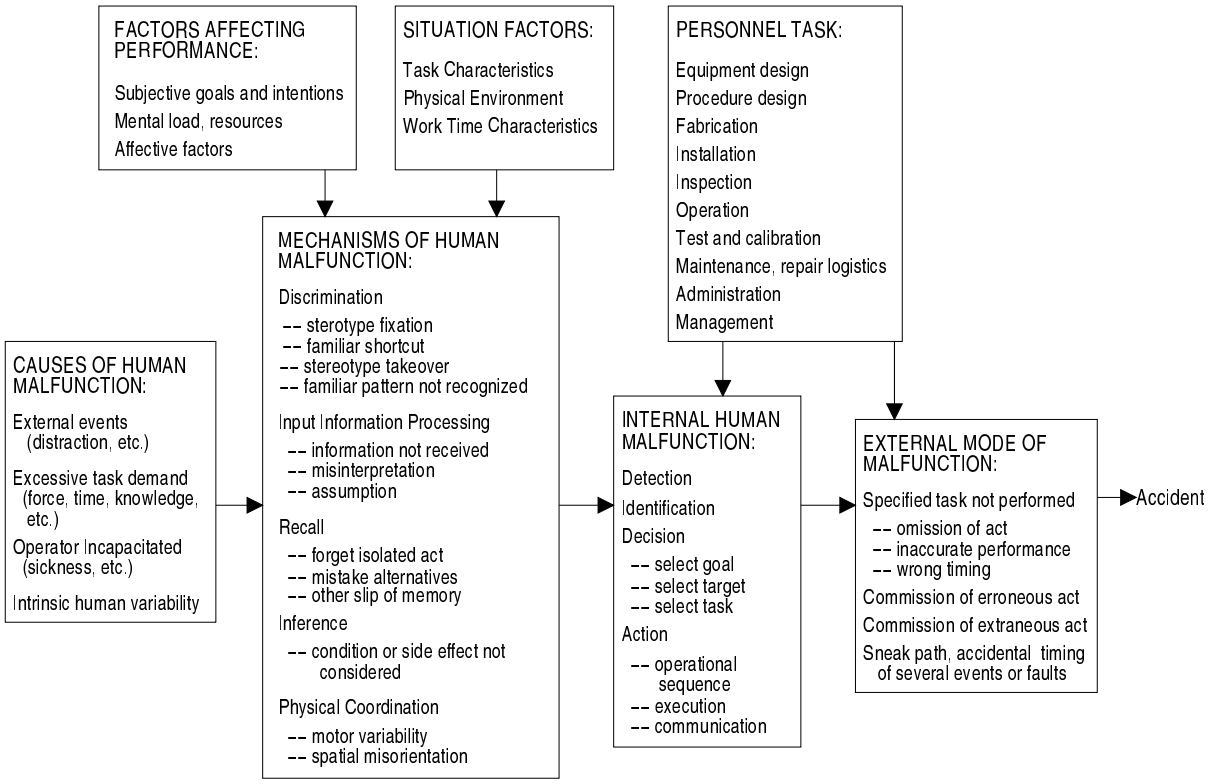
Human Error Models

- Categorize errors by external manifestations
- Categorize by type of task
  - Simple, vigilance, emergency response, control, complex
  - Coordinating, scanning, recognizing, problem solving, planning ...
  - Usually consider performance−shaping factors such as task structure, stress, design of displays and controls
- Categorize by cognitive mechanisms
  - Instead of focusing on task and environment characteristics, consider psychological mechanisms used by operator in performing tasks.
    - Interaction of psychological factors with features of work environment
    - Requires only a limited number of basic concepts

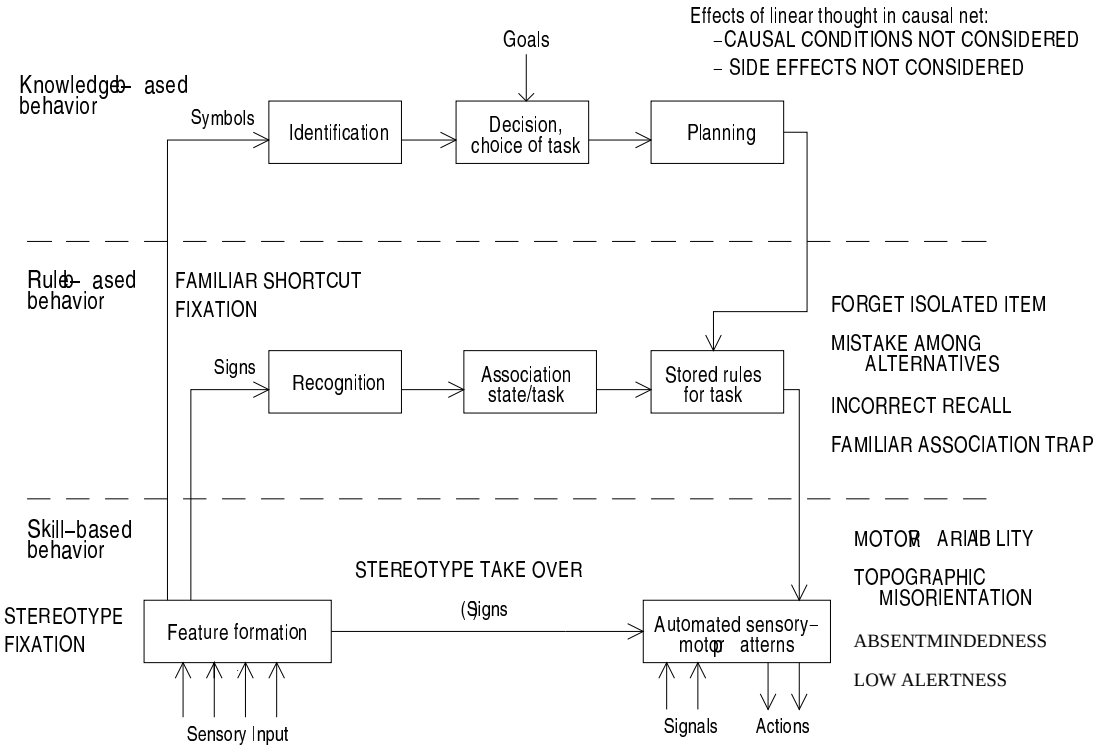
Common Features of Cognitive Models

- Most based on Bartlett's ''schemas''
  - Internal representations of regularities of the world
  - An organized structure of knowledge
  - Our way of understanding and dealing with world
- Slips vs. Mistakes (Don Norman)
  - Mistake is an error in intention (error in planning)
  - Slip is error in carrying out the intention
- Human−Task Mismatch (Rasmussen)
  - Errors are an integral part of learning
  - Should be considered human−task or human−system mismatches
- Skill−Rules−Knowledge framework (Rasmussen)
- Human skills needed to solve problems also lead to errors
  - If eliminate possibility of human error, may eliminate ability to solve problems.

Rasmussen Model of Human−Task Mismatch



Skill−Rules−Knowledge Hierarchy



Social Psychology Models

Engineering models: look at human behavior in terms of tasks

Psychology models: relate human cognition to performance

Social Psychology models: include individual value systems and sense of personal responsibility

Safety Information System

- Studies have ranked this second in importance only to top management concern for safety.

- Contents
  - Updated System Safety Program Plan
  - Status of activities
  - Results of hazard analyses
  - Tracking and status information on all known hazards.
  - Incident and accident information including corrective action.
  - Trend analysis data.

- Information collection
- Information analysis
- Information dissemination

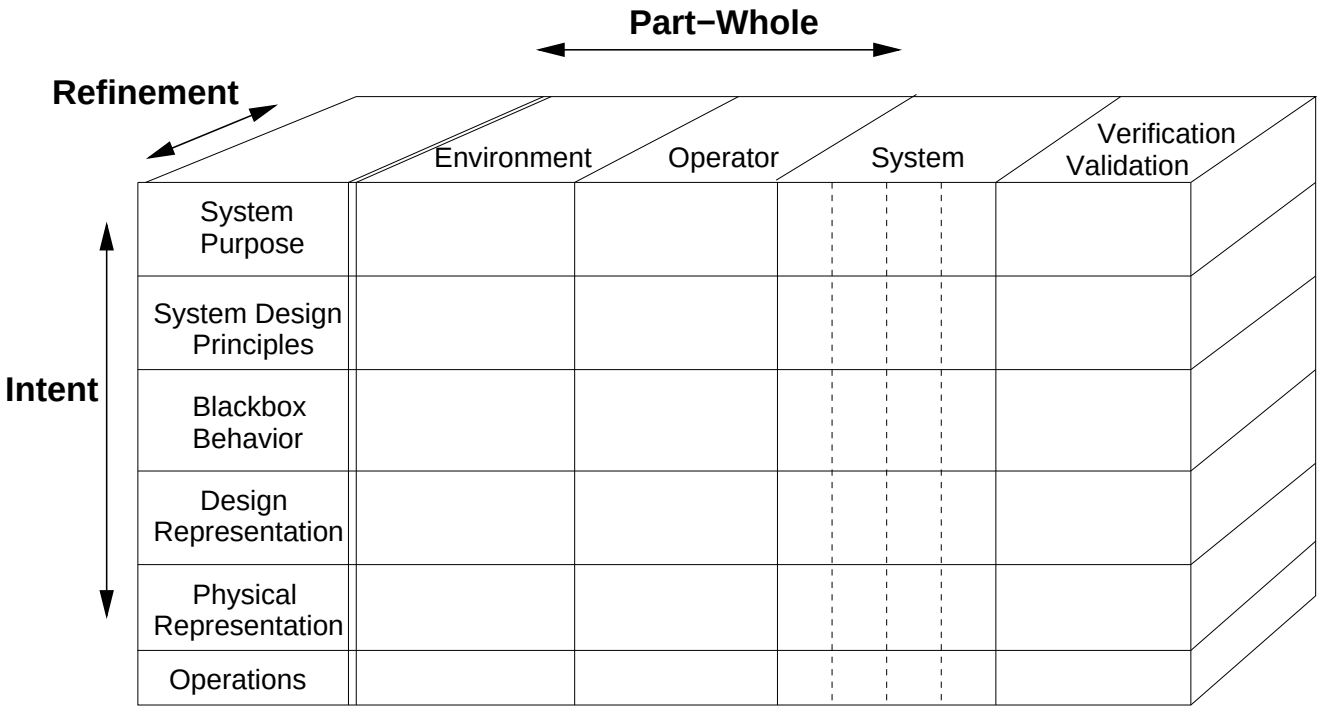
Intent Specifications

- Bridge between disciplines
- Support for human problem solving
- Traceability
- Support for upstream safety efforts
- Integration of safety information into decision−making environment
- Assistance in software evolution

Intent Specifications (2)

- Hierarchical abstraction based on ‘‘why’’ (design rationale) as well as what and how.
  - Design decisions at each stage mapped back to to requirements and constraints they are derived to satisfy
  - Earlier decisions mapped to later stages of process
  - Results in record of progression of design rationale from high−level requirements to component requirements and designs.
  - Provides traceability of intent information

Intent Specifications



- Each level supports a different type of reasoning about system.
- Mappings between levels provide relational info necessary to reason across hierarchical levels.

	Environment	Operator	System and components	V&V
Level 0	Project management plans, status information, safety plan, etc.			
Level 1 System Purpose	Assumptions Constraints	Responsibilities Requirements I/F requirements	System goals, high–level requirements, design constraints, limitations	Preliminary Hazard Analysis Reviews
Level 2 System Principles	External interfaces	Task analyses Task allocation Controls, displays	Logic principles, control laws, functional decomposition and allocation	Validation plan and results, System Hazard Analysis
Level 3 Blackbox Models	Environment models	Operator Task models HCI models	Blackbox functional models Interface specifications	Analysis plans and results, Subsystem Hazard Analysis
Level 4 Design Rep.		HCI design	Software and hardware design specs	Test plans and results
Level 5 Physical Rep.		GUI design, physical controls design	Software code, hardware assembly instructions	Test plans and results
Level 6 Operations	Audit procedures	Operator manuals Maintenance Training materials	Error reports, change requests, etc.	Performance monitoring and audits

Level 1: System Purpose

- Introduction
- Historical Perspective
- Environment Description
- Environment Assumptions
  - Altitude information is available from intruders with a minimum precision of 100 feet.
  - All aircraft have legal identification numbers.
- Environment Constraints
  - The behavior or interaction of non−TCAS equipment with TCAS must not degrade the performance of the TCAS equipment.
- System Functional Goals
  - Provide affordable and compatible collision avoidance system options for a broad spectrum of National Airspace System users.

Level 1: System Purpose (2)

- High−Level Requirements

  [1.2] TCAS shall provide collision avoidance protection for any two aircraft closing horizontally at any rate up to 1200 knots and vertically up to 10,000 feet per minute.

  Assumption: Commercial aircraft can operate up to 600 knots and 5000 fpm during vertical climb or controlled descent (and therefore the planes can close horizontally up to 1200 knots and vertically up to 10,000 fpm.

- Design and Safety Constraints

  [SC5] The system must not disrupt the pilot and ATC operations during critical phases of flight nor disrupt aircraft operation.

  [SC5.1] The pilot of a TCAS−equipped aircraft must have the option to switch to the Traffic−Advisory−Only mode where TAs are displayed but display of resolution advisories is prohibited.

  Assumption: This feature will be used during final approach to parallel runways when two aircraft are projected to come close to each other and TCAS would call for an evasive maneuver.

Example Level 1 Safety Constraints for TCAS

SC−7 TCAS must not create near misses (result in a hazardous level of vertical separation) that would not have occurred had the aircraft not carried TCAS.

SC−7.1 Crossing maneuvers must be avoided if possible.

↓ 2.36, 2.38, 2.48, 2.49.2

SC−7.2 The reversal of a displayed advisory must be extremely rare.

↓ 2.51, 2.56.3, 2.65.3, 2.66

SC−7.3 TCAS must not reverse an advisory if the pilot will have insufficient time to respond to the RA before the closest point of approach (four seconds or less) or if own and intruder aircraft are separated by less than 200 feet vertically when 10 seconds or less remain to closest point of approach.

↓ 2.52

Level 1: System Purpose (3)

- System Limitations

  L.5 TCAS provides no protection against aircraft with nonoperational or non−Mode C transponders.

- Operator Requirements

  OP. 4 After the threat is resolved the pilot shall return promptly and smoothly to his/her previously assigned flight path.

- Human−Interface Requirements

- Hazard and other System Analyses

Hazard List for TCAS

H1: Near midair collision (NMAC): An encounter for which, at the closest point of approach, the vertical separation is less than 100 feet and the horizontal separation is less than 500 feet.

H2: TCAS causes controlled maneuver into ground
e.g. descend command near terrain

H3: TCAS causes pilot to lose control of the aircraft.

H4: TCAS interferes with other safety−related systems
e.g. interferes with ground proximity warning

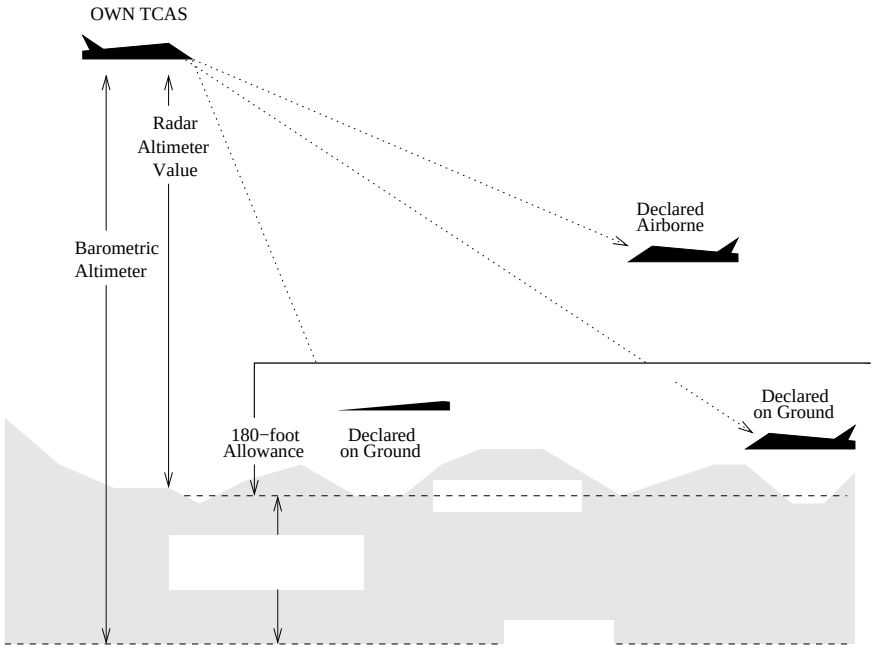
TCAS does not display a resolution advisory.

- TCAS unit is not providing RAs.
  - <Self−monitor shuts down TCAS unit>
  - Sensitivity level set such that no RAs are displayed.
  - ...
- No RA inputs are provided to the display.
  - No RA is generated by the logic
    - Inputs do not satisfy RA criteria
      - Surveillance puts threat outside corrective RA position.
        - Surveillance does not pass adequate track to the logic
          - <Threat is non−Mode C aircraft> → L.5
          - <Surveillance failure> → 1.23.1
        - to be calculated>
      - Altitude reports put threat outside corrective RA position
        - Altitude errors put threat on ground
          - <Uneven terrain> ↓ 2.19
          - <Intruder altitude error>
          - <Own Mode C altitude error> → 1.23.1
          - <Own radar altimeter error> → 1.23.1
        - Altitude errors put threat in non−threat position.
        - ...
      - <Intruder maneuver causes logic to delay RA beyond CPA> → SC4.2 ↓ 2.35
    - ...
  - <Process/display connectors fail>
- <Display is preempted by other functions> → SC4.8 ↓ 2.22
- <Display hardware fails> → 1.23.1

TCAS displays a resolution advisory that the pilot does not follow.

- Pilot does not execute RA at all.
  - Crew does not perceive RA alarm.
    - <Inadequate alarm design> → 1.4 to 1.14 ↓ 2.74, 2.76
    - <Crew is preoccupied>
  - <Crew does not believe RA is correct.> → OP.1
  - ...
- Pilot executes the RA but inadequately
  - <Pilot stops before RA is removed> → OP.10
  - <Pilot continues beyond point RA is removed> → OP.4
  - <Pilot delays execution beyond time allowed> → OP.10

2.19 When below 1700 feet AGL, the CAS logic uses the difference between its own aircraft pressure altitude and radar altitude to determine the approximate elevation of the ground above sea level (see Figure 2.5). It then subtracts the latter value from the pressure altitude value received from the target to determine the approximate altitude of the target above the ground (barometric altitude − radar altitude + 180 feet). If this altitude is less than 180 feet, TCAS considers the target to be on the ground (↟1.SC4.9). Traffic and resolution advisories are inhibited for any intruder whose tracked altitude is below this estimate. Hysteresis is provided to reduce vacillations in the display of traffic advisories that might result from hilly terrain (↟ FTA−320). All RAs are inhibited when own TCAS is within 500 feet of the ground.



Example Level−2 System Design for TCAS

SENSE REVERSALS ↓Reversal−Provides−More−Separation$_{m-301}$

2.51 In most encounter situations, the resolution advisory sense will be maintained for the duration of an encounter with a threat aircraft.

↑ SC−7.2

However, under certain circumstances, it may be necessary for that sense to be reversed. For example, a conflict between two TCAS−equipped aircraft will, with very high probability, result in selection of complementary advisory senses because of the coordination protocol between the two aircraft. However, if coordination communications between the two aircraft are disrupted at a critical time of sense selection, both aircraft may choose their advisories independently.

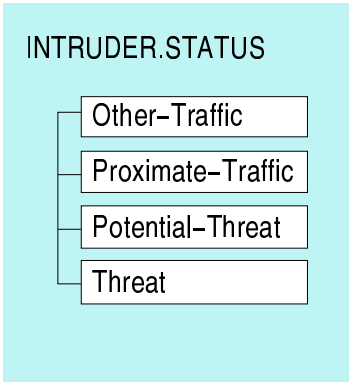
↑ FTA−1300

This could possibly result in selection of incompatible senses.

↑ FTA−395

2.51.1 [Information about how incompatibilities are handled]

Level 3 Modeling Language Example



Threat → Other−Traffic

		OR			
AND	Alt−Reporting **in−state** Lost	T	T	T	.
	Bearing−Valid$_{m-478}$	F	.	T	.
	Range−Valid$_{v-398}$	.	F	T	.
	Proximate−Traffic−Condition$_{m-498}$	.	.	F	.
	Potential−Threat−Condition$_{m-494}$	.	.	F	.
	Other−Aircraft **in−state** On−Ground	.	.	.	T

Description: A threat is reclassified as other traffic if its altitude reporting has been lost (↑2.13) and either the bearing or range inputs are invalid; if its altitude reporting has been lost and both the range and bearing are valid but neither the proximate no r potential threat classification criteria are satisfied; or the aircraft is on the ground (↑2.12) .

Mapping to Level 2: ↑2.23, ↑2.29

Mapping to Level 4: ↓4.7.1, Traffic−Advisory