EUROPEAN STAMP WORKSHOP AND CONFERENCE (ESWC) 2020

WMG, THE UNIVERSITY OF WARWICK

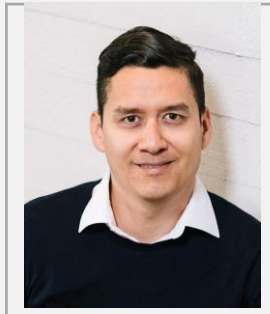
21-22 OCTOBER 2020

ESWC

European STAMP Workshop and Conference

CATAPULT High Value Manufacturing

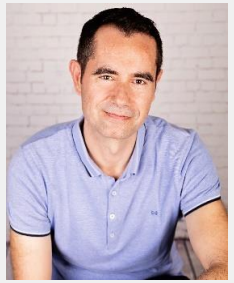
WMG THE UNIVERSITY OF WARWICK



Osiris Valdez Banda
Assistant Professor of Marine Technology at Aalto University

Martin Rejzek
Deputy head Safety-Critical Systems Research Lab, Zurich University of Applied Sciences

Svana Helen Björnsdóttir
Visiting Researcher and a Doctoral Candidate at Reykjavik University



Nektarios Karanikas
Associate Professor in Health, Safety and Environment, Queensland University of Technology

Robert Jan de Boer
Campus Director, Northumbria University
Senior consultant Intergo

Stefan Wagner
Professor and director of the Institute of Software Engineering, University of Stuttgart

WMG, THE UNIVERSITY OF WARWICK

21-22 OCTOBER 2020

INTRODUCTION TO SYSTEMS-THEORETIC ACCIDENT MODEL AND PROCESSES

Dr John Thomas,
Executive Director,
Engineering Systems Laboratory,
Safety and Cybersecurity Group,
Massachusetts Institute of Technology, USA

Massachusetts Institute of Technology

A Systems Approach to Safety using STAMP/STPA

Models and Comparisons

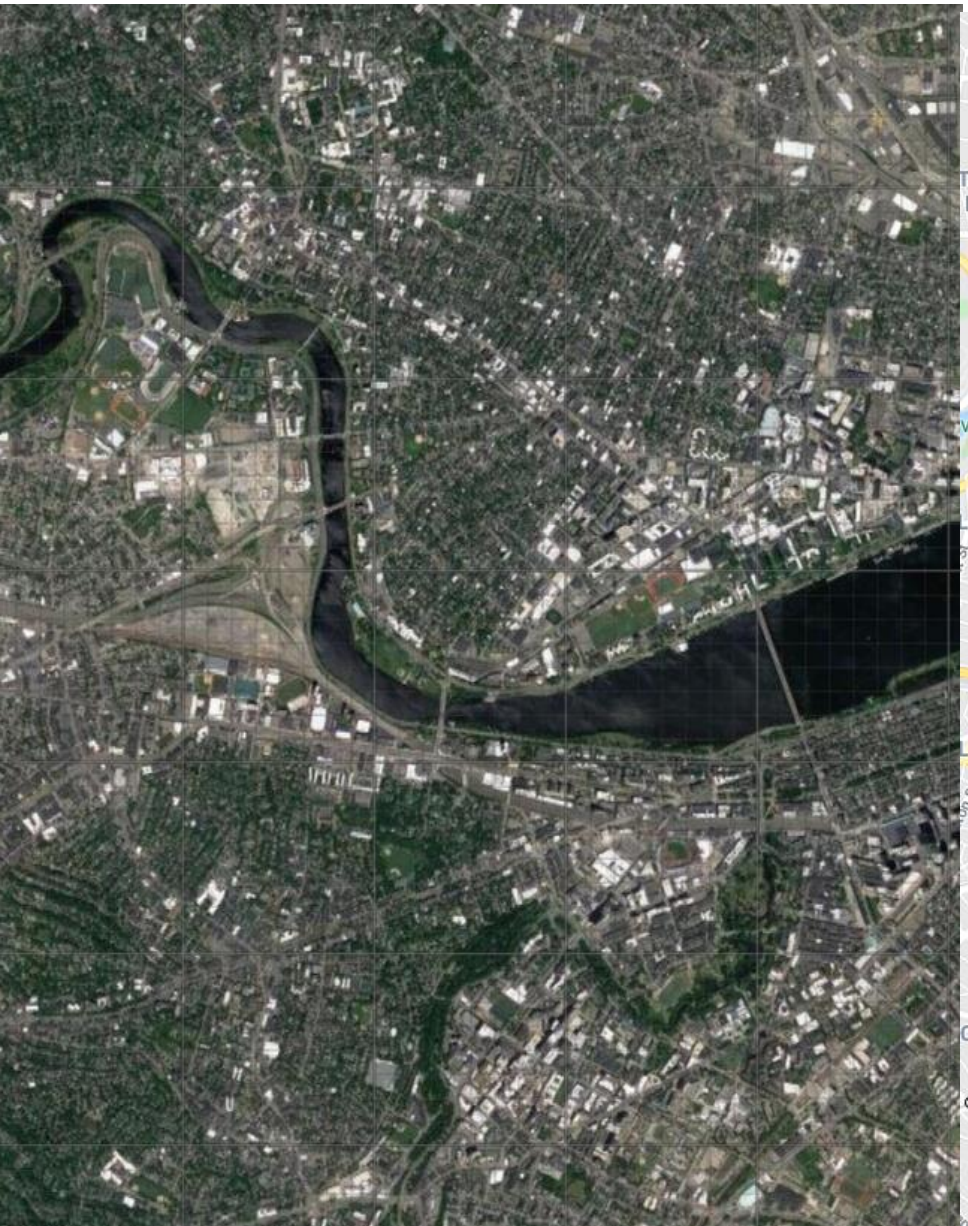
Dr. John Thomas

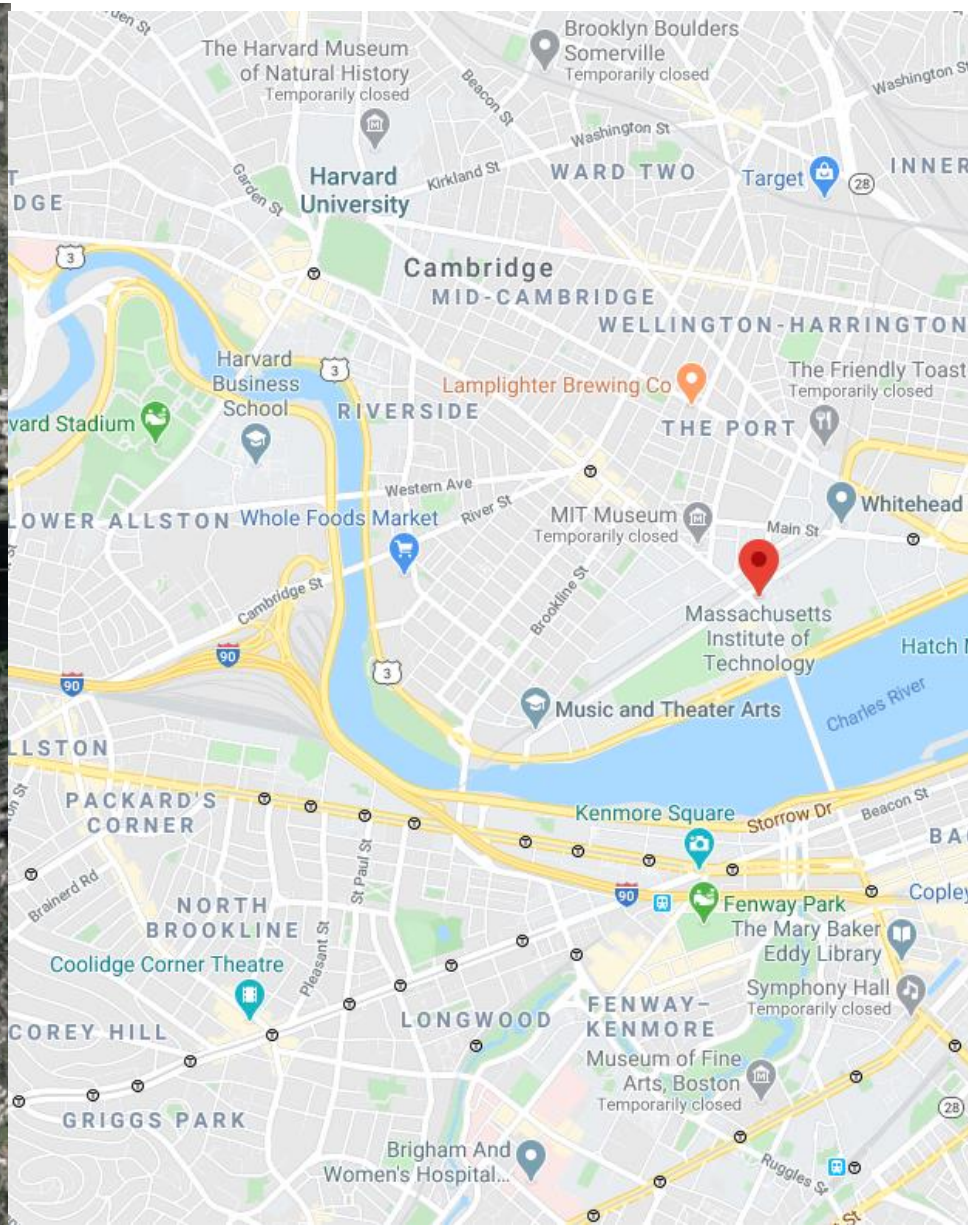
Questions or comments? Contact me! JTHOMAS4@MIT.EDU

Reality

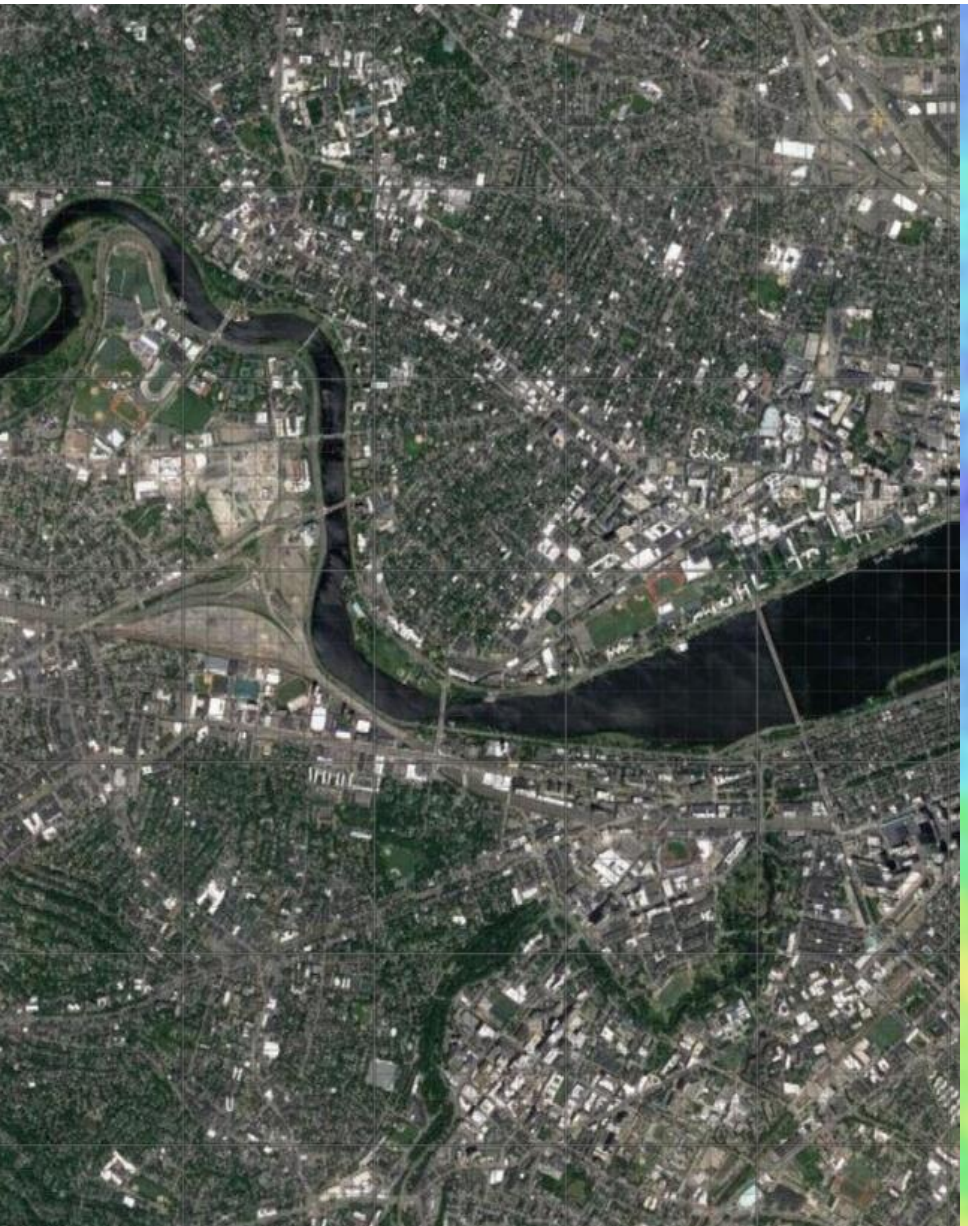
Abstracting

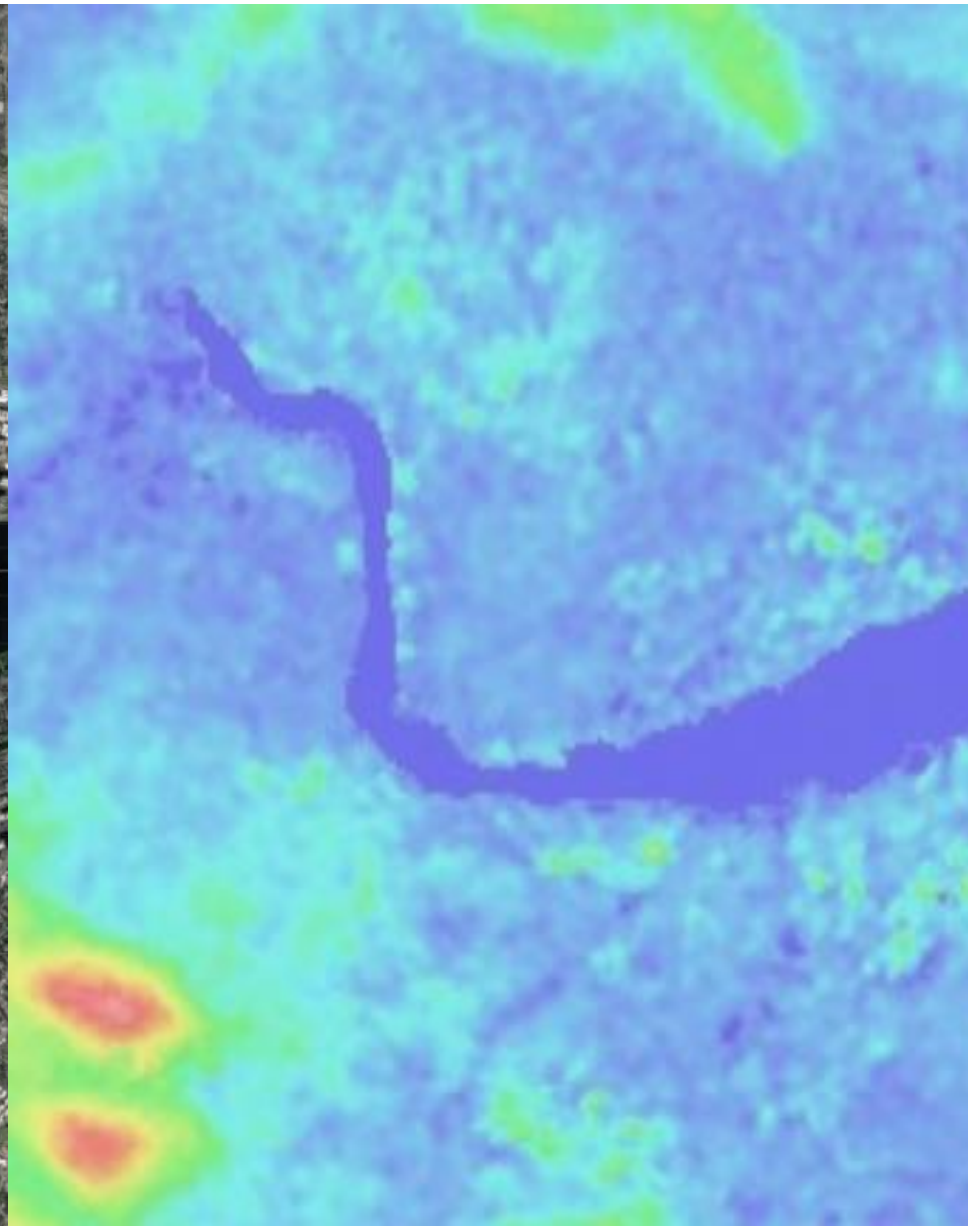
Model





Reality → Abstracting → Model

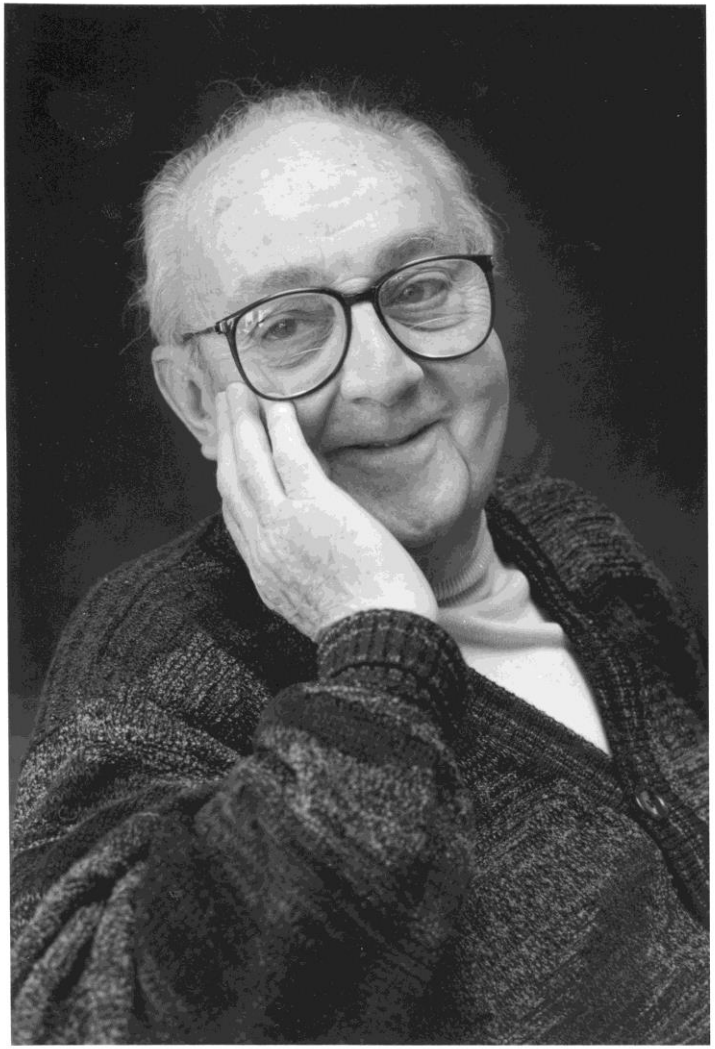




Famous George Box quote

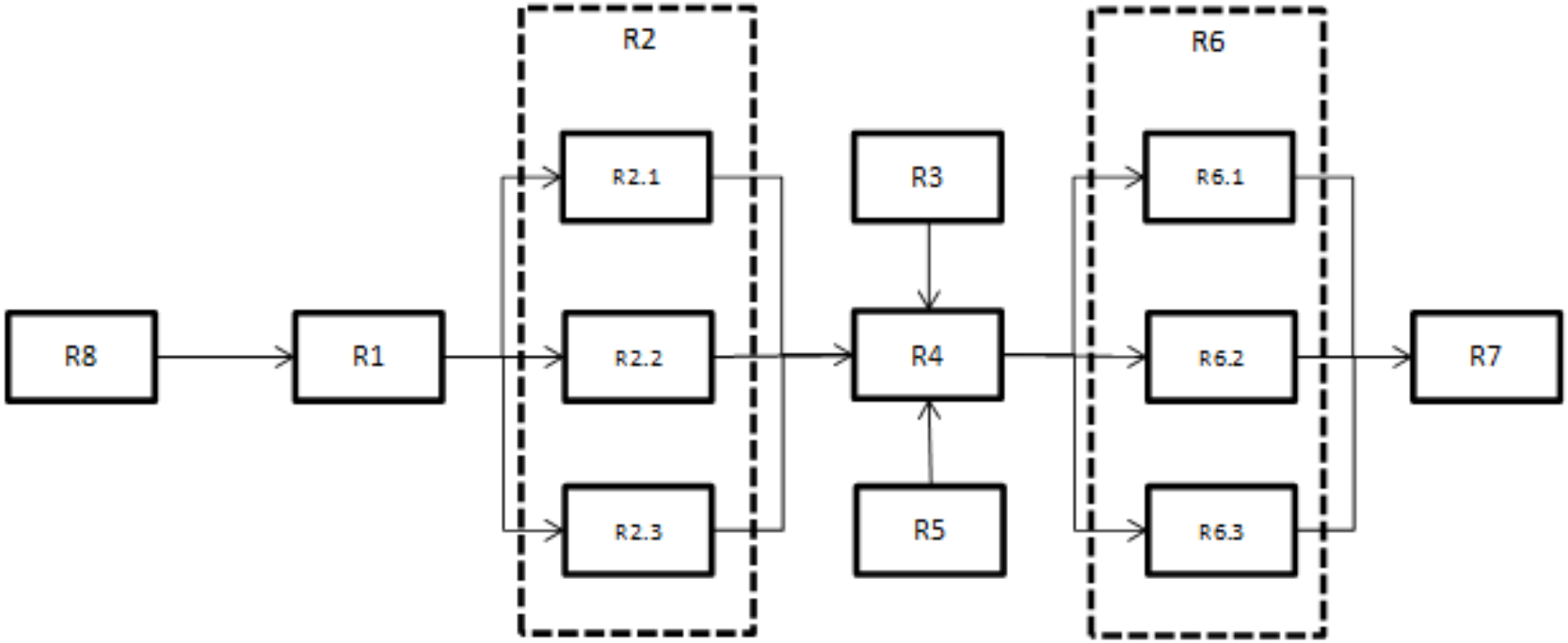
"Essentially, **all models are wrong but some are useful**."

- George Box



System Models

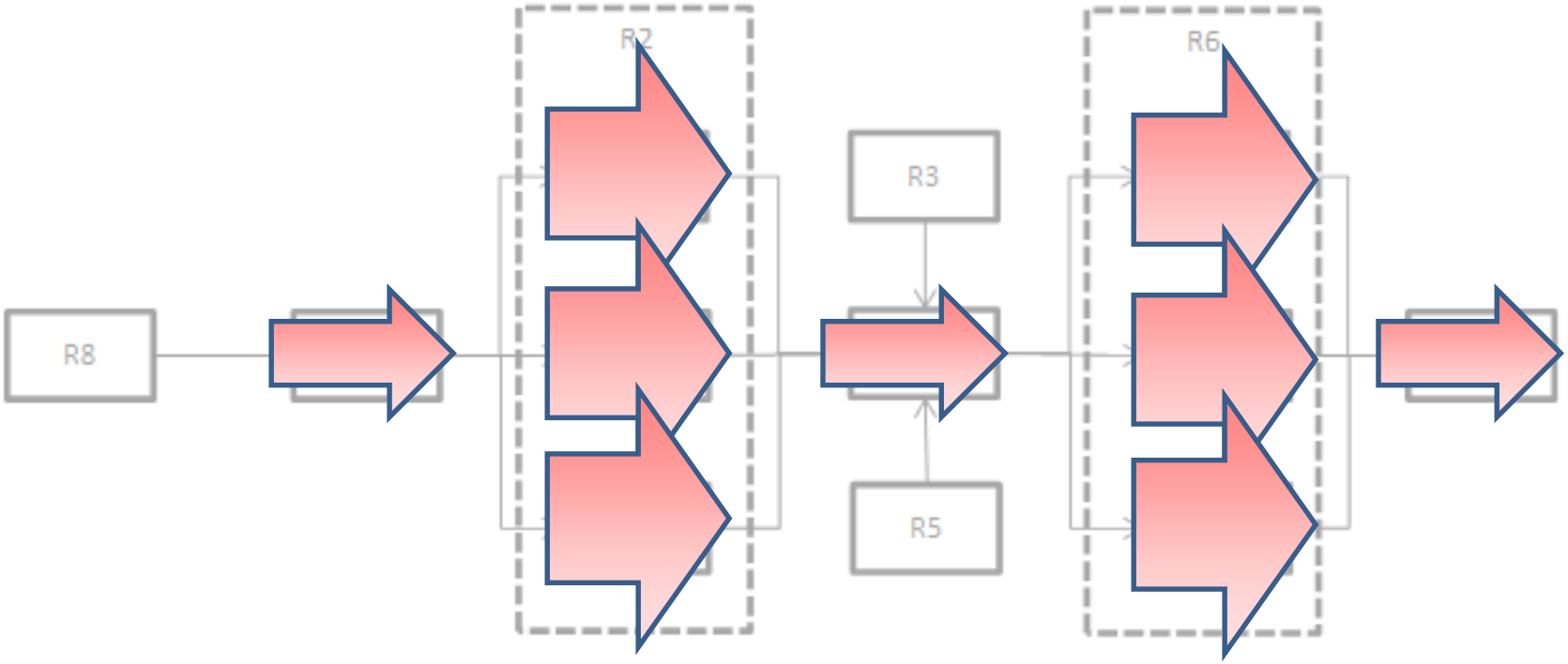
Example: Reliability Block Diagram (RBD)



Emphasizes redundancy
Often used in Fault Tree Analysis and Reliability Assessment

System Models

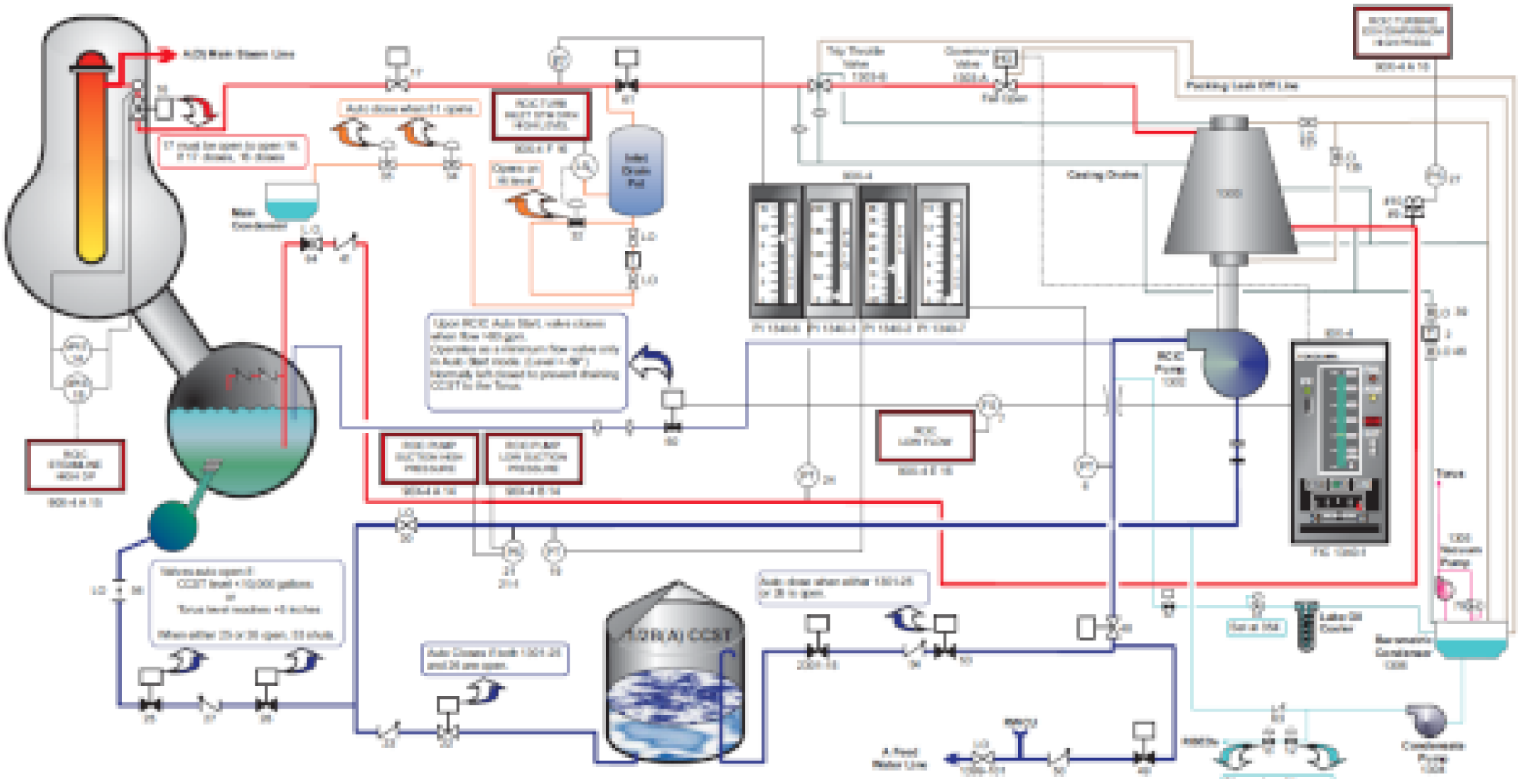
Example: Reliability Block Diagram (RBD)



Emphasizes redundancy
Often used in Fault Tree Analysis and Reliability Assessment

System Models

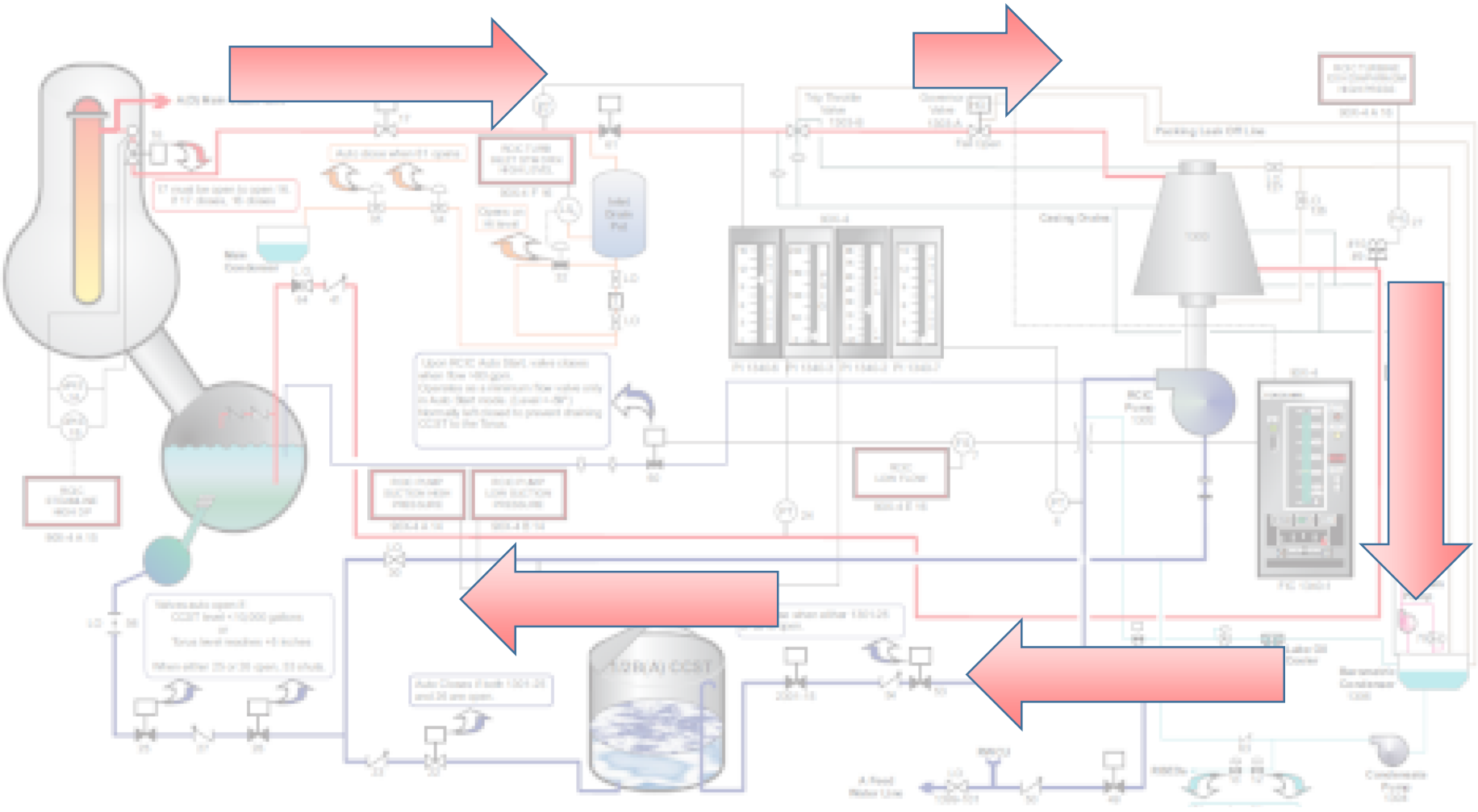
Example: Piping and Instrumentation Diagram (P&ID)



Emphasizes physical flows
Often used in HAZOP

System Models

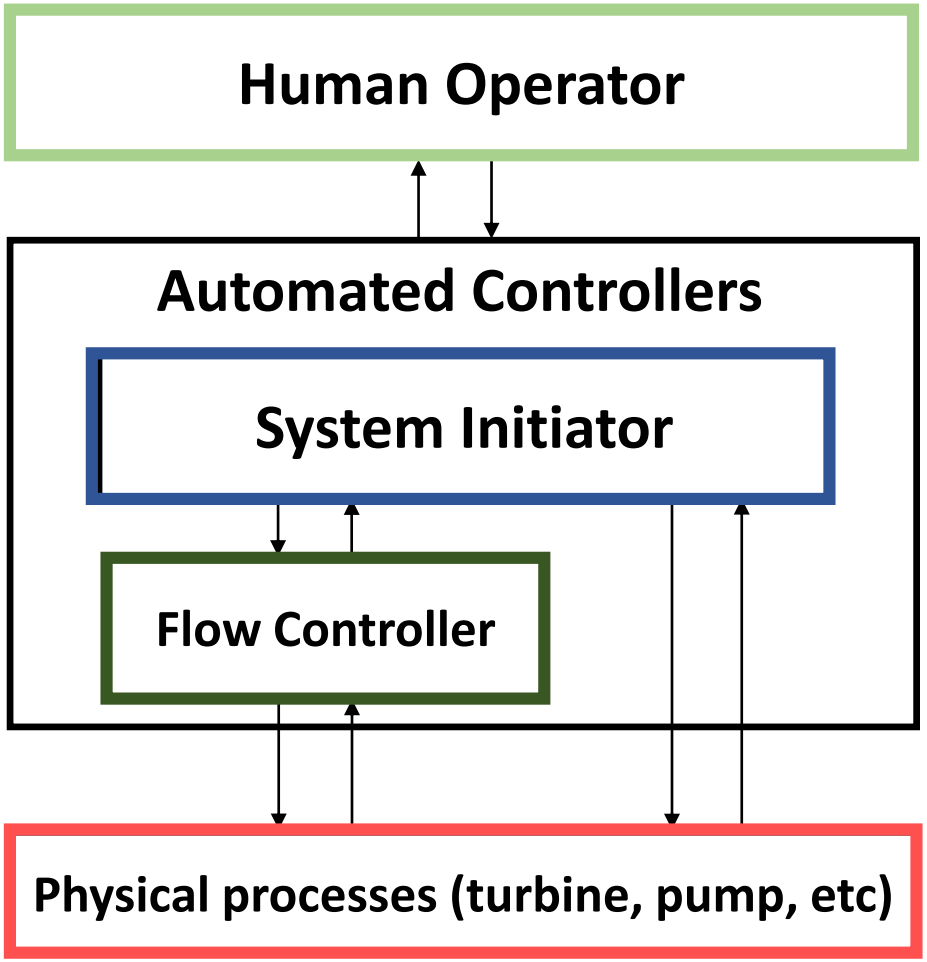
Example: Piping and Instrumentation Diagram (P&ID)



Emphasizes physical flows
Often used in HAZOP

System Models

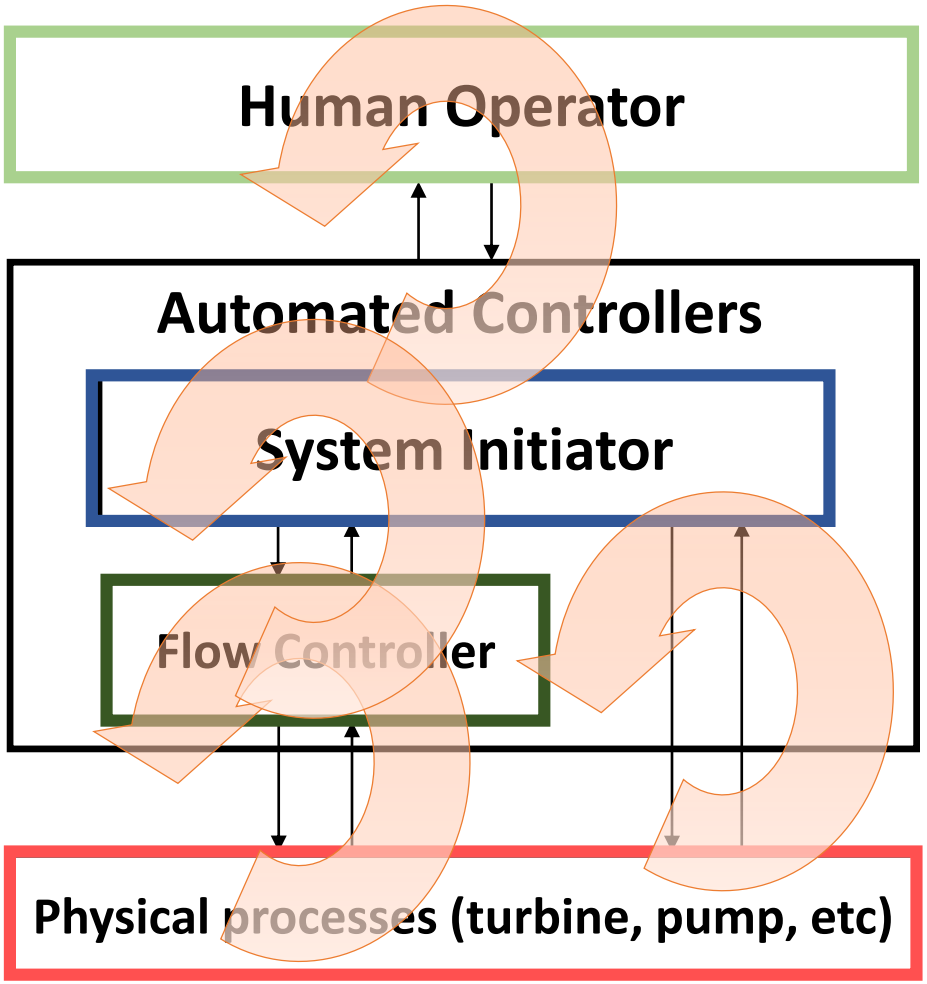
Example: Control Structure



Emphasizes control relationships
Used in STAMP / STPA

System Models

Example: Control Structure

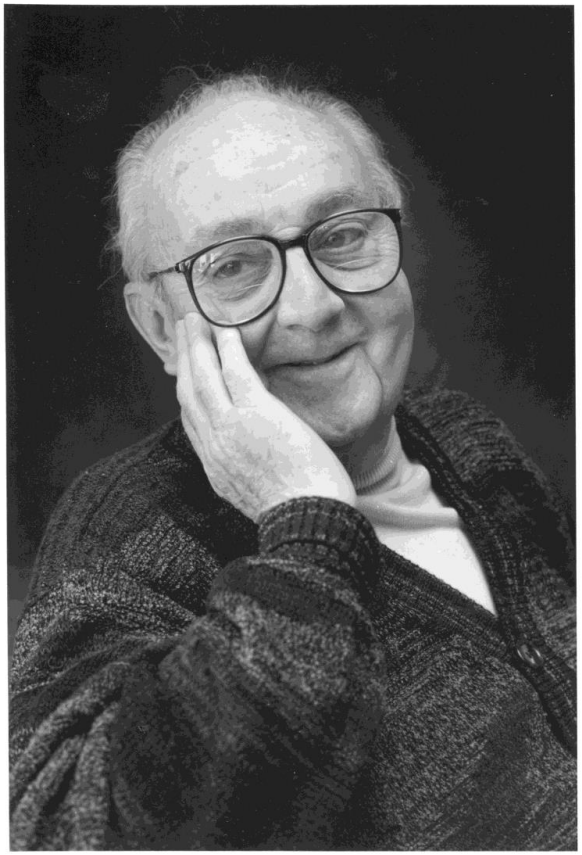


Emphasizes control relationships
Used in STAMP / STPA

System Models

- All system models emphasize certain information, abstract away other info. All are “wrong”.

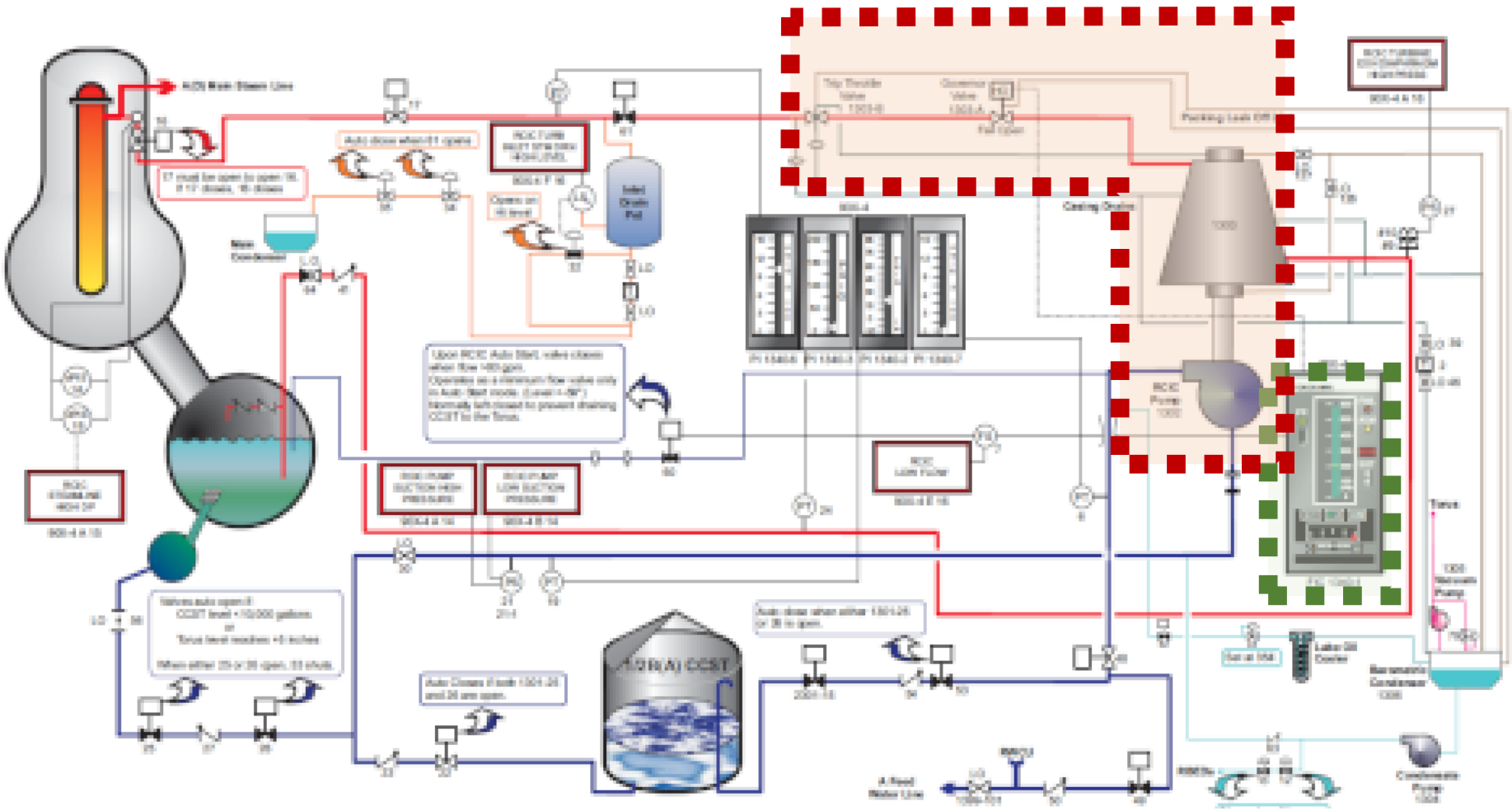
- But are they useful?



STPA analyzes a control structure

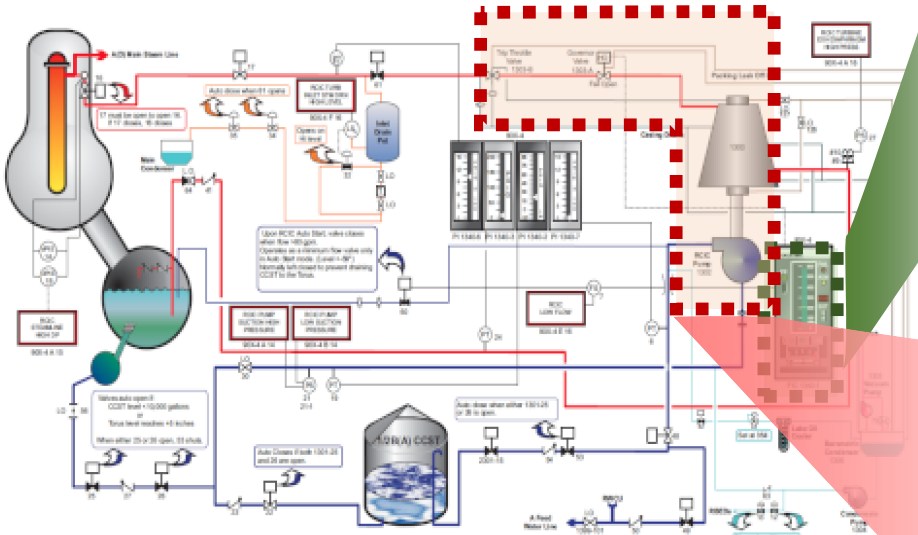
What is a control structure?

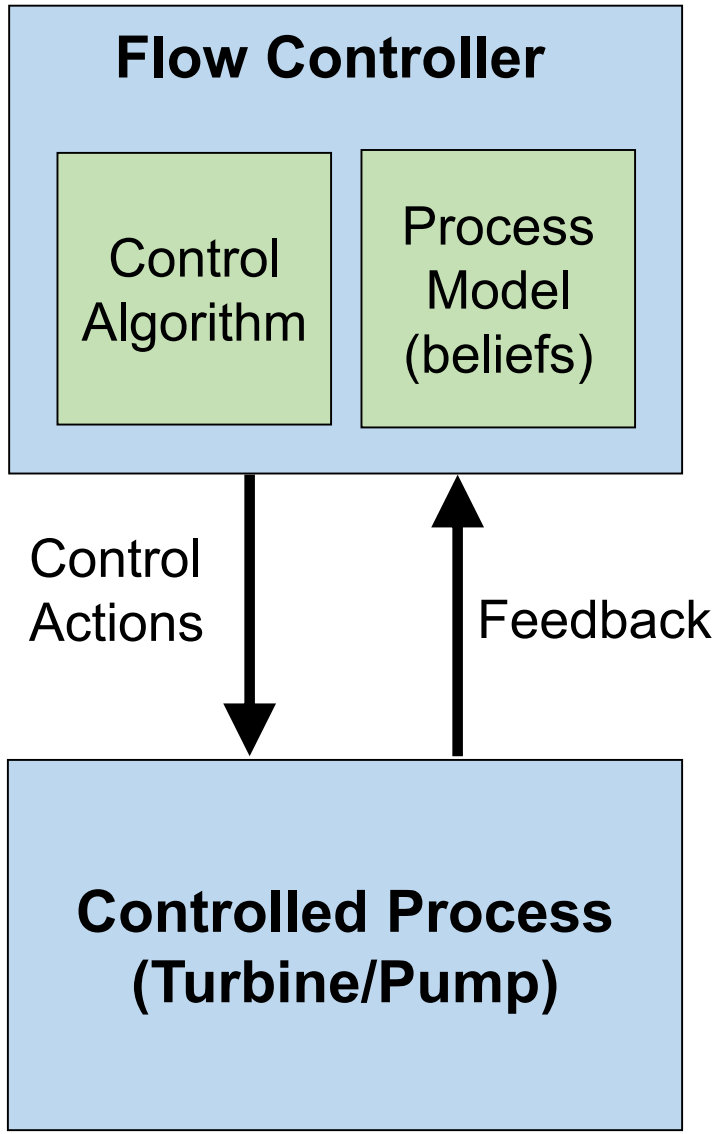
Enabling Abstraction



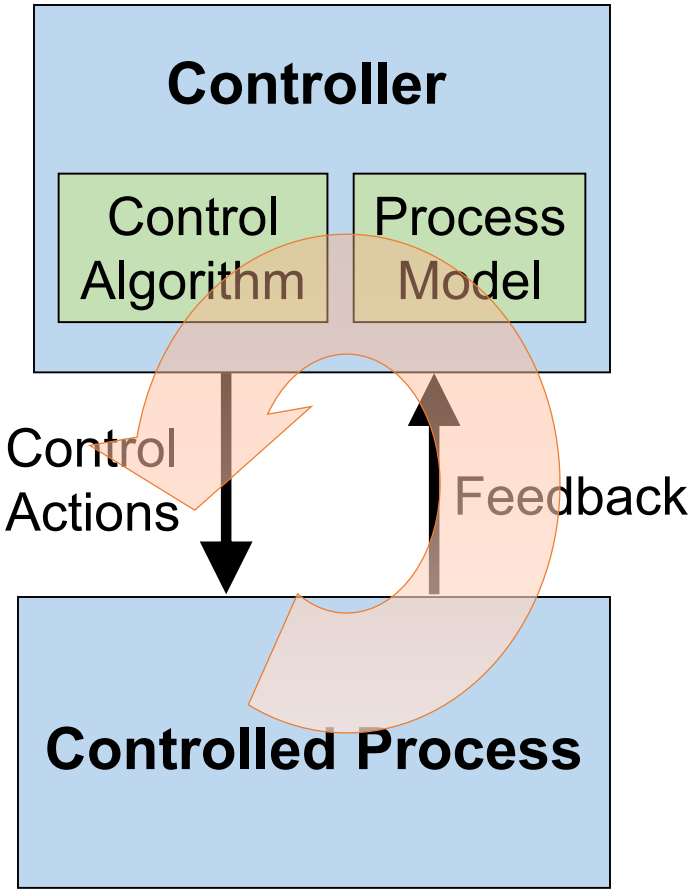
Thomas, 2020

Enabling Abstraction Control Structure



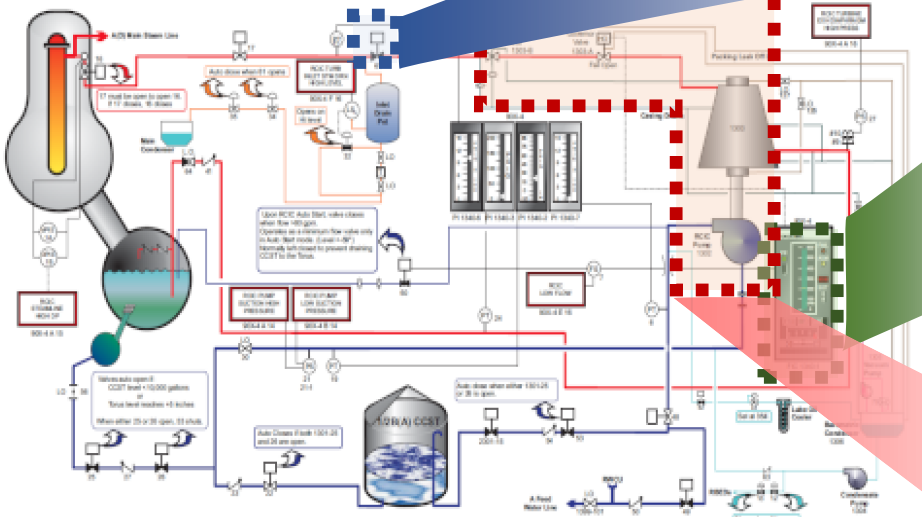


Basic control loop



- **Control actions** are provided to affect a controlled process
- **Feedback** may be used to monitor the process
- **Process model** (beliefs) formed based on feedback and other information
- **Control algorithm** determines appropriate control actions given current beliefs

Enabling Abstraction



Automated Controllers

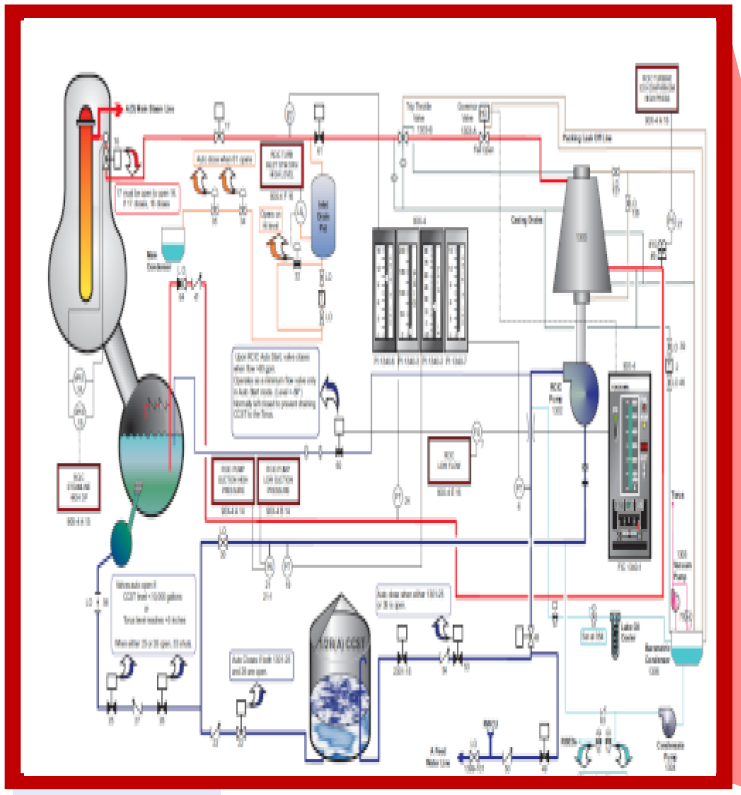
System Initiator

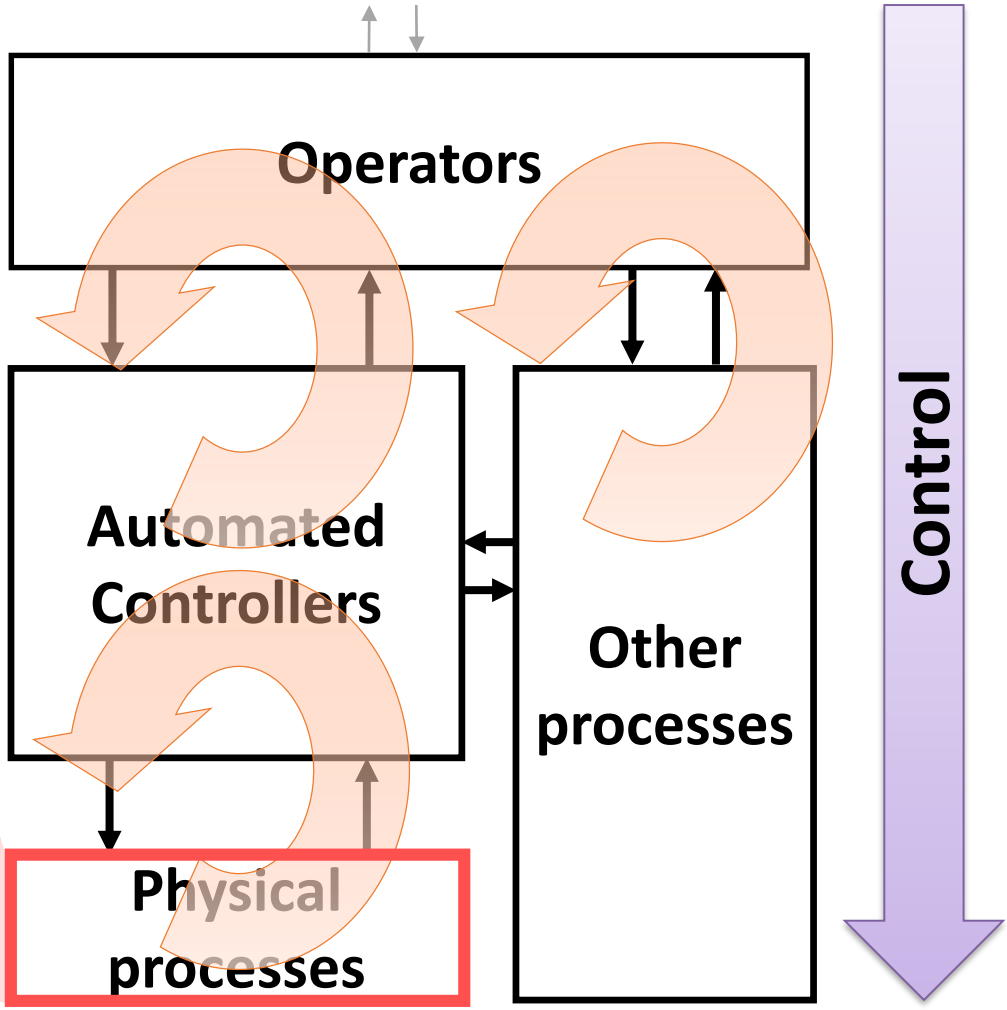
Flow Controller

Physical processes (turbine, pump, etc)

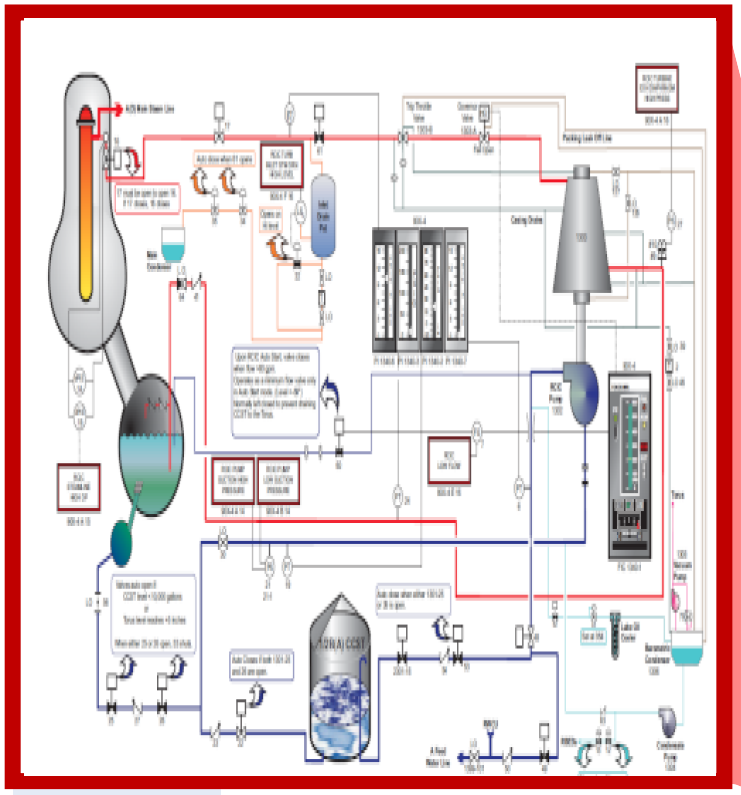
Control

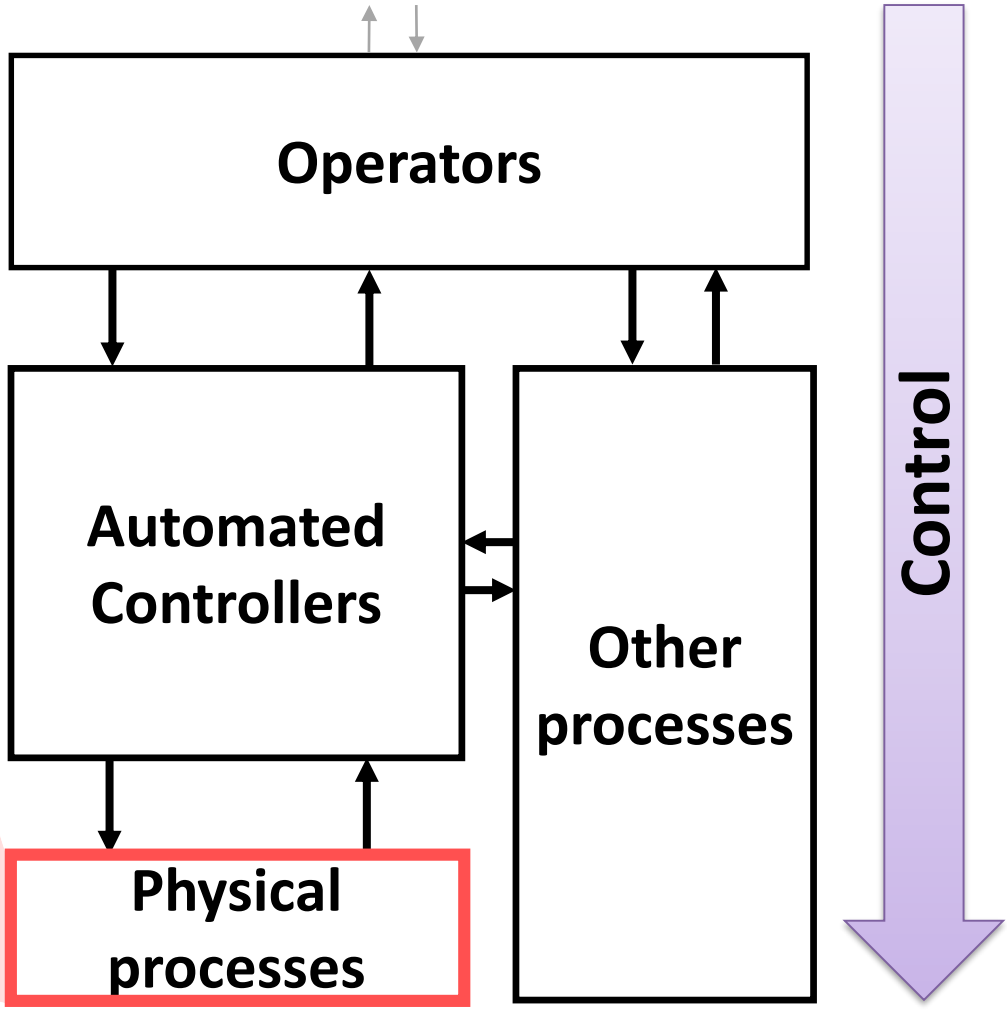
Abstraction





Abstraction

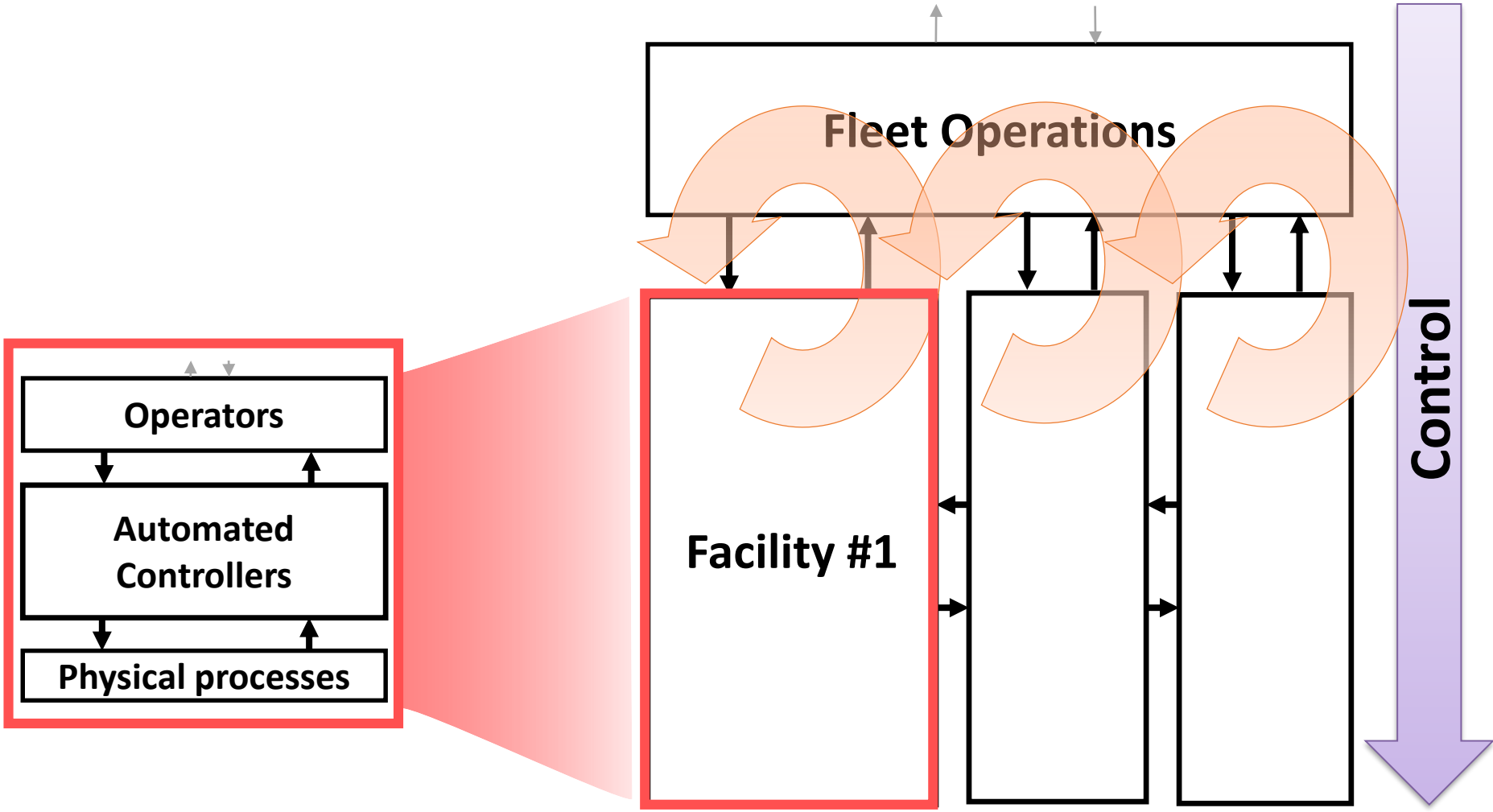




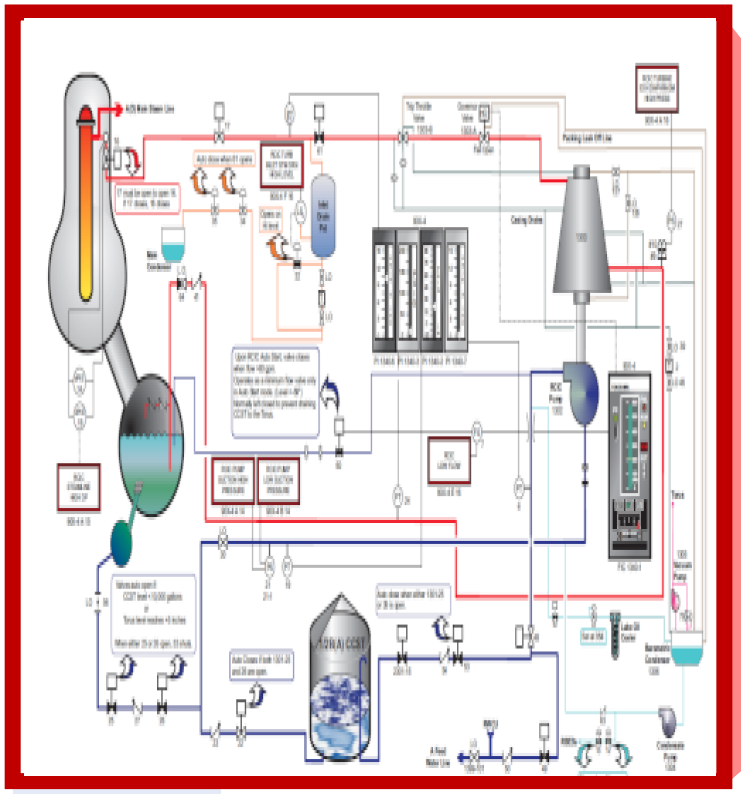
Component view

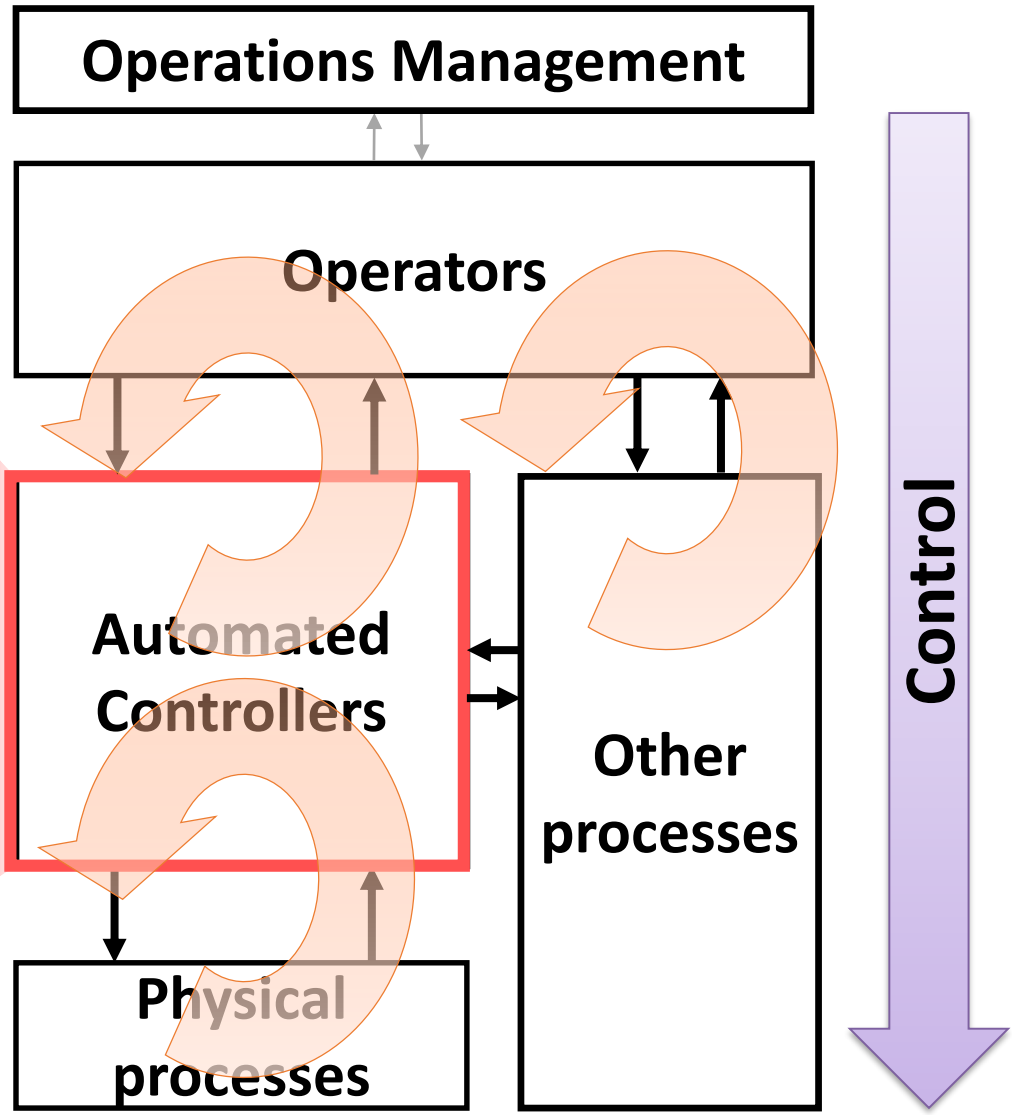
Systems view

Enabling abstraction



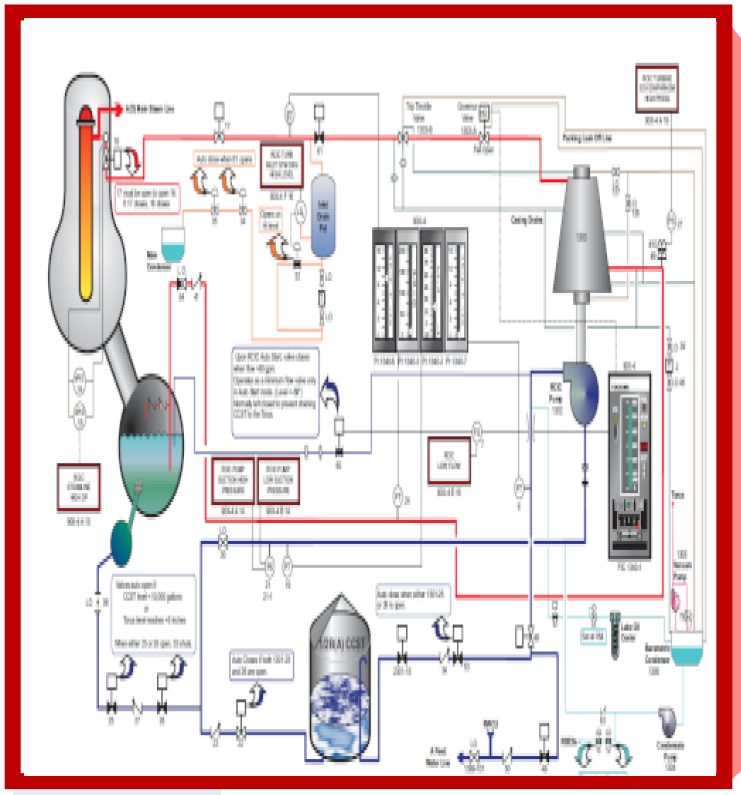
Enabling abstraction





Thomas, 2017

Enabling abstraction



Operations Management

Operators

Automated Controllers

Other processes

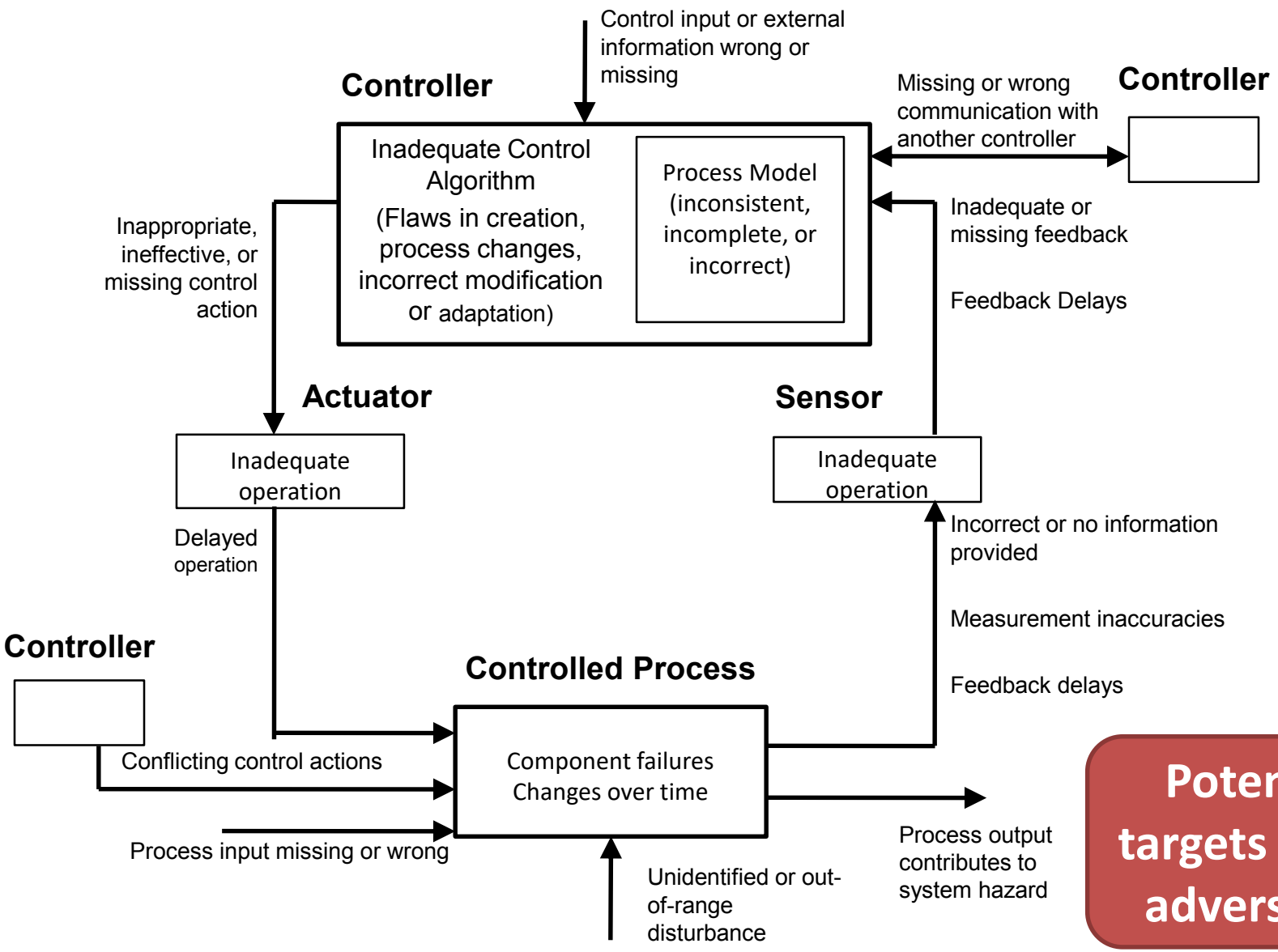
Physical processes

Control

Component view

Systems view

Some Factors in Causal Scenarios

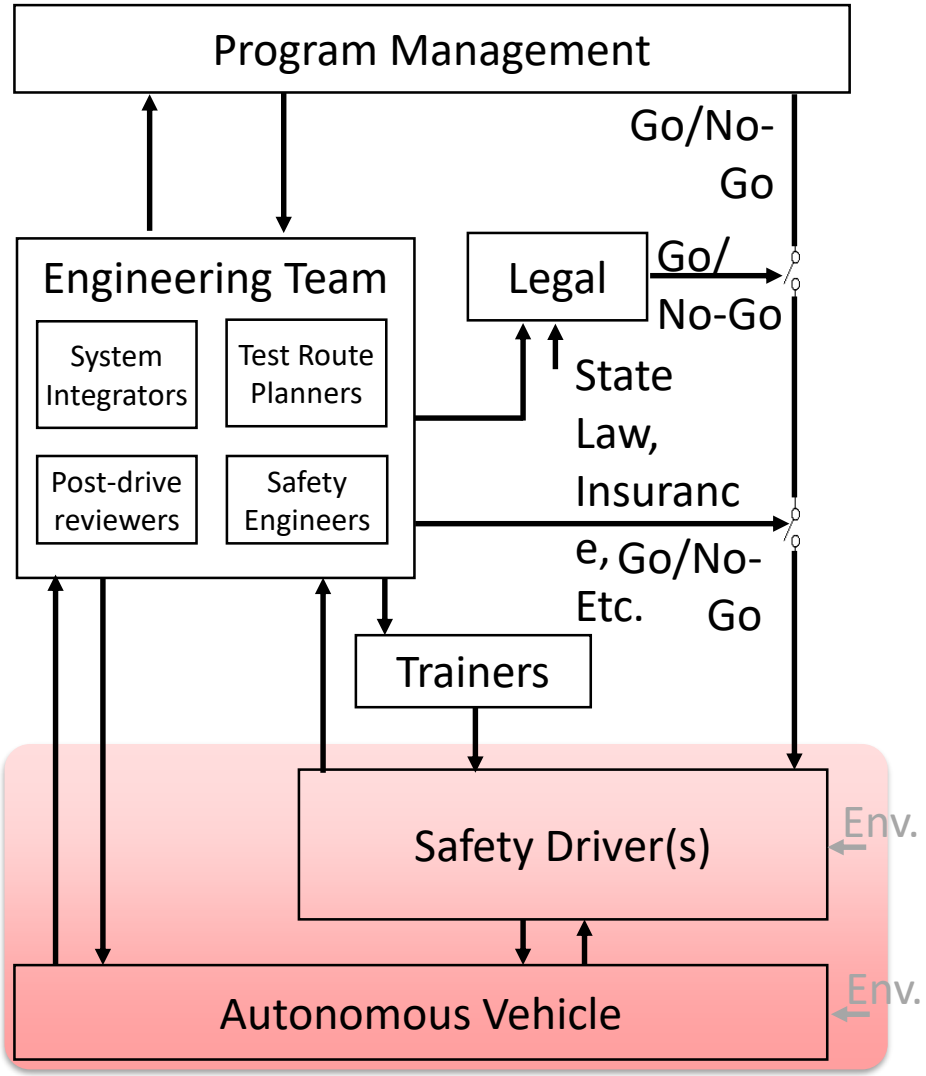


Potential targets for an adversary?

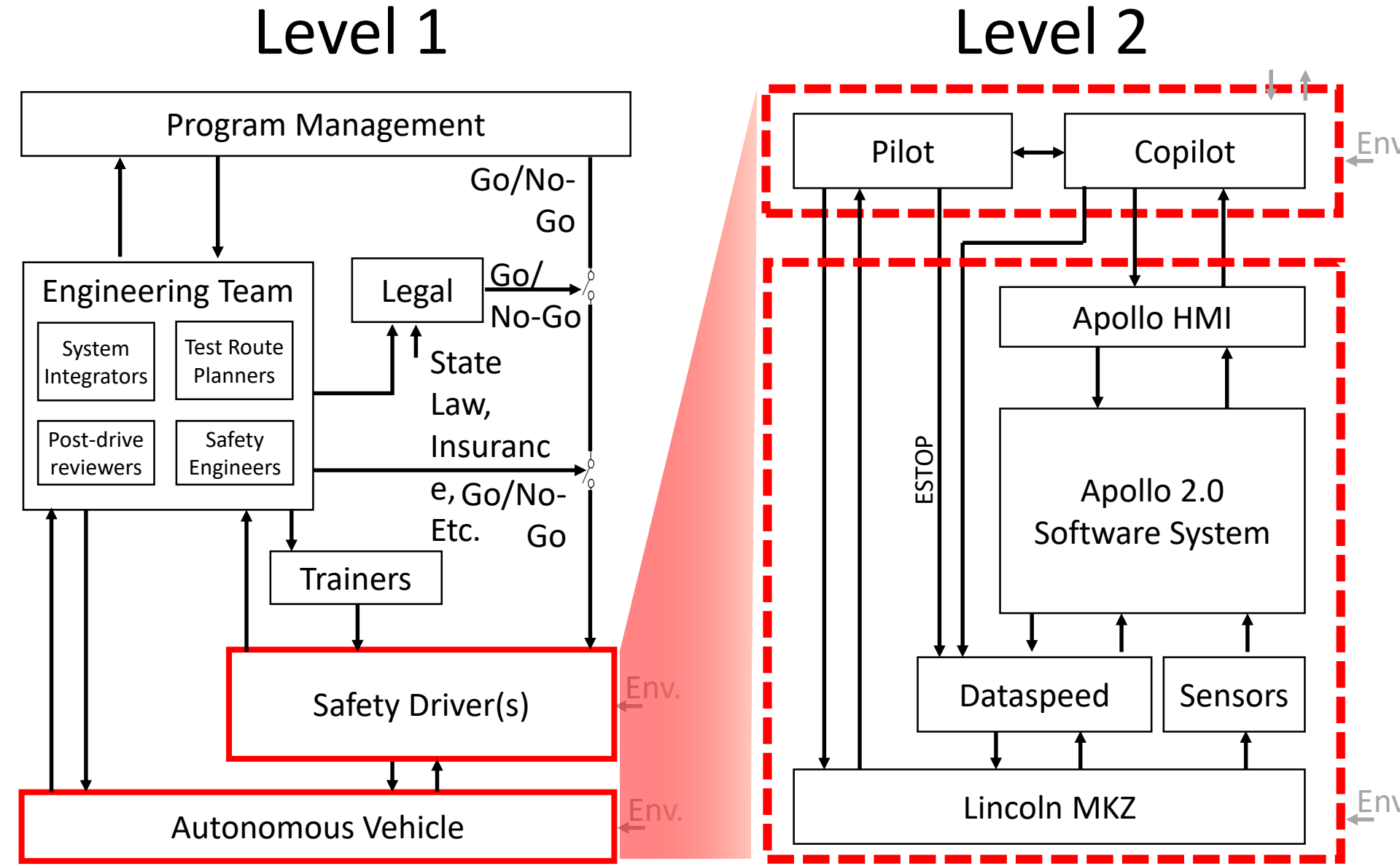
(Leveson, 2012)

Control Structure Refinement

Level 1

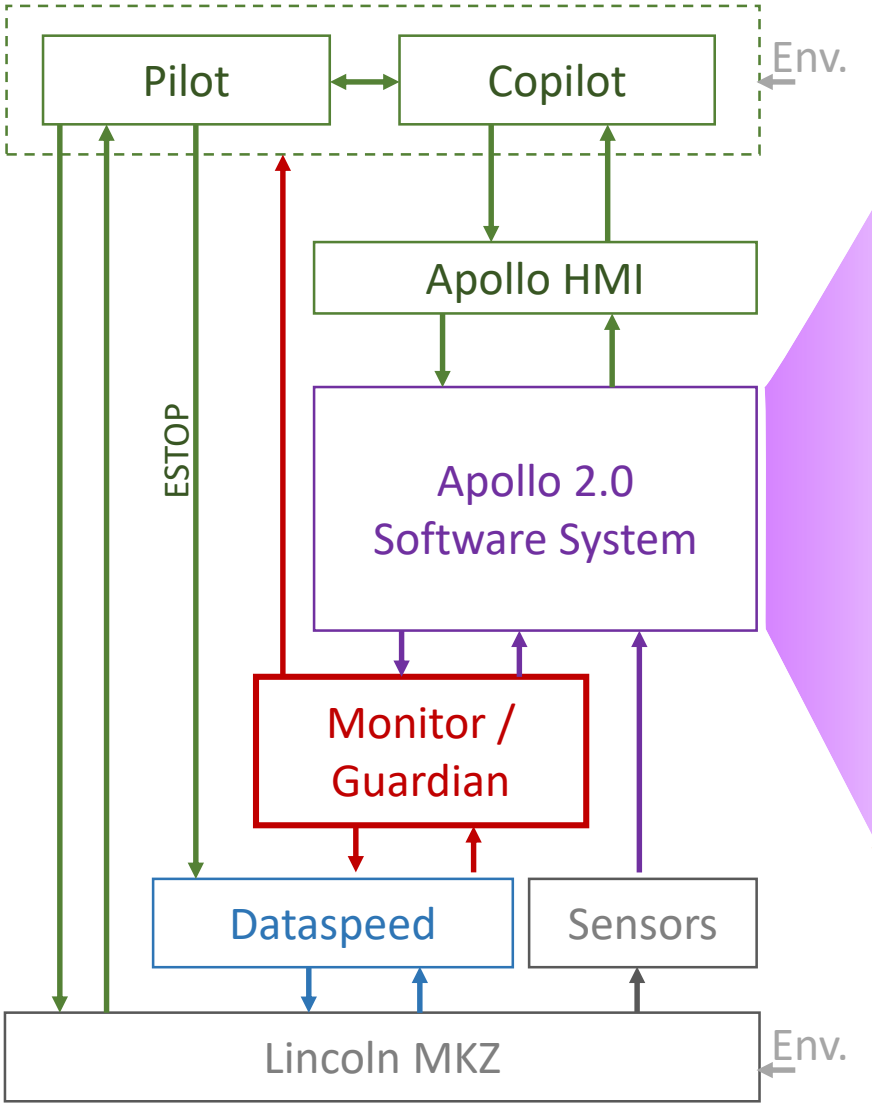


Control Structure Refinement

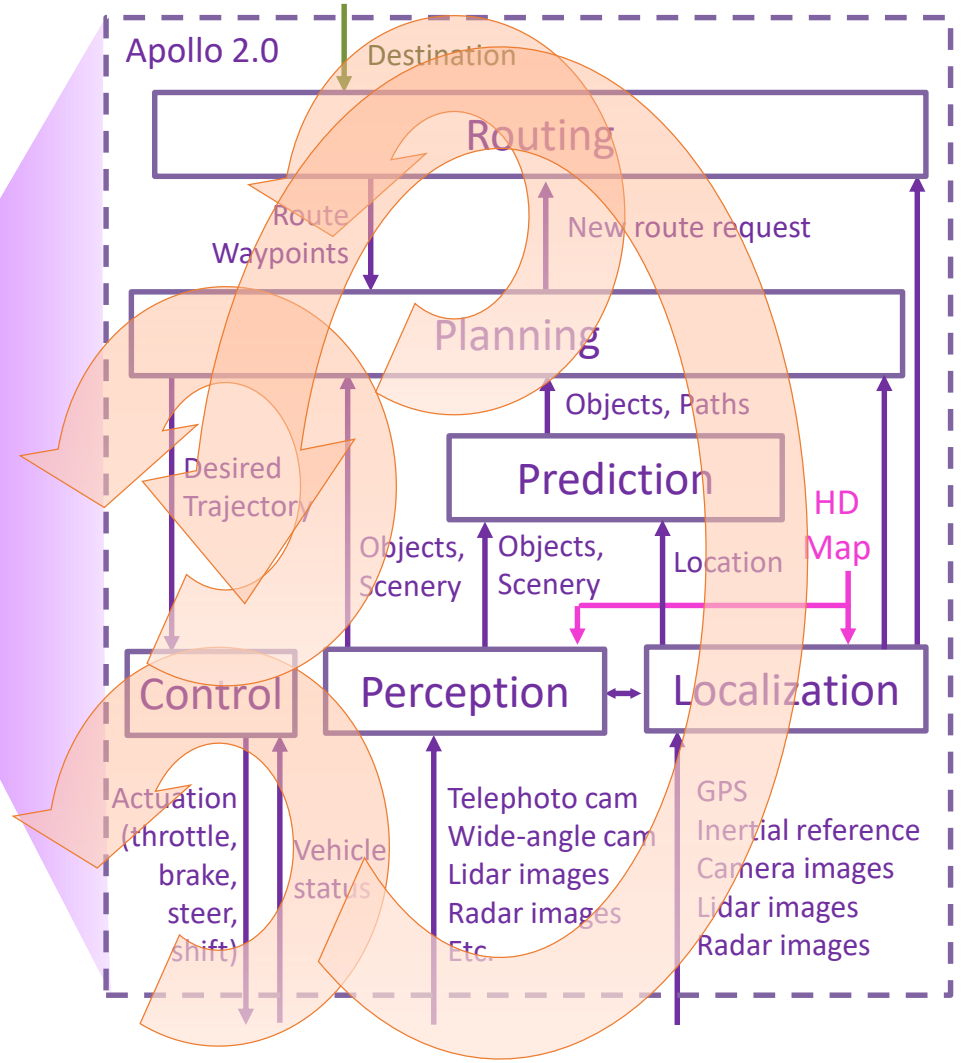


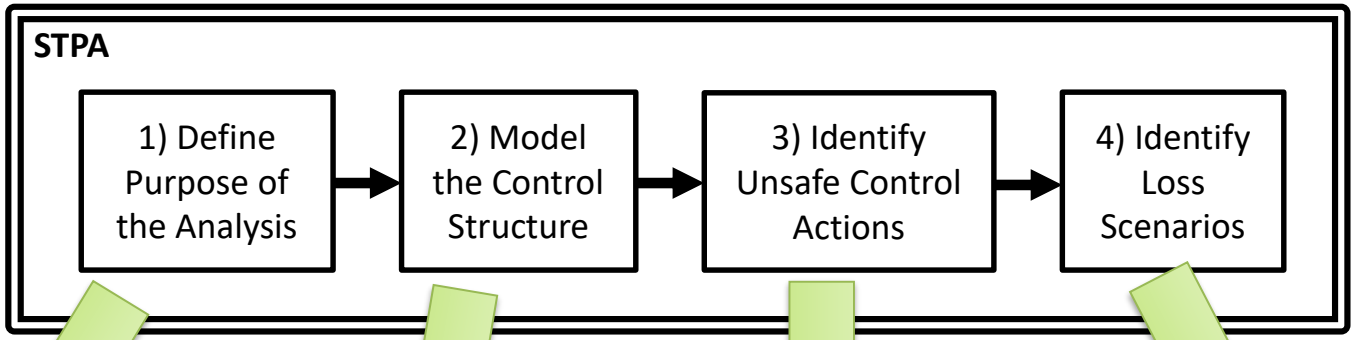
Control Structure Refinement

Level 2



Level 3



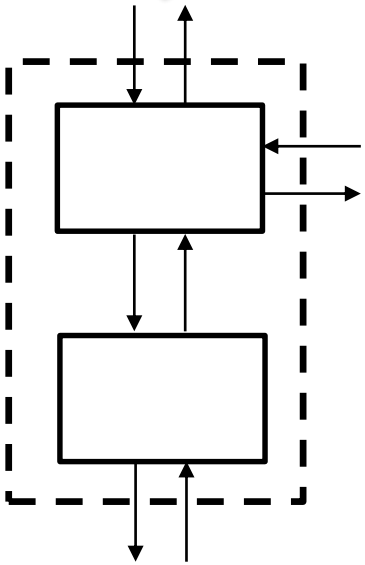


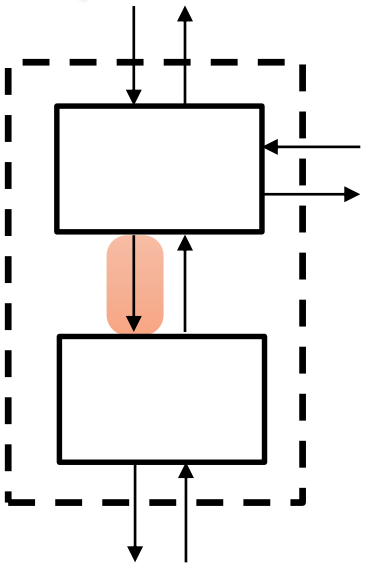
Identify Losses, Hazards

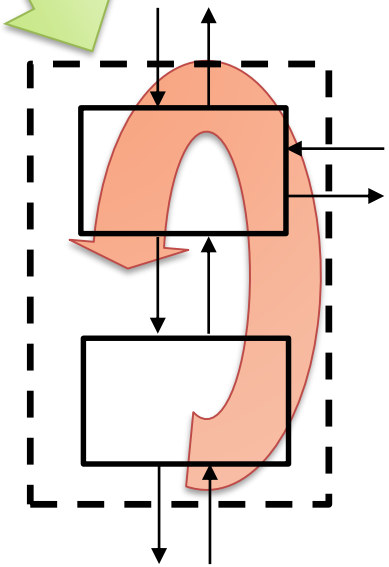
Define System boundary

Environment

System







Losses to prevent

Model

Behavior to prevent

How could behavior occur

(Leveson and Thomas, 2018)

Massachusetts Institute of Technology

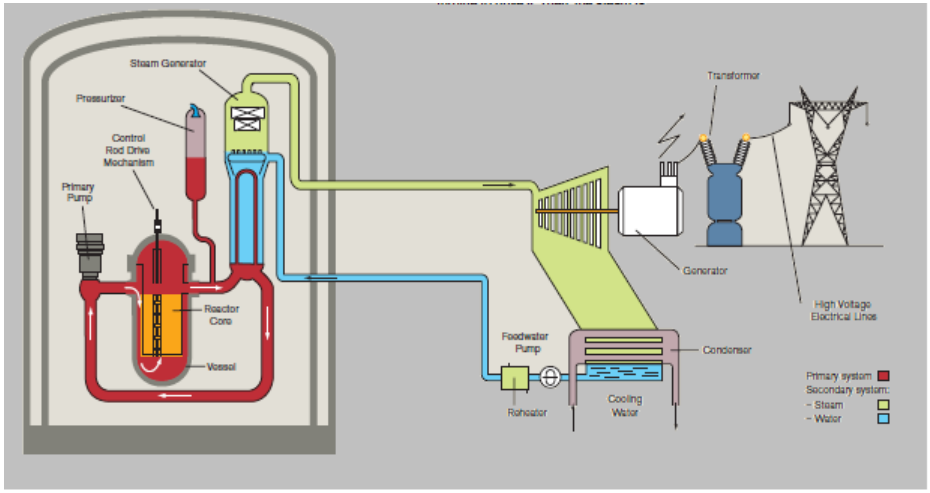
STPA vs. HAZOP vs. FTA vs. FMEA

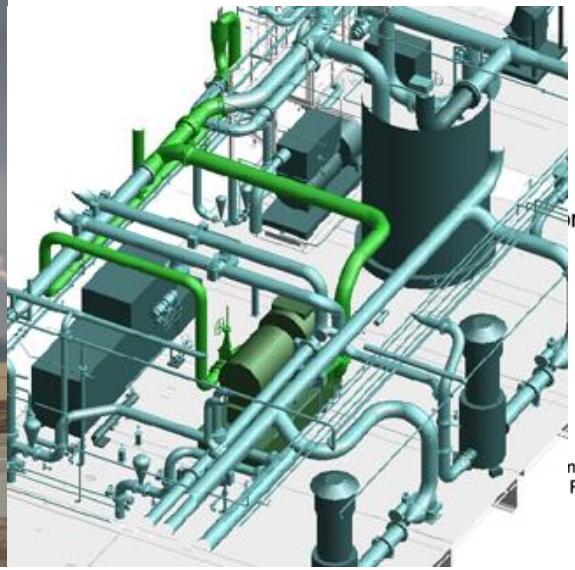
Cooling System Case Study

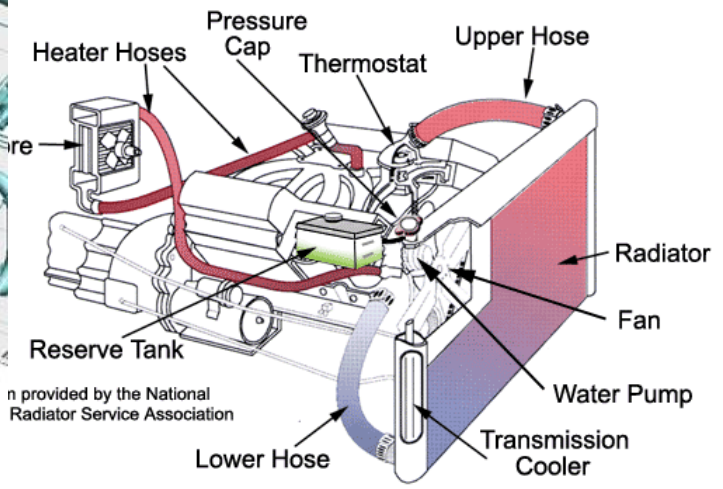
Disclaimer

This presentation is not meant to exhaustively demonstrate STPA. The STPA results presented are selected to show key differences between methods.

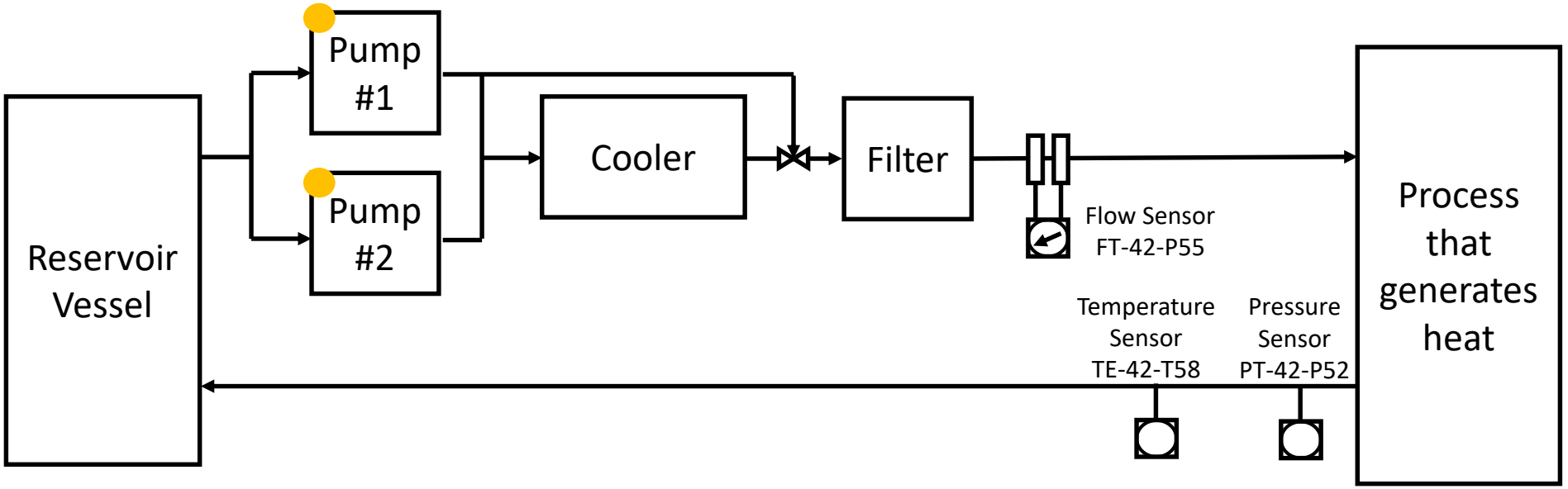
Examples of Cooling systems





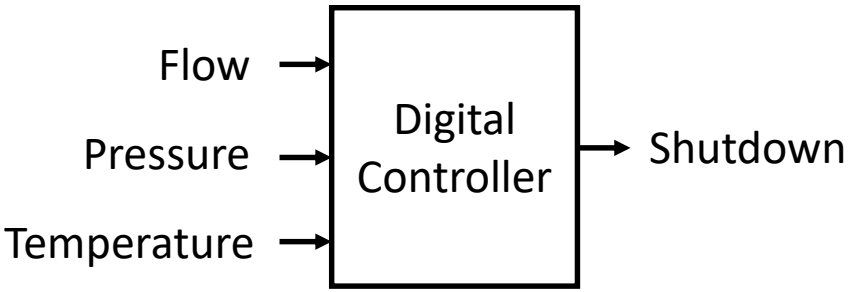


Cooling System (Old System)



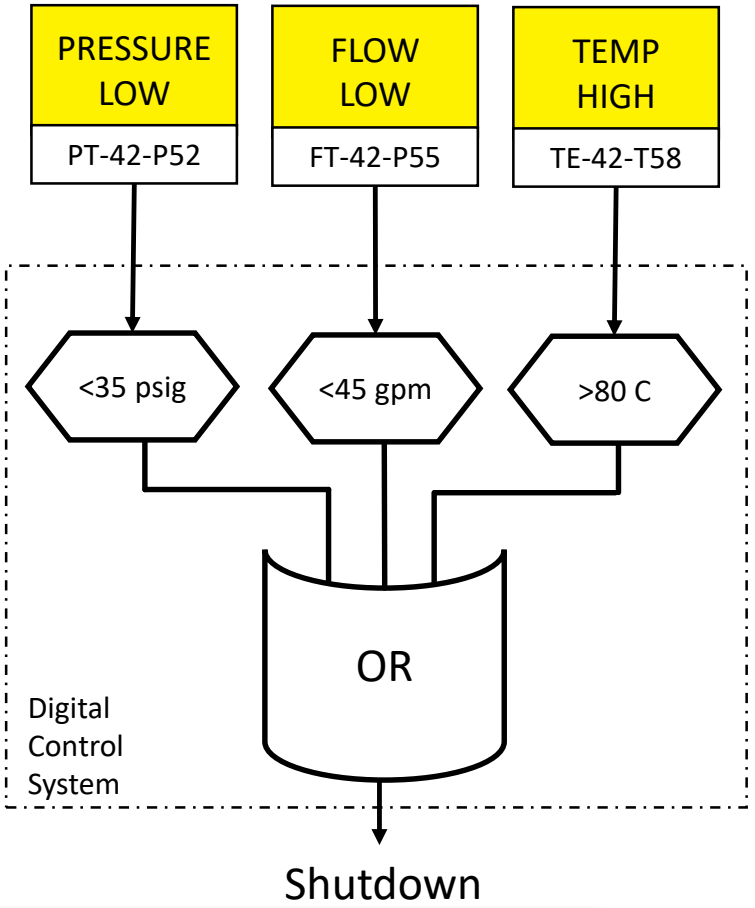
Each pump sized for 100% capacity
Second pump on standby

Manual Control (Maintenance)



Loss of Cooling detection
Old System

- Controller applies a threshold to each sensor
  - Pressure below 35 psig
  - Flow below 45 gpm
  - Temp above 80 C
- Shutdown any time inadequate cooling condition is indicated
  - Low pressure
  - Low flow
  - High temp



Problem: Inadvertent Shutdown (from single sensor failure)
~$1m economic loss **each time**

Digital Modification

Purpose:

- To resolve the problems associated with the existing control systems, each existing system is replaced with a new, state-of-the-art, Digital Control System (DCS).
- These new systems employ, as a minimum, redundant input signal devices, redundant digital signal processors, and redundant output devices.

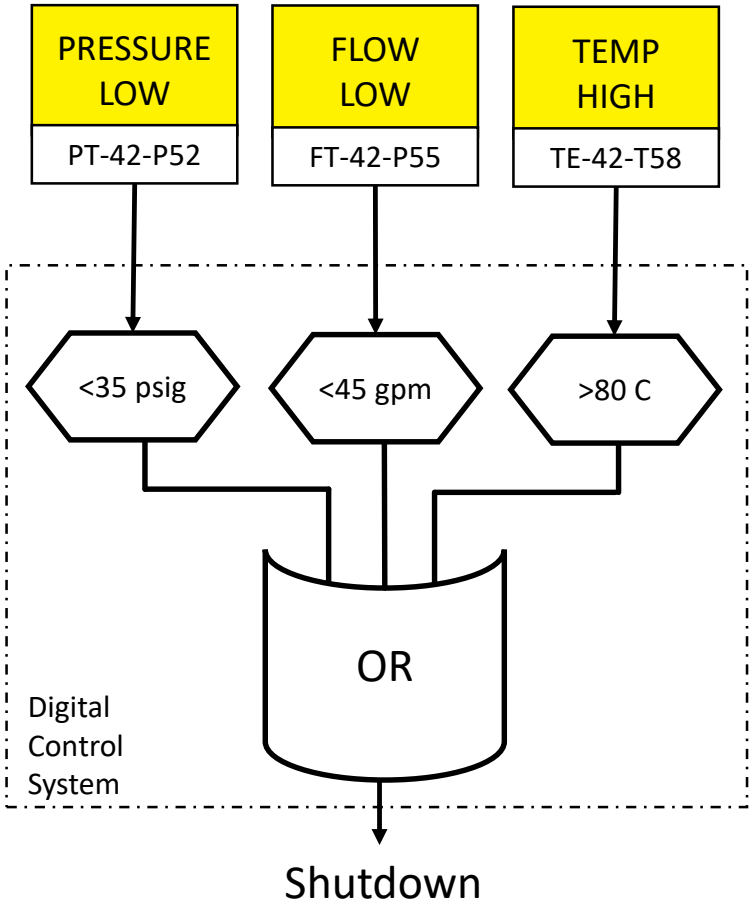
System Requirements Spec:

- DCS will include, among other parameters, protection from **loss of cooling**, which will be measured by low cooling flow, low cooling pressure, or high cooling temperature. DCS will provide automatic Shutdown on loss of cooling.
- The system will identify faulted instruments and will have protection from actuation of any of the shutdown features, including cooling flow, due to a faulted instrument. The system will send a shutdown signal if all instruments for a given parameter are faulted.

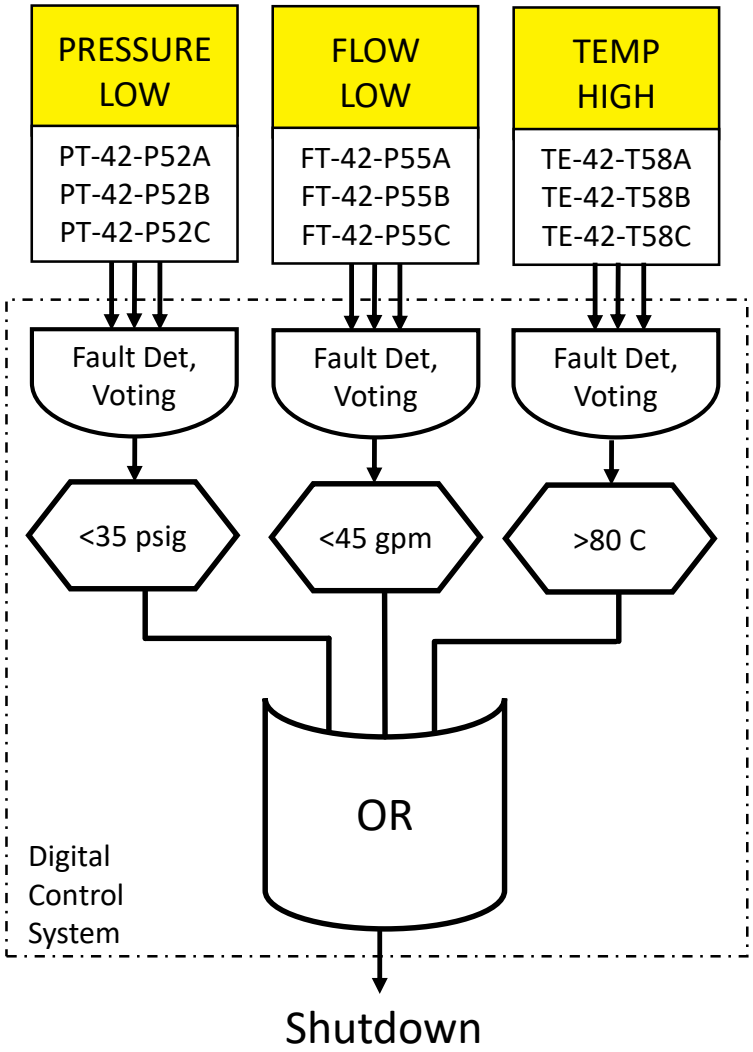
Cost to upgrade: ~$1m
Worth it to prevent an Inadvertent Shutdown!

Loss of Cooling detection

Old System



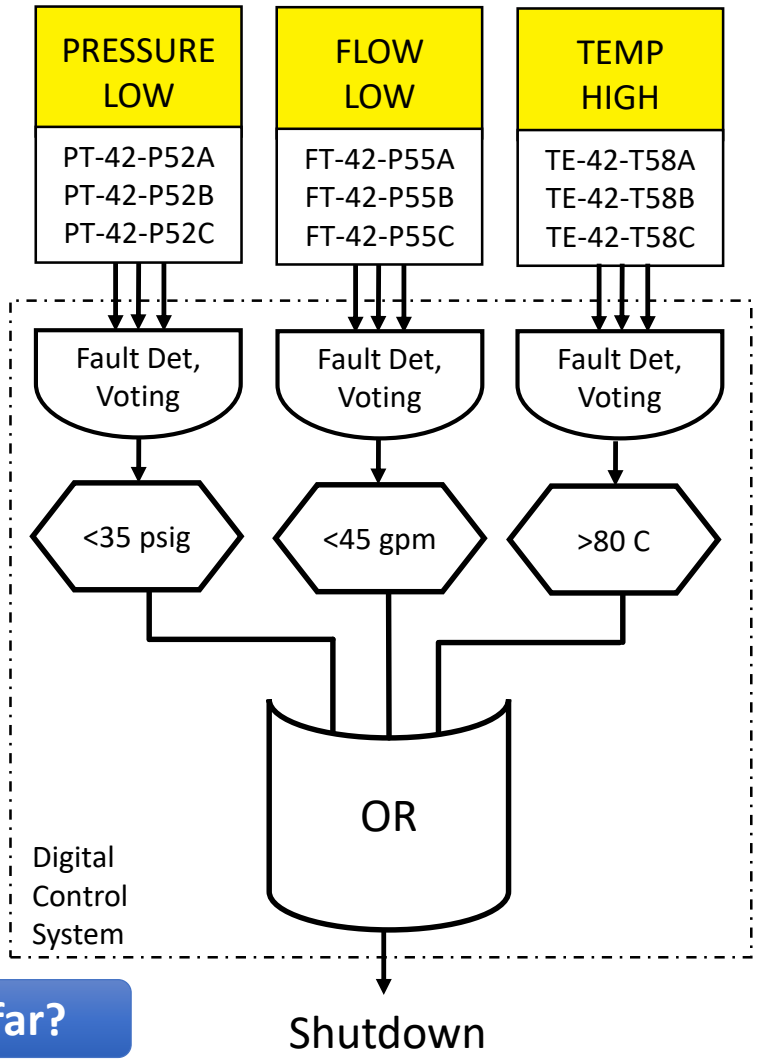
New System



Loss of Cooling detection:
New System

Fault detection and voting

- The Digital Controller will detect faulted instruments and automatically remove them from the logic.
- The logic is designed to identify a faulted instrument by measuring the output either high or low outside the calibrated range.
- 1oo3 logic is used:
  - When one instrument is faulted, it is automatically bypassed by the logic. Bypassing an instrument automatically changes the voter logic to use the remaining two instruments.
  - If a second instrument is faulted, it is also automatically bypassed and the voter logic uses the remaining valid instrument.
  - If all three instruments for a channel are faulted, the logic is designed to send a shutdown signal.
- The setpoints for detection of faulted instrument are 3.8 mA low and 20.32 mA high.



Does this make sense so far?

HAZOP (actual)

- 120 person-hours total

HAZOP Excerpt (simplified)

Parameter	Guideword	Deviation	Cause	Effect	Protective Systems
Temperature	Higher	Temp. above 84 C	Coolant Blockage, Process overheating, etc.	Damage to Equipment, Loss of Production, etc.	DCS logic, 3x temp sensors
Temperature	Lower	Temp. below 50 C	Process underheating, etc.	No meaningful effect (low temp is good)	
Flow	Higher	Flow above 50 gpm	Both pumps on simultaneously	No meaningful effect (high flow is good)	
Flow	Lower	Flow below 45 gpm	Coolant Blockage, etc.	Damage to Equipment, Loss of Production, etc.	DCS logic, 3x flow sensors

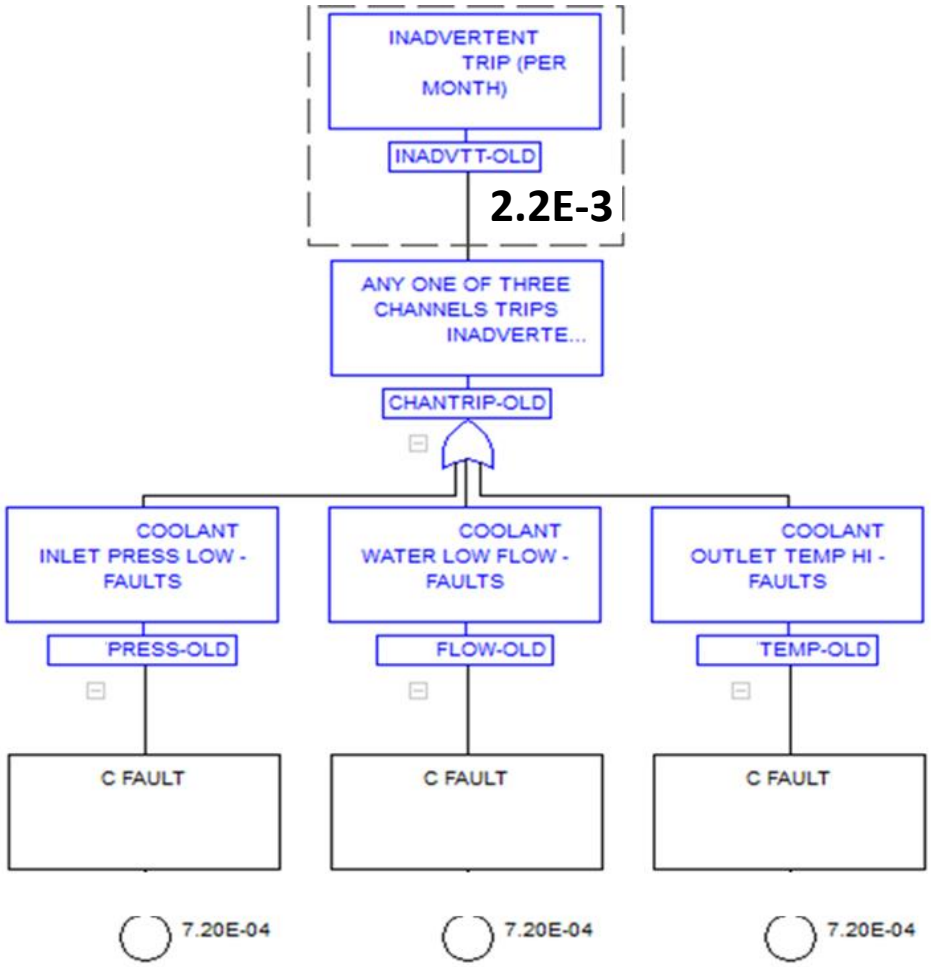
Actual HAZOP: 120 person-hours total

FMEA Excerpt (simplified)

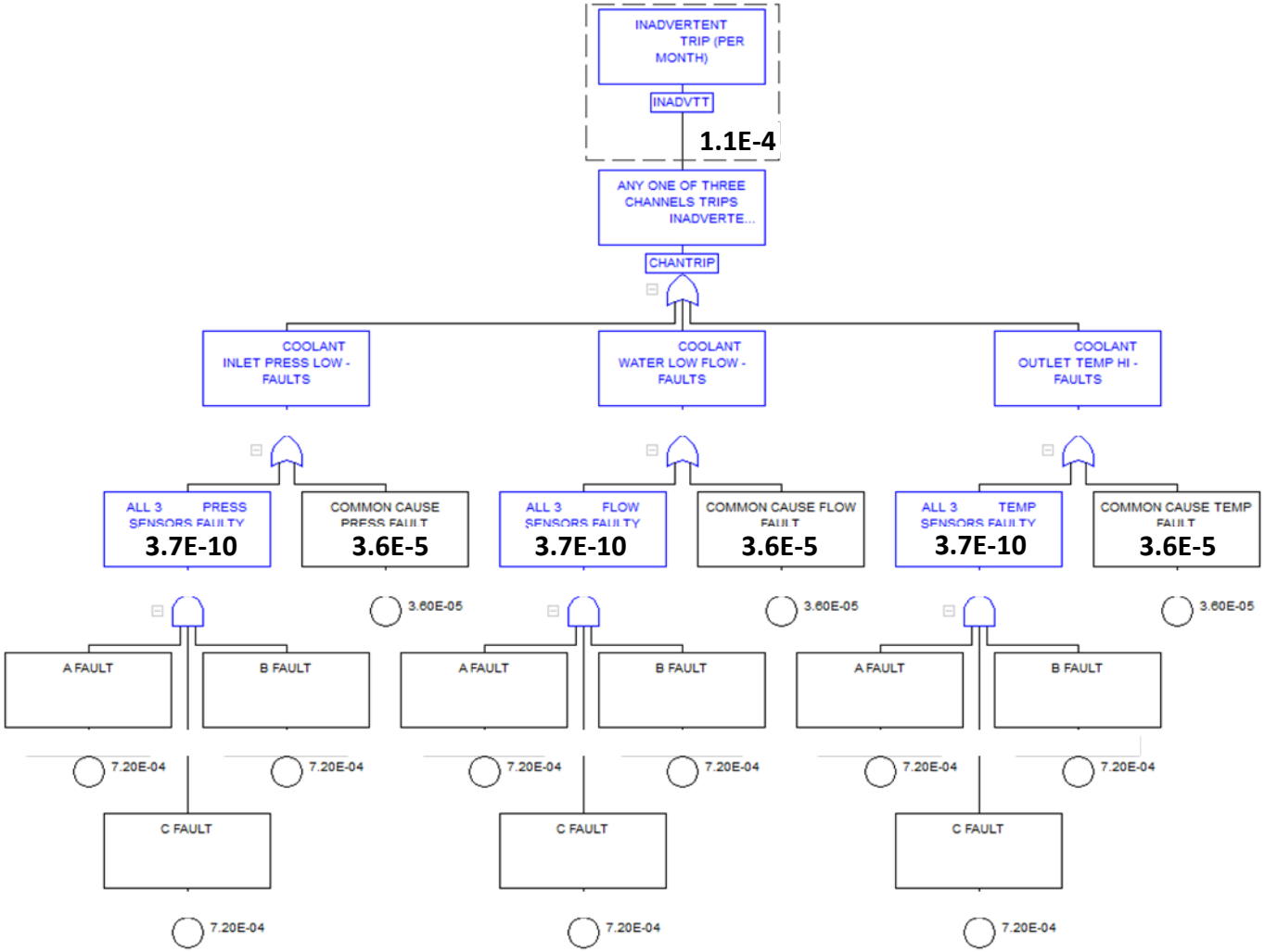
Component	Failure Mode	Failure Mechanism	Effect	Mitigations
Temperature Sensor TE-42-T58	Fail high	[…]	Unnecessary shutdown by DCS (false positive)	3x Temp Sensors, DCS logic protects from single or dual sensor failures
Temperature Sensor TE-42-T58	Fail low	[…]	Undetected loss of cooling: Damage to equipment, Loss of production (false negative)	3x Temp Sensors, DCS logic protects from single or dual sensor failures
Flow Sensor FT-42-P55	Fail high	[…]	Undetected loss of cooling: Damage to equipment, Loss of production (false negative)	3x Flow Sensors, DCS logic protects from single or dual sensor failures
Flow Sensor FT-42-P55	Fail low	[…]	Unnecessary shutdown by DCS (false positive)	3x Flow Sensors, DCS logic protects from single or dual sensor failures

Actual FMEA: 200+ pages, 1,000+ person-hours

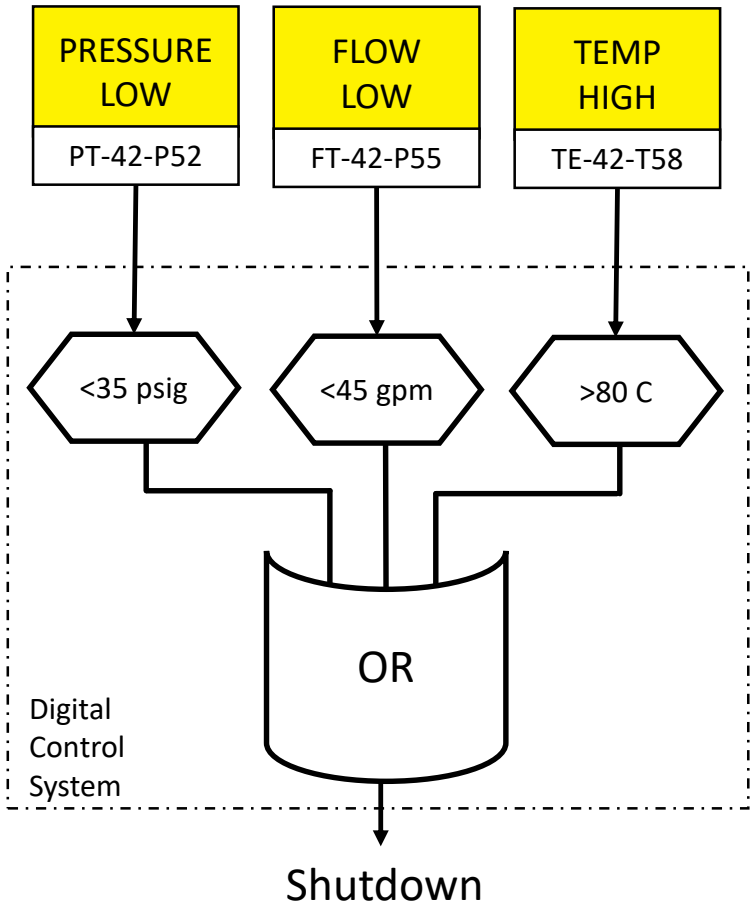
Loss of Stator Cooling detection
Old System



Loss of Cooling detection:
New System



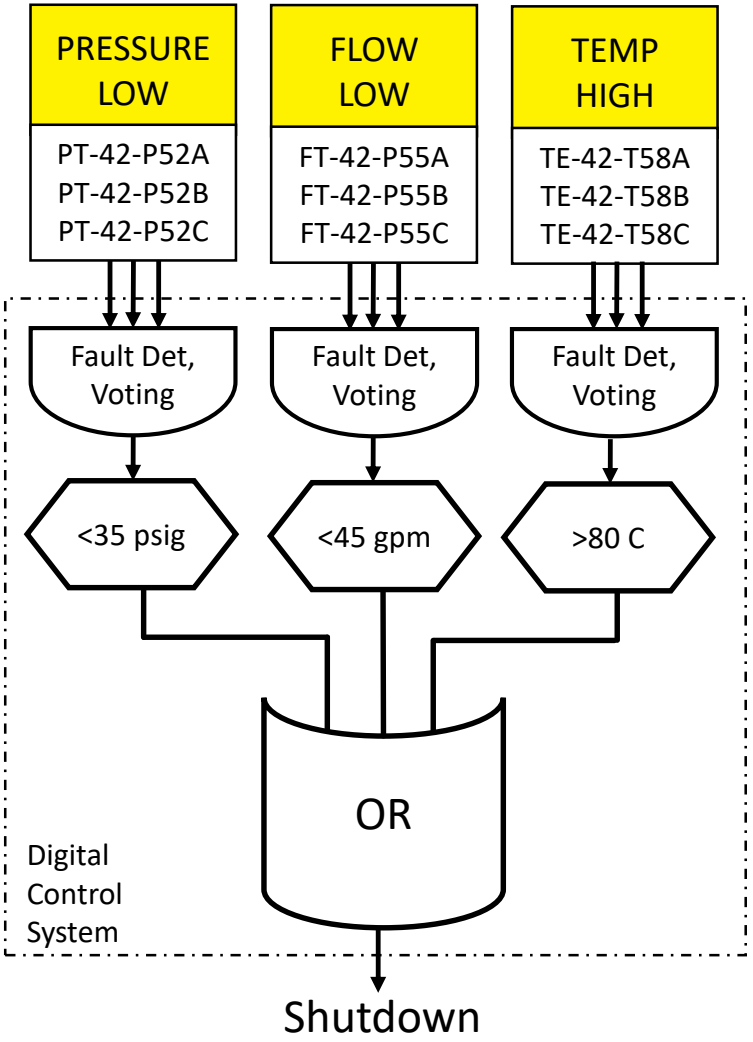
Old System



$P(IS/m) = 2.2 \times 10^{-3}$

(~Once in 38 years)

New System

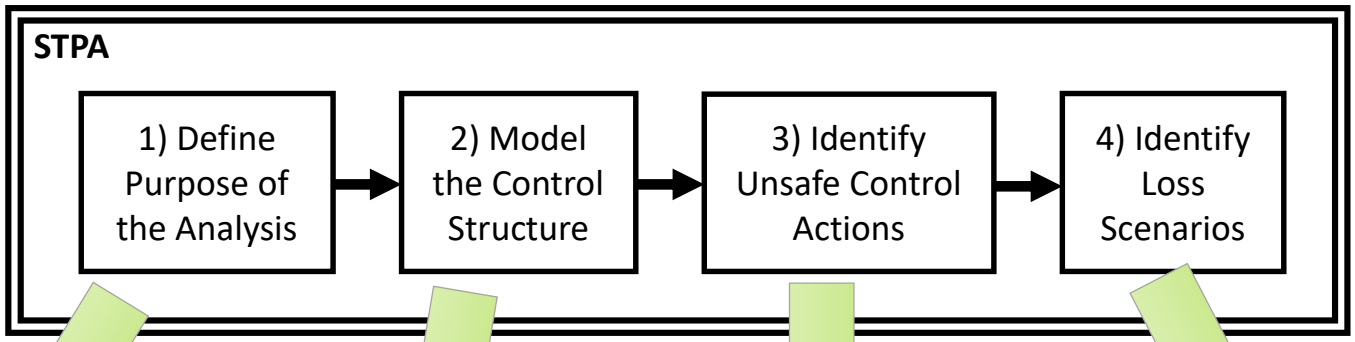


$P(IS/m) = 1.1 \times 10^{-4}$

(~Once in 757 years)

IS = Inadvertent Shutdown

Let's try STPA!

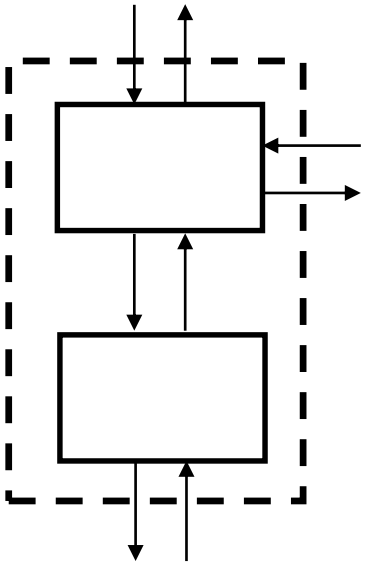


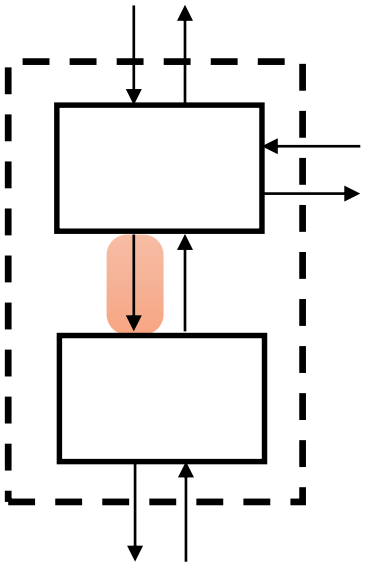
Identify Losses, Hazards

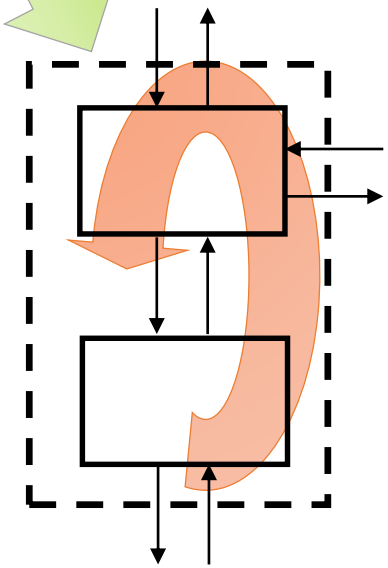
Define System boundary

Environment

System







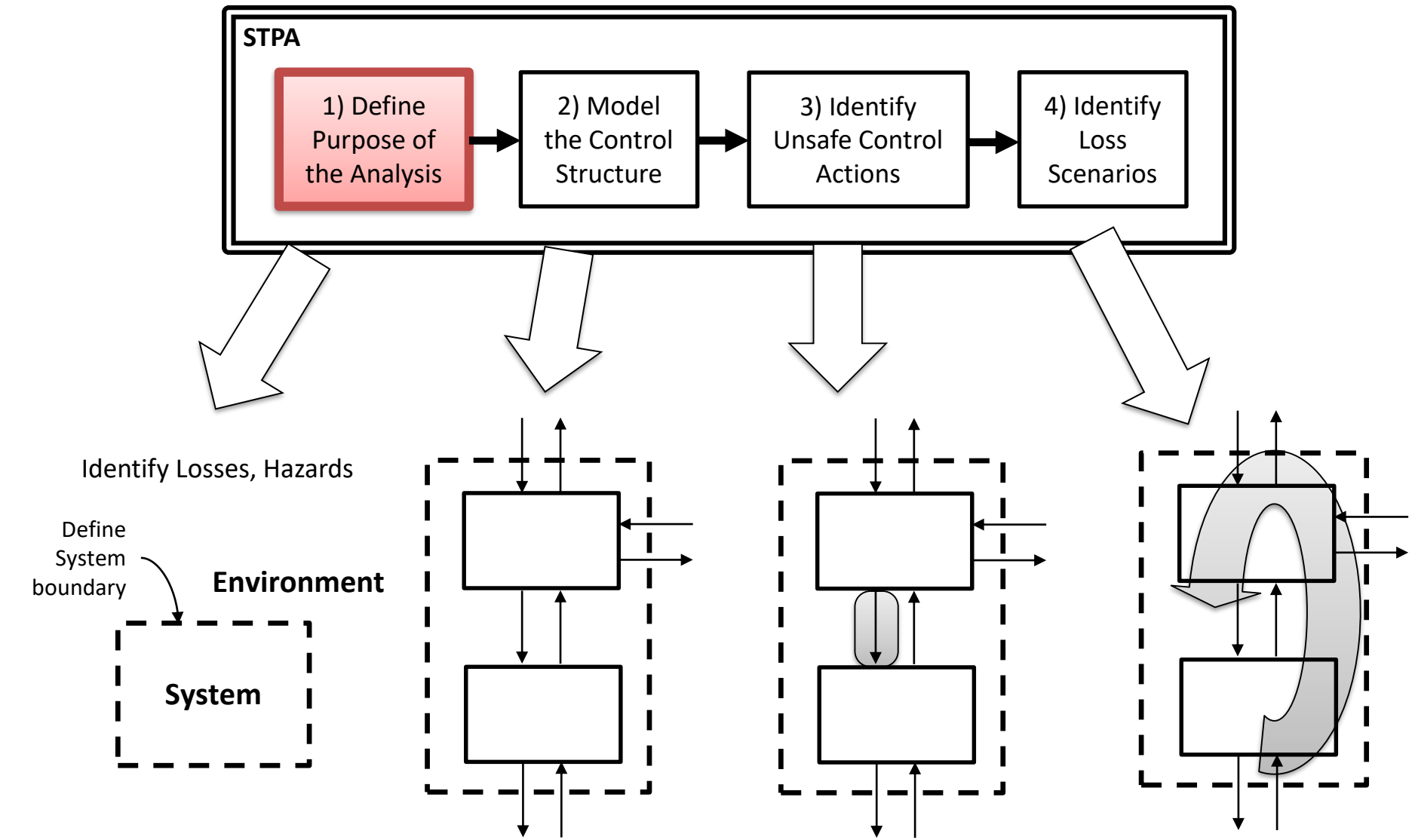
Losses to prevent

Model

Behavior to prevent

How could behavior occur

(Leveson and Thomas, 2018)

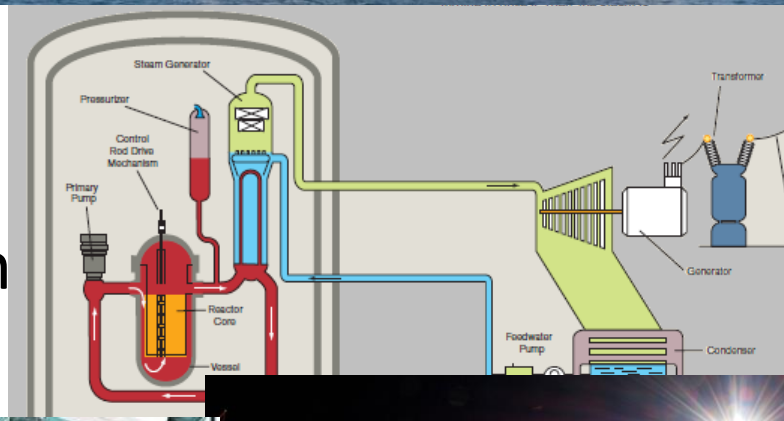


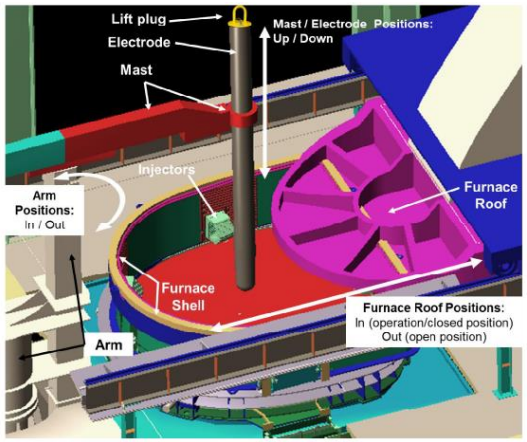
(Leveson and Thomas, 2018)

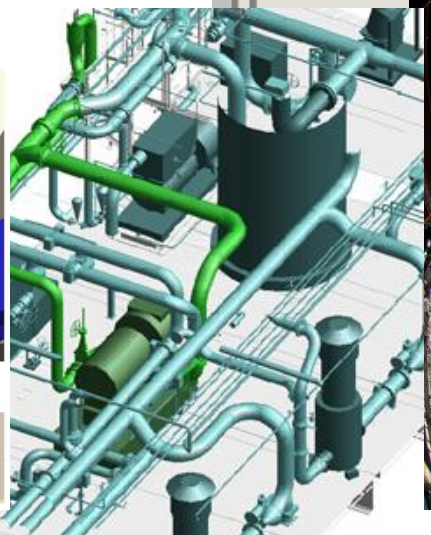
STPA Losses

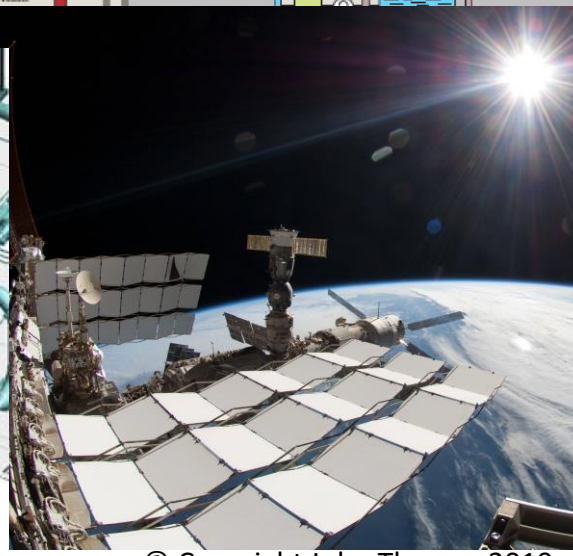
Example Losses

- L-1: Loss of life or injury
- L-2: Equipment damage
- L-3: Environmental contamination
- L-4: Loss of mission
- Etc.







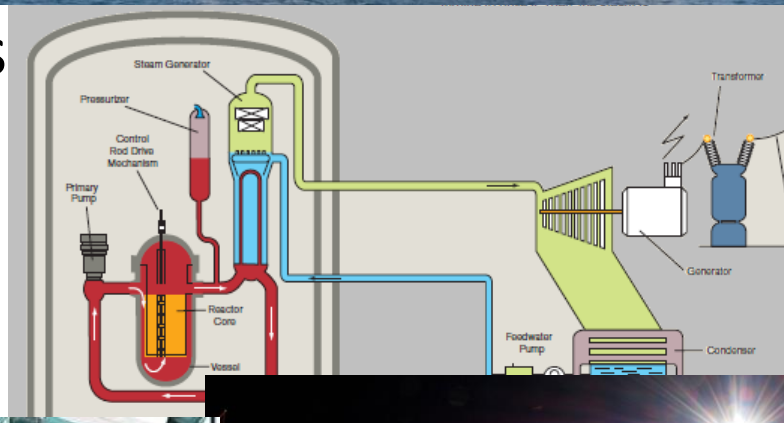


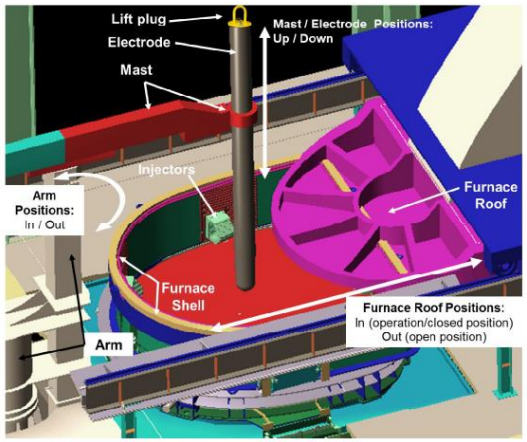
(John Thomas, 2014)

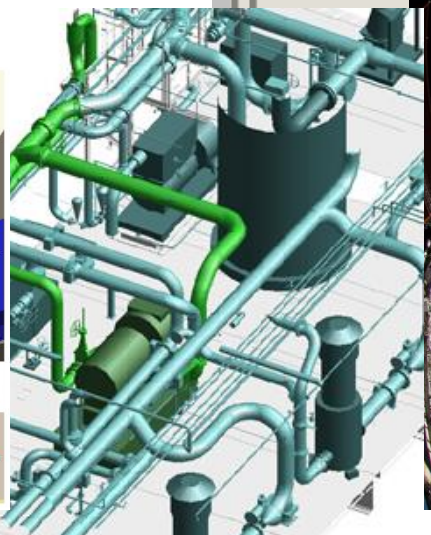
STPA System Hazards

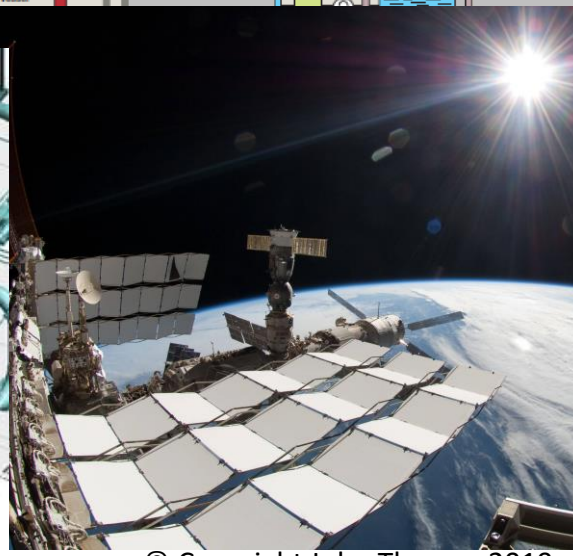
Example System Hazards

- H-1: Plant releases toxic materials
- H-2: Plant unable to produce X
- H-3: Plant is physically damaged
- Etc.

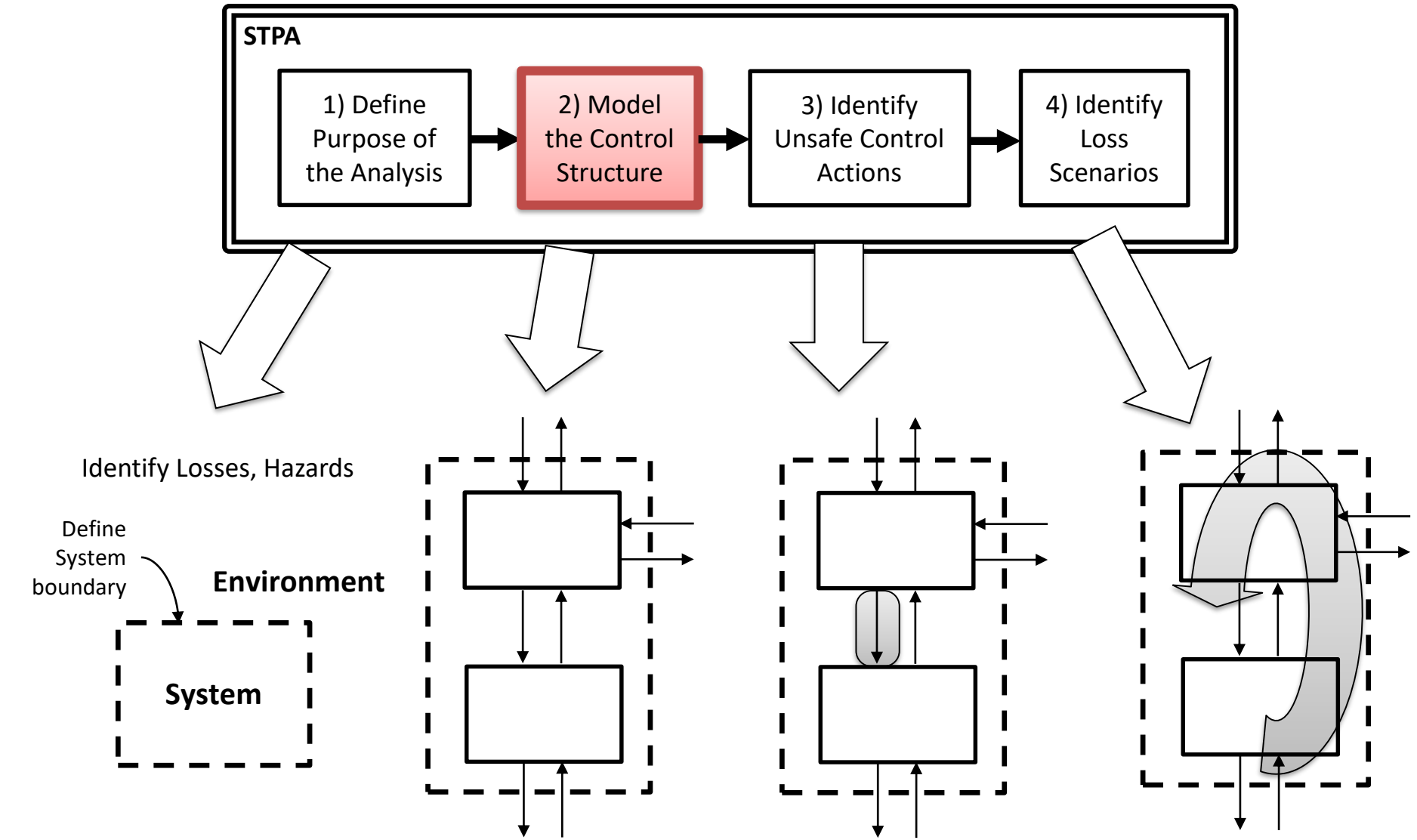






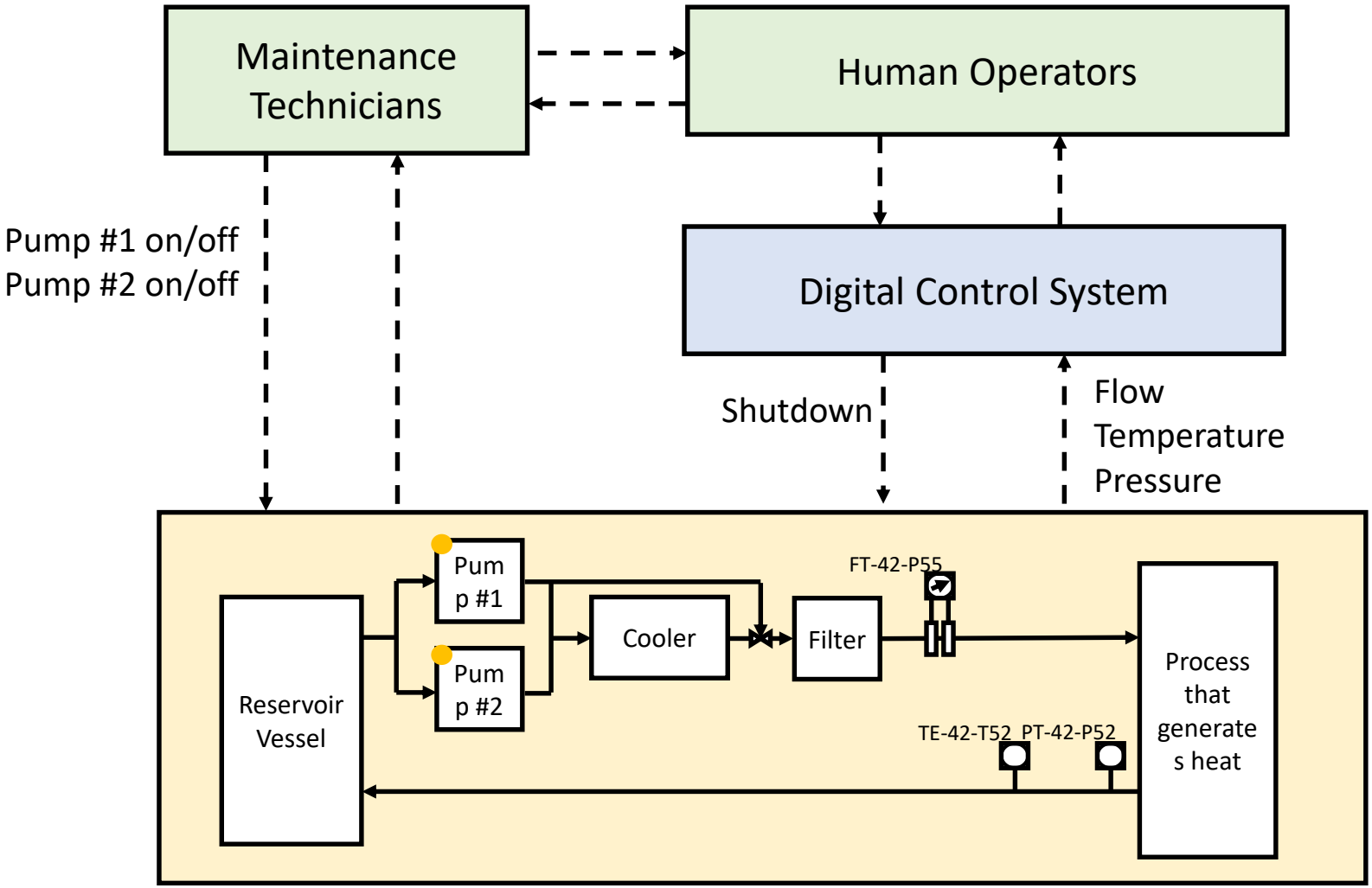


(John Thomas, 2014)



(Leveson and Thomas, 2018)

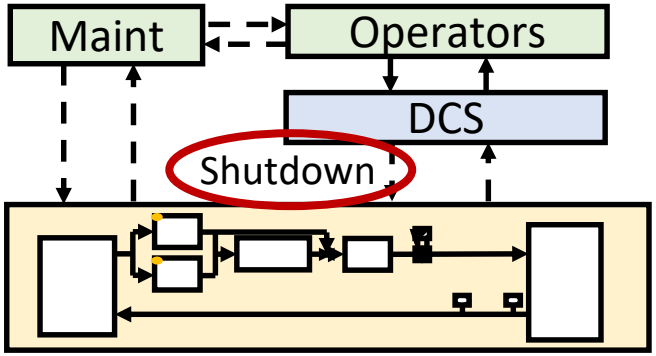
Control Structure



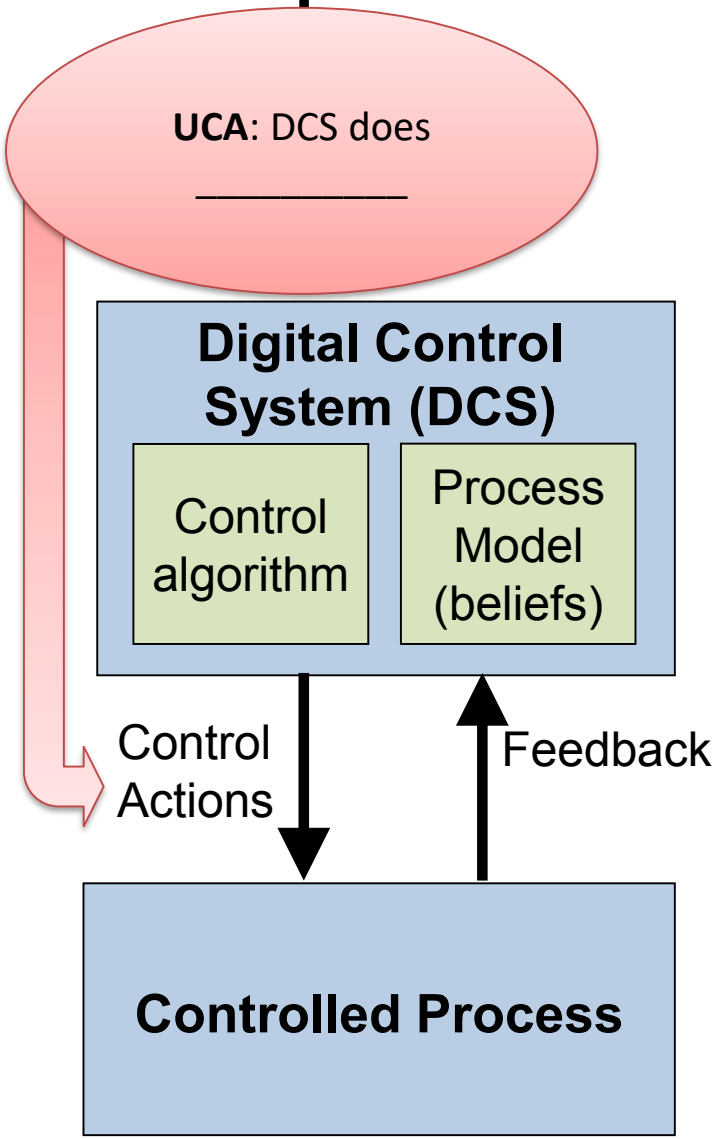
Asking the right questions

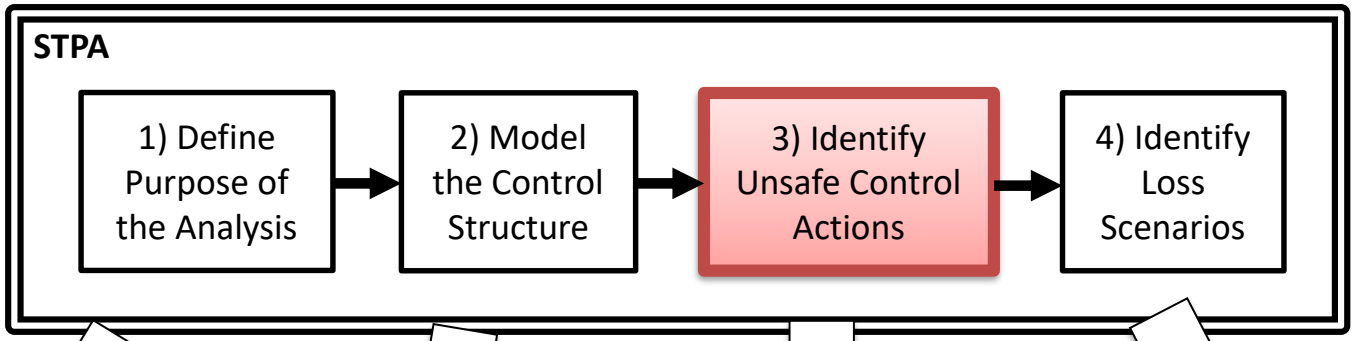
Loss: Loss of Mission (unnecessary shutdown)

Question: What DCS control actions can cause unnecessary shutdown?



UCA: DCS does _____



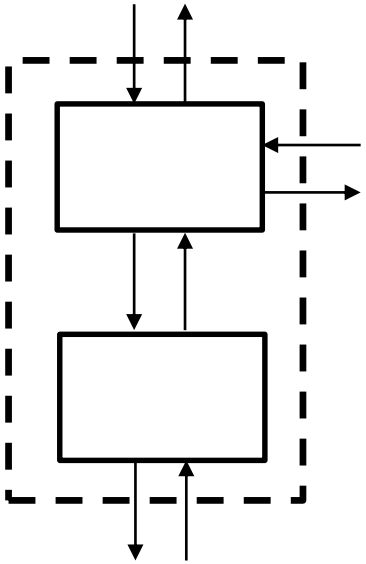


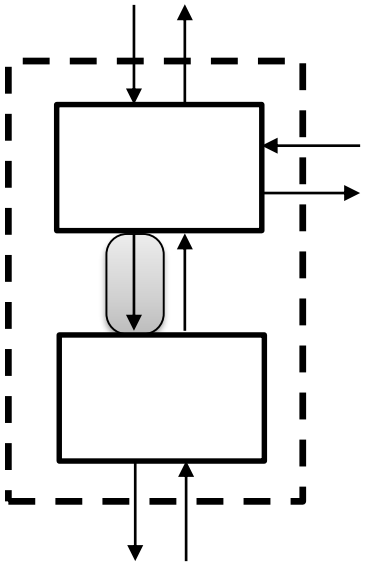
Identify Losses, Hazards

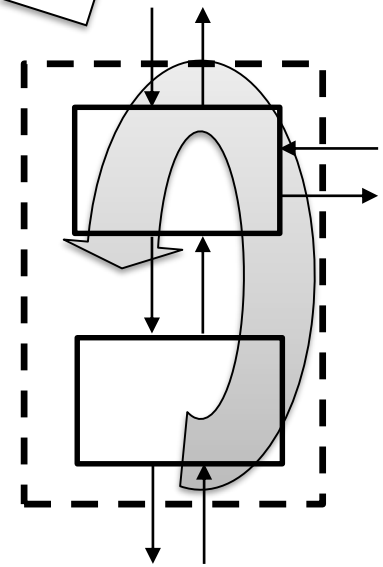
Define System boundary

Environment

System

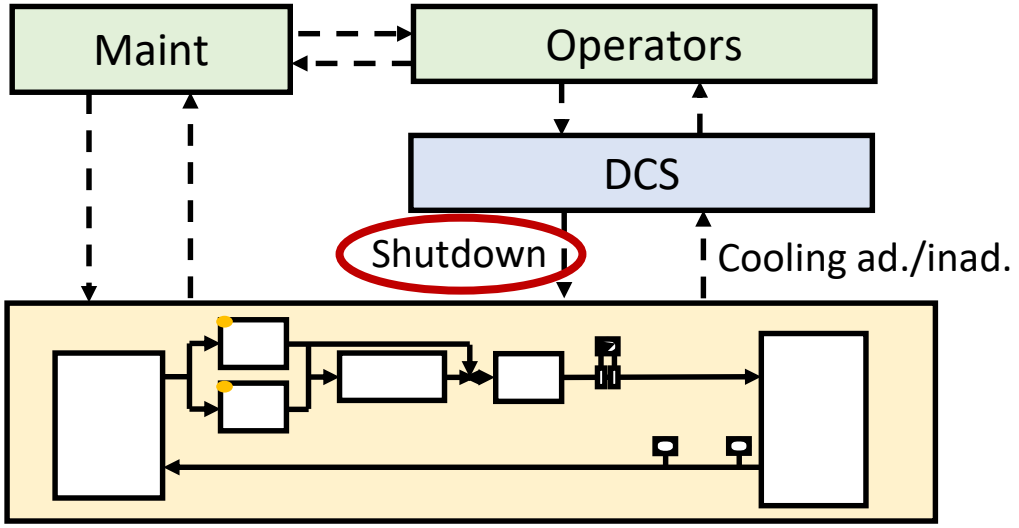






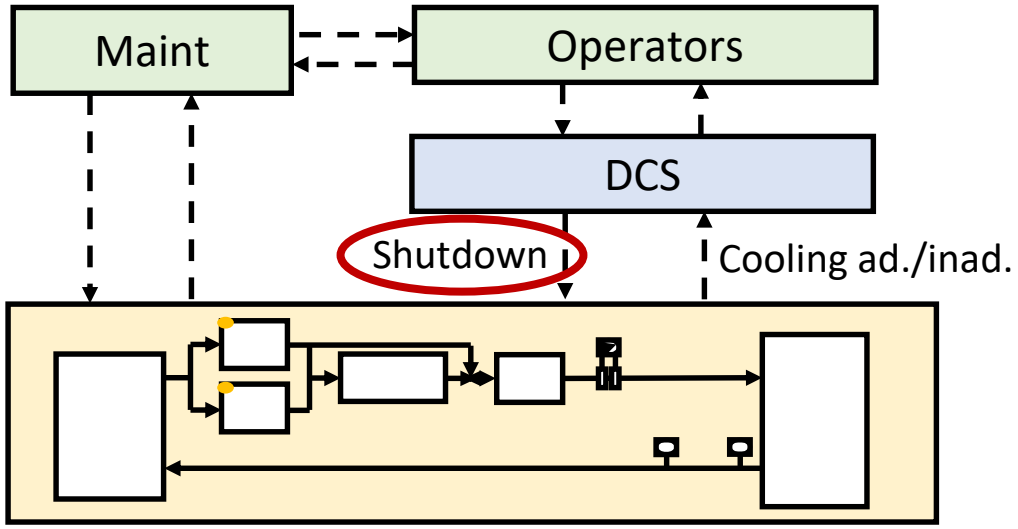
(Leveson and Thomas, 2018)

Unsafe Control Actions



	Not providing causes hazard	Providing causes hazard	Too early, too late, out of order	Stopped Too Soon / Applied too long
Shutdown Cmd				

Unsafe Control Actions

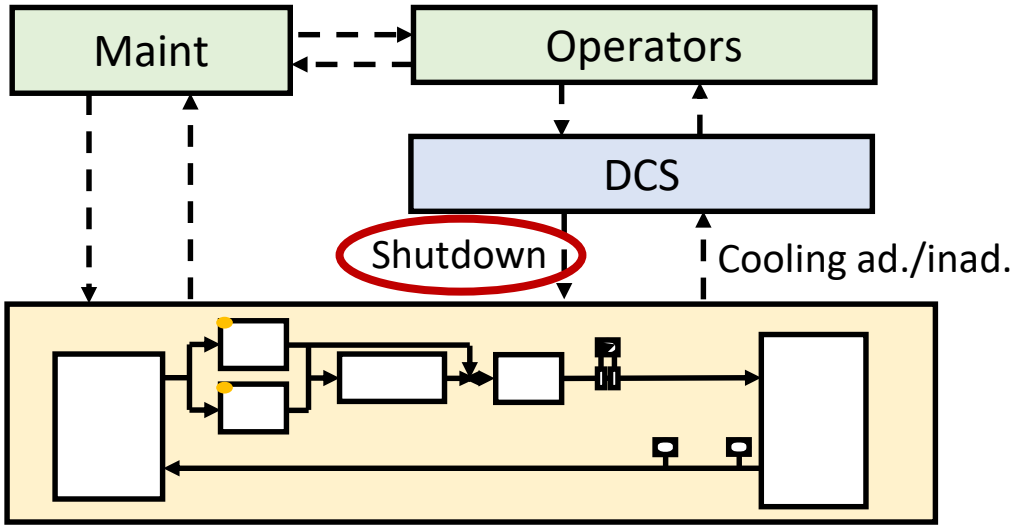


L-1: Loss of life or injury
L-4: Loss of mission (unnecessary shutdown)

	Not providing causes hazard	Providing causes hazard	Too early, too late, out of order	Stopped Too Soon / Applied too long
Shutdown Cmd	**Controller does not provide Shutdown Cmd when _____**	**Controller provides Shutdown Cmd when _____**	**Controller provides Shutdown Cmd before _____ Controller provides Shutdown Cmd after _____**	**Controller continues providing Shutdown Cmd too long after_____**

Note: This short example is incomplete, for demonstration only!

Unsafe Control Actions



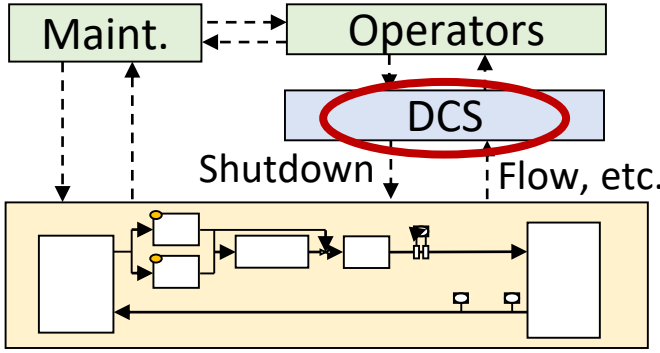
	Not providing causes hazard	Providing causes hazard	Too early, too late, out of order	Stopped Too Soon / Applied too long
Shutdown Cmd	**Controller does not provide Shutdown Cmd when cooling is inadequate* [H-1]**	**Controller provides Shutdown Cmd when cooling is adequate* [H-2]**	**[…]**	**[…]**

Cooling is inadequate* = low pressure OR low flow OR high temp

Note: This short example is incomplete, for demonstration only!

Asking the right questions

Loss: Loss of Mission (unnecessary shutdown)



Question: What DCS control actions can cause unnecessary shutdown?

UCA: DCS provides Shutdown Cmd when cooling is adequate*

Question: What DCS beliefs would cause it to provide Shutdown Cmd when cooling is adequate?

Digital Control System (DCS)

Control algorithm

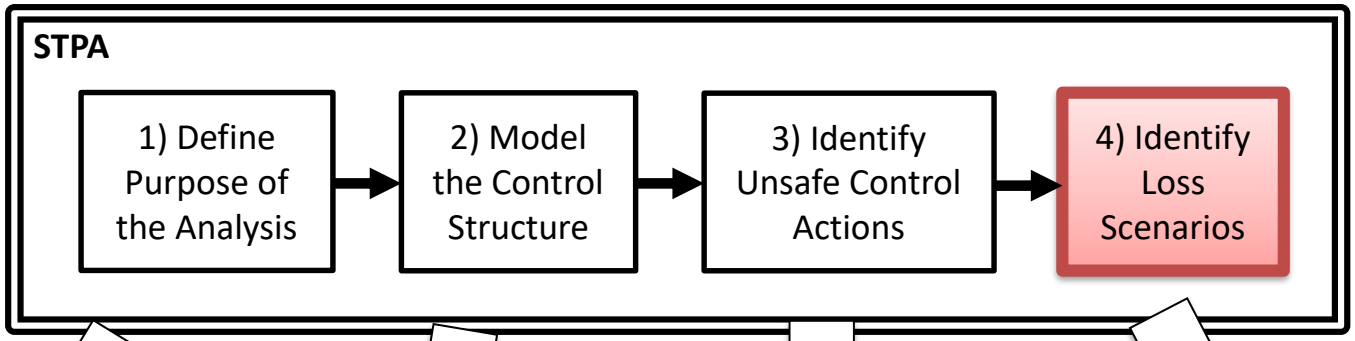
Process Model (beliefs)

Process Model: DCS believes _____

Control Actions

Feedback

Controlled Process

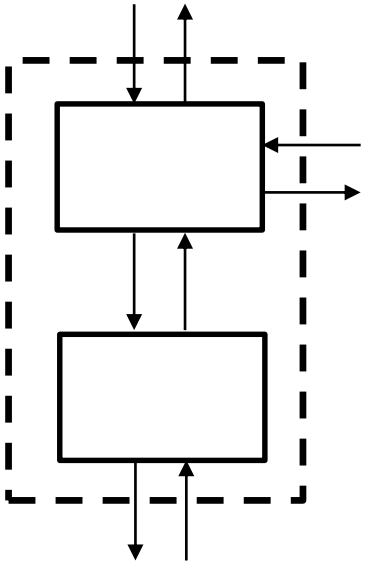


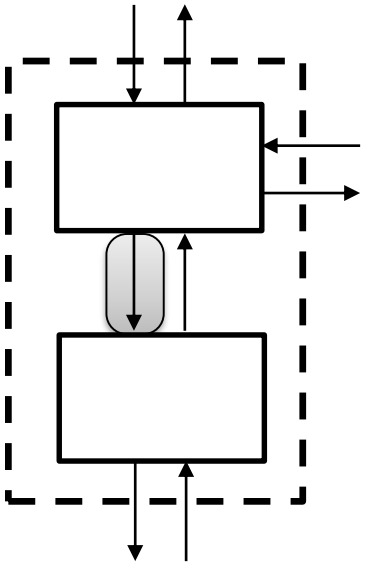
Identify Losses, Hazards

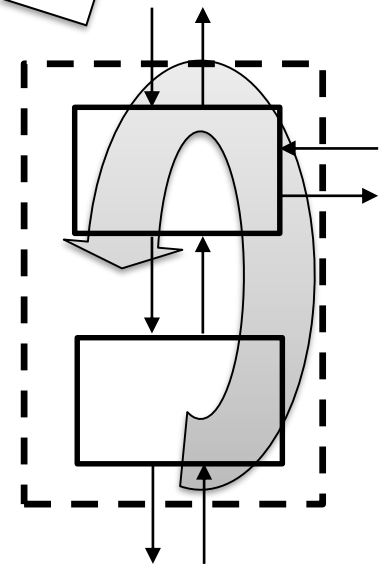
Define System boundary

Environment

System

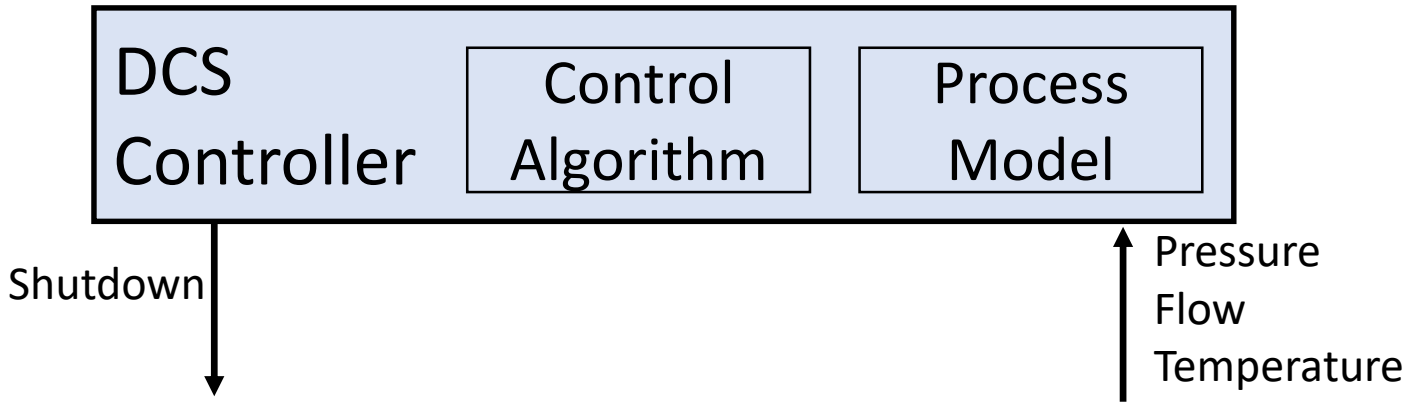






(Leveson and Thomas, 2018)

Controller Analysis (Let's do this together!)



DCS output

UCA-2: DCS provides Shutdown Cmd when cooling is adequate* [H-2]

DCS process model

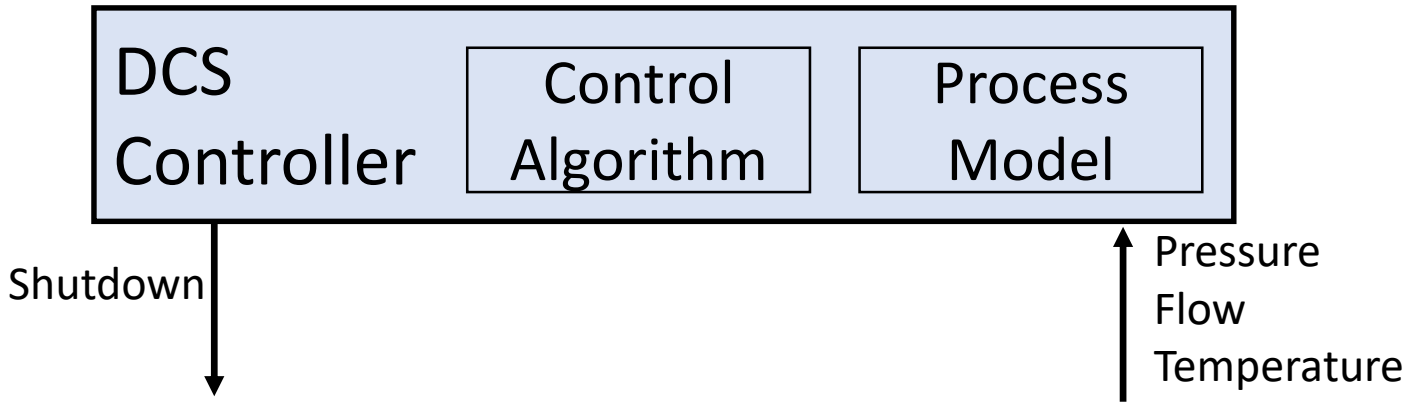
PM-1: Controller believes pressure is too low [UCA-2]

DCS input

F-1: All three pressure sensors report low pressure [PM-1]
< 15.2mA, aka < 45gpm

Note: This short example is incomplete, for demonstration only!

Controller Analysis (Let's do this together!)



DCS output

UCA-2: Controller provides Shutdown Cmd when cooling is adequate* [H-2]

DCS process model

PM-1: Controller believes Pressure is too low

PM-2: Controller believes Flow is too low

PM-3: Controller believes Temperature is too high

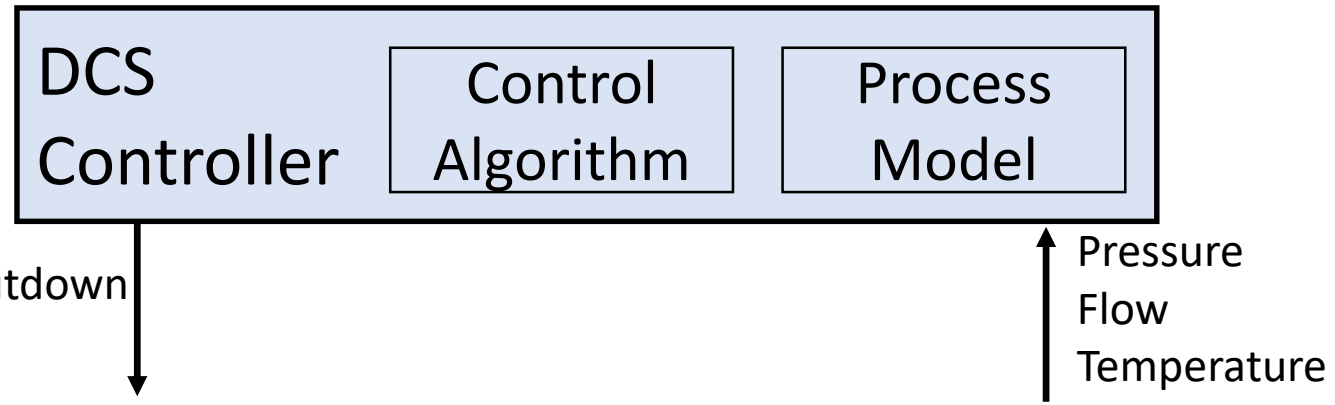
PM-4: Controller believes all three Flow sensors are faulted

DCS input

F-1: ?

Note: This short example is incomplete, for demonstration only!

Controller Analysis (Let's do this together!)



DCS output

UCA-2: Controller provides Shutdown Cmd when cooling is adequate* [H-2]

DCS process model

PM-1: Controller believes Pressure is too low

PM-2: Controller believes Flow is too low

PM-3: Controller believes Temperature is too low

PM-4: Controller believes all three Flow sensors are faulted

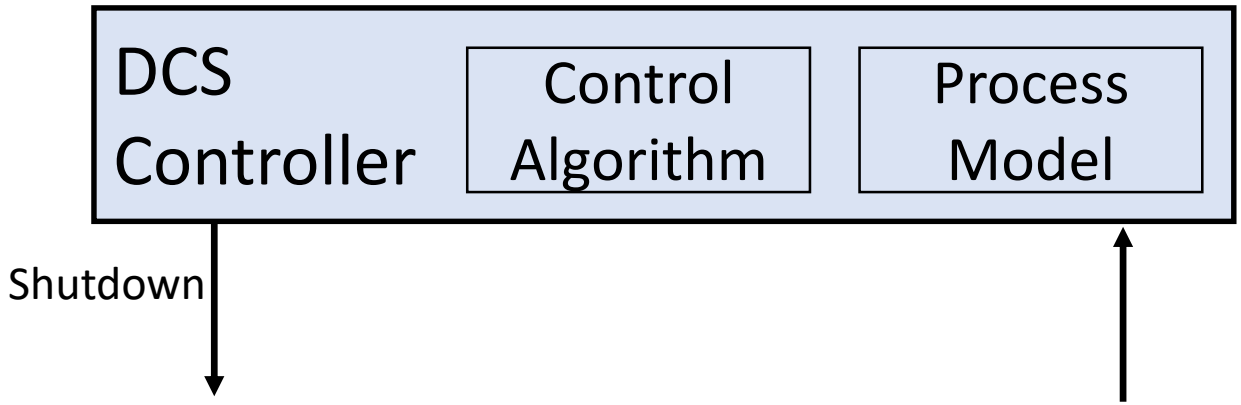
DCS input

F-1: All three flow sensors report out of range low
<3.8mA, aka < 0 gpm

F-2: All three flow sensors report out of range high
>20.38mA, aka ? gpm

Note: This short example is incomplete, for demonstration only!

Controller Analysis (Let's do this together!)



DCS output | **DCS process model** | **DCS input**

UCA-2: Controller provides Shutdown Cmd when cooling is adequate* [H-2]

PM-4: Controller believes all three flow sensors have failed [UCA-2]

F-2: All three flow sensors out of range high [PM-4]
>20.38mA, aka X gpm

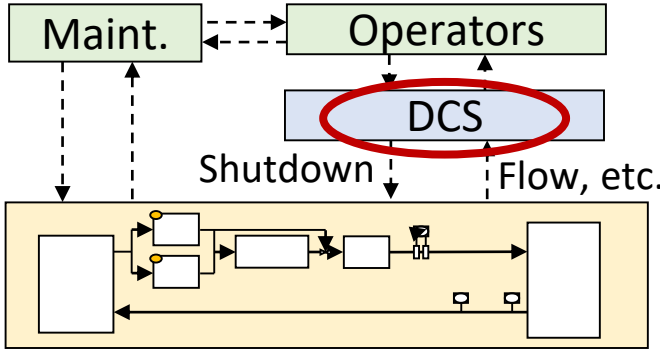
New question: What's X?

(Need to make sure it won't happen inadvertently!)

Note: This short example is incomplete, for demonstration only!

Asking the right questions

Loss: Loss of Mission (unnecessary shutdown)



Question: What DCS control actions can cause unnecessary shutdown?

UCA: DCS provides Shutdown Cmd when cooling is adequate*

Question: What DCS beliefs would cause it to provide Shutdown Cmd when cooling is adequate?

PM: DCS believes all flow sensors are faulted

Question: What DCS inputs would cause DCS to incorrectly believe all flow sensors are faulted?

FB: All Flow Sensors report > 20.32mA (X gpm)

Question: What would cause all flow sensors > X gpm when cooling is adequate?

CP: _____

Digital Control System (DCS)

Control algorithm

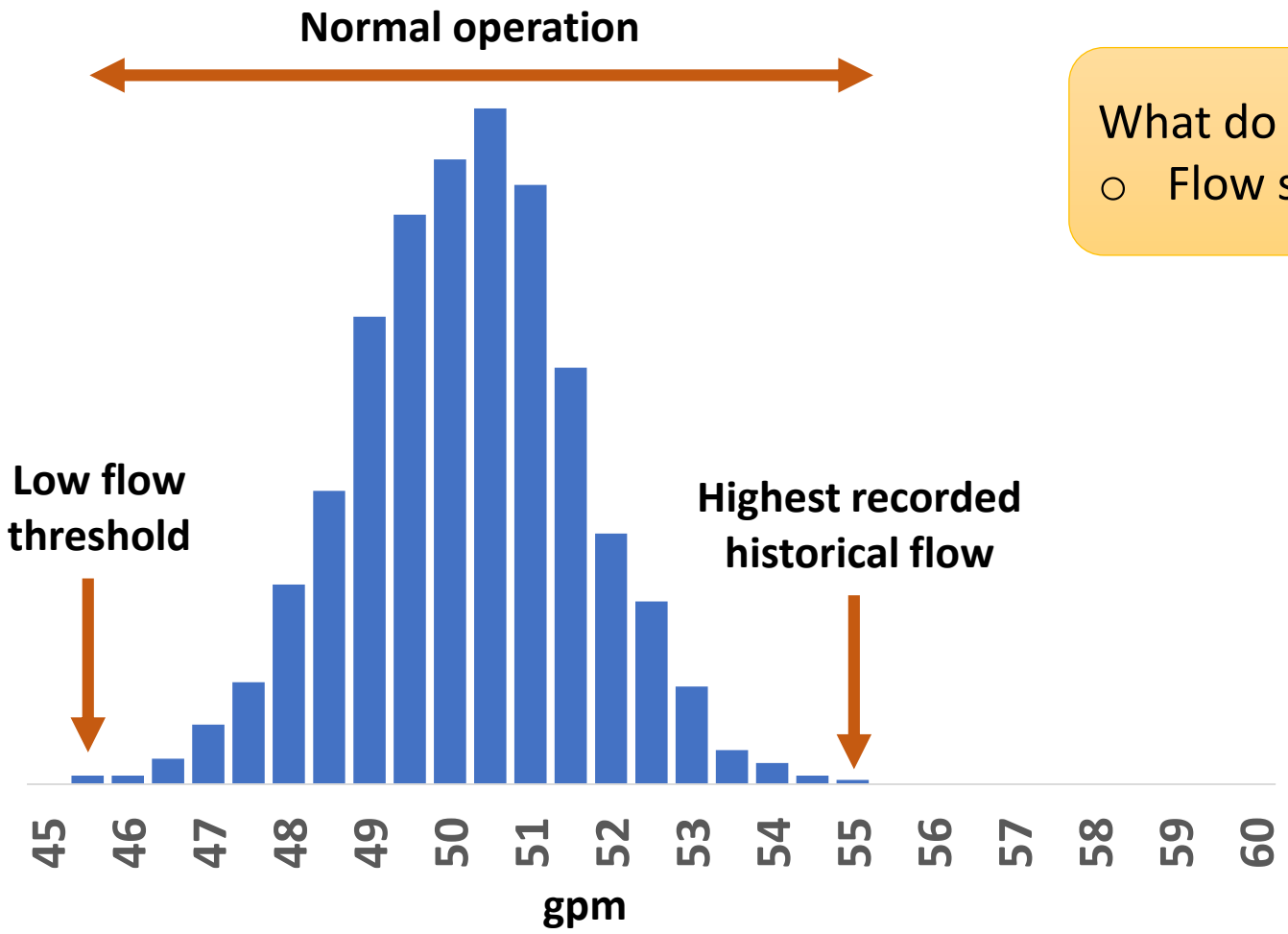
Process Model (beliefs)

Control Actions

Feedback

Controlled Process

What is the flow sensor max range?
Answer based on historical data:

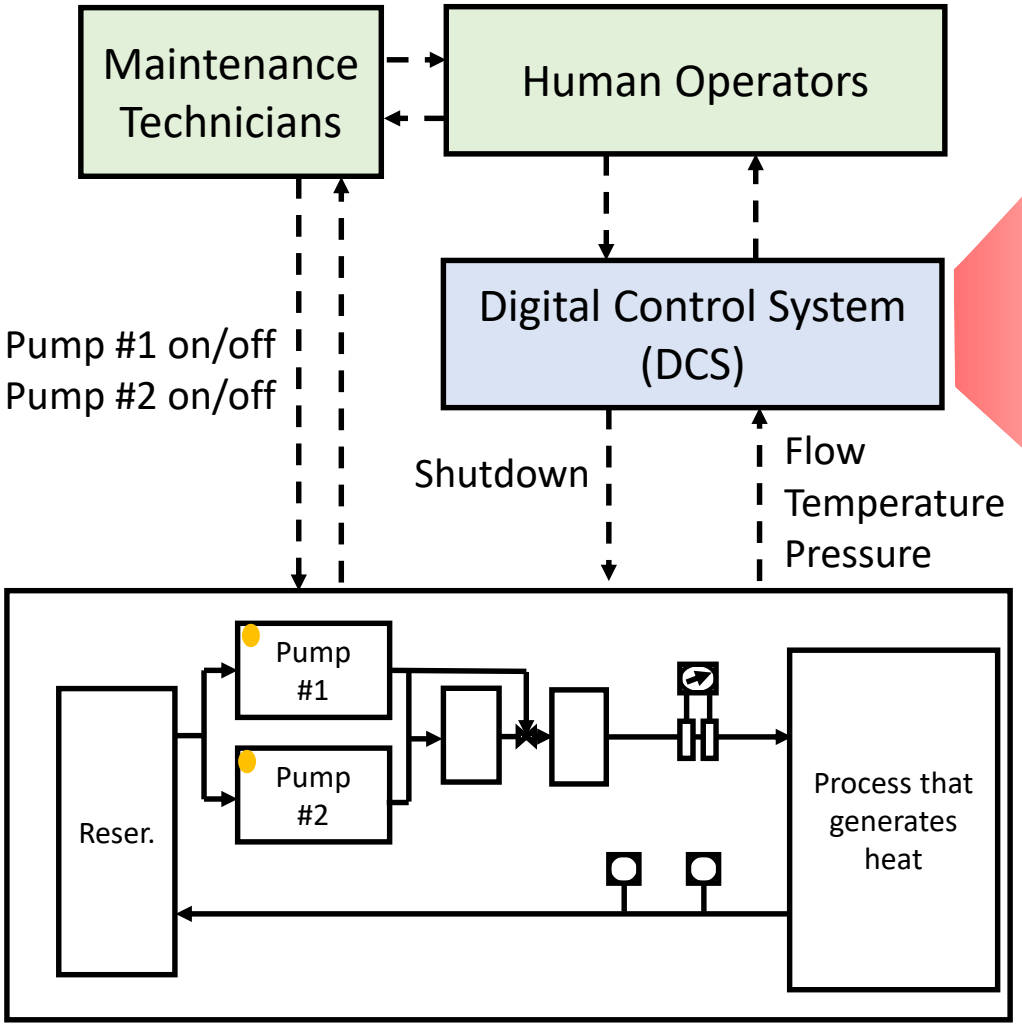


What do you think they chose?

- Flow sensor max: ? gpm

Historical flow data (sampled regularly over many years)

Faulted Instrument Detection

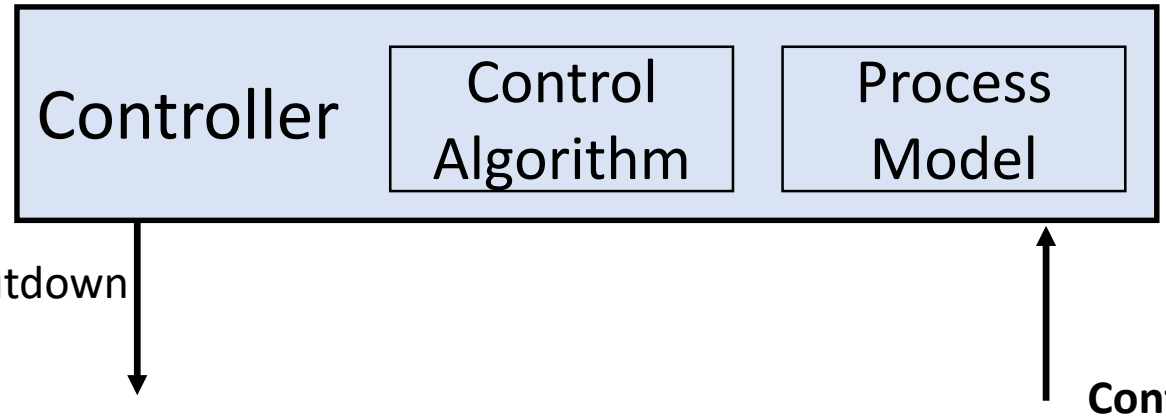


DCS requirements for faulted instrument detection:

Shall detect faulted instrument when signal is:

- <3.8 mA (<0gpm)
- >20.32 mA (>60gpm)

Controller Analysis (Let's do this together!)



Controller output

UCA-2: Controller provides Shutdown Cmd when cooling is adequate [H-2]

Controller process model

PM-4: Controller believes all 3 flow sensors are faulted

Controller input

F-3: all 3 flow sensors indicate high out of range [>20mA; 60gpm]

- **This will occur any time both pumps are running (planned every 9 months)**

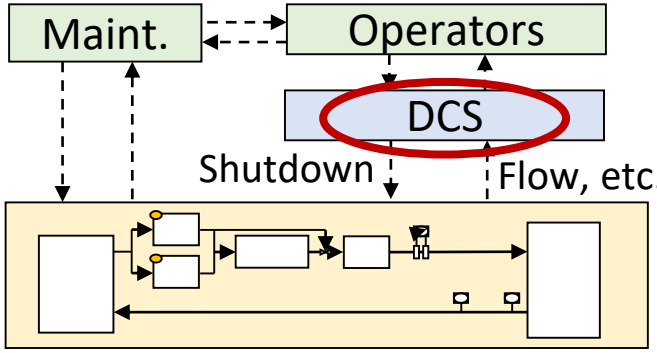
Aha! The design is flawed—components interacting exactly as designed will inadvertently shutdown the system!

This will occur even if all component requirements are met, no components fail, and all human procedures are followed!

Note: This short example is incomplete, for demonstration only!

Asking the right questions

Loss: Loss of Mission (unnecessary shutdown)



Question: What DCS control actions can cause unnecessary shutdown?

UCA: DCS provides Shutdown Cmd when cooling is adequate*

Question: What DCS beliefs would cause it to provide Shutdown Cmd when cooling is adequate?

AHA!

This will occur any time both pumps are running (planned every 9 months)

Digital Control System (DCS)

Control algorithm

Process Model (beliefs)

PM: DCS believes all flow sensors are faulted

Question: What DCS inputs would cause DCS to incorrectly believe all flow sensors are faulted?

FB: All Flow Sensors report >20.32mA, aka >60 gpm

Control Actions

Feedback

Question: What would cause all flow sensors > 60 gpm when cooling is adequate?

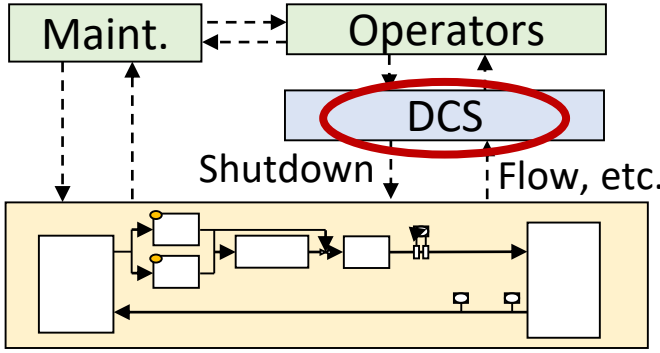
Controlled Process

CP: Both pumps turned on together (maintenance activity)

John Thomas, 2020

Asking the right questions

Loss: Loss of Mission (unnecessary shutdown)



Question: What DCS control actions can cause unnecessary shutdown?

UCA: DCS provides Shutdown Cmd when cooling is adequate*

Question: What DCS beliefs would cause it to provide Shutdown Cmd when cooling is adequate?

PM: DCS believes all flow sensors are faulted

Question: What DCS inputs would cause DCS to incorrectly believe all flow sensors are faulted?

FB: All Flow Sensors report >20.32mA, aka >60 gpm

Question: What would cause all flow sensors > 60 gpm when cooling is adequate?

CP: Both pumps turned on together (maintenance activity)

New upgrade: Expect ~$1m loss every 9 months <u>with no component failures!</u>

Digital Control System (DCS)

Control algorithm

Process Model (beliefs)

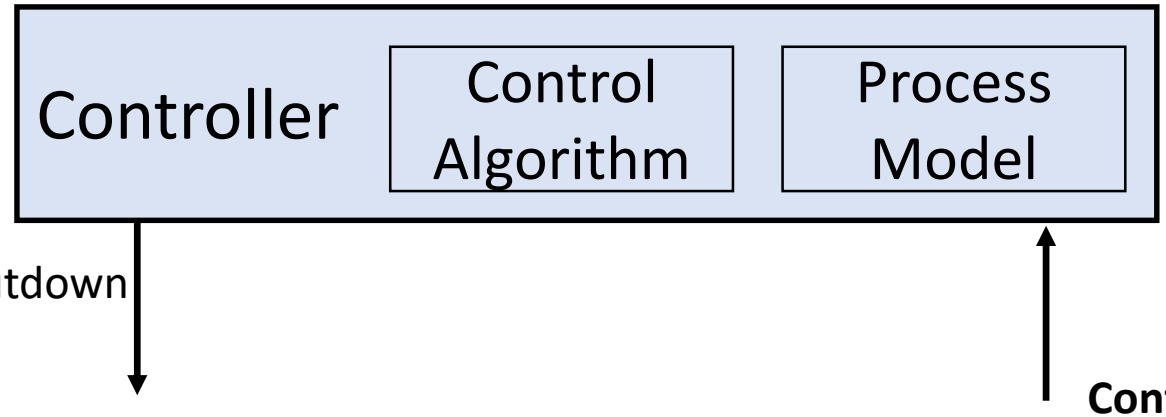
Control Actions

Feedback

Controlled Process

John Thomas, 2020

Controller Analysis (Let's do this together!)



Controller output

UCA-2: Controller provides Shutdown Cmd when cooling is adequate [H-2]

Controller process model

PM-4: Controller believes all 3 flow sensors are faulted

Controller input

F-3: all 3 flow sensors indicate high out of range [>20mA; 60gpm]

- **This will occur any time both pumps are running (planned every 9 months)**

Design solutions?

New requirements?

Maintenance, operator solutions?

Note: This short example is incomplete, for demonstration only!

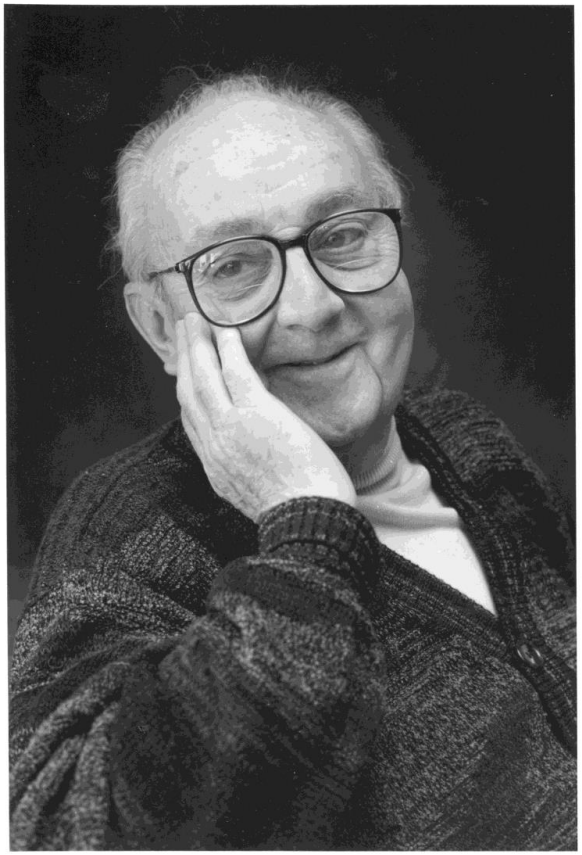
Hindsight
Why did others miss this?

- These analyses (HAZOP, FMEA, FTA, STPA) were all done blind—without any knowledge of the flaw—by separate teams
- In hindsight and after an incident, we may all find a place to update an old analysis (HAZOP, FMEA, FTA, STPA) to include a newly discovered flaw
- That is not sufficient!
- The challenge we have is NOT whether we can correct omissions in hindsight with full knowledge of the flaws!
- The real challenge—and the real test of any method—is whether the technique consistently discovers these flaws **before an incident**, and before we have the benefit of hindsight!

System Models

- All system models emphasize certain information, abstract away other info. All are “wrong”.

- But are they useful?



STPA Consistency

This study has been replicated 10 times, same result*

- STPA Team Sizes
  - Ranging from 1 to 4+ people
- STPA Team Backgrounds
  - All teams were blind to the flaw and incident
  - No prior STPA experience
  - Professionals from process industry
  - Professionals from non-process industries
  - Students with no professional experience
- STPA Facilitator
  - Options tested: Totally blind; partially blind; no facilitator (instead more STPA training)
- STPA Training Time
  - Ranging from 2hrs to 30hrs
- STPA Analysis Time
  - Ranging from 30 minutes to 8 hours
- Materials provided to STPA Teams
  - General description of system as available before incidents
  - No knowledge or info about the flaws or the incidents

All teams successfully used STPA to discover the exact flaw and scenario!

*Plus 100s of studies using different examples

Analysis Method Comparison

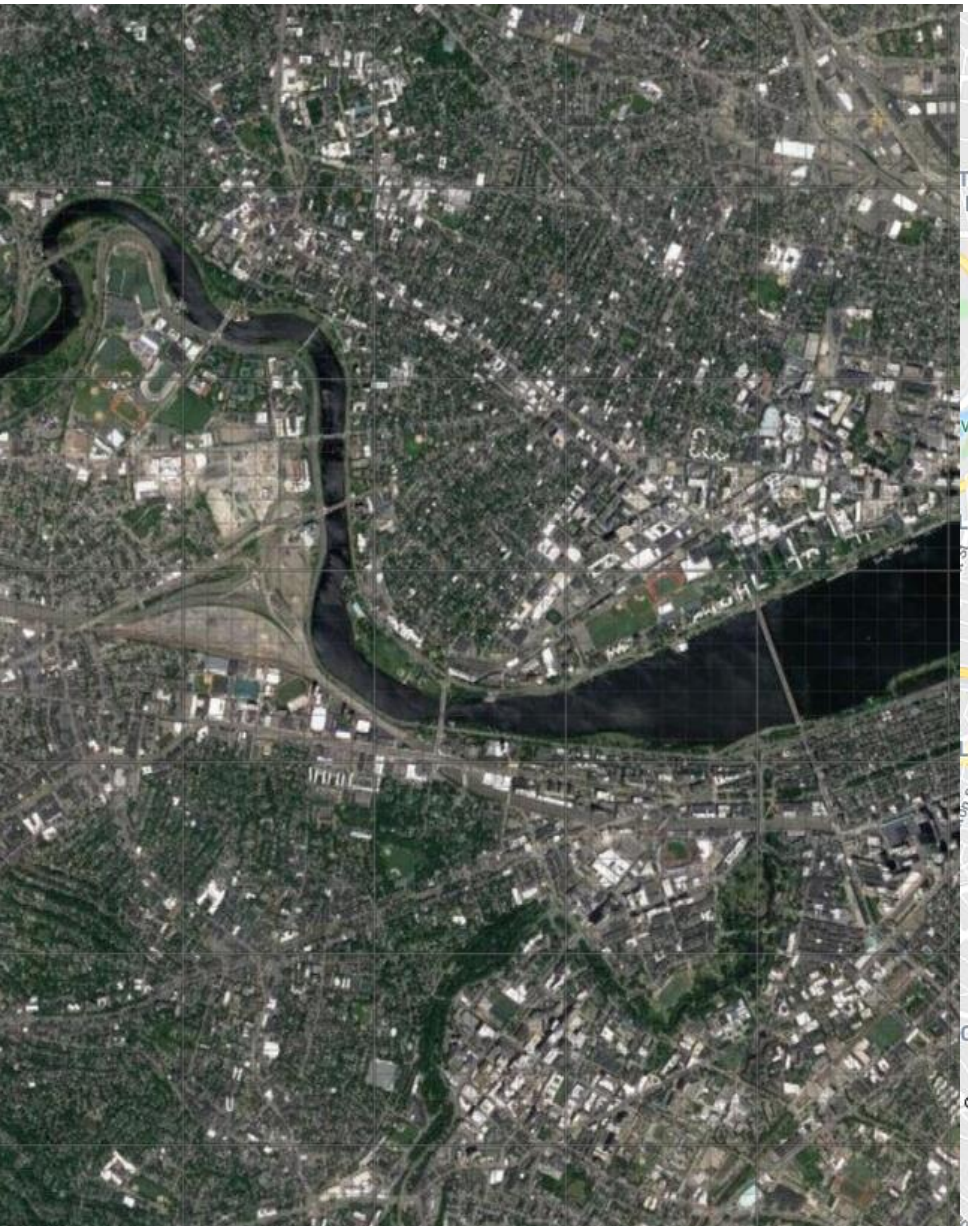
- STPA*
  - **Time:** 1 to 32 person-hours
  - **Team:** STPA novices (but trained by STPA expert trainer/facilitator)
  - **Results:** All seven independent STPA teams anticipated the flaw and the incident scenario
- HAZOP*
  - **Time:** 120 person-hours
  - **Team:** Professional HAZOP practitioners from process industry
  - **Results:** HAZOP did not anticipate the flaw
- FMEA*
  - **Time:** 1,000+ person hours
  - **Team:** Industry professionals with FMEA training and prior FMEA experience
  - Note: Considered many more design details than other teams in this study
  - **Results:** FMEA did not anticipate the flaw
- Fault Tree Analysis (FTA)*
  - **Time: [**Not Available]
  - **Team:** Industry FTA professionals with decades of experience and specialization in FTA development
  - **Results:** FTA did not anticipate the flaw

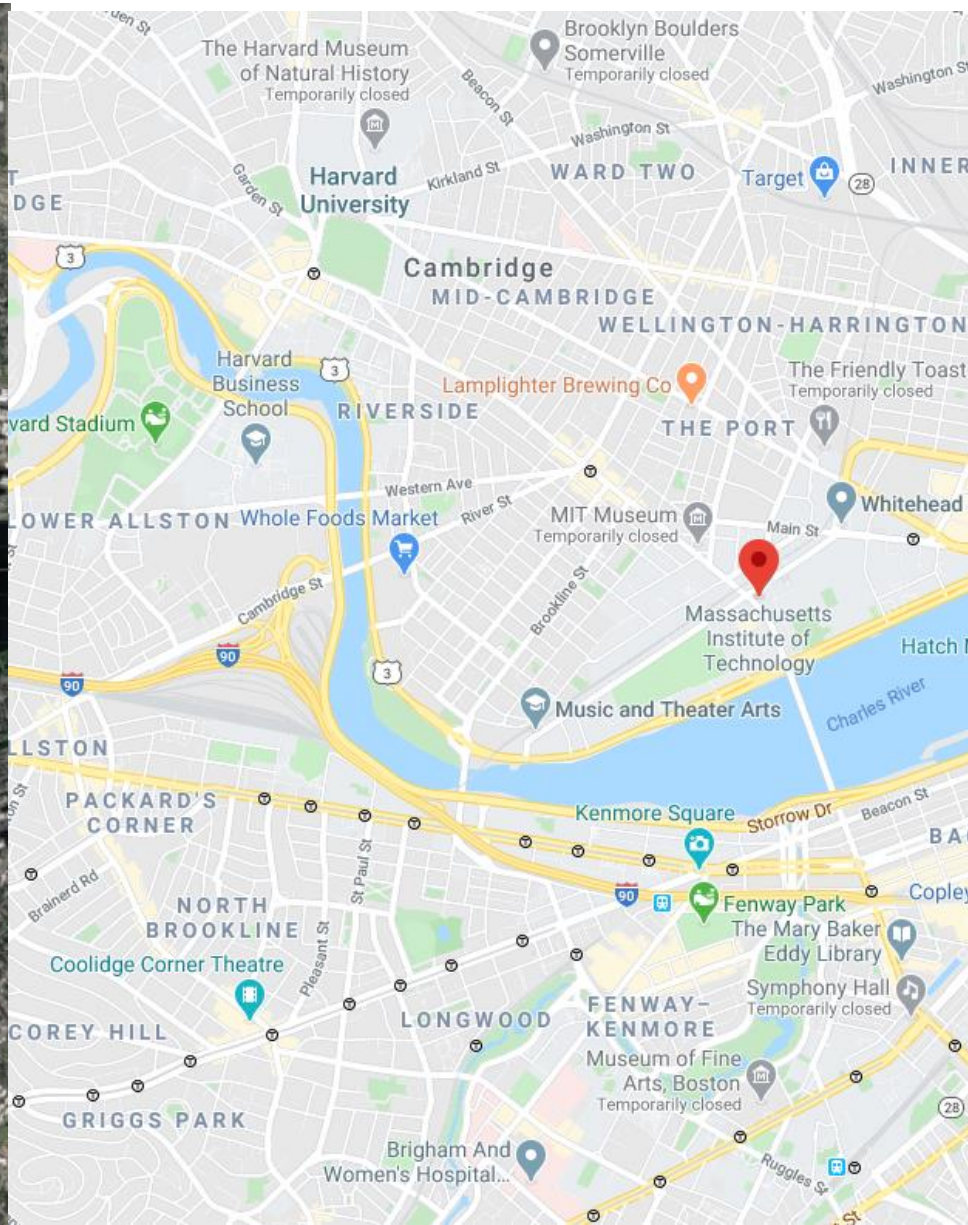
*These results only reflect one data point and are not intended as recommendations. For example, it is NOT recommended to limit STPA to 32 person-hours.

Reality

Abstracting

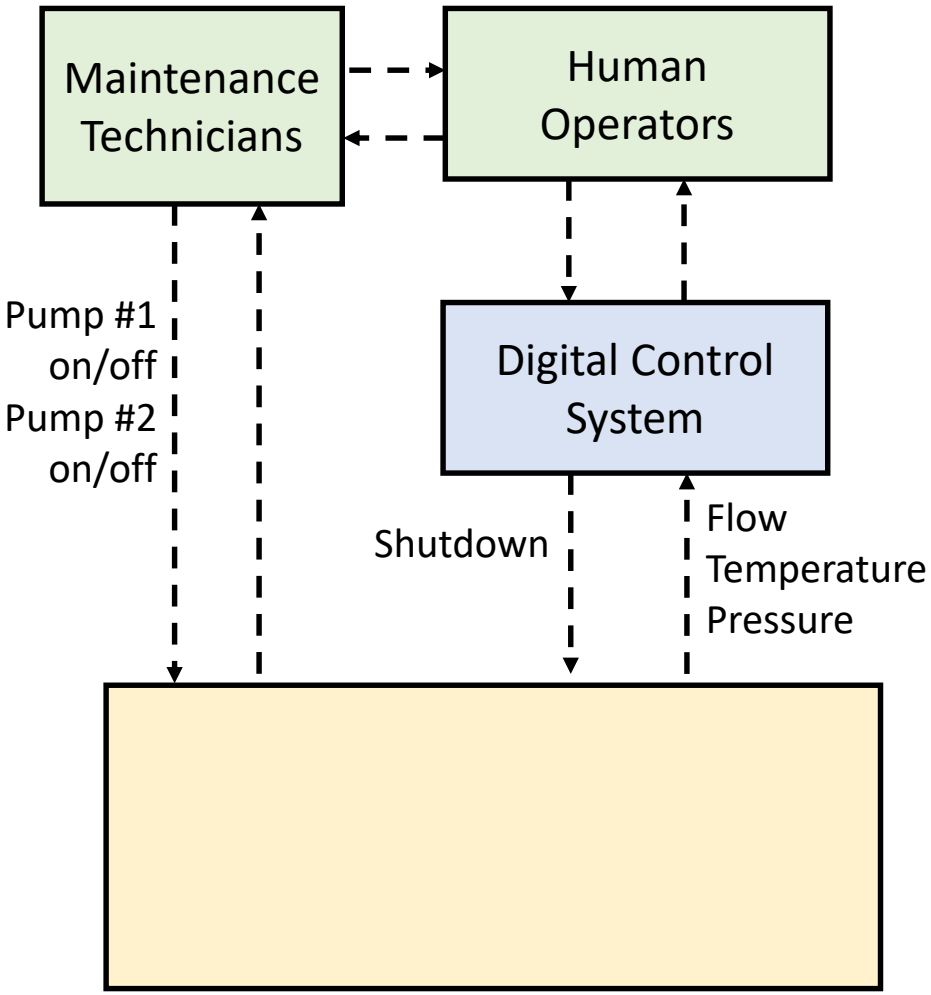
Model



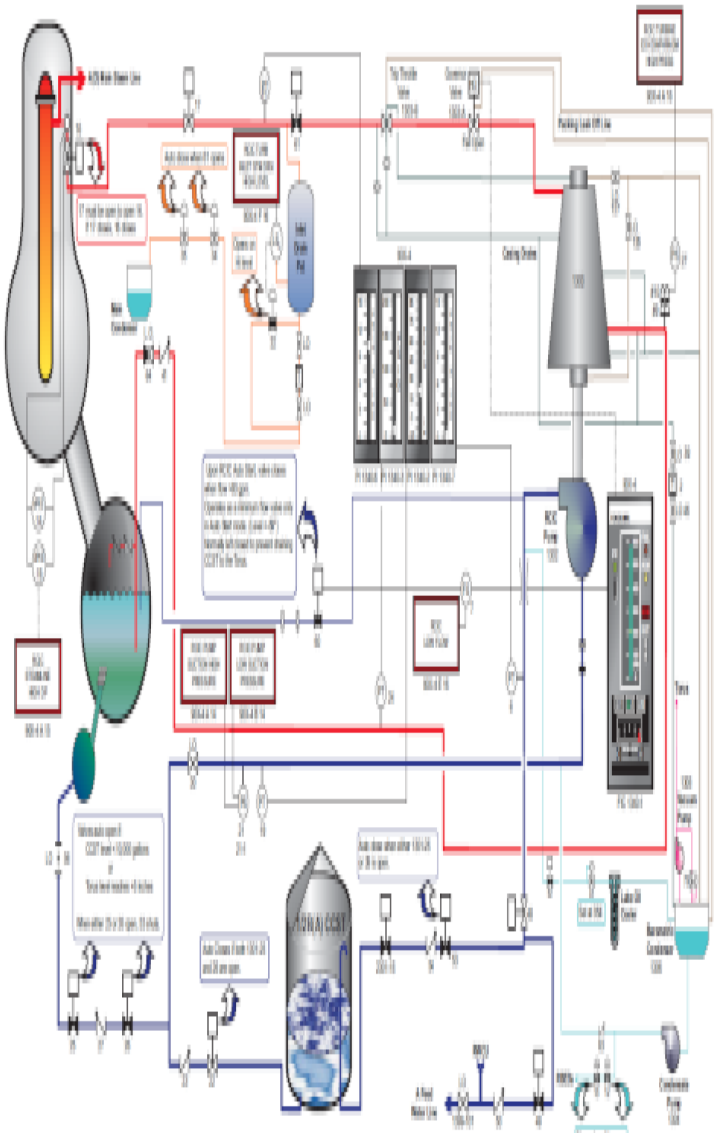


System Models

Control Structure



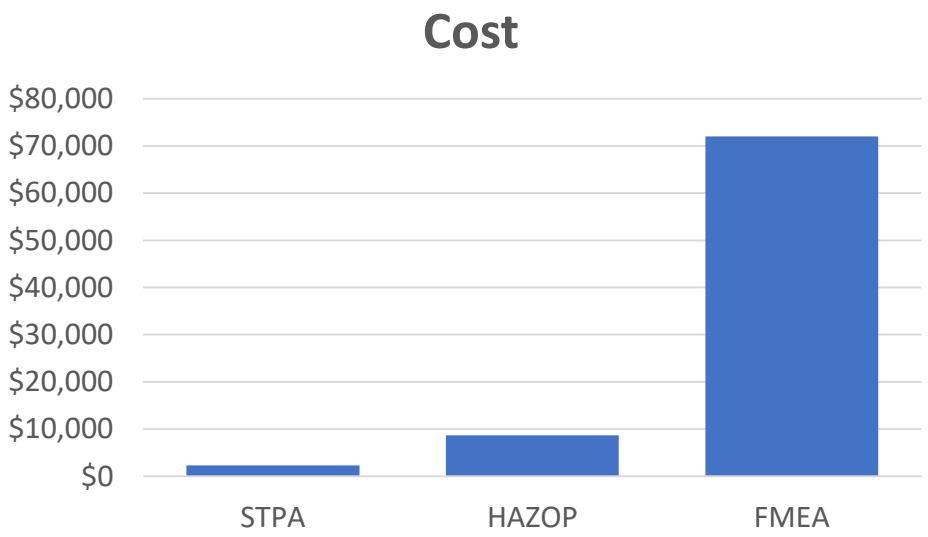
P&ID



Thomas, 2020

Return on Investment (ROI)

- Let's assume $75/hr labor
- STPA cost for this application: $2,400
- Value added: Identifying and addressing the $1m design flaw
- ROI*: $1m / $2,400 = 416



*These results only reflect one data point—this specific project. Further work is underway to explore the generalizability of these results.

Sample of STPA use worldwide, all industries (incomplete, <10%)

Massachusetts Institute of Technology

STPA: The most popular approach you haven't tried?

Countries:

Argentina
Australia
Austria
Belgium
Brazil
Canada
China
Cyprus
Czech Republic
Denmark
England
Estonia
Finland
France
Germany
Greece
Hong Kong
Iceland
India
Ireland
Israel
Italy
Japan
Kenya
Korea
Kosovo
Kuwait
Malaysia
Mexico
Nepal
Netherlands
New Zealand
Nigeria
Norway
Pakistan
Poland
Portugal
Saudi Arabia
Scotland
Serbia
Singapore
South Korea
Spain
Sverige
Sweden
Switzerland
Taiwan
Thailand
Turkey
UK
United Arab Emirates (UAE)
USA

Industries:

Academia
Accelerator Engineering
Accelerator-based research
Accident investigation
Aeronautics
Aerospace
Agriculture
Air Force
Air Traffic Control
Air Transportation
Aircraft
Analytics and Simulation
Automation
Automotive
Aviation
BioPharmaceutical
Chemical
Civil Engineering
Clinical Research
Cloud Computing
Collegiate Sports
Communication
Computer Science
Computing
Construction
Consulting
Consumer Goods
Consumer Products
Content Delivery Network (CDN)
Critical Infrastructure
Critical Infrastructures
Cyber operations
Cybersecurity
Dam Safety
Decision Analysis
Defense
Disaster Risk Management
Diving and Hyperbarics
Education
Electric Power
Electrical & Computer Engineering
Elevator industry
Embedded Software Testing
Energy
Engineering Services
Enterprise Software
Entertainment
Environmental
Ergonomics
Fertilizer Manufacturing
FFRDC
Financial
Firefighting
Fitness
Food
Food processing
Gas
Government
Grid Energy Storage
Ground Combat Systems (Live Fire)
Healthcare
Higher Education
Home Appliances
Hospitals
Human Factors
Hydropower
Industrial
Industrial Automation
Industrial Control
Industrial equipment
Information security
Information Technology (IT)
Infrastructure
Insurance
Internet
Internet of Things (IoT)
IV&V
Labor
Labor Organization
Labor Unions
Life sciences R&D
Logistics
Logistics and Aviation
Manufacturing
Manufacturing Process
Automation
Maritime
Medical
Medical Devices
Medicine
Metals
Military
Military
Acquisition
Military Aviation
Military Defense
Mining
National Security
Natural disasters
Naval
News
Non-profit R&D
Nuclear
Nuclear Energy
Nuclear enginering
Nuclear Power
Nuclear Utility
Nuclear Weapon
Surety
Oil
Oil & gas
Open Standards
Open Systems
Oversight
Particle Accelerators
Patient Safety
Petrochemical
Petroleum
Pipelines
Pharmaceutical (clinical)
Pharmaceuticals
Power
PRA consultants
Private Investigations
Process
Process industry
Processing
Public Sector
R&D
Rail Traffic Control and Safety
Railroads
Real estate
Refining
Regs
Research
Road Traffic Management
Road transport
Robotics
Rotating Equipment
Safety
Safety Assurance
Safety Consulting
Safety engineering
Safety Management
Satellite Operator
Security
Sediment Management
Semiconductor
Ship Design
Shipbuidling
Shipping
Software
Space
Steel
Structural engineering
Supply Chain Management
Surface Transportation System
Engineering
System Safety
Systems Engineering
Telecoms
Test and eval
Think tank
Trade Association
Traffic Control and Safety
Training
Transportation
Turnaround & Innovation
Consulting
University
Videographer
Web development
Web provider
Web standards

STPA in Industry Standards

- ISO/PAS 21448: SOTIF: Safety of the Intended Functionality
  - STPA used assess safety of automotive systems
- ASTM WK60748
  - “Standard Guide for Application of STPA to Aircraft”
- SAE AIR6913
  - “Using STPA during Development and Safety Assessment of Civil Aircraft”
- RTCA DO-356A
  - “Airworthiness Security Methods and Considerations”
  - STPA-sec used for cybersecurity of digital systems
- IEC 63187
  - “Functional safety - Framework for safety critical E/E/PE systems for defence industry applications”
- SAE J3187
  - “Recommended Practice for STPA in Automotive Safety Critical Systems”
- EPRI 3002012755
  - Recommends STPA for digital I&C
- NIST SP800-160 Vol2
  - “Developing Cyber Resilient Systems: A Systems Security Engineering Approach”
  - “Attack scenarios can be represented as part of a model-based engineering effort […] based on identification of loss scenarios from System-Theoretic Process Analysis (STPA).

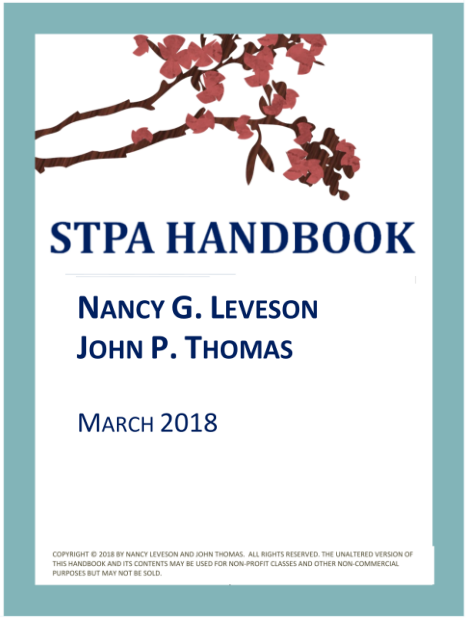
STPA Common Mistakes

- Not getting properly educated in STPA
  - A short presentation is not enough!
  - Formal training needed
- Implementing STPA without an expert STPA facilitator
  - Example mistake: We already have a facilitator with decades of experience facilitating fault tree analysis. Just give us a couple days to "bring him up to speed on the STPA methodology".
- Limiting STPA to a simple system or simple problem with obvious answers

Massachusetts Institute of Technology

For more information

- Google: “STPA Handbook”
  - How-to guide for practitioners applying STPA
  - Free PDF
  - Same book used in our professional/industry STPA training classes
- Website: mit.edu/psas
- Email: JTHOMAS4@MIT.EDU



Free PDF

VIRTUAL BUTTON AND HUMAN MACHINE INTERFACE (HMI) SYSTEM SAFETY EVALUATION USING STPA

Mark Vernacchia,
Technical Fellow, Principal System Safety Engineer for all propulsion systems,
General Motors worldwide

WMG, THE UNIVERSITY OF WARWICK

21-22 OCTOBER 2020

Virtual Button and Graphical Interface System Safety Evaluation Using STPA

Mark A. Vernacchia
GM Technical Fellow, Principal System Safety Engineer – Propulsion Systems

Jesse Johnston
System Safety Engineer – Non-Motion Controls

Global Safety, Systems, & Integration
General Motors Company

European STPA Conference

October 22, 2020

Introduction

- Examine how STPA can be used to explore safety concerns associated with interactions between human operators and virtual controls
- Inclusion of humans as control structure elements enables representation as a “human controller”
- Discuss how to organize STPA generated system safety requirements, and how these requirements can be documented and used by system engineers
- Touch on organizational challenges for STPA

Overview of Topics

- What Are “Virtual Controls?” Why Popular?
- Mechanical/Electrical vs. Virtual Control Interface Differences
- STPA Related Activities:
  - Early use of STPA in Concept Stage is important
  - Deciding if a virtual control is safety critical
  - Use of “human controller” aspects Mental Models
    - Process State
    - Process Behavior
    - Environment

Overview of Topics

- Virtual Control Example Evaluation Using STPA
  - Operating Conditions
  - Control Structure and Unsafe Control Actions (UCAs)
  - Causal Scenarios and Requirements Leveraging Mental Models
- Lessons to Share

What Are "Virtual Controls?" Why Are They Becoming Popular?

- "Virtual controls" are controls that do not require physical actuation of a moving part
  - Touch (*focus for this presentation*)
  - Voice
  - Gestures
- Switches, buttons, dials, etc. can be "virtualized"
  - Cost savings
  - Greater design freedom
  - Modern approach desired by customers

Virtual Control Examples

- Cell phones – smart phones replaced most physical buttons with virtual controls, with fewer buttons as design evolves
- Home automation – “smart homes” replace lighting, temperature and other controls with virtual controls
- Automotive examples
  - Audio controls
  - Climate controls
  - Lighting controls
  - …with even more on the horizon

Safety Considerations: Benefits of Replacing Physical With Virtual

- Mechanical key ignition switches and electrical start-stop switches have various fault mechanisms
  - Vehicle vibration and frictional forces influence operation
  - Cable connections and wire harness faults occur
- Elimination of key ignition mechanical and electrical components can eliminate these types of problems
- Hardware verification testing reduction opportunity

Differences in Safety Considerations

Physical Controls	vs.	Virtual Controls (touch)
• Switches, buttons, dials, etc.	→	• Rendered graphics
• Wiring	→	• Display + wiring
• Physical location	→	• Screen layout

Common Considerations

- Software control logic
- Accidental or erroneous activation

STPA Related Activities: Early Use of STPA in Concept Stage is Important

- Very beneficial in the early stages of the design as it provides a way to do exploratory analysis when all potential causes / effects of misbehaviors not known
- Enables review of the anticipated operating scenarios for these virtual control devices and facilitates a discussion about better use scenarios
- Facilitates discussion between system safety engineers and system design engineers as to which requirements are safety related versus performance and/or functional related

STPA Related Activities: Deciding if a Virtual Control is Safety Critical

- ISO-26262 (Automotive Safety Standard)
  - Provides guidance to determine requirements that prevent or manage potential hazards so that the system is “free from unreasonable risk”
  - Focuses on harm to humans (a.k.a. STPA “losses”)
- Preliminary Hazard Analysis
  - HAZOP (Hazard and Operability) – performed to identify potential hazards that might lead to accidents (harm)
  - HARA (Hazard Analysis and Risk Assessment) – analyzes severity, exposure and controllability of hazards, allowing assignment of Automotive Safety Integrity Levels (ASIL)

STPA Related Activities: "Human Controller" Aspects in STPA Evaluation

- Referencing work by Megan France and John Thomas to write scenarios for actions performed by the human operator, using "Human Controller Model" construct

- Human Controller Model focuses on three aspects:
  - The controller's (driver or occupant) goals & how they make decisions based on what they expect
  - The flaws in how a human controller thinks about system and its environment
  - The influence of human experiences and the expectations related to processing sensory feedback/input

Byton M-Byte

Weltmeister EX5

NIO ES8

Tesla “Dog Mode”

Byton M-Byte

Weltmeister EX5

Tesla “Dog Mode”

Let’s look at an example where we may want to replace a Key Ignition or a Start/Stop push button device with a Virtual Control

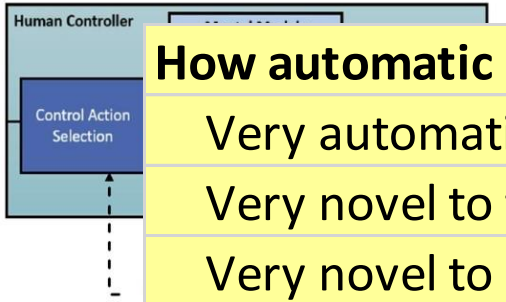
STPA Related Activities: Key or Start/Stop Button Virtualization Example

- Is it safety critical?
  - Yes, it controls the propulsion state of the vehicle (confirm by HAZOP and HARA)

- Operating Contexts:
  - How does driver turn on ignition?
  - How does driver turn off ignition?
  - Impact on living things left in vehicle if driver leaves the vehicle unattended?
  - Entering vehicle, driving vehicle, exiting vehicle, post crash

STPA Human Controller

CONTROL ACTION SELECTION



Control Action Selection

What were operator's goals?

- To drive the vehicle
- To have propulsion become active without physical button

How automatic or novel was the behavior?

- Very automatic to enable propulsion - no driver action
- Very novel to turn vehicle <OFF>
- Very novel to allow vehicle to continue to run <Let Run>

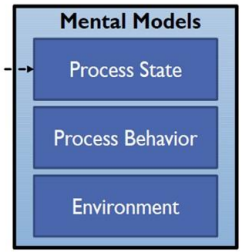
How might operator's mental models affect decisions?

- Mental model of how to start car may impact driver interaction
- MM of how to turn vehicle <OFF> may influence driver interaction
- MM of how to get to <ACC> may influence driver interaction
- MM of how to turn vehicle <OFF> at speed may influence driver interaction

External factors that might affect decision?

- Vehicle automatically goes into propulsion with specific driver action
- Vehicle automatically shuts <OFF> only after driver exits

STPA Human Controller



Mental models are ***partial representations.***

- *Information may be purposefully omitted*
- *"Unknowns" may be known or unknown*
- *Information may be incorrect or outdated*

What does the operator believe about the system?

Mental Model of Process State

- Beliefs about modes and mode changes
- Believes about the current process stage, for processes with multiple stages
- Beliefs about system variables (eg. true/false)

Mental Model - Process State

Beliefs about modes and mode changes

- There will be a button to start or stop the vehicle propulsions system
- There will be a button to stop the engine while driving in emergency
- There will be an accessory button to listen to radio with propulsion not active
- Once <Let Run> option is selected the vehicle will never exit that mode without

multiple stages

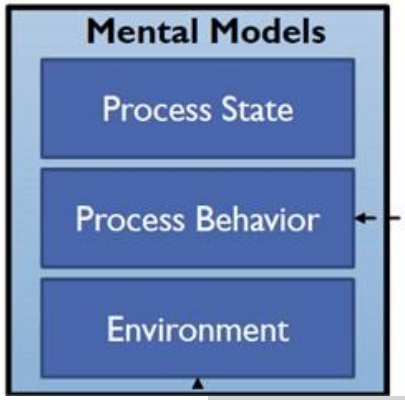
Beliefs about system variables (e.g. true/false)

- Driver believes stepping on the brake pedal will stop the vehicle even with excessive acceleration from propulsion system

There will be button to stop the engine while driving in emergency

System Safety

STPA Human Controller

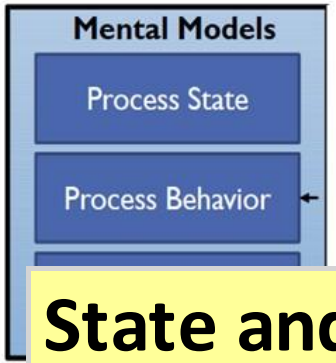


Mental Model of Process Behavior

- Beliefs about what the system can do
- Beliefs about how the system will behave in a particular mode or stage of operation
- Beliefs about if-then relationships between operator input and system output

Mental Model - Process Behavior
Beliefs about what the system can do
Start automatically
Turn <OFF> automatically
Vehicle will not rollaway on an incline if driver exits without turning <OFF>
Vehicle will activate a <Let Run> mode automatically because it has one
Beliefs about if-then relationships between operator input and system output
If the driver performs a specific action the vehicle will turn propulsion <ON>
Performing that specific action after propulsion is <ON> will not do anything

STPA Human Controller



- State and behavior of other controllers
- Social and organizational relationships

Mental Model - Environment
Changes in environmental conditions
Driver entering the vehicle
Driver exiting the vehicle
State and behavior of other controllers
Shift by Wire system is operational and available
A brake override feature is operational and available
Social and organizational relationships
Passenger expectation of vehicle operation

State and behavior of other controllers

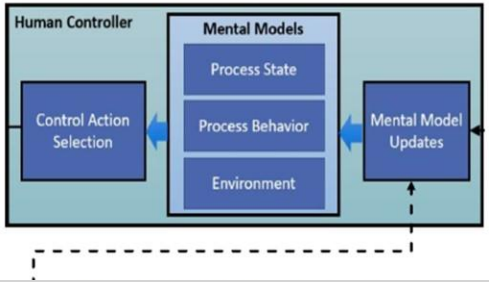
Shift by Wire system is operational and available

A brake override feature is operational and available

System Safety

STPA Human Controller

MENTAL MODEL UPDATES



Consider whether input/feedback was correctly perceived & interpreted

- Feedback presented to driver in clear unambiguous manner
- Feedback clearly shows vehicle operating state
- Clinic data to be gathered regarding feedback method effectiveness

Consider whether input/feedback was observed (salience, expectations)
NONE at this time
Consider whether input/feedback was correctly perceived & interpreted
Feedback presented to driver in clear unambiguous manner
Feedback clearly shows vehicle operating state
Clinic data to be gathered regarding feedback method effectiveness

System Safety

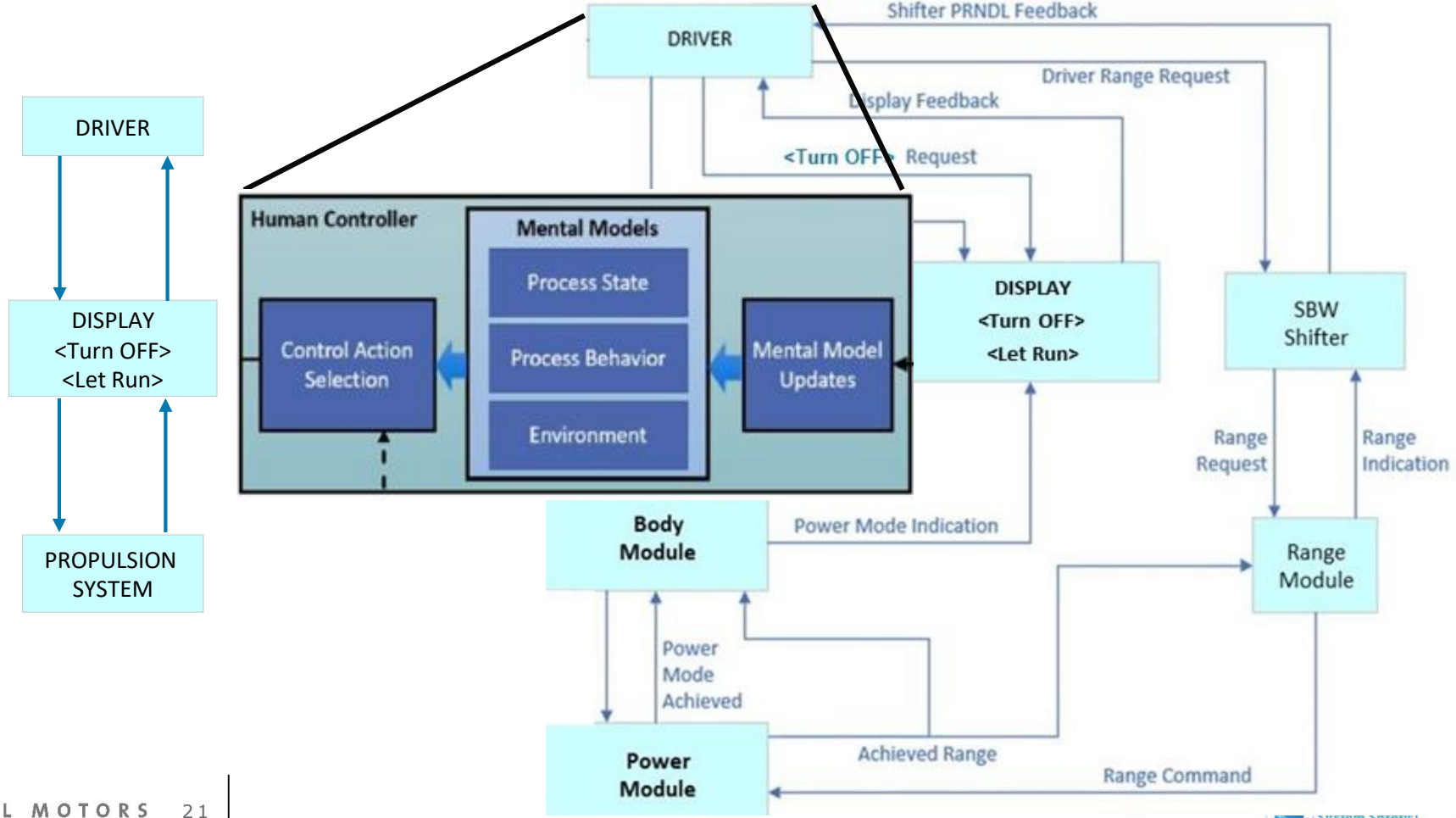
STPA Human Controller – Accidents, Hazards, and Operating Contexts

EXAMPLE ACCIDENTS		
People injured when car collides with obstacle	A1	
People injured when car collides with another car	A2	
People injured when car collides with pedestrian	A3	
People injured when car interior overheats or gets too cold	A4	

EXAMPLE POTENTIAL HAZARDS		
Car rolls away as driver exits	H1	A1, A2, A3
Propulsion cannot be deactivated when experiencing unintended acceleration	H2	A1, A2, A3
Car overheats with occupants inside	H3	A4
Car gets too cold with occupants inside	H4	A4

EXAMPLE OPERATING CONTEXTS / SCENARIOS
While Entering Vehicle
While Seated in Driver Seat
While Exiting Vehicle
While in Gear and Stationary
While in Gear and Moving

STPA Control Structure Example

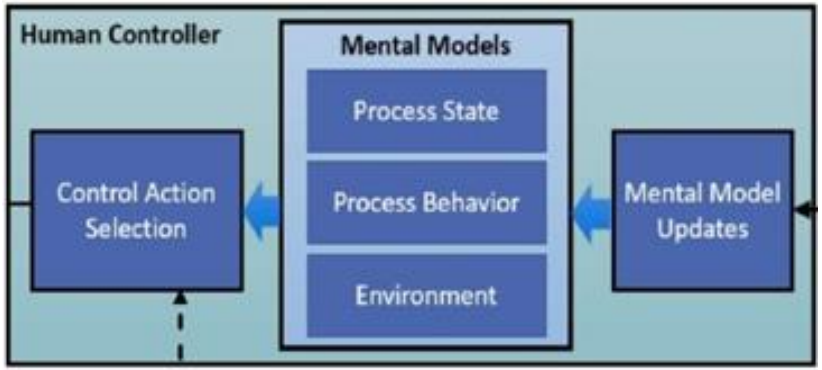


STPA UCA Approach

	"NOT Providing" Cause Hazard	"Providing" Cause Hazard	Incorrect Timing Incorrect Order	Stopped Too Soon Applied Too Long
	Missing Not Followed	Providing When Not Expected Provided More/Less than Required Provided Content Results in Control Conflict	Provided Too Early/Late When Required Provided Before/After When Required Provided Content in Wrong Order Provided Opposite of What Expected	Providing Unstable or Osscilating Content Providing Truncated Content Providing Stuck Content

"NOT Providing" Cause Hazard	"Providing" Cause Hazard	Incorrect Timing Incorrect Order	Stopped Too Soon Applied Too Long
Missing Not Followed	Providing When Not Expected Provided More/Less than Required Provided Content Results in Control Conflict	Provided Too Early/Late When Required Provided Before/After When Required Provided Content in Wrong Order Provided Opposite of What Expected	Providing Unstable or Osscilating Content Providing Truncated Content Providing Stuck Content

STPA Causal Scenario Development Utilizing Mental Model Framework



Control Action Selection

- Replacement of mechanical devices is very novel idea
- Anticipate need for instruction and guidance cues

Process State

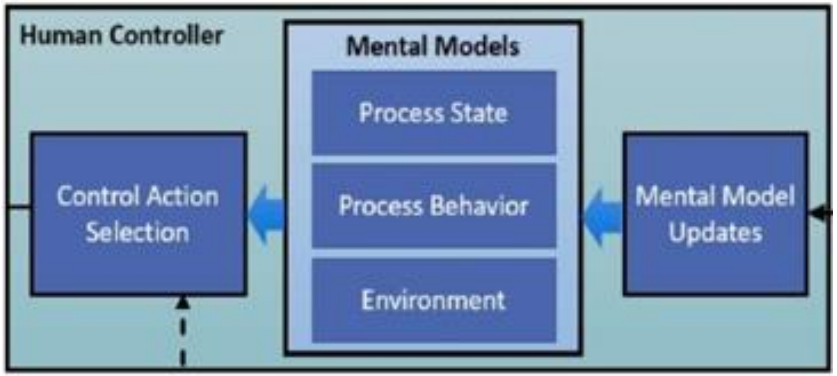
- There will be mechanical means to start / stop
- There is a way to turn propulsion off in emergency

Process Behavior

- The system will start and turn off automatically

System Safety

STPA Causal Scenario Development Utilizing Mental Model Framework



Environment

- How system will behave when driver enters or exits vehicle
- Other vehicle systems (e.g., shift by wire, braking, etc.) operation will not be affected/changed

Metal Model Updates

- Feedback of propulsion state needs to be clear
- Feedback mechanisms evaluated for effectiveness

STPA Example Causal Scenarios and Potential Requirements

	"NOT Providing" Cause Hazard		
Control Action(s)	**Missing Not Followed**		
Requests propulsion <ON> initially	UCA-14: Vehicle propulsion does not turn <ON> when driver wants to begin	CS-6b: Driver thinks propulsion <ON> will occur automatically	SR-24a: Information in the driver's manual shall inform operator of all methods to start and shut down vehicle
			SR-32: System shall display brief summary of operating instructions to the driver when they enter the vehicle (instruction presentation may be deactivated by vehicle settings menu selection)
			SR-22: System shall provide driver notification that brake pedal must be applied to start vehicle

STPA Example Causal Scenarios and Potential Requirements

	"NOT Providing" Cause Hazard	
Control Action(s)	**Missing Not Followed**	
Requests <OFF> while Vehicle is Moving	UCA-8: Driver does not know how to turn propulsion <OFF> while moving [H2]	
	CS-24: System fault keeps <Let Run> mode <OFF>	SR-21a: System shall provide driver notification of power mode status (i.e., Off, Prop, Let Run, etc.)
		SR-21b: System shall provide driver notification of change of power mode status
Requests <Let Run>	UCA-19: <Let Run> mode does not activate when requested [H3] [H4] [H5]	

Lessons to Share From This Example

- STPA evaluation can lead to suggestions to redesign initial propulsion activation strategy
  - Original strategy may have been to wait until the driver makes an initiating action before presenting propulsion activation information
  - How to develop a user interface to guide the driver through “OFF” or “Let Run” options

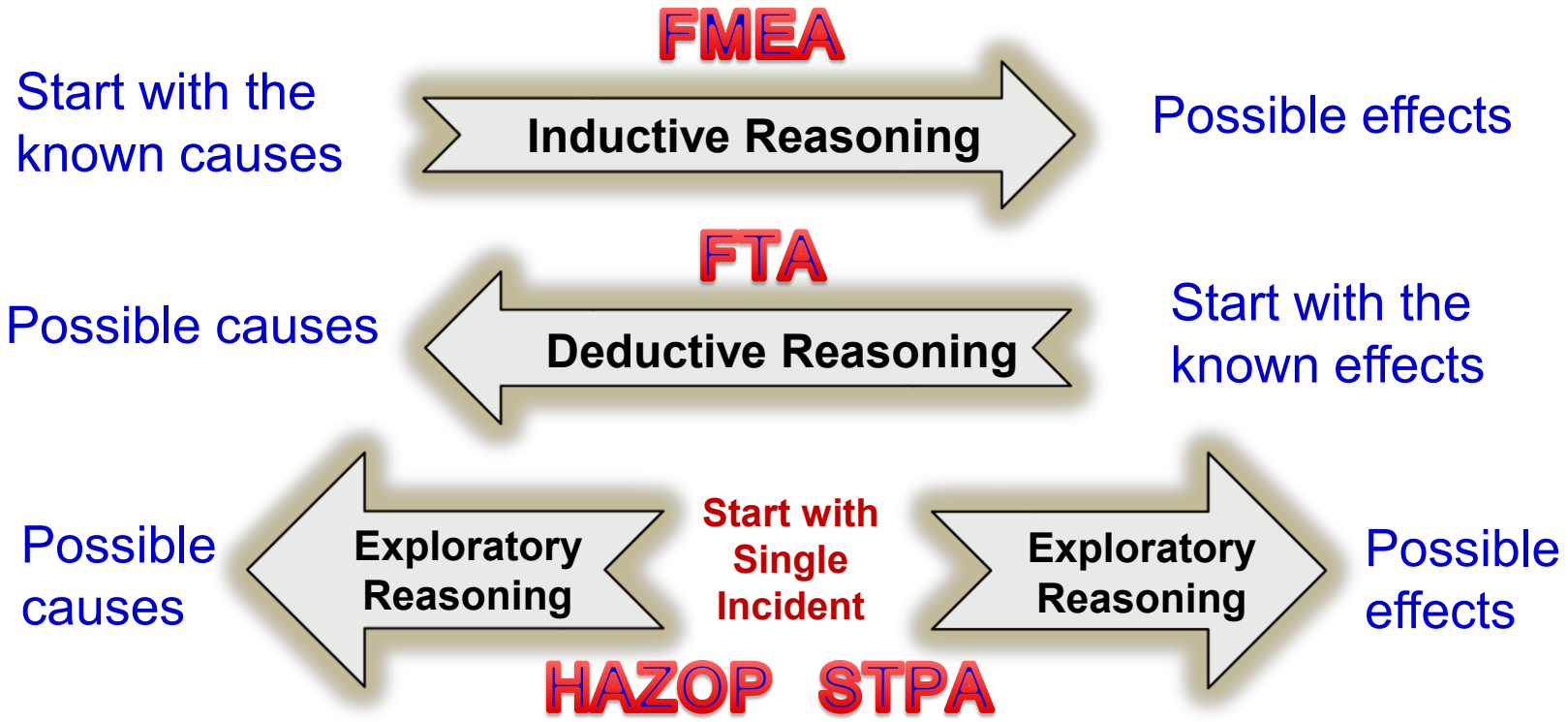
Lessons to Share From This Example

- Assess, and redesign if necessary, the control structure based on potential shortcomings or trade-off study feedback.
  - Requirements to prevent or manage potential hazardous states for driver and occupants due to an erroneous or inadvertent driver action
  - Requirements for shift-by-wire or brake system to secure vehicle upon driver exit can be defined early

Lessons to Share From This Example

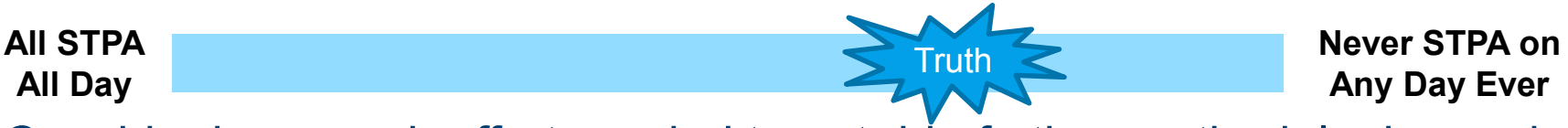
- Joint use of STPA between system safety engineers and system design groups helps to:
  - …think beyond a “failed” component perspective (e.g., FMEA or FTA)
  - …consider a “controls” design perspective and system usage scenarios that could lead to control actions being improperly executed or not executed
  - …evaluate causal scenarios that enable UCAs to occur and prevent or manage these scenarios by defining appropriate requirements

STPA Adoption in Industry – Challenges - Evaluation Methodologies



STPA Adoption in Industry - Challenges

- Comparison of STPA to other safety analysis methodologies
  - Need to choose your "spectrum" philosophy



  - Consider how much effort needed to get rid of other methodologies and if organization would even entertain that idea
  - STPA is an "exploratory" to complement "inductive" (FMEA) and "deductive" (FTA) evaluation methodologies
  - Consider how STPA would/could feed or flow into other methods
  - Emphasize STPA can be very effective in finding missing requirements especially early in concept phase on systems
  - STPA strong for "non-failure" controls-based system design evaluations

STPA Adoption in Industry - Challenges

- Finding an initial application for STPA
  - Learn STPA to a working level
  - Look for an area with the greatest need
  - Propose STPA as an alternative to struggling methodology
    - Used STPA as alternative to a DFMEA effort to deal with human factors
  - Operate below the “radar”
    - Be focused
    - Do not alienate people with grandiose statements
    - Be respectful of people’s concerns and opinions

Virtual Button & Graphical Interface System Safety Evaluation Using STPA

Questions??

CATAPULT High Value Manufacturing

WMG THE UNIVERSITY OF WARWICK

WMG, THE UNIVERSITY OF WARWICK

21-22 OCTOBER 2020

OPENAPS: STPA OF AN OPEN SOURCE MEDICAL DEVICE

Shaun Mooney,
Divisional Manager,
Codethink Ltd, UK

Safety Analysis of Linux Powered Open Source Medical Device

Shaun Mooney
shaun.mooney@codethink.co.uk

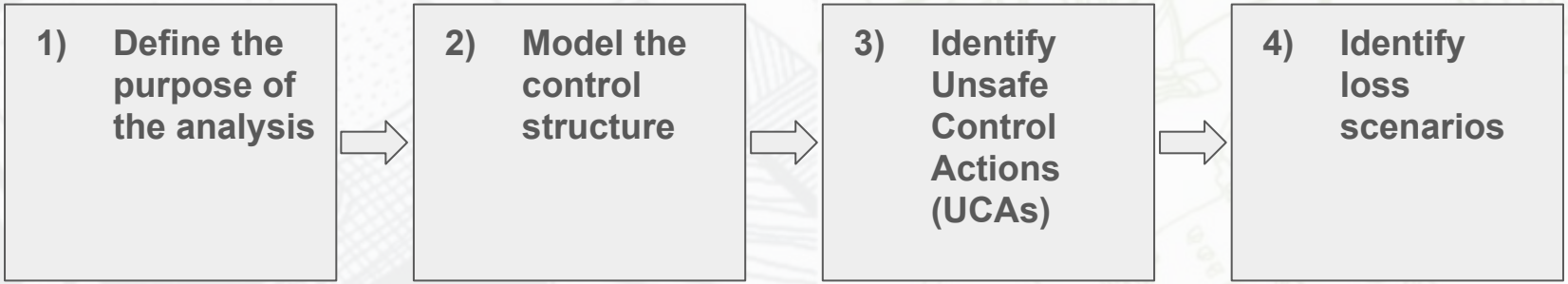
OpenAPS (Artificial Pancreas System)

- https://openaps.org/
- https://github.com/openaps

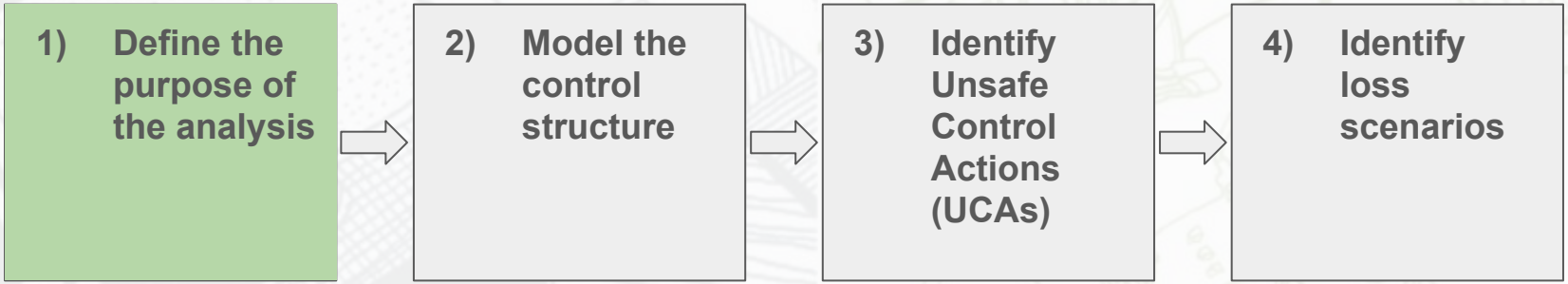
OpenAPS Software

- oref0
  - Javascript implementation of the main calculation algorithm
- OpenAPS
  - Python “core utilities” needed to build an OpenAPS system upon
  - Interface between algorithm and system functions
- Docs
  - Installation shell scripts

STPA Method


1) Define the purpose of the analysis
2) Model the control structure
3) Identify Unsafe Control Actions (UCAs)
4) Identify loss scenarios

1) Define the Purpose of the Analysis



1) Define the Purpose of the Analysis

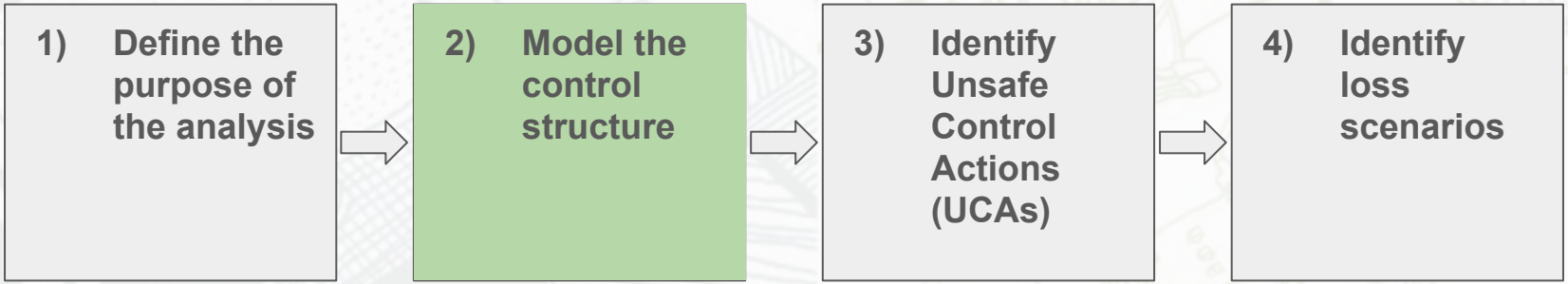
Losses

1. Loss of life or injury
2. Loss of quality of life

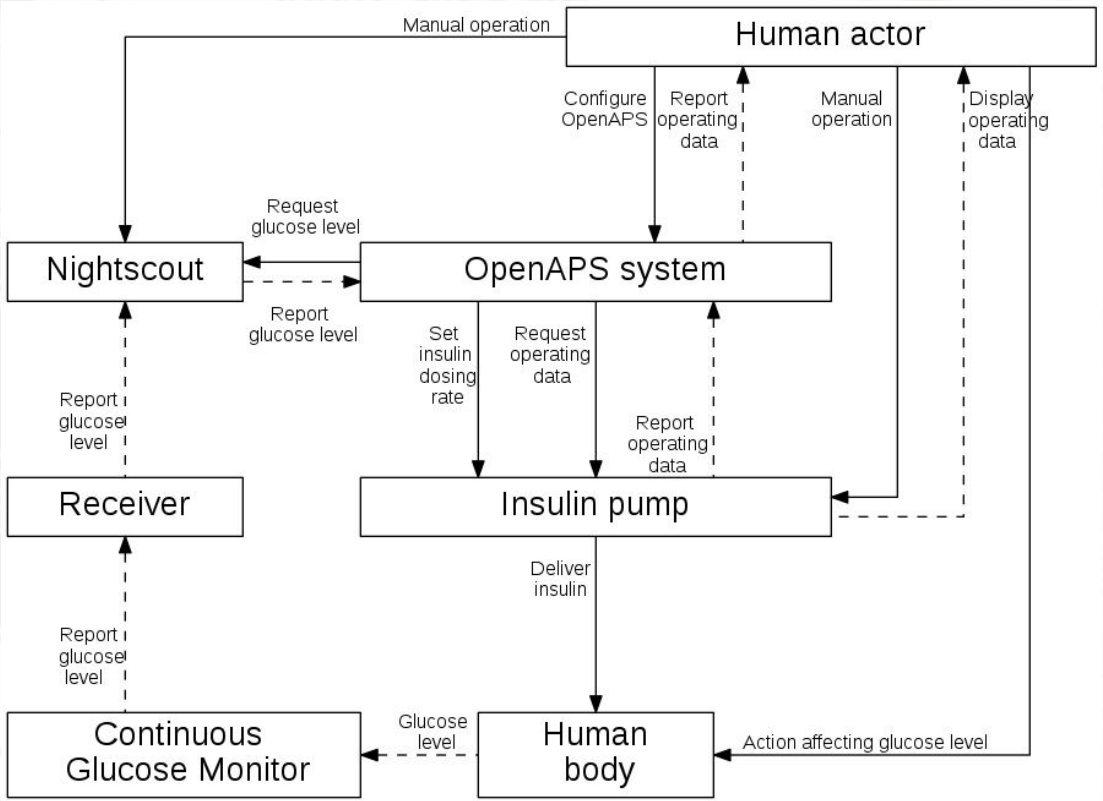
Hazards

1. Human blood glucose levels become too low
2. Human blood glucose levels become too high

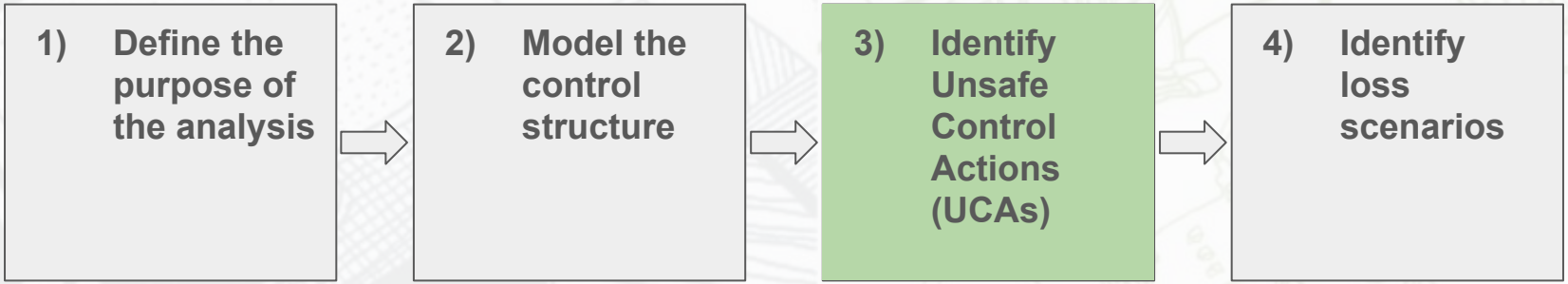
2) Model the Control Structure



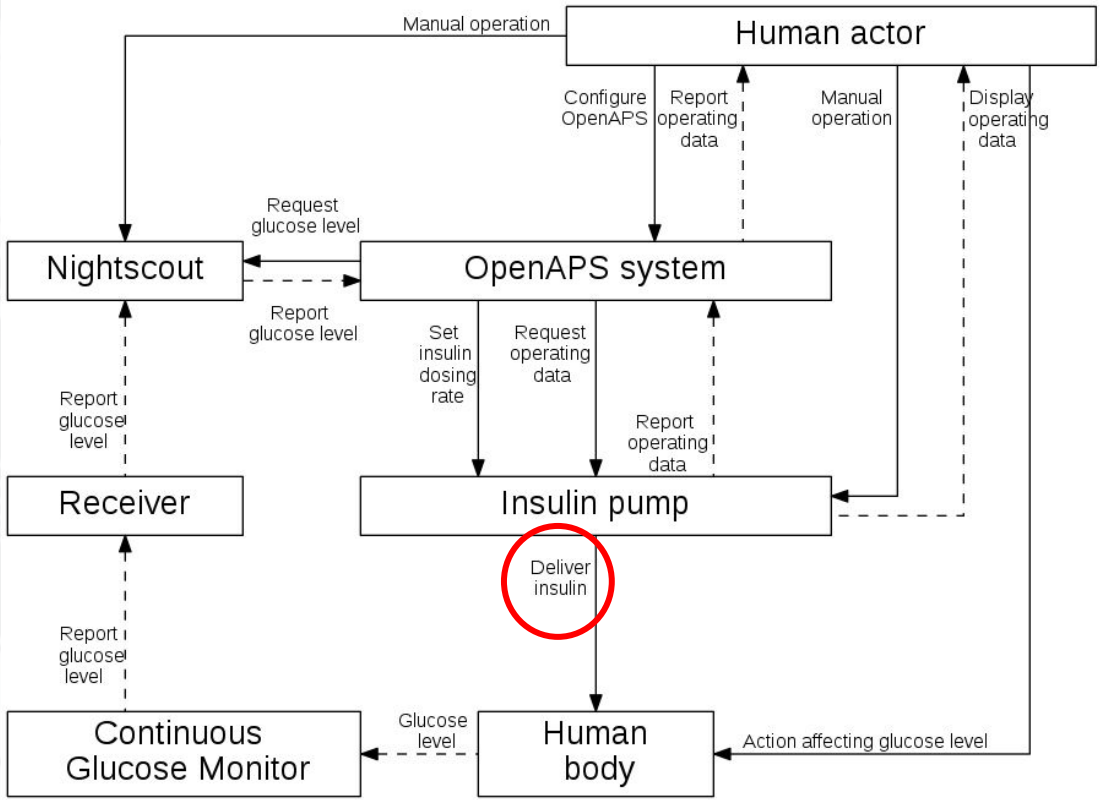
2) Model the Control Structure



3) Identify Unsafe Control Actions (UCAs)

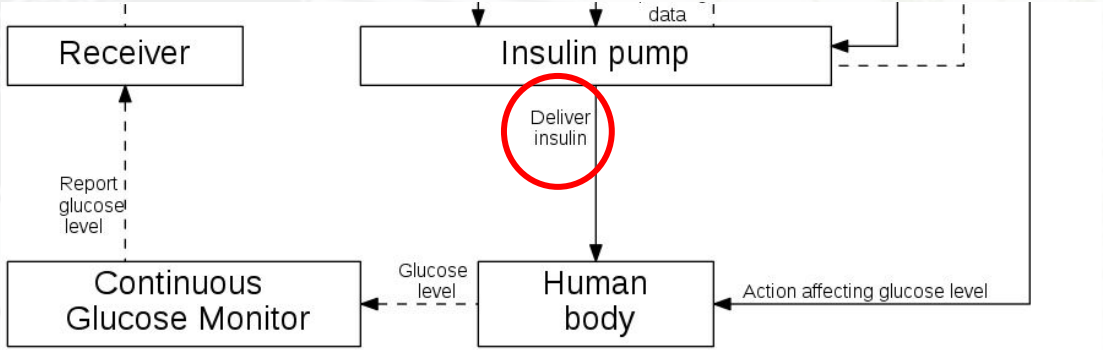


3) Identify Unsafe Control Actions (UCAs)

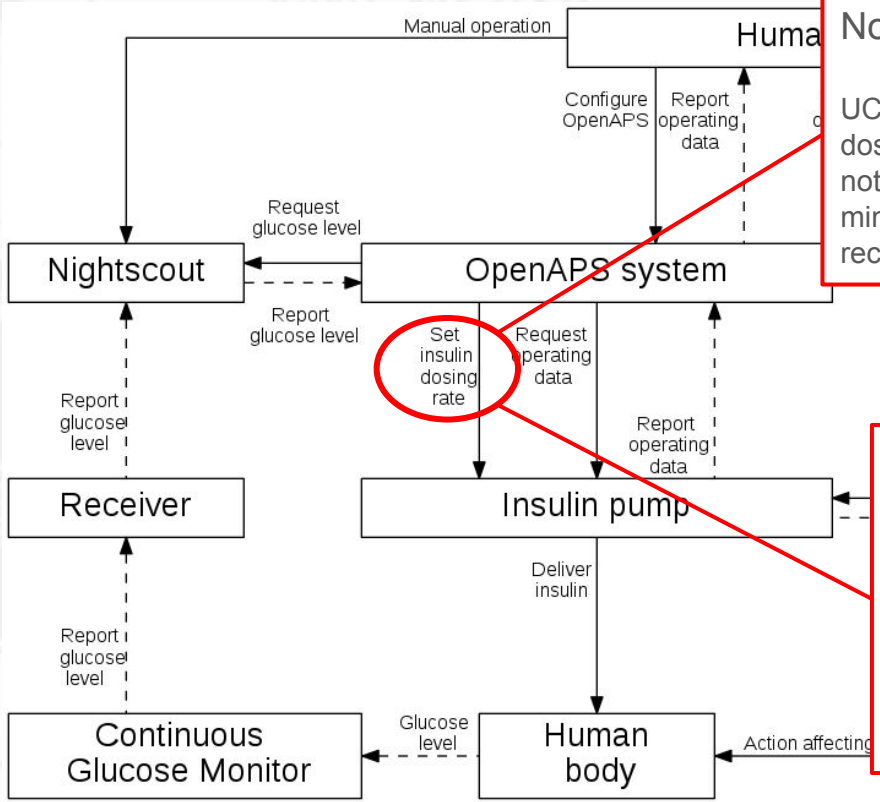


3) Identify Unsafe Control Actions (UCAs)

Not providing	Providing	Too late / too early	Too long / too short



3) Identify Unsafe Control Actions (UCAs)



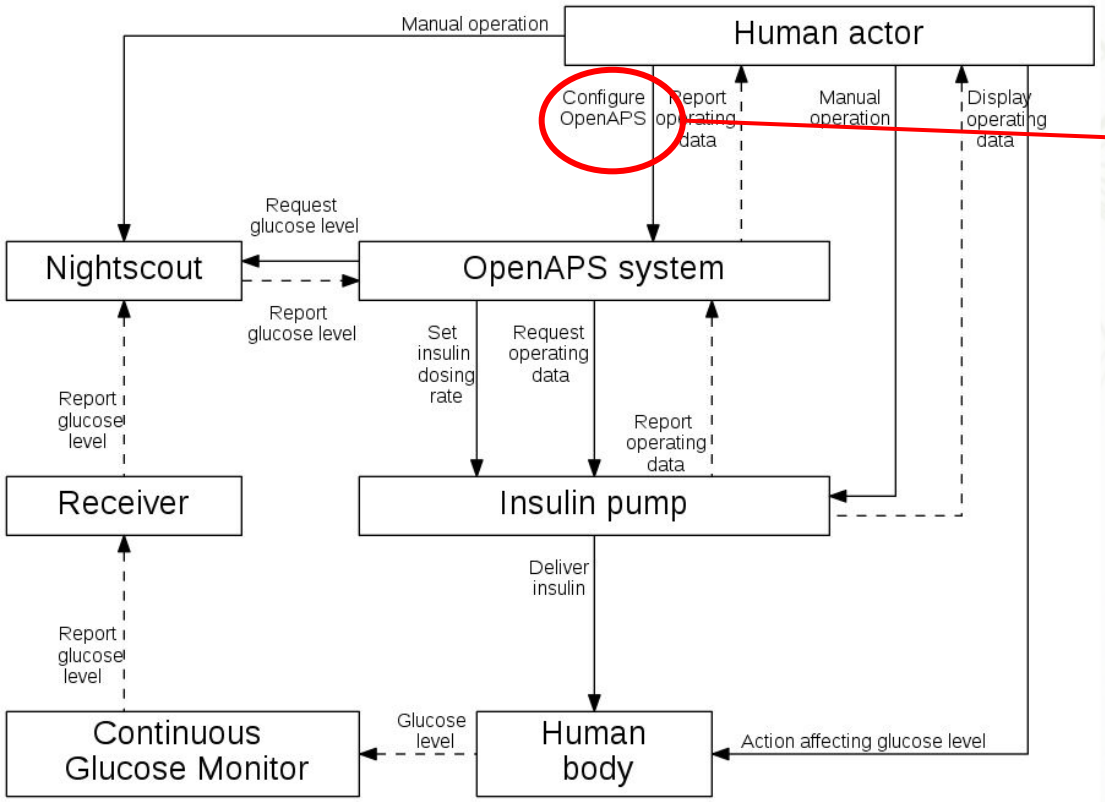
Not Providing

UCA: OpenAPS does not set the insulin dose when the current insulin dosage is not optimal and it has been over X minutes since the last dose recommendation

Providing

UCA: OpenAPS algorithm recommends too high insulin dosing when current glucose levels are low and not likely to rise (based on active insulin and human insulin pump input)

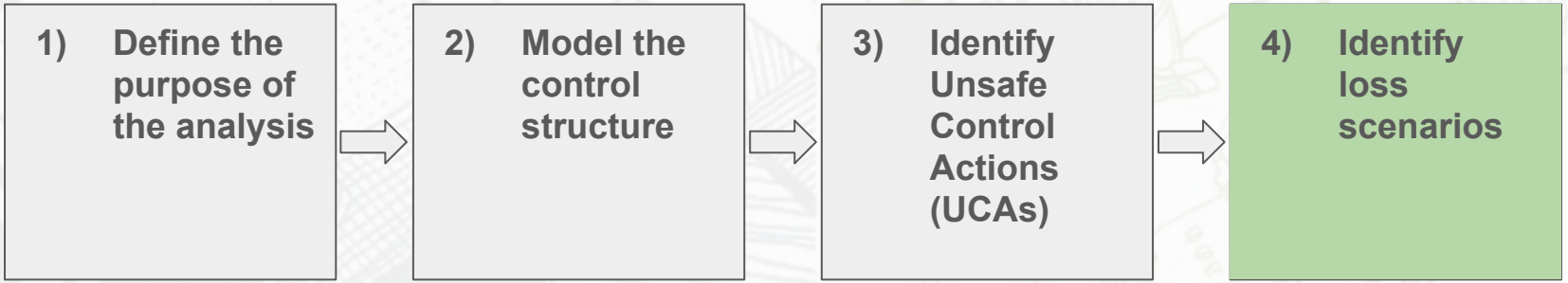
3) Identify Unsafe Control Actions (UCAs)



Not Providing

UCA : Human actor does not update the OpenAPS configuration when there is a change in the loop setup

4) Identify Loss Scenarios



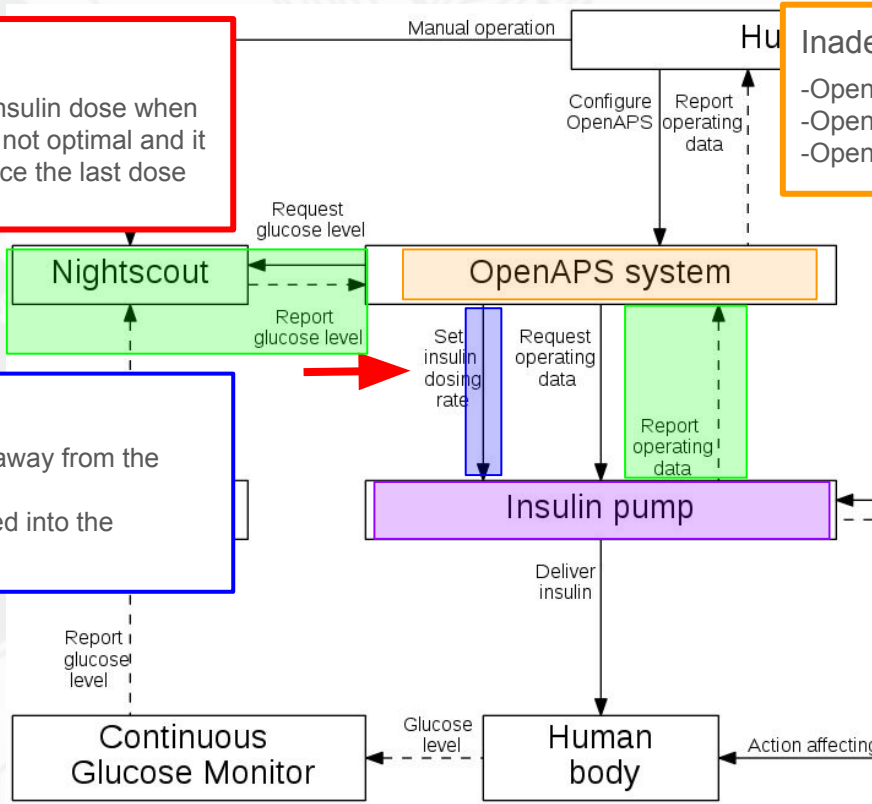
4) Identify Loss Scenarios

Not Providing UCA

OpenAPS does not set the insulin dose when the current insulin dosage is not optimal and it has been over X minutes since the last dose recommendation

Inadequate control path

-The insulin pump is too far away from the OpenAPS rig
-The radio stick is not plugged into the OpenAPS rig properly



Inadequate controller behaviour:

-OpenAPS algorithm not running
-OpenAPS out of power
-OpenAPS clock is set to the wrong time

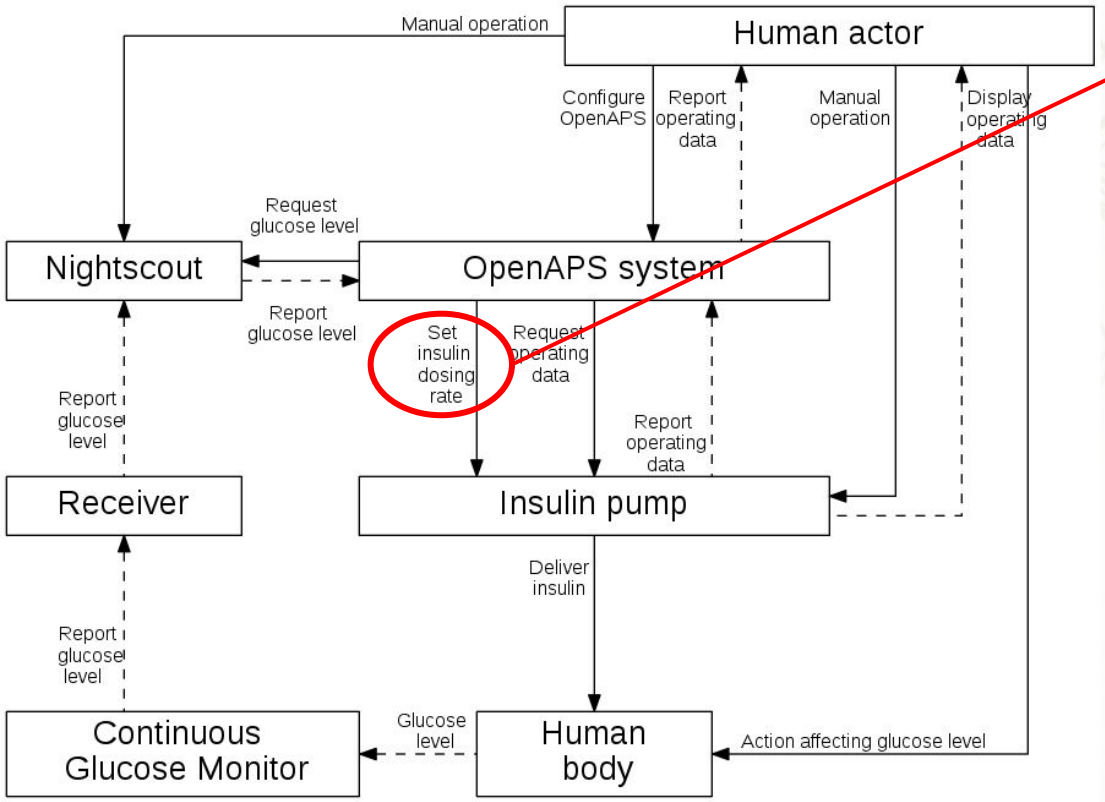
Inadequate feedback:

-OpenAPS has gone out of range from NightScout and not received updated info
-OpenAPS has had wireless interference from the insulin pump and not received updated info

Inadequate controller behaviour

-Liquid in the tubing connector has caused the infusion set to get blocked
-The insulin pump is out of battery

Loss Scenario Examples

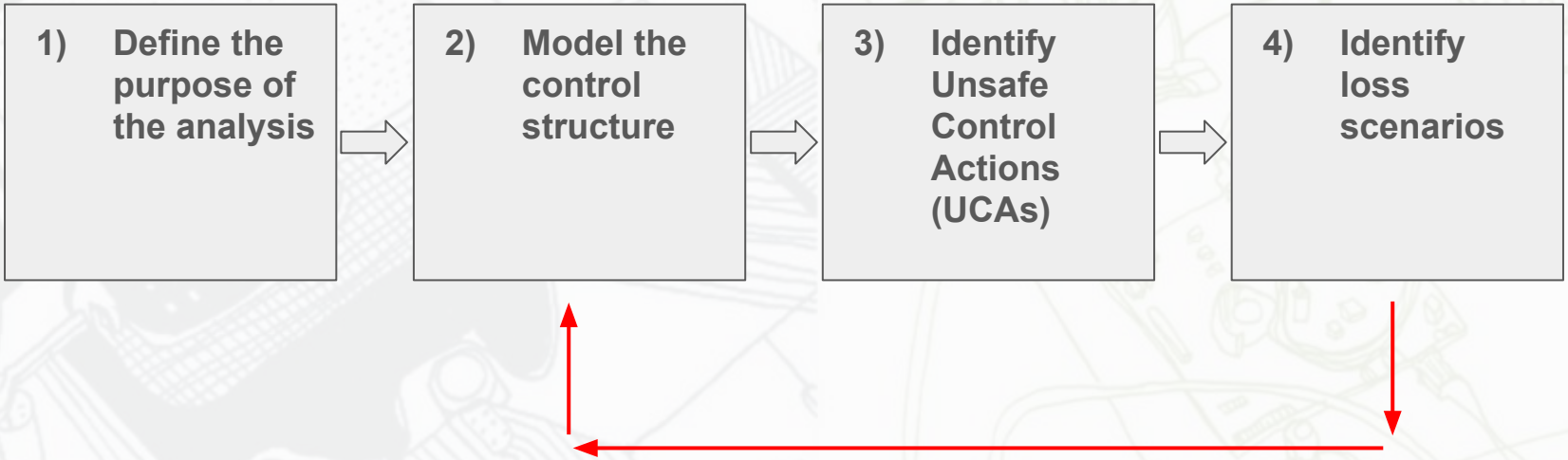


UCA : OpenAPS algorithm recommends too low temporary basal dosing recommendation when current glucose levels are high and not likely to fall (bases on current glucose levels, active insulin and human insulin pump input)

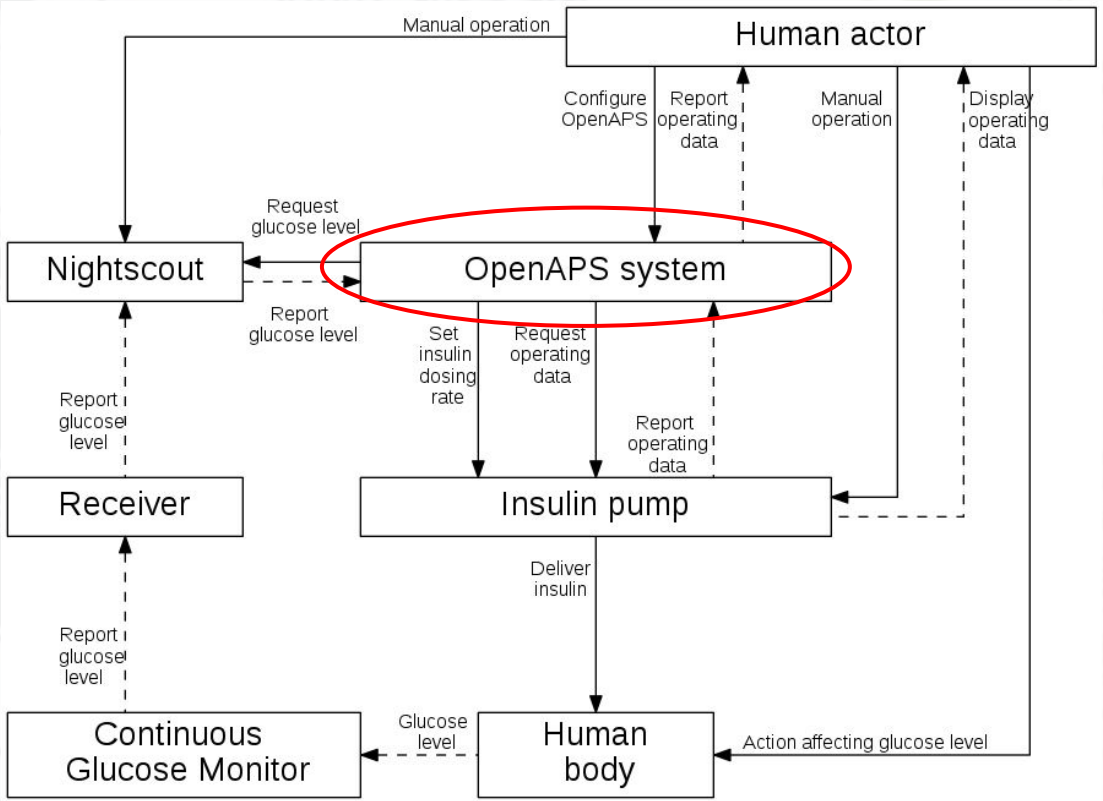
How could this happen?

- Flawed insulin to glucose calculation
- OpenAPS is pulling the wrong glucose data or misreading the glucose data from NightScout
- Human actor input too few carbs
- Insulin pump not dosing properly but OpenAPS is not informed of this
- Human actor has configured OpenAPS with the wrong DIA / ISF / carb ratio settings

STPA Method - Iterate

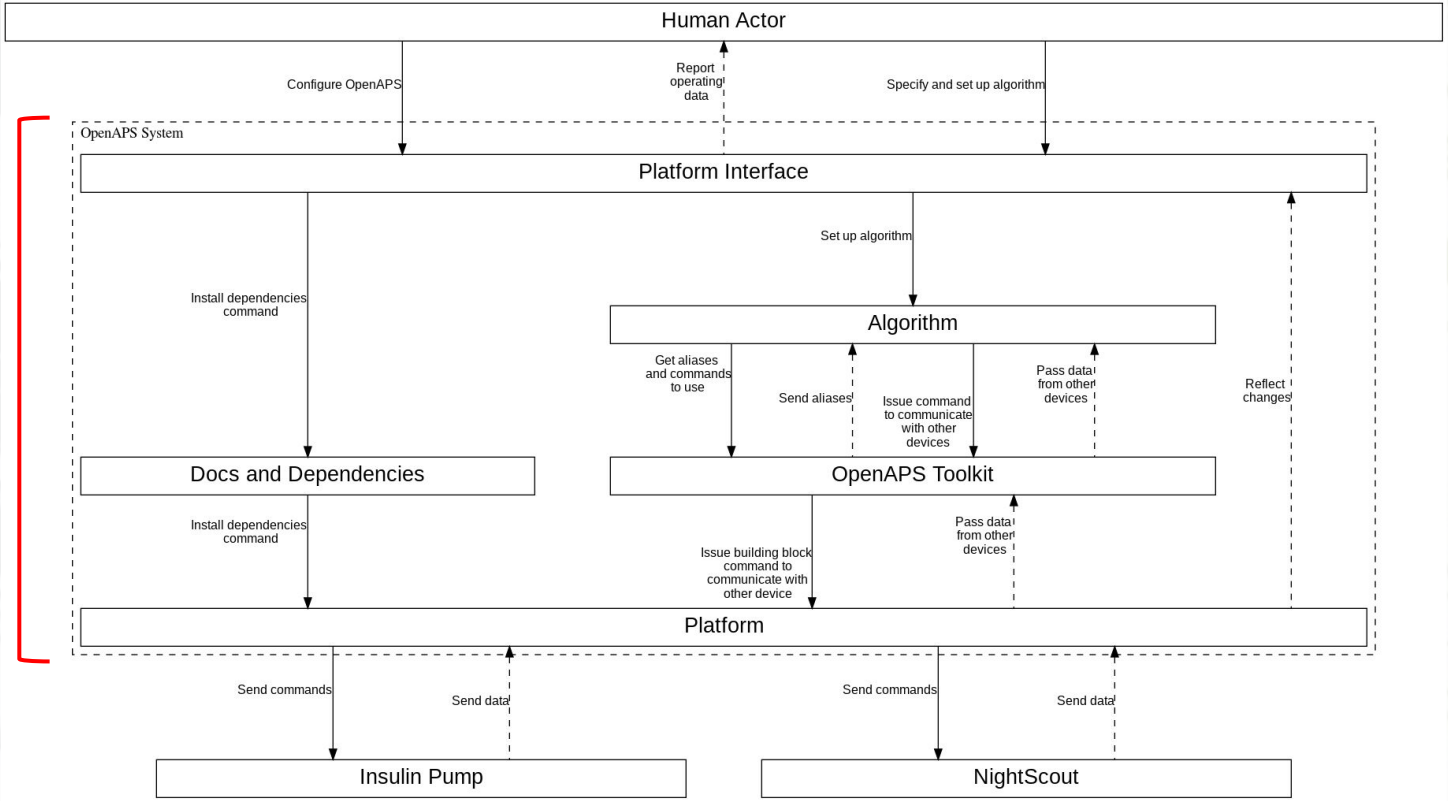


Next Level Control Diagram

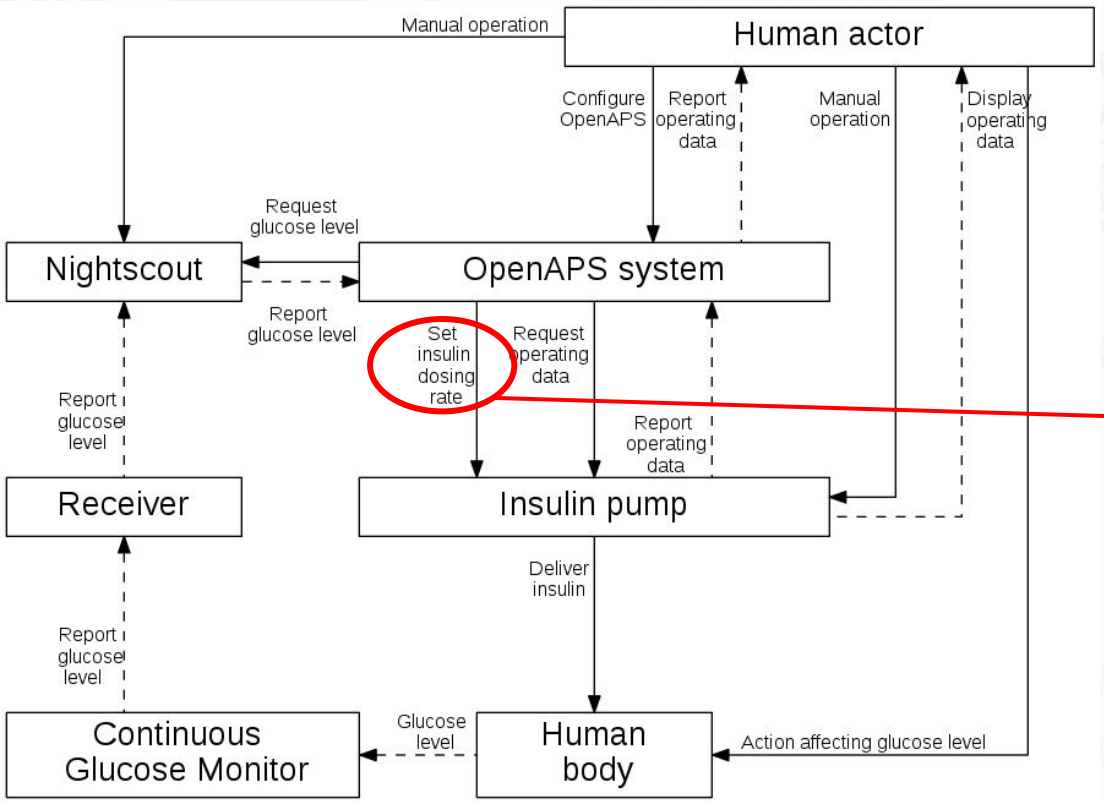


Next Level Control Diagram

OpenAPS System



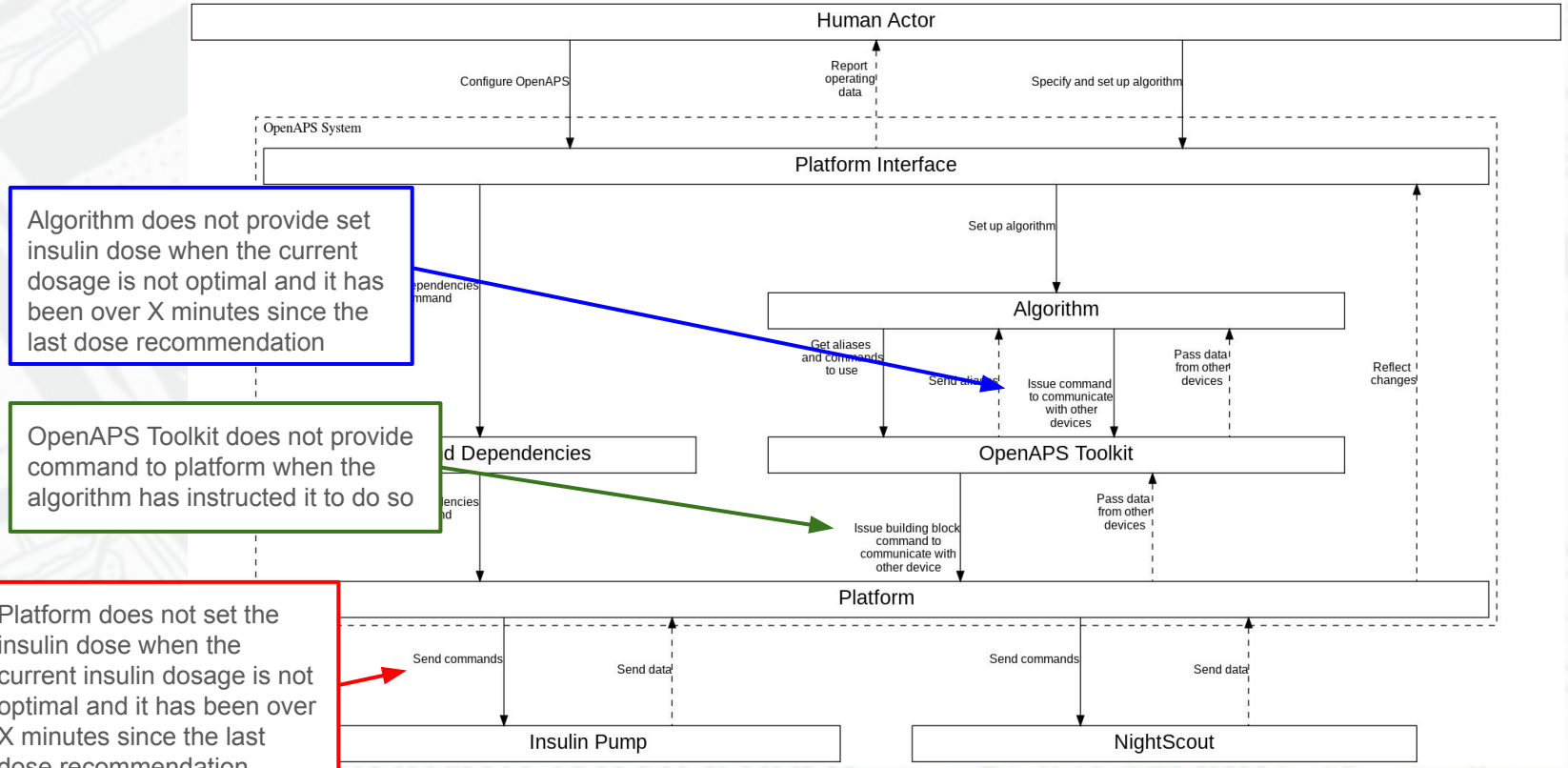
High Abstraction to Low Abstraction



Not Providing

OpenAPS does not set the insulin dose when the current insulin dosage is not optimal and it has been over X minutes since the last dose recommendation

Next Level Control Diagram

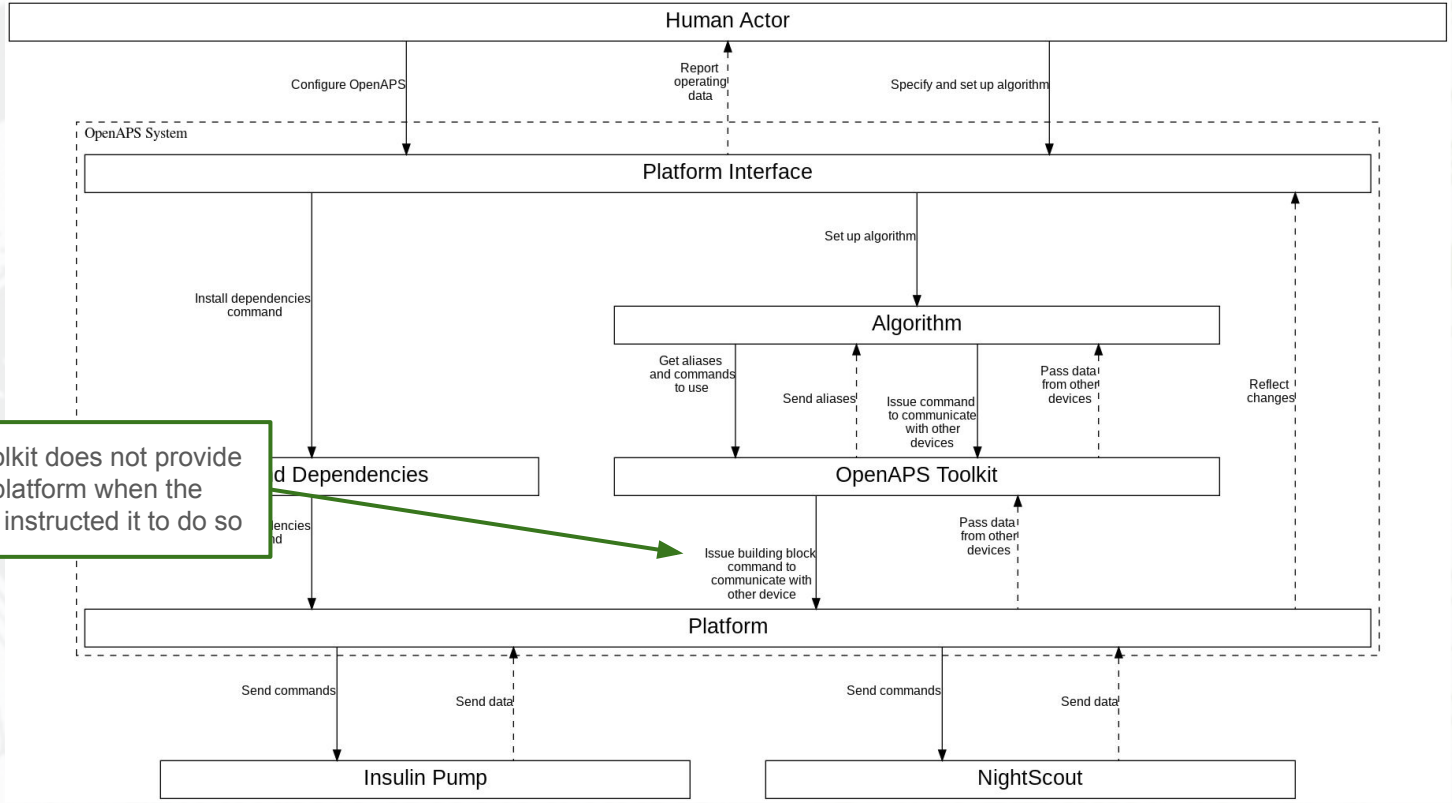


Algorithm does not provide set insulin dose when the current dosage is not optimal and it has been over X minutes since the last dose recommendation

OpenAPS Toolkit does not provide command to platform when the algorithm has instructed it to do so

Platform does not set the insulin dose when the current insulin dosage is not optimal and it has been over X minutes since the last dose recommendation

Next Level Control Diagram



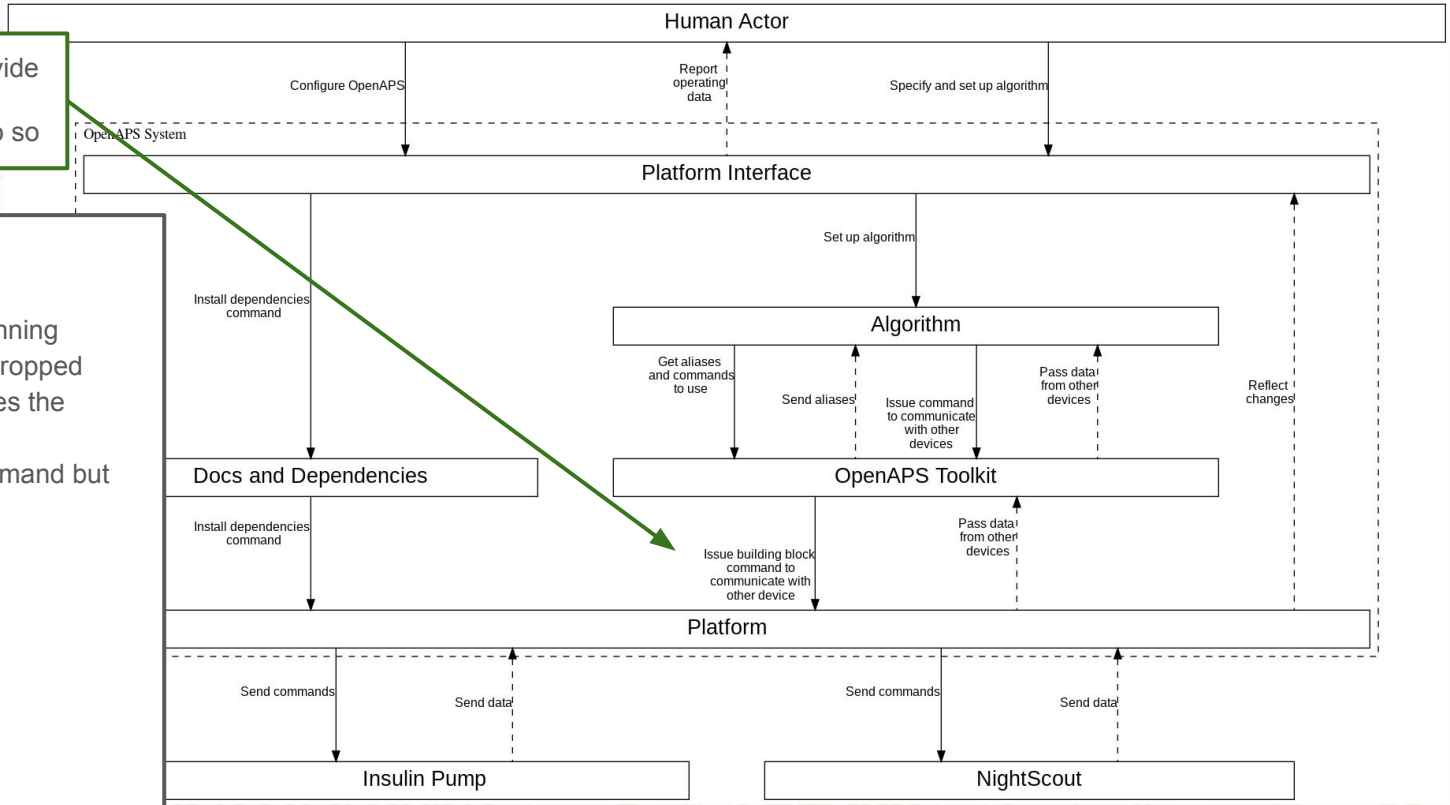
OpenAPS Toolkit does not provide command to platform when the algorithm has instructed it to do so

Next Level Control Diagram

OpenAPS Toolkit does not provide command to platform when the algorithm has instructed it to do so

How could this happen?

- OpenAPS Toolkit is not running
- The command has been dropped
- Memory corruption changes the command
- Platform receives the command but does not execute it
- etc.



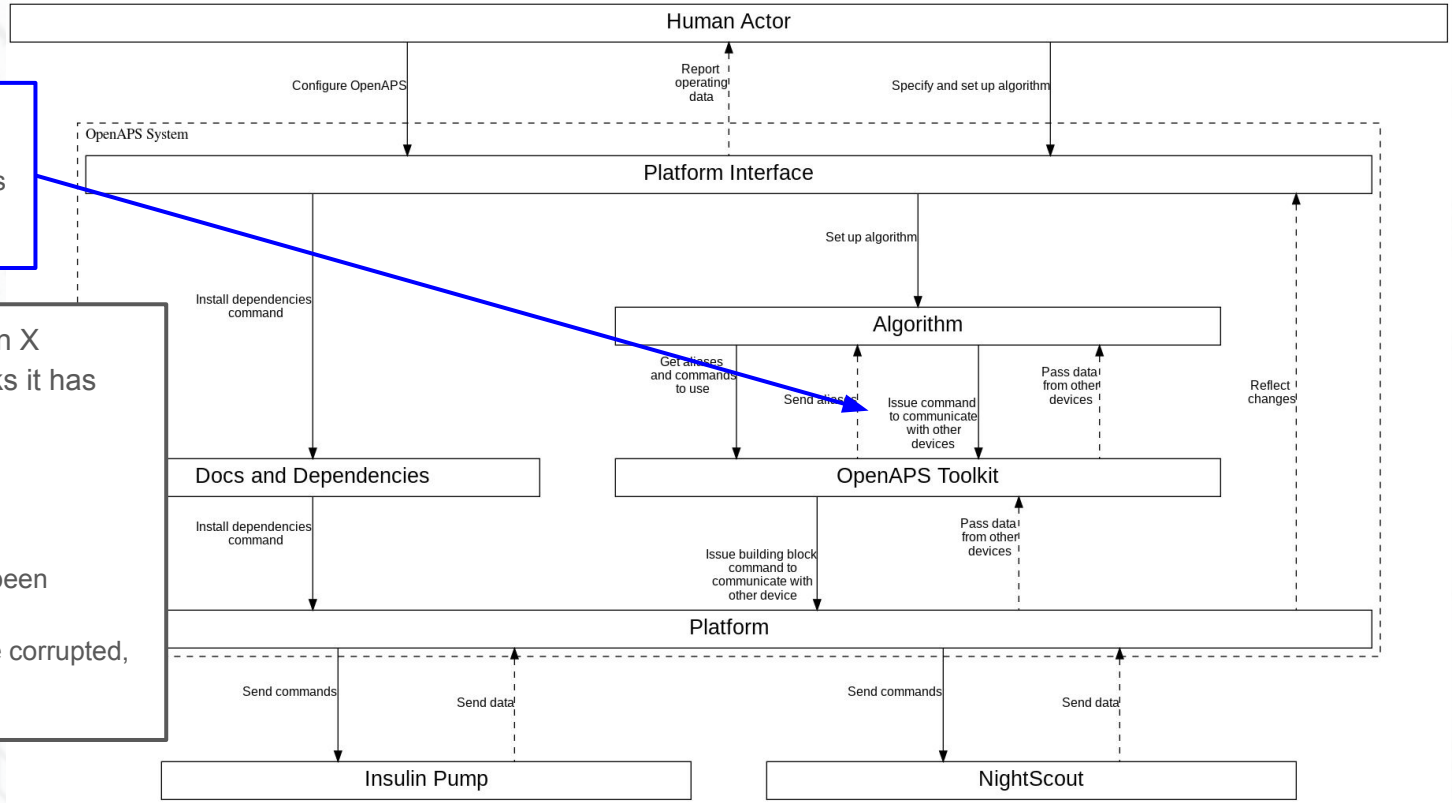
Next Level Control Diagram

Algorithm does not provide set insulin dose when the current dosage is not optimal and it has been over X minutes since the last dose recommendation

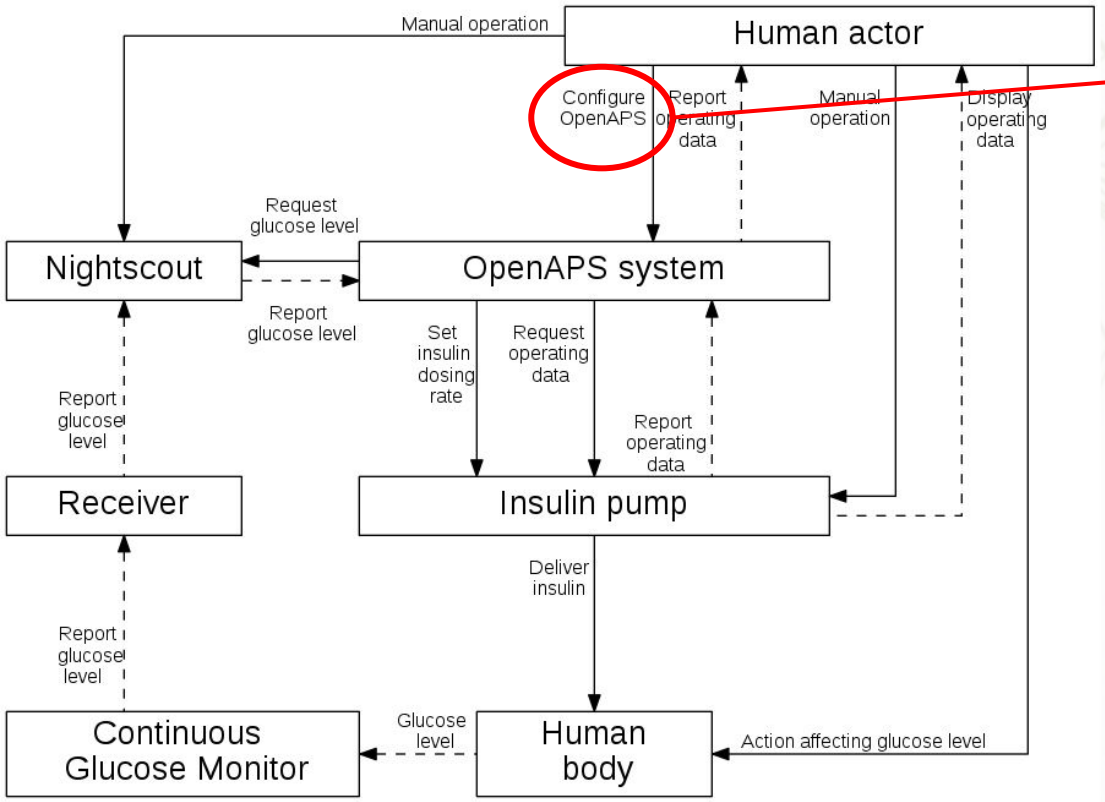
Scenario: it has been more than X minutes, but the algorithm thinks it has been less.

How could this happen?

- Clock source - clock has been updated, corrupted etc.
- Local variable storing time corrupted, overwritten



Example

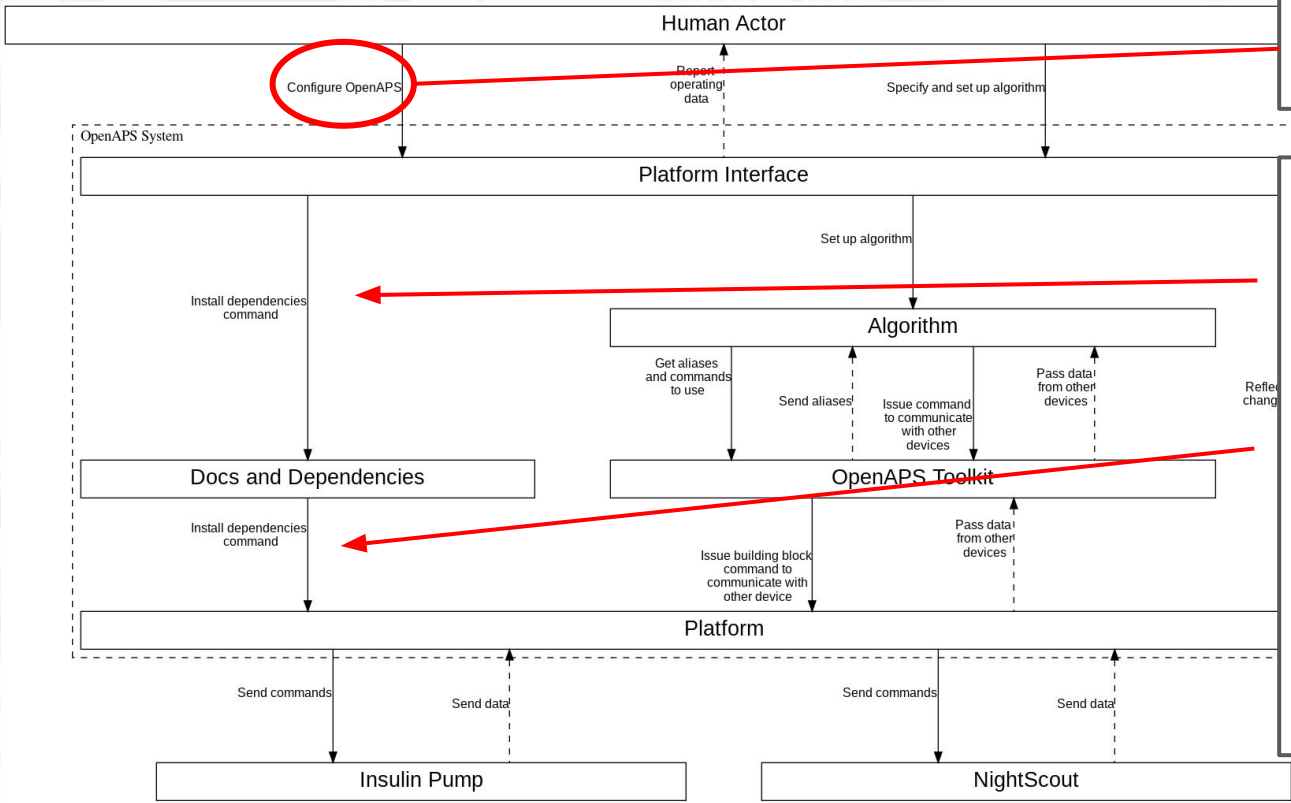


UCA : Human actor does not install all the dependencies necessary for OpenAPS

How could this happen?

- Flawed “controller” - human believes the wrong thing
- Connectivity issues on control path
- Feedback issues
  - install script does not give enough information
  - Human does not correctly interpret script output
- Installed correctly but automatic updates overwrite current library version

Example

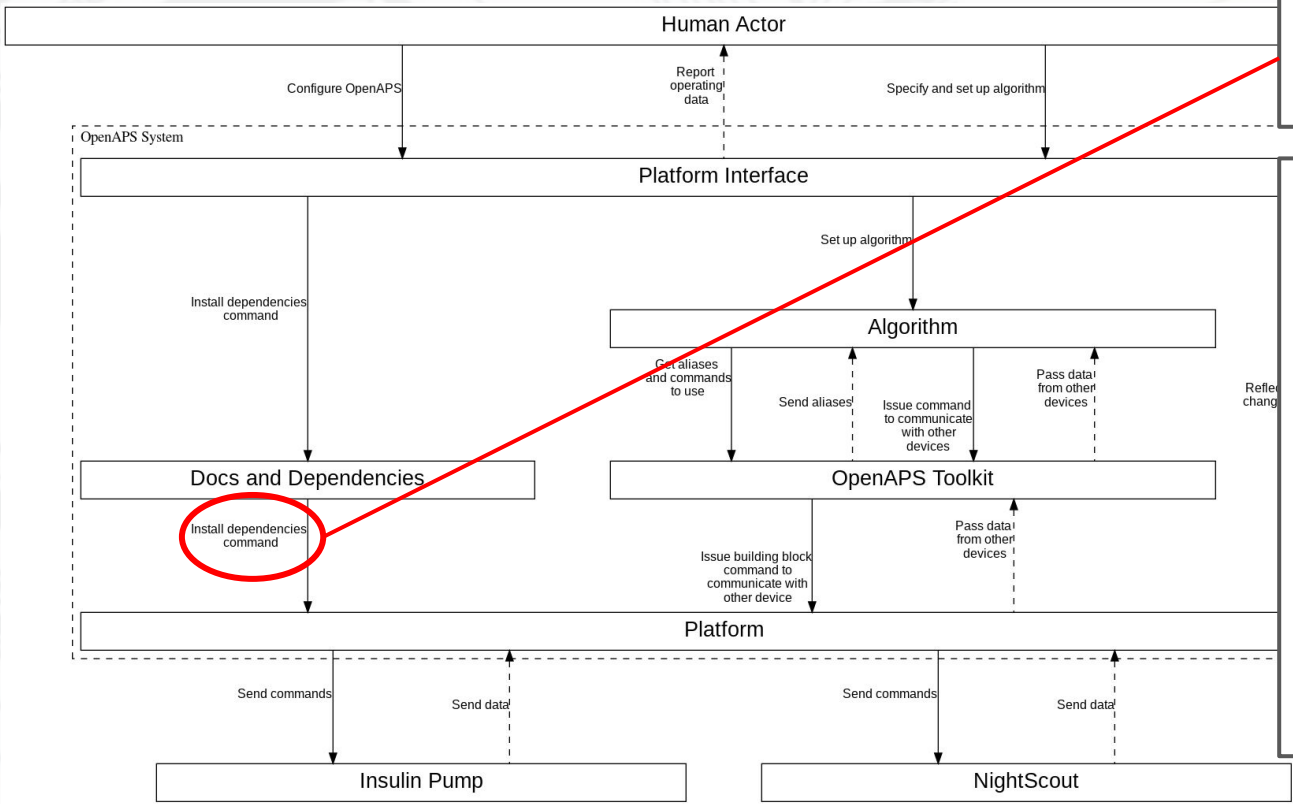


UCA : Human actor does not install all the dependencies necessary for OpenAPS

Next Level UCAs

- UCA: platform interface does not install dependencies when the human instructs it to do so
- UCA: Installation script does not install dependencies correctly when the platform instructs it to

Example

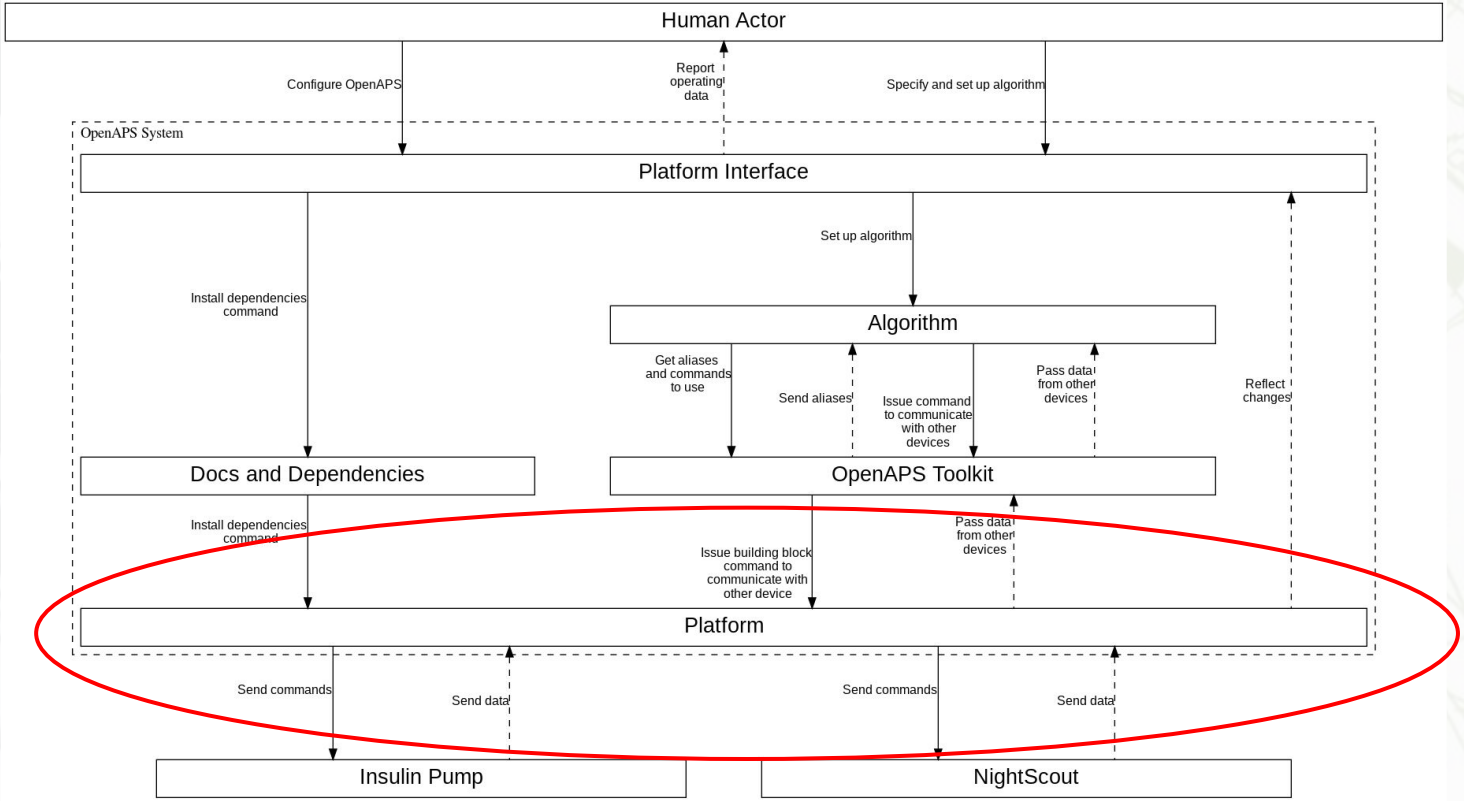


UCA : Installation script does not install dependencies correctly when the platform instructs it to

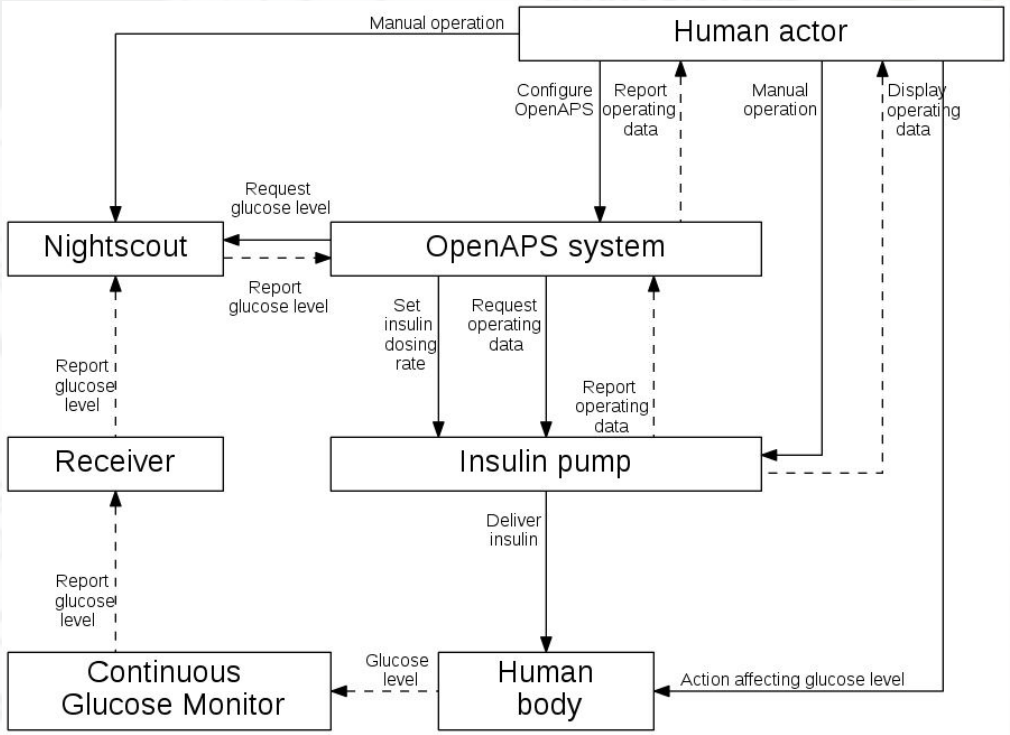
How could this happen?

- Connectivity issues
- apt-get not updated
- Automatic system updates install different version of a package

The Next-Next Level



Expanding the Analysis



Missing Analysis

- Software Developer
- Source Code
- Test and Validation System

ect...

Analysis Highlights

- Analyse complex interactions between human, hardware and software, with possible conflicting control and information
  - Create recommendations for control hierarchy and test if the system actually meets these requirements
- Analyse interaction between software modules and platform, in order to interface with the safety architecture working group
  - Create recommendations for software and hardware architecture
- Analyse software build and deploy strategy
  - Can create an automated system and use containers and reproducible build techniques to ensure correct software and dependencies are deployed
- Analyse the algorithm
  - Create recommendations to verify correctness (validation, tests etc)

Join the Open Source Community

- https://elisa.tech/
- Talk to us on the STPA slack channel
  - stampnetwork.slack.com/