CIS 855: High-Assurance Systems

Introduction to Safety Concepts

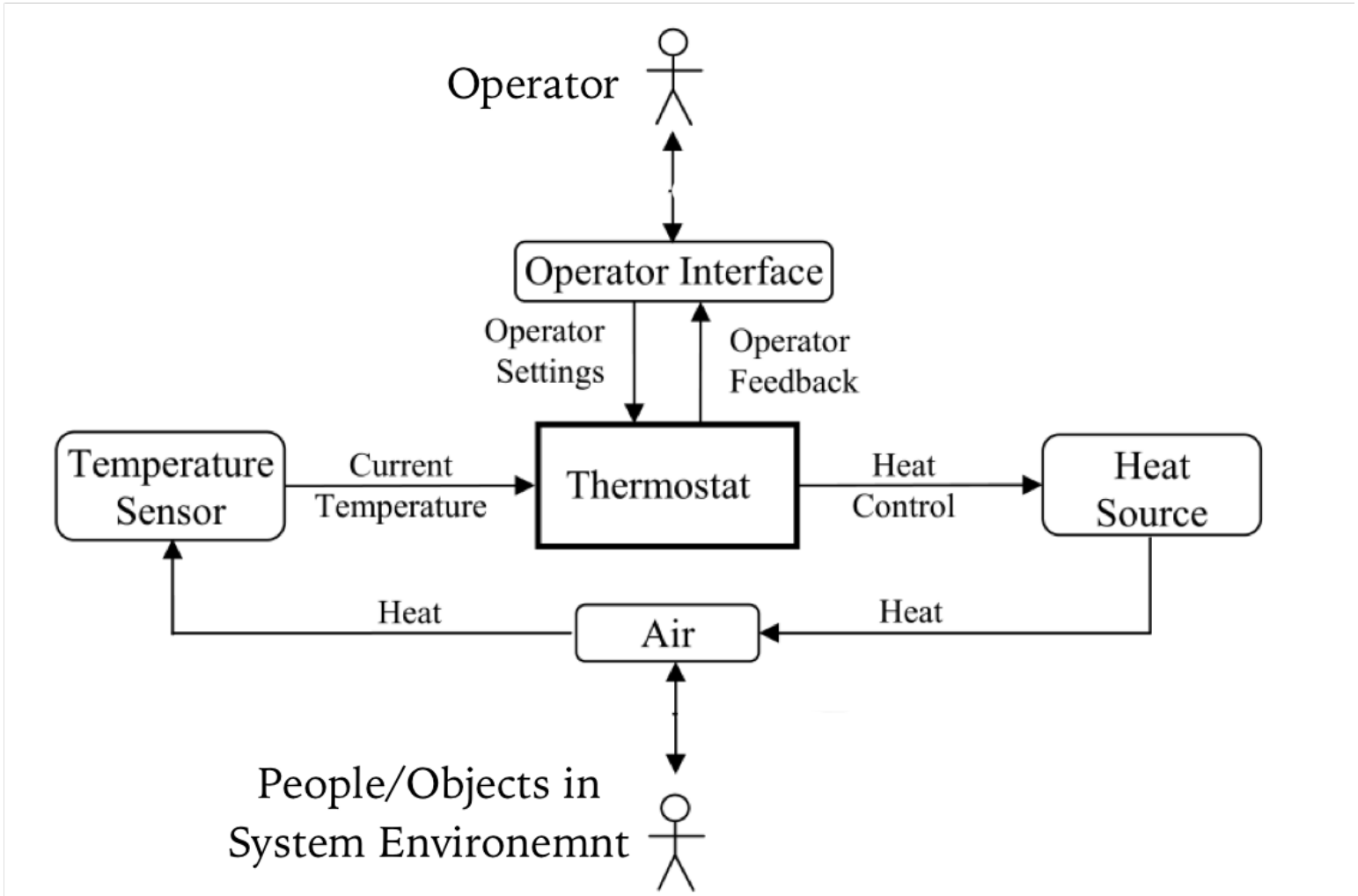
Lecture: Control Loop Concepts using Temperature Controller Example

Objectives

- Understand the basic elements of a control loop – a fundamental concept in…
  - Embedded system design
  - Safety engineering
- Begin to consider the notion of system boundaries and the role they play in requirements and safety engineering
- See some hints about how control loops can be used in hazard analysis (safety analysis)

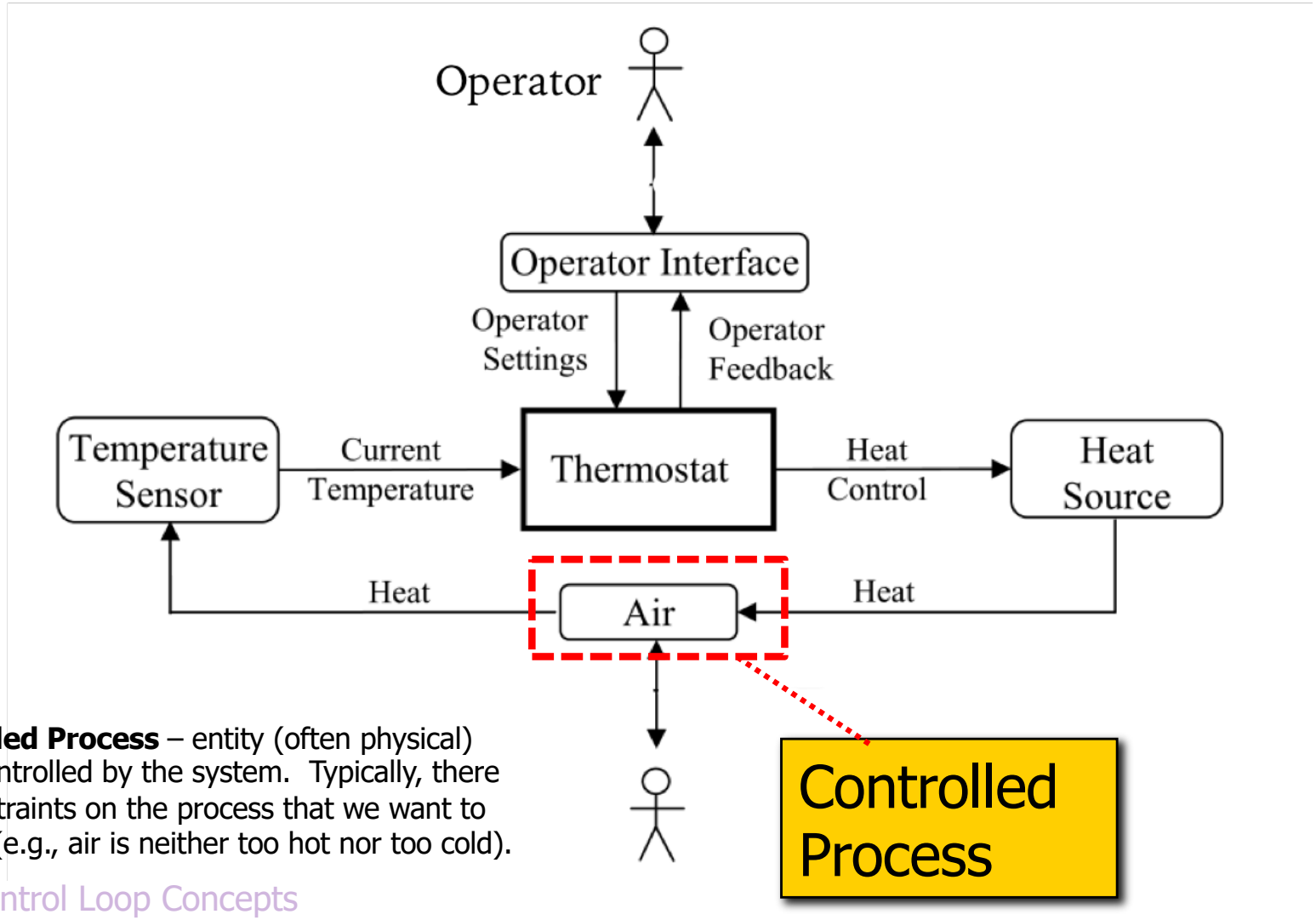
Simple Example System

Simple "Temperature Control" illustrates many core concepts of cyber-physical systems



Common Components in a Safety Critical System

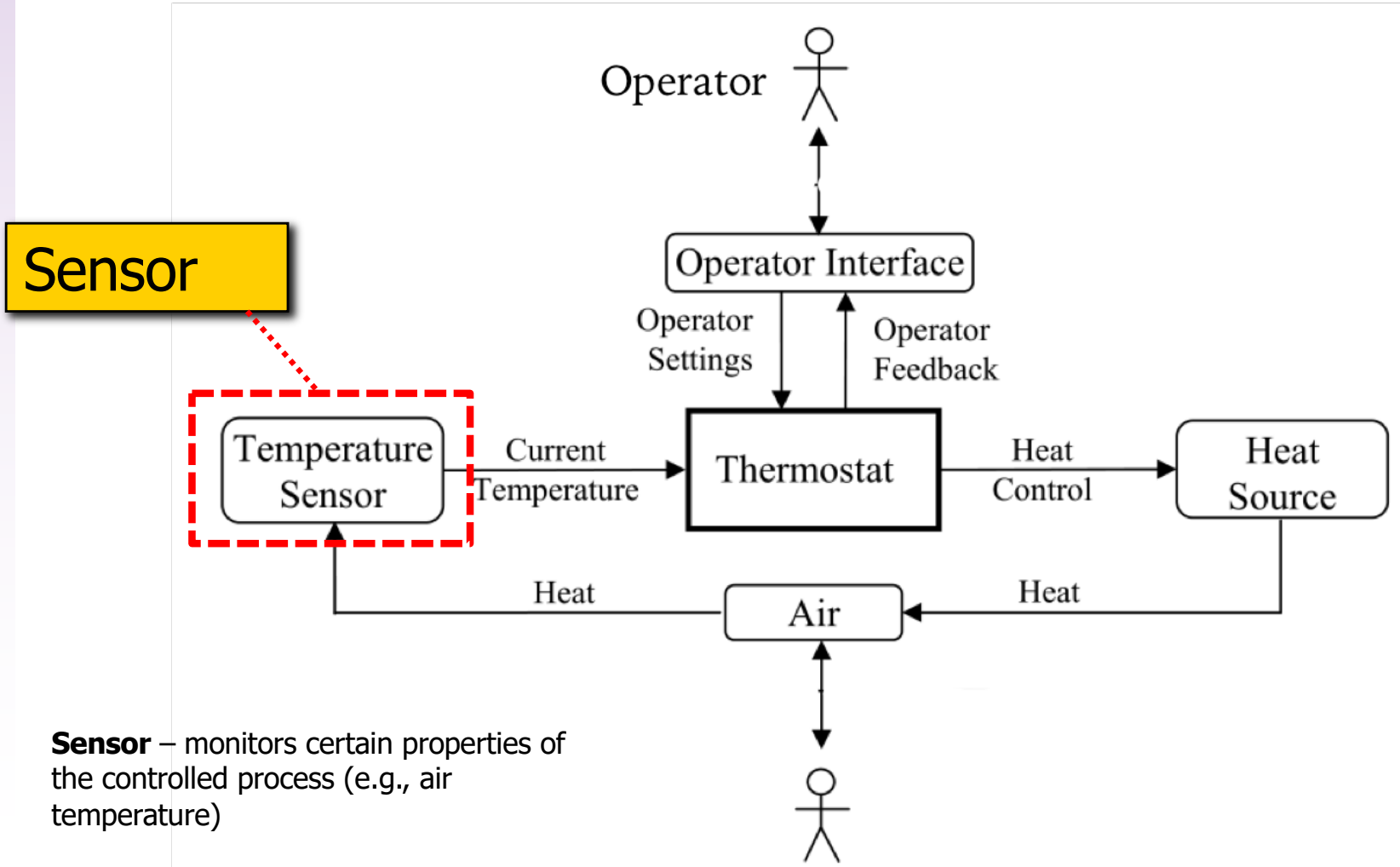
Controlled Process – the thing in the "physical world" that the system aims to control



Controlled Process – entity (often physical) being controlled by the system. Typically, there are constraints on the process that we want to enforce (e.g., air is neither too hot nor too cold).

Common Components in a Safety Critical System -- Sensor

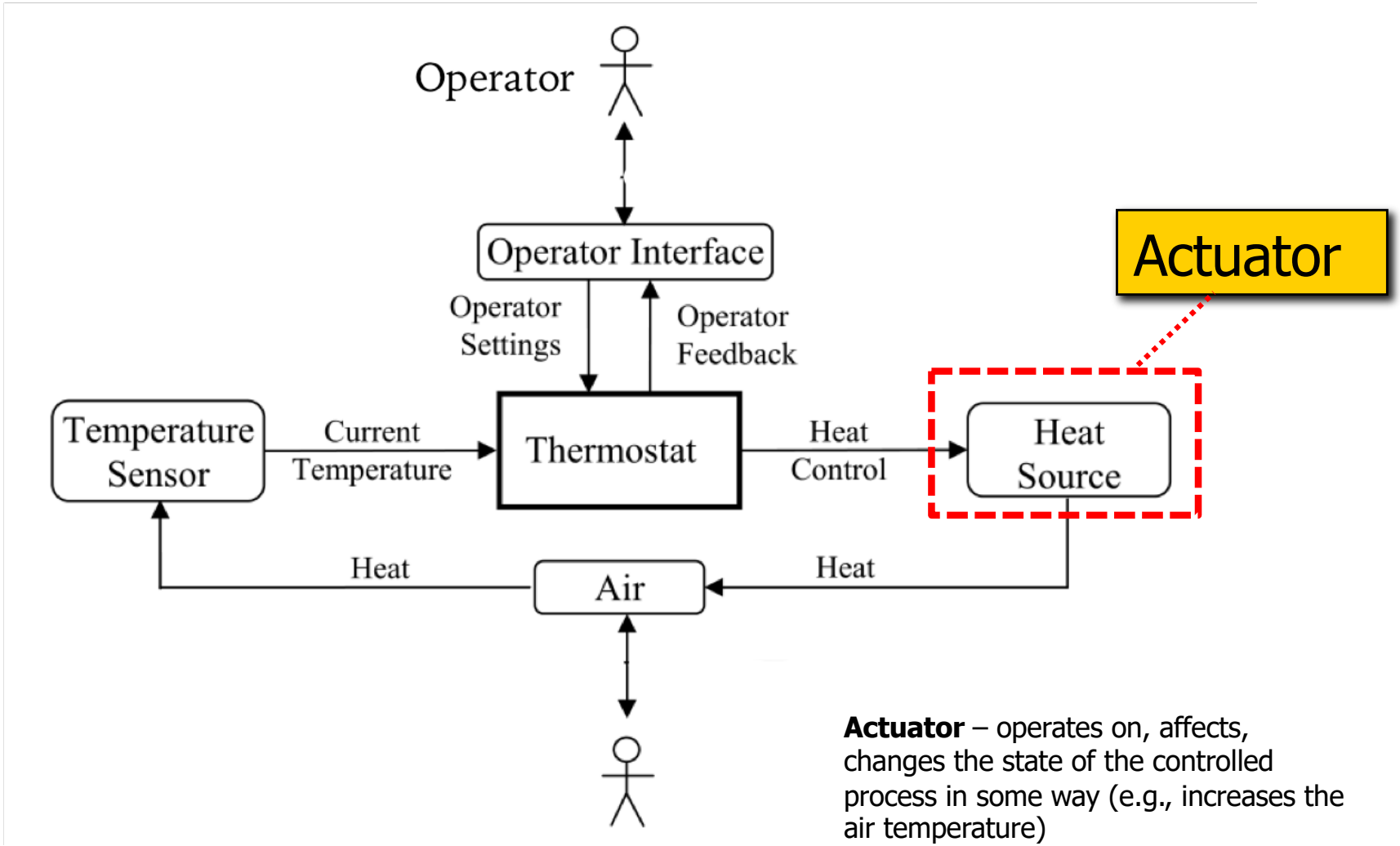
Sensor



Sensor – monitors certain properties of the controlled process (e.g., air temperature)

Common Components in a Safety Critical System -- Actuator

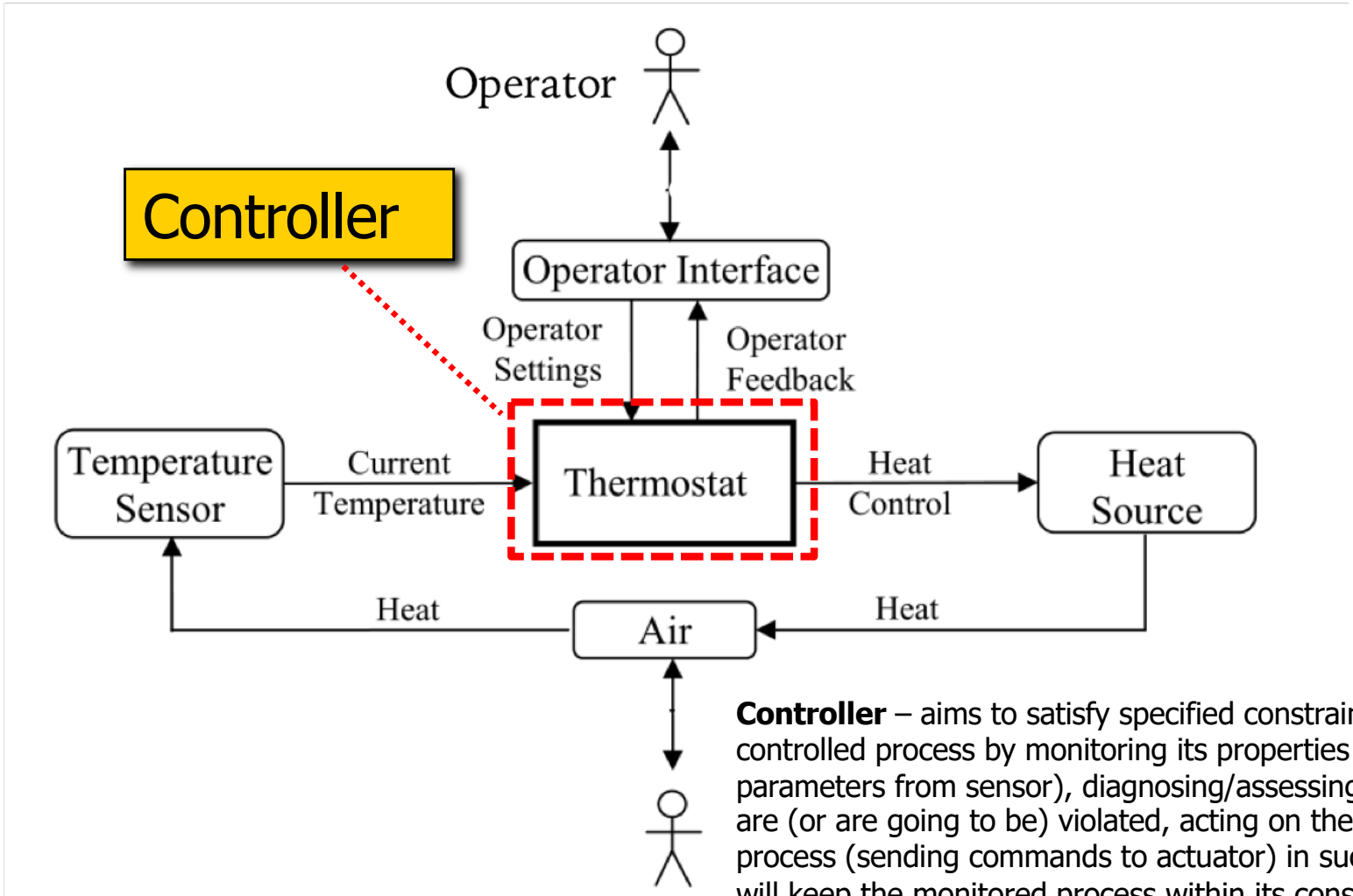
Actuator



Actuator – operates on, affects, changes the state of the controlled process in some way (e.g., increases the air temperature)

Common Components in a Safety Critical System -- Controller

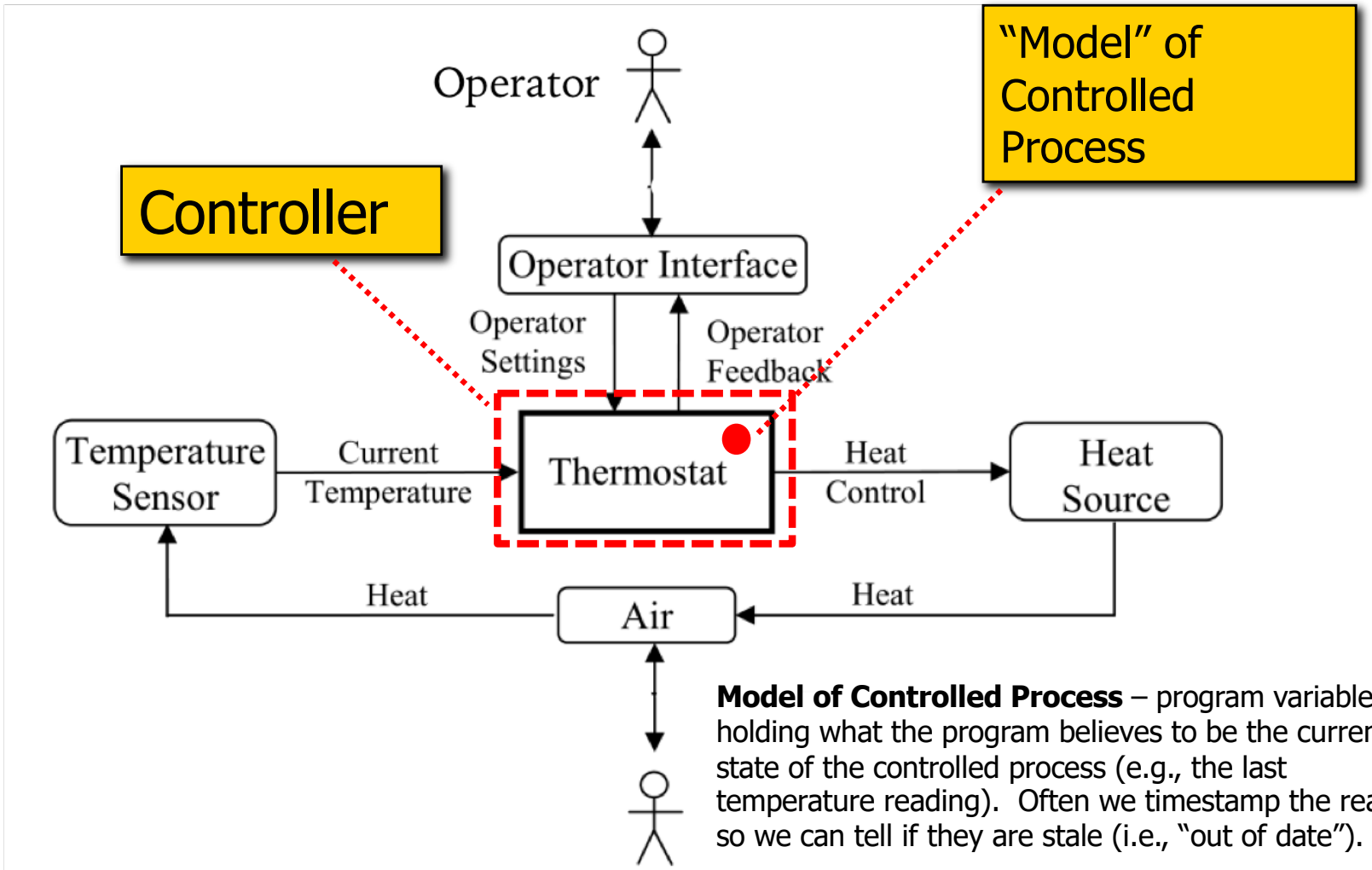
Controller



Controller – aims to satisfy specified constraints on the controlled process by monitoring its properties (reading parameters from sensor), diagnosing/assessing if constraints are (or are going to be) violated, acting on the controlled process (sending commands to actuator) in such as way that will keep the monitored process within its constraints (or restore it to the constraints)

Common Components in a Safety Critical System -- Controller

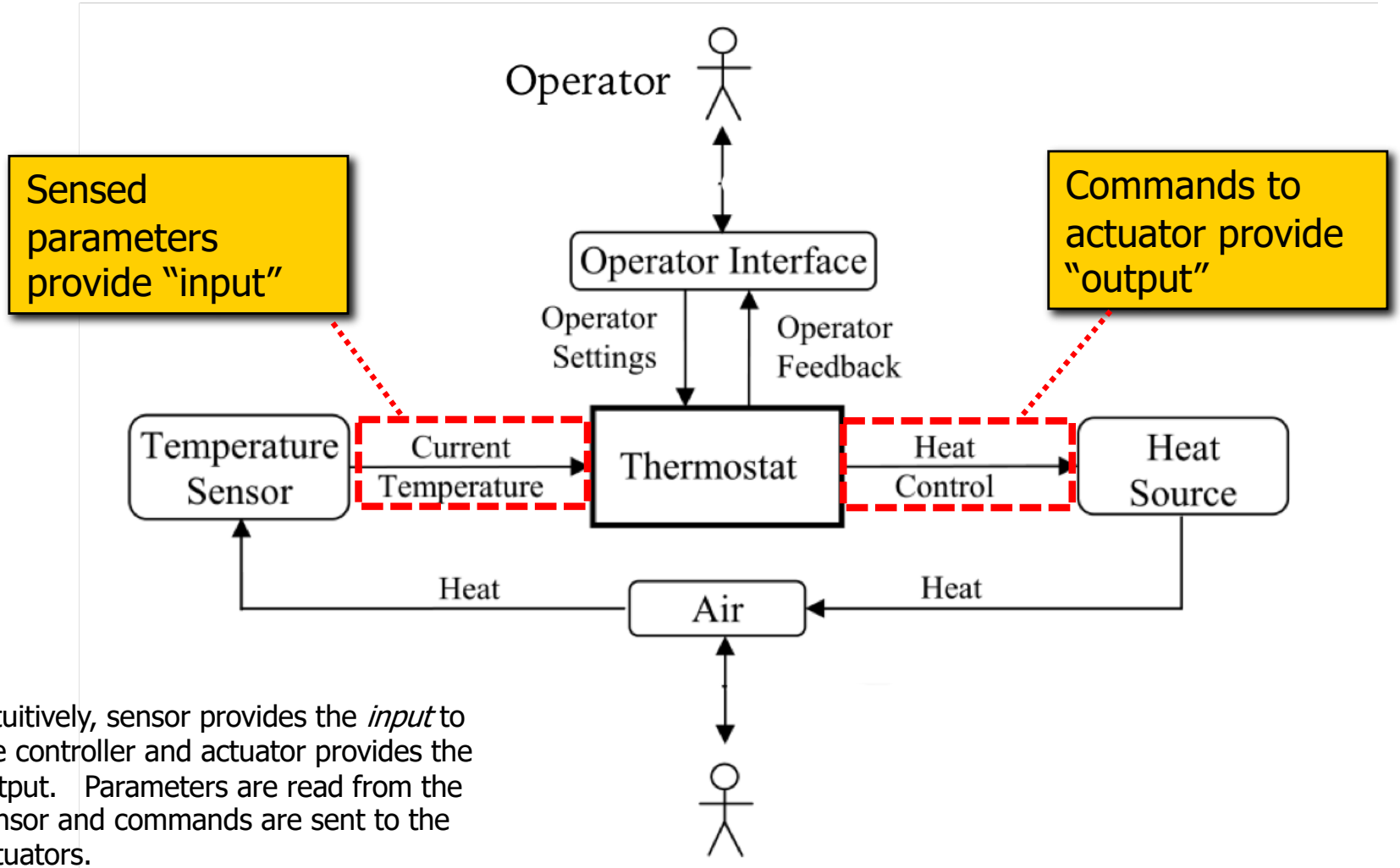
Controller



Model of Controlled Process – program variables holding what the program believes to be the current state of the controlled process (e.g., the last temperature reading). Often we timestamp the readings so we can tell if they are stale (i.e., "out of date").

Common Components in a Safety Critical System -- Controller

The controller "inputs" and "outputs" associated with the controlled process

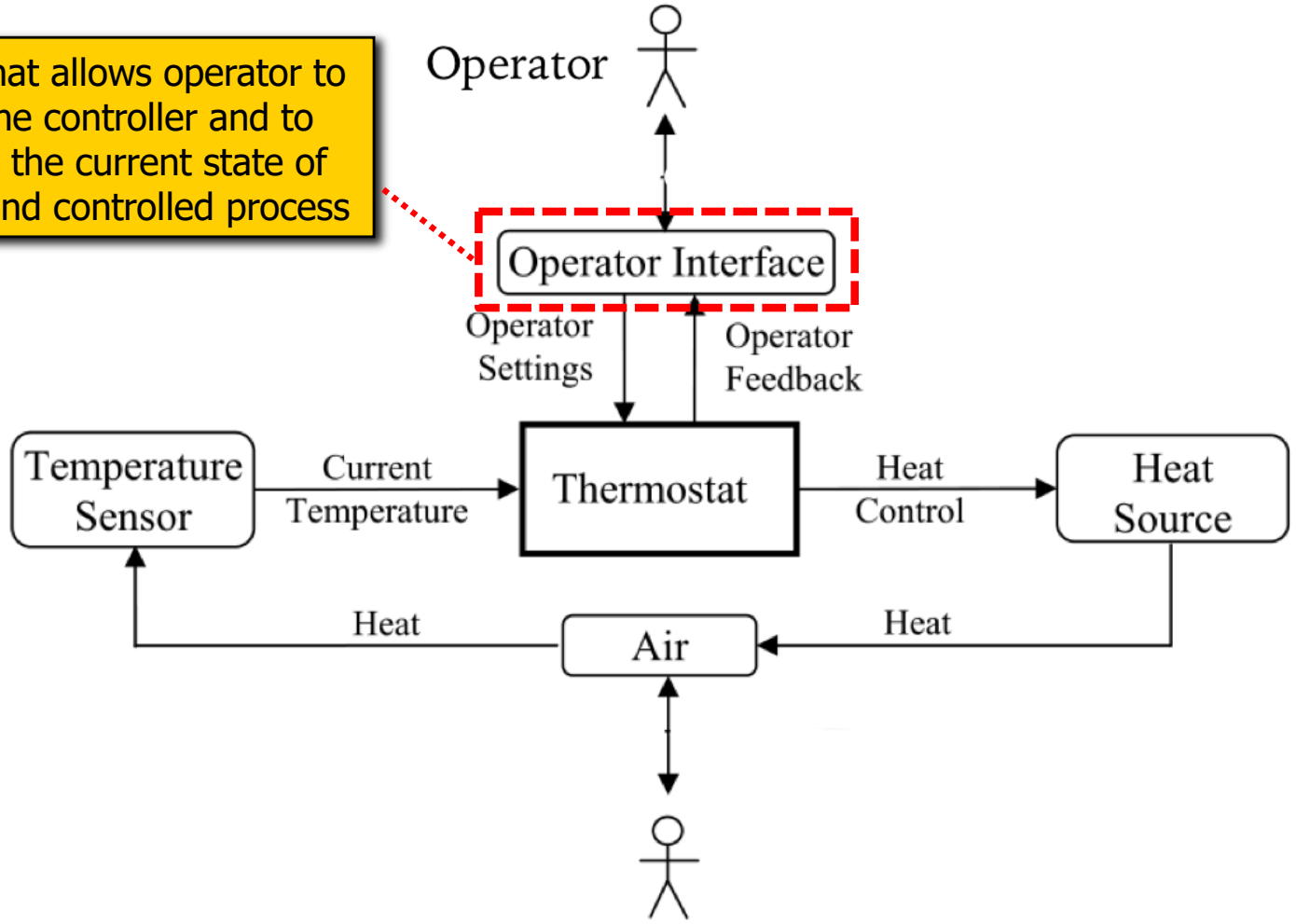


Intuitively, sensor provides the *input* to the controller and actuator provides the output. Parameters are read from the sensor and commands are sent to the actuators.

Common Components in a Safety Critical System - Operator Interface

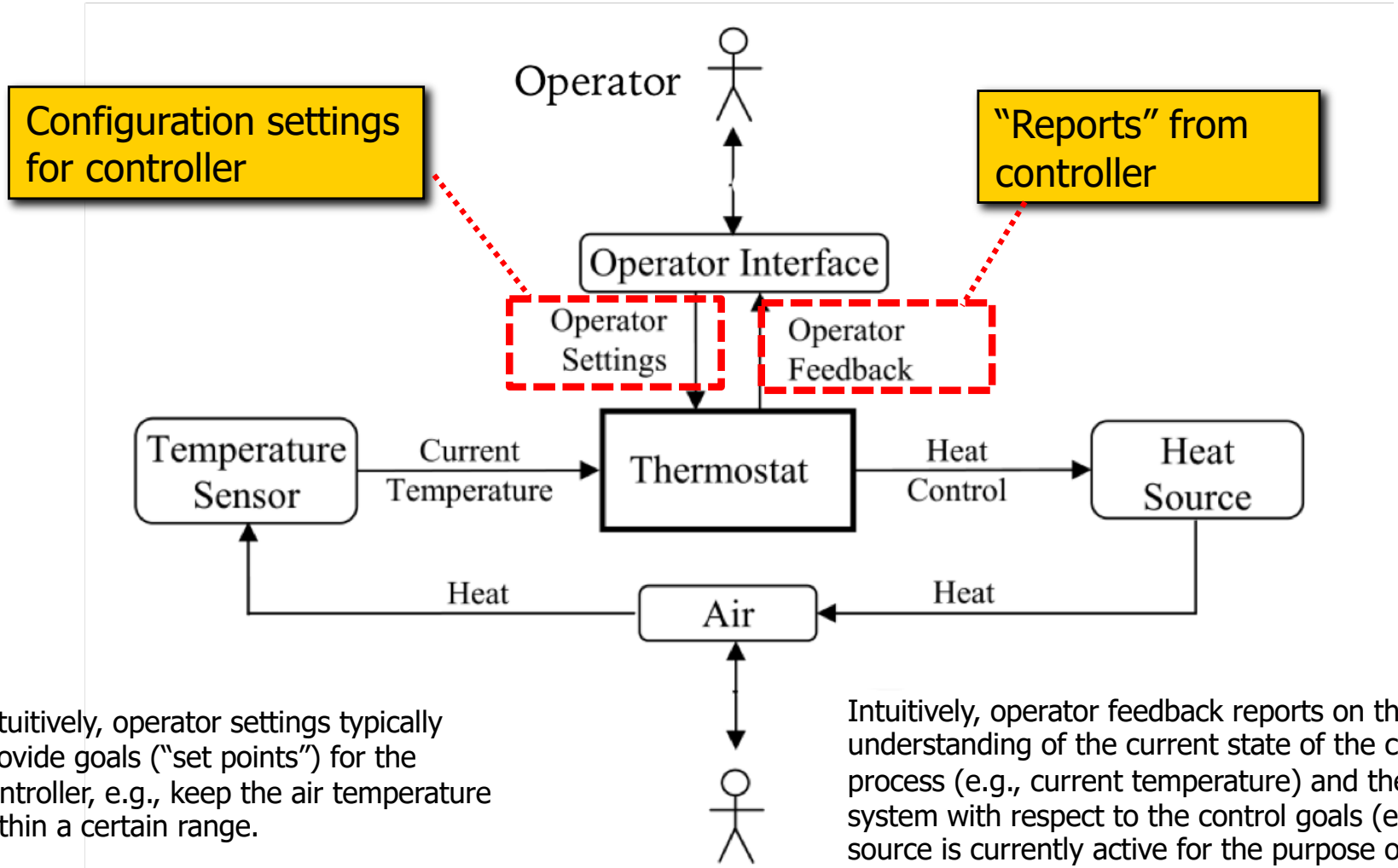
Operator Interface

Interface that allows operator to configure the controller and to understand the current state of controller and controlled process



Common Components in a Safety Critical System - Operator Interface

Inputs / Outputs from the Controller associated with the Operator



Intuitively, operator settings typically provide goals ("set points") for the controller, e.g., keep the air temperature within a certain range.

Intuitively, operator feedback reports on the controller's understanding of the current state of the controlled process (e.g., current temperature) and the status of the system with respect to the control goals (e.g., the heat source is currently active for the purpose of warming the air to get it up to within the desired temperature range).

Common Components in a Safety Critical System

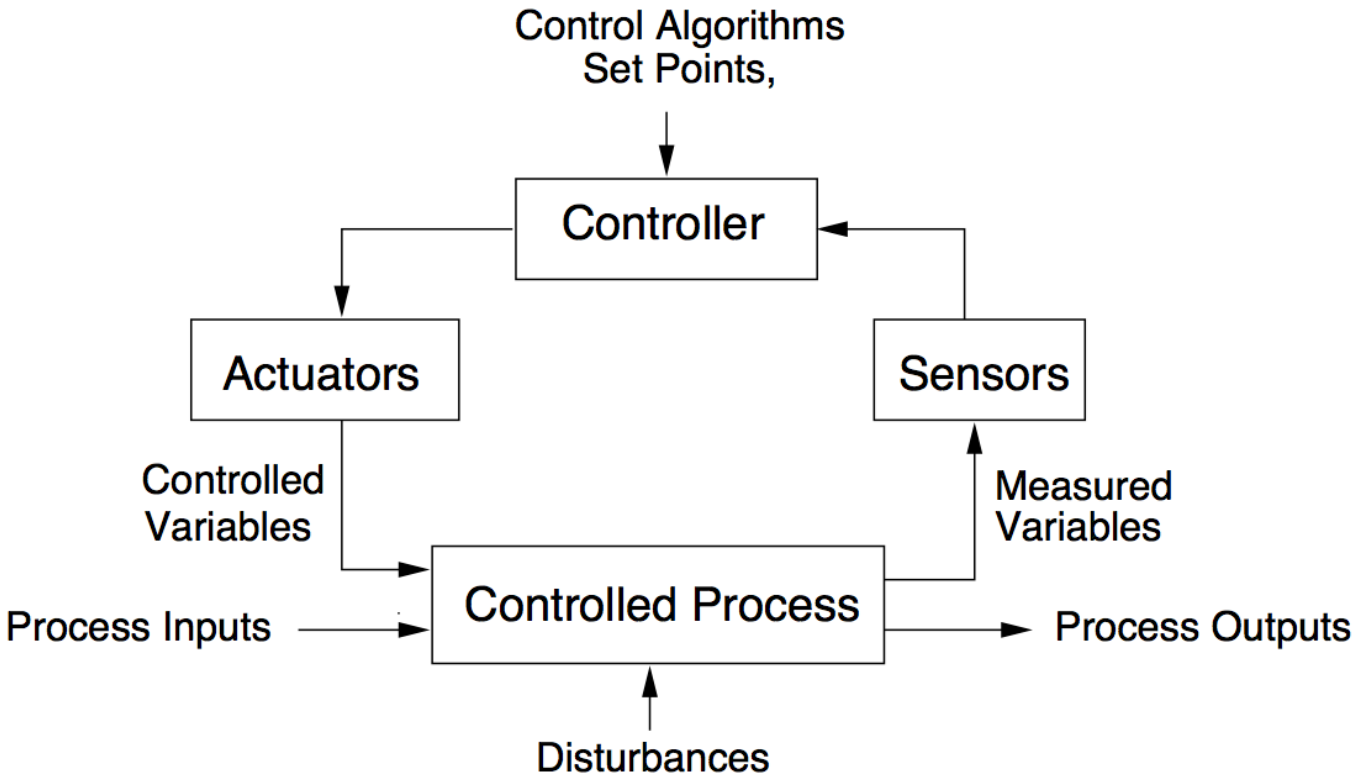


Figure 3.2: A standard control loop.

From "Safer World", Nancy Leveson

Control Loop Concepts

Common Components in a Safety Critical System

Mapping the standard control loop diagram to the Temp Control example…

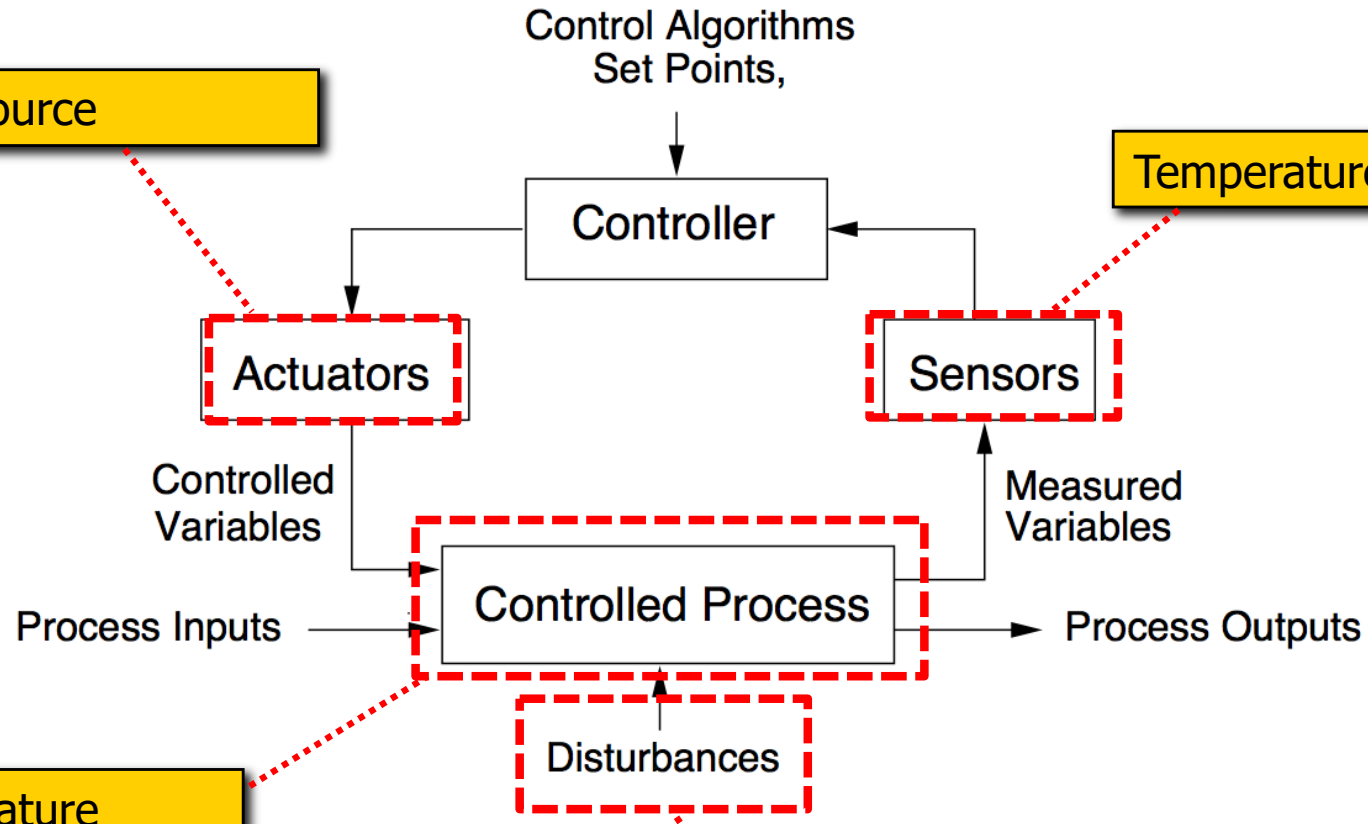


Figure 3.2: A standard control loop.

Heat Source

Temperature sensor

Air Temperature

Natural cooling of the air

From "Safer World", Nancy Leveson

Common Components in a Safety Critical System

Mapping the standard control loop diagram to the Temp Control example…

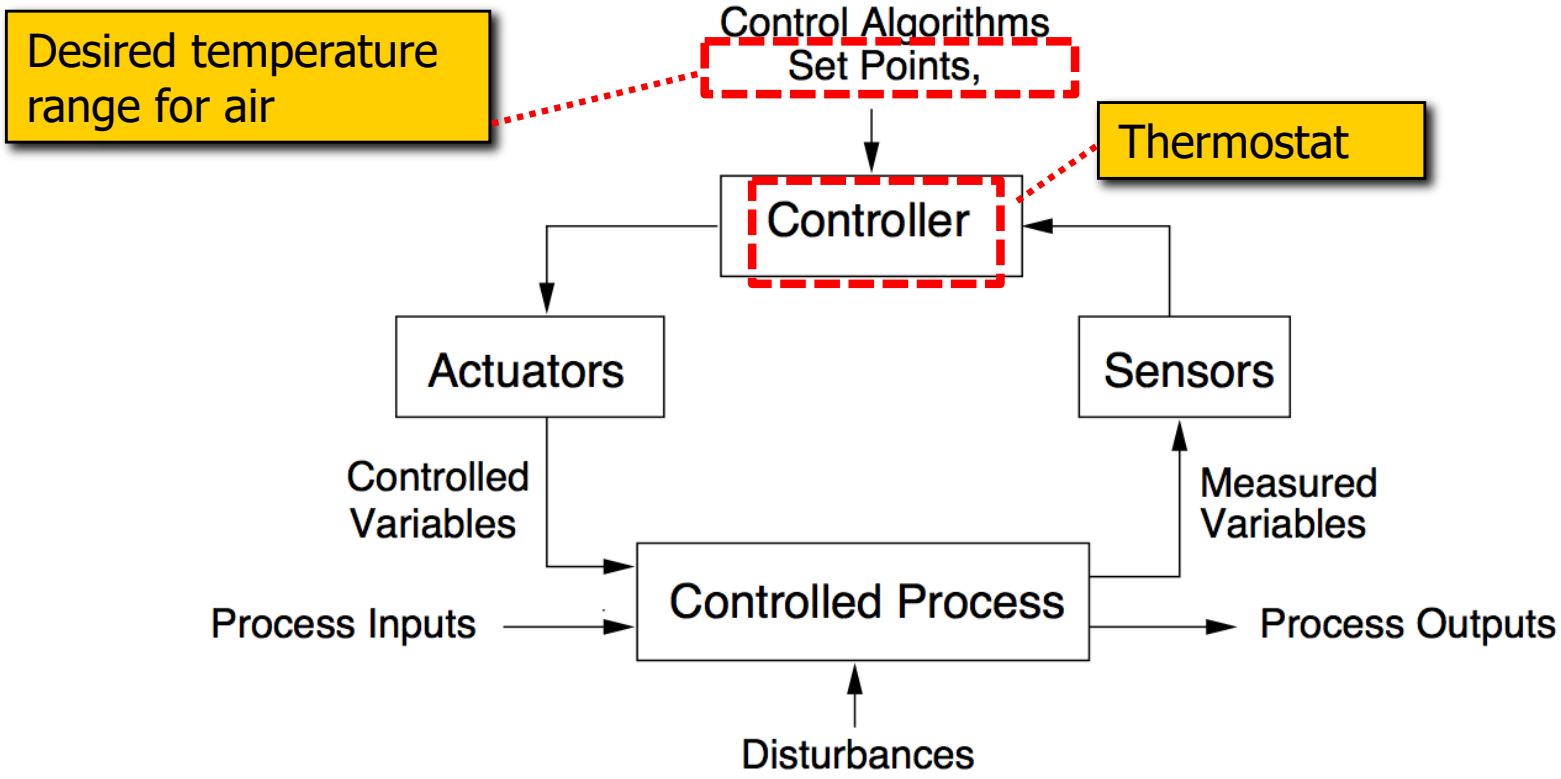
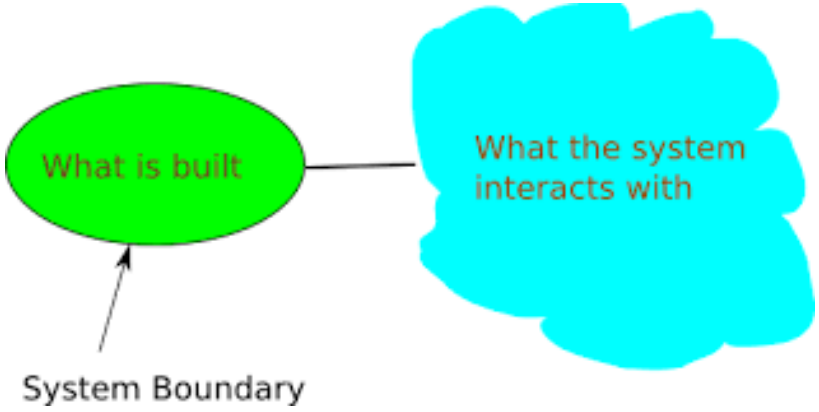


Figure 3.2: A standard control loop.

From "Safer World", Nancy Leveson

Boundaries

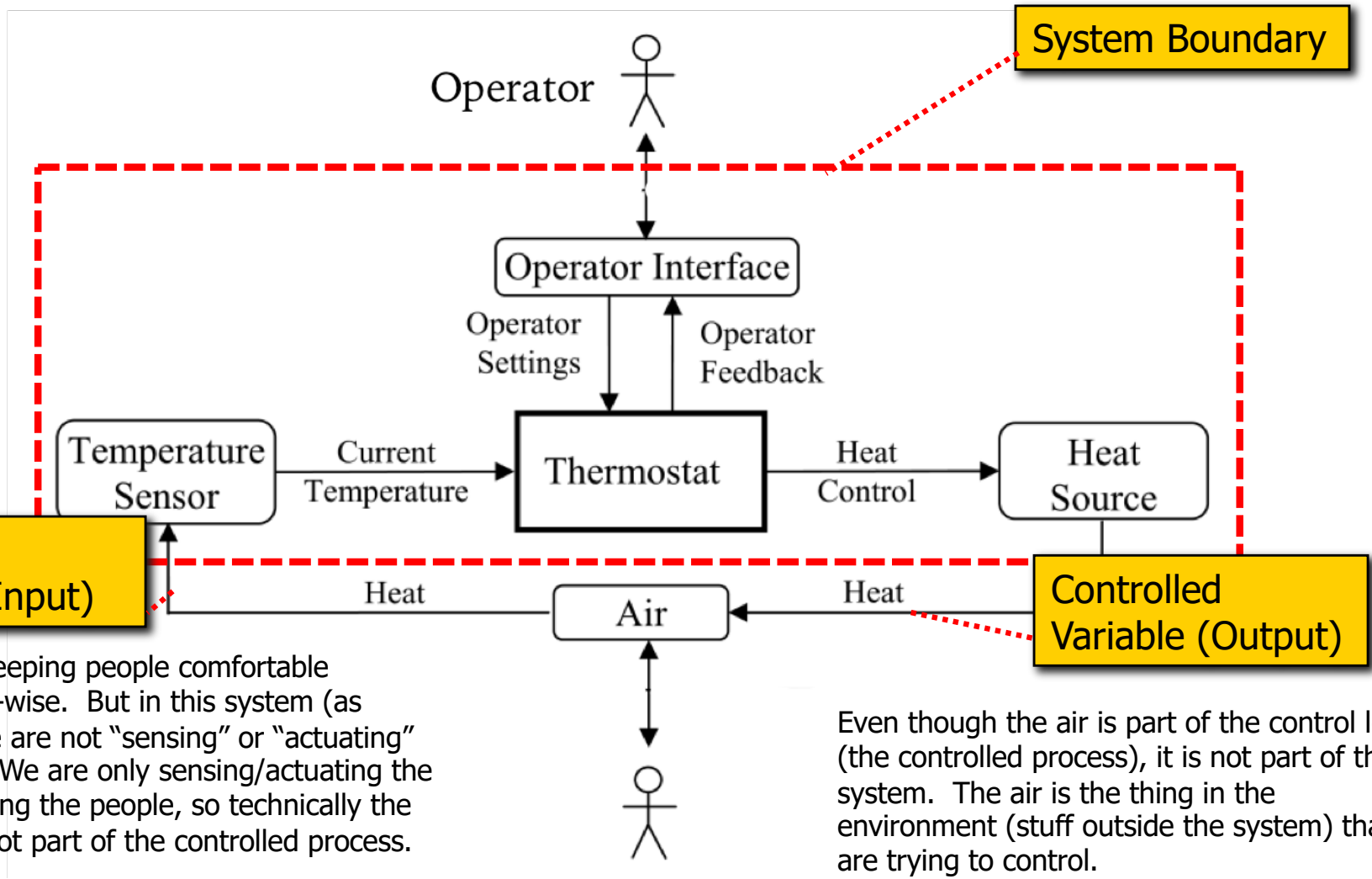
- Identifying boundaries is an important part of safety/security-critical development
- **Scoping of Responsibilities** -- The system boundary scopes what the system manufacturer is responsible for versus what is assumed by the environment into which the system is deployed
- **Scoping of Assumptions** -- An environment boundary may be described to indicate the entities that the system may interact with, and may also indicate how "far out into the world" the environment assumptions reach



System Boundary

Some subtleties… Are the people part of the controlled process?

Precisely defining the *system boundary* is one of the important steps in engineering a safety-critical. The engineer needs a very clear understanding on what exactly is being monitored & controlled, as well as the constraints that are to be maintained.

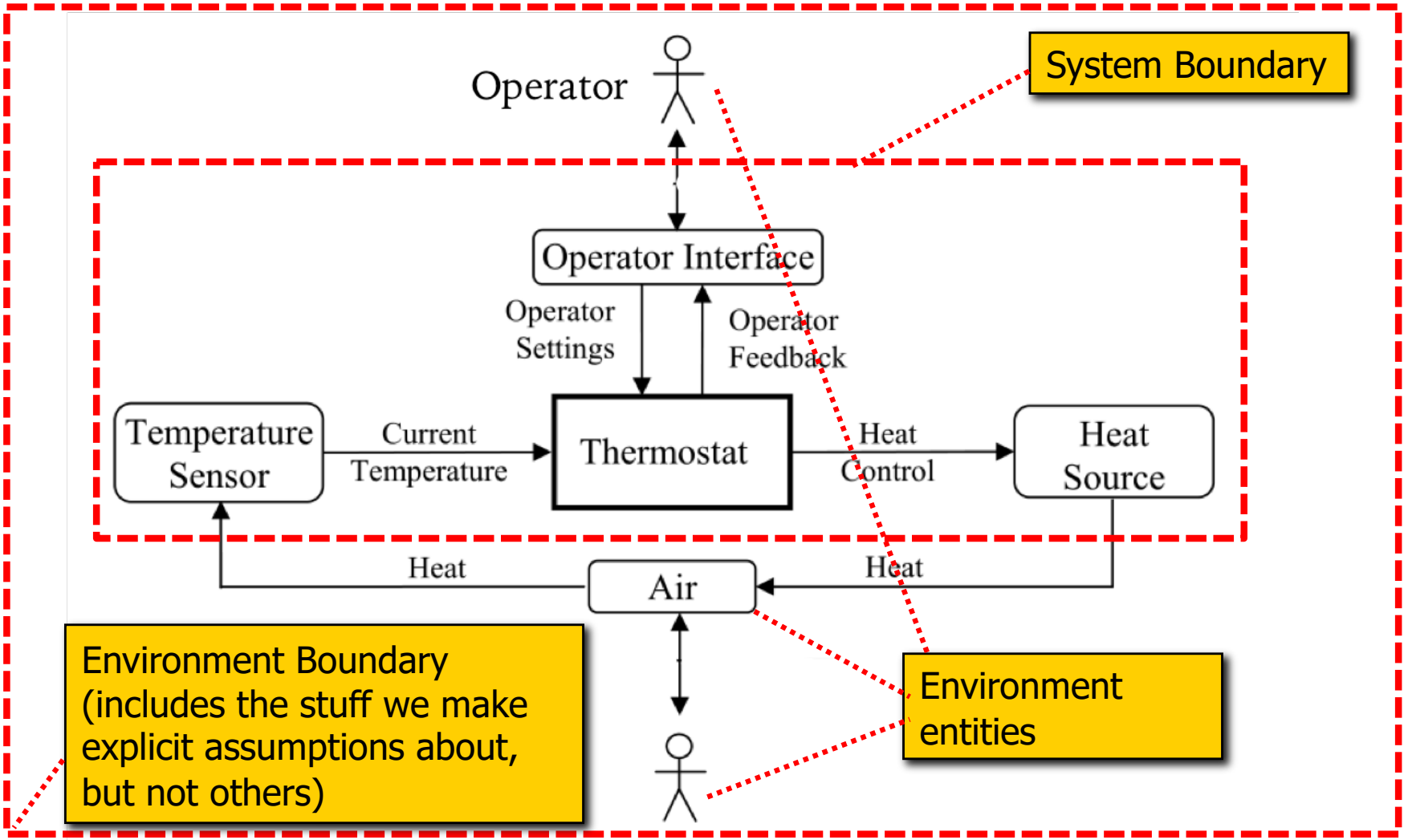


Of course, keeping people comfortable temperature-wise. But in this system (as depicted) we are not "sensing" or "actuating" the people. We are only sensing/actuating the air surrounding the people, so technically the people are not part of the controlled process.

Even though the air is part of the control loop (the controlled process), it is not part of the system. The air is the thing in the environment (stuff outside the system) that we are trying to control.

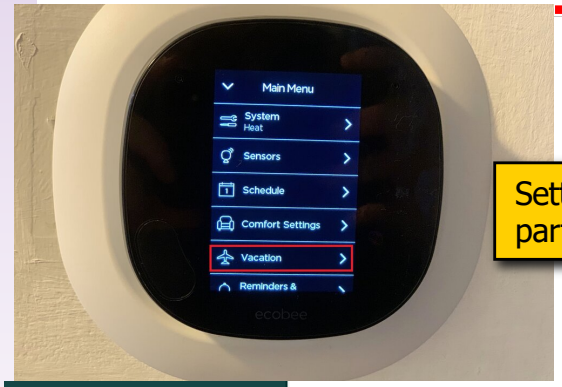
Environment+System as a System

There are many things in the environment of the system but our assumptions (e.g., about how fast the air cools) and our use cases capturing interactions between the system and entities in the environment only address a limited number of things. We enumerate those as part of our environment description.



Environment Boundary

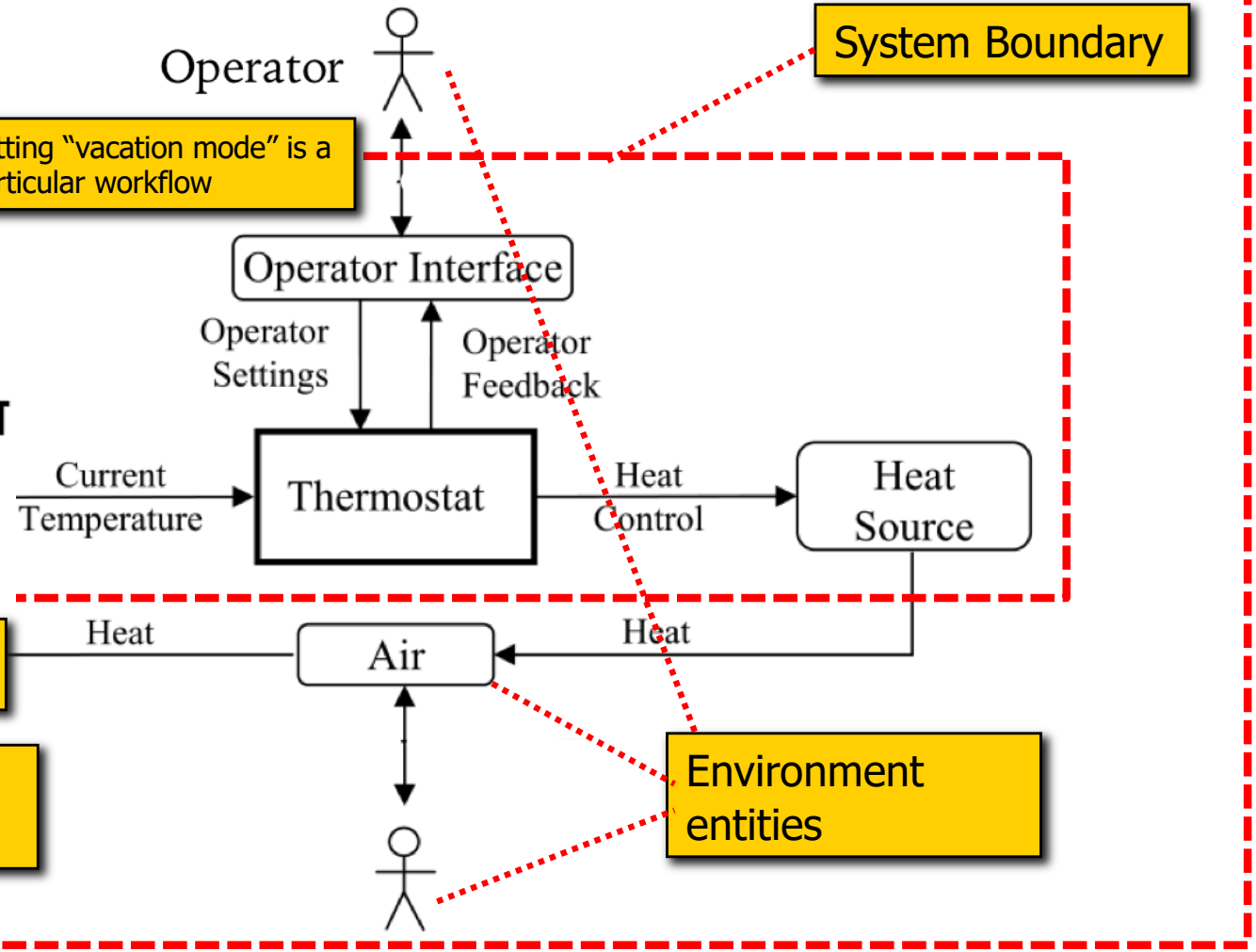
In some sense the manufacturered system plus the environment can itself be seen as a system of both computational, physical, and human actors. The recognized "workflows" associated with the operator are analogous to "algorithms" that the larger "system" is following.



HOW TO SET ECOBEE SMART THERMOSTAT VACATION MODE

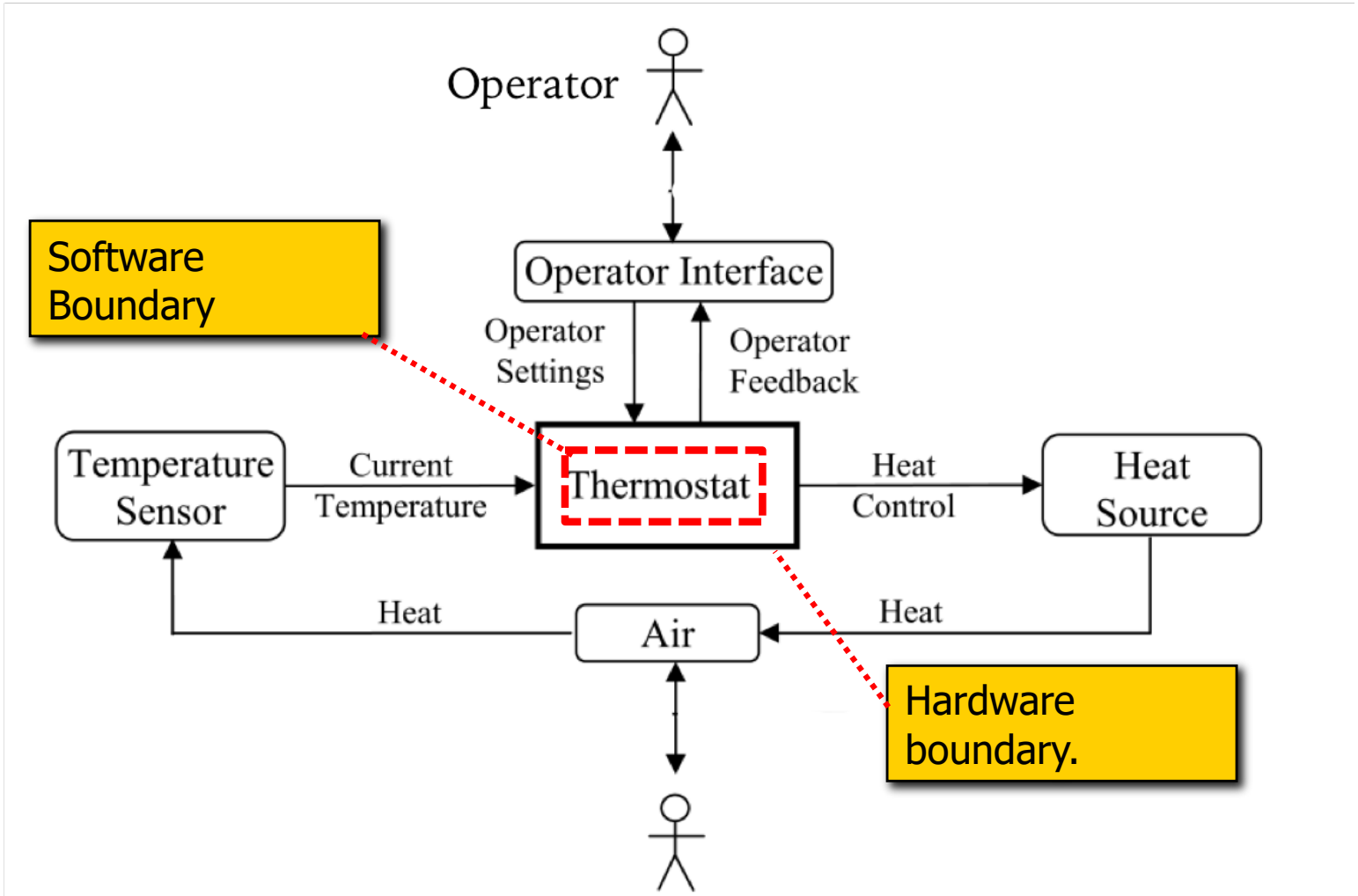
Setting "vacation mode" is a particular workflow

Owner's manual instructions provide the "algorithm" to follow



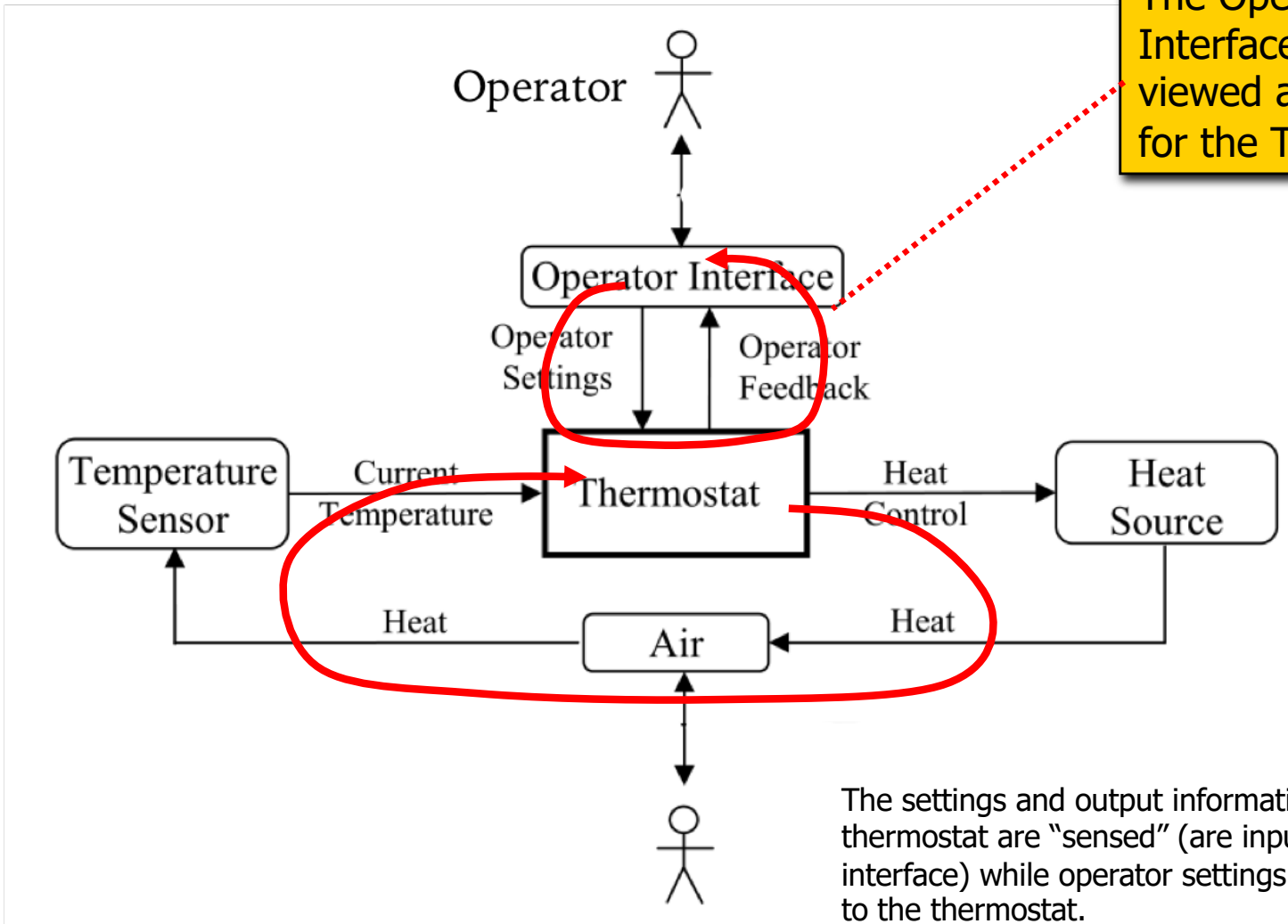
Hardware vs Software Boundary

There are other important notions of boundaries – such as the boundary of the software vs the boundary of the hardware. Consider that for the thermostat component, we likely have software running on a processor within the larger set of hardware elements for the thermostat.



Common Components in a Safety Critical System

Some subtleties… notice that we have two loops…

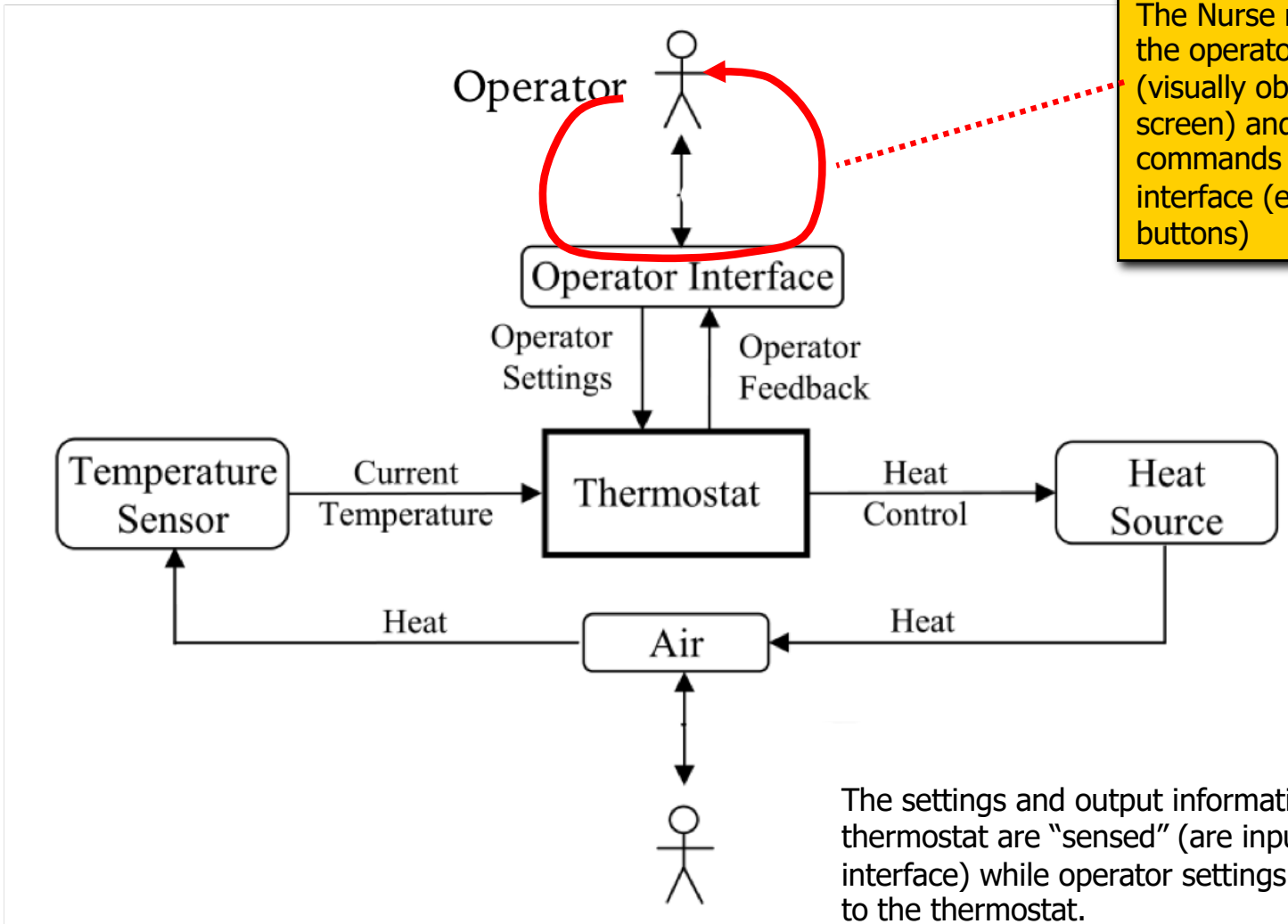


The Operator Interface can be viewed as a controller for the Thermostat.

The settings and output information from the thermostat are "sensed" (are inputs to the operator interface) while operator settings are "commands" to the thermostat.

Common Components in a Safety Critical System

Some subtleties… …and maybe even a third loop…

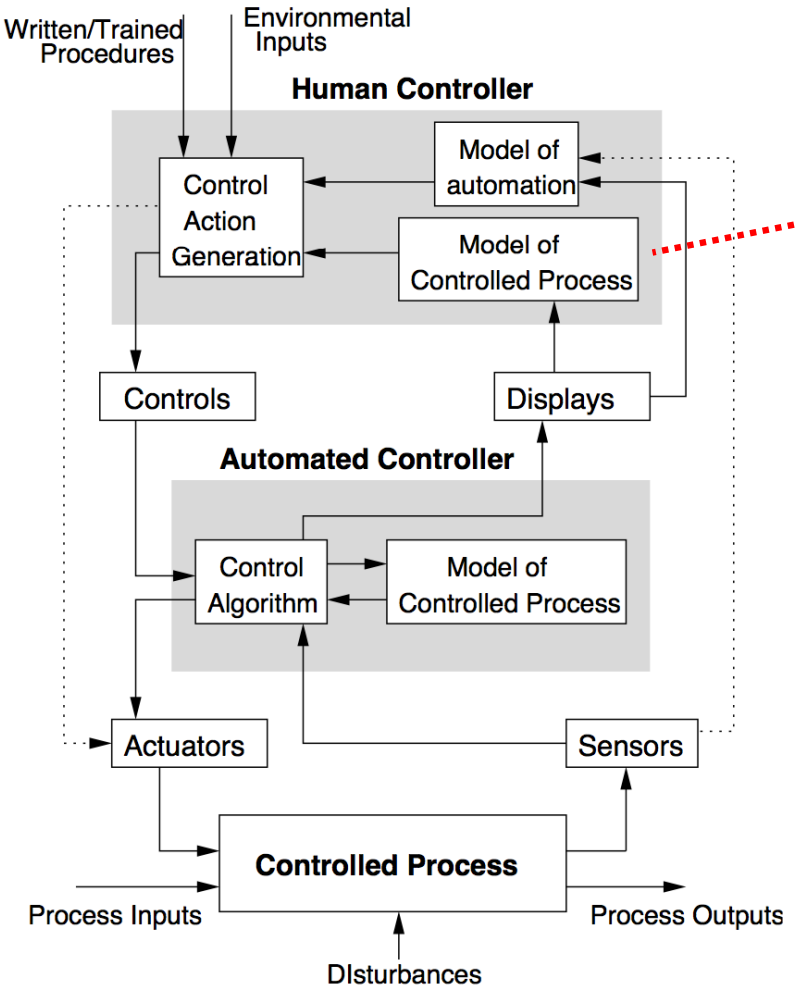


The Nurse monitors (senses) the operator interface (visually observing output screen) and controls (sends commands to) the operator interface (e.g., pushing buttons)

The settings and output information from the thermostat are "sensed" (are inputs to the operator interface) while operator settings are "commands" to the thermostat.

Control Loops

Leveson's *Safer World* book emphasizes a methodology for reasoning about system safety based on control loops called STPA (System Theoretic Process Analysis)



The "control logic" of the human is impacted by their training, the "mental model" that they have of the controlled process and automated controller, etc.

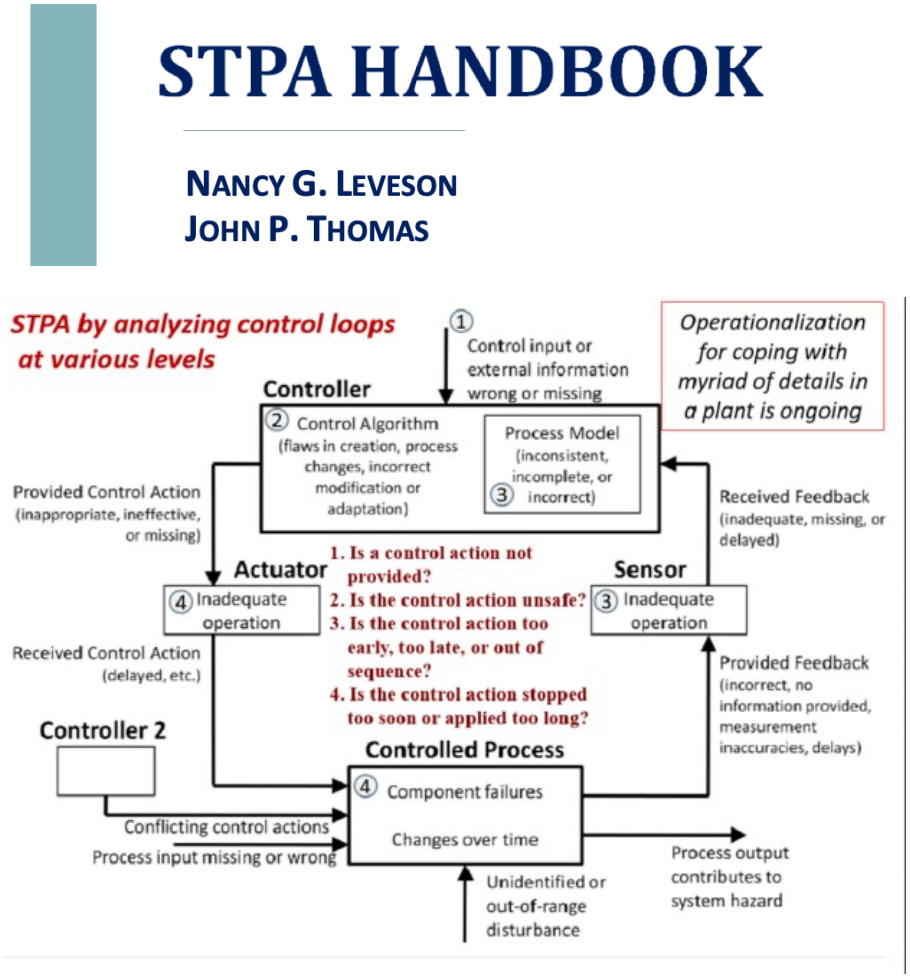
Many safety issues arise because the mental models of the human operator are not aligned with the actual state of the automated controller or controlled process.

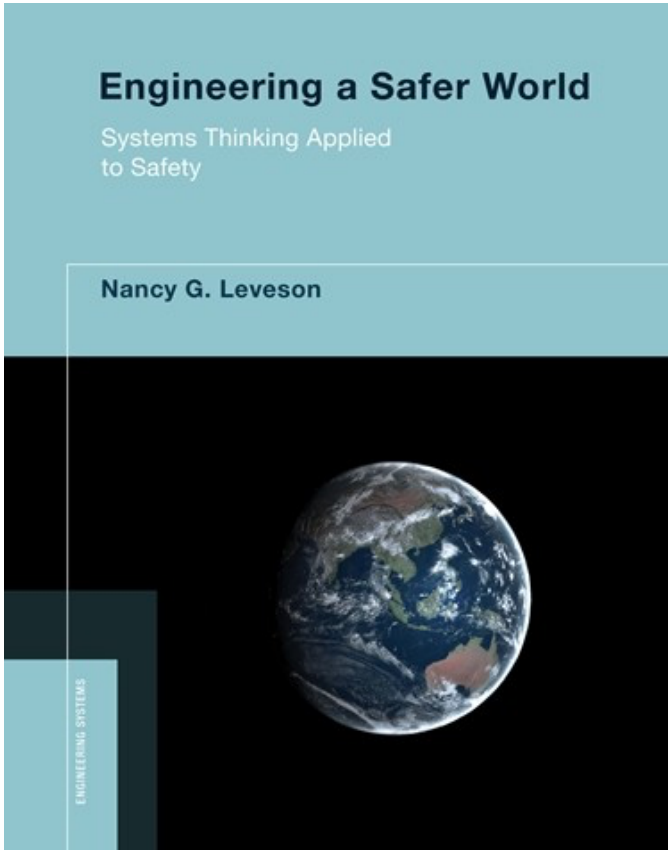
Figure 8.8: A human controller controlling an automated controller controlling a physical process

From "Safer World", Nancy Leveson

Using Control Loops to Support Safety Analysis

Leveson's System Theoretic Process Analysis (STPA) is a technique for hazard/safety analysis based on systematic examination of control loops within a system



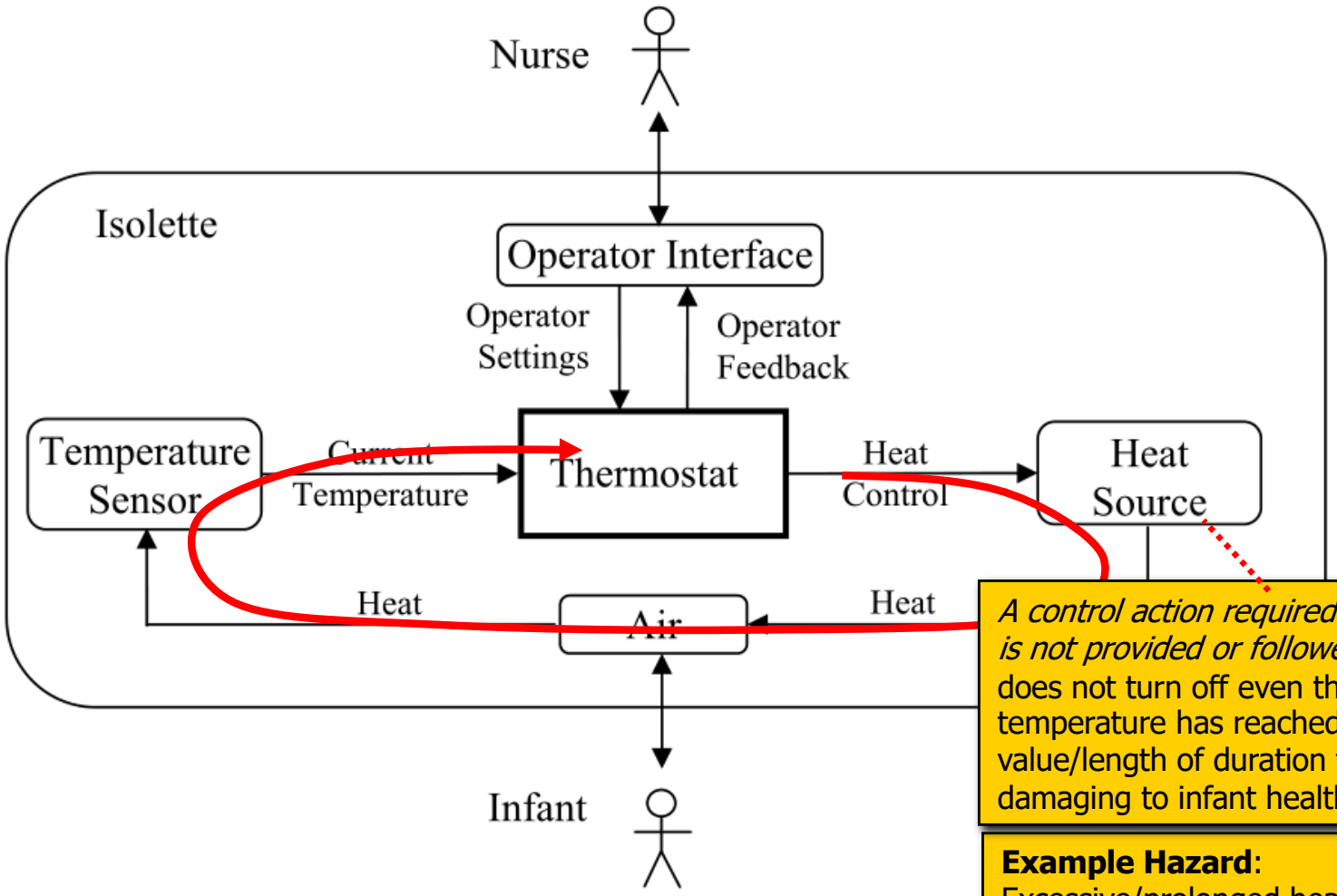


We'll use STPA in this course as our primary hazard analysis technique because it is very intuitive, and it ties in well to control algorihm design, requirements engineering, and testing.

http://www.flighttestsafety.org/images/STPA_Handbook.pdf

STPA in a Nutshell

Step 1: Working from the control loops in the system, identify the potential for inadequate control of the system that could lead to a hazardous state

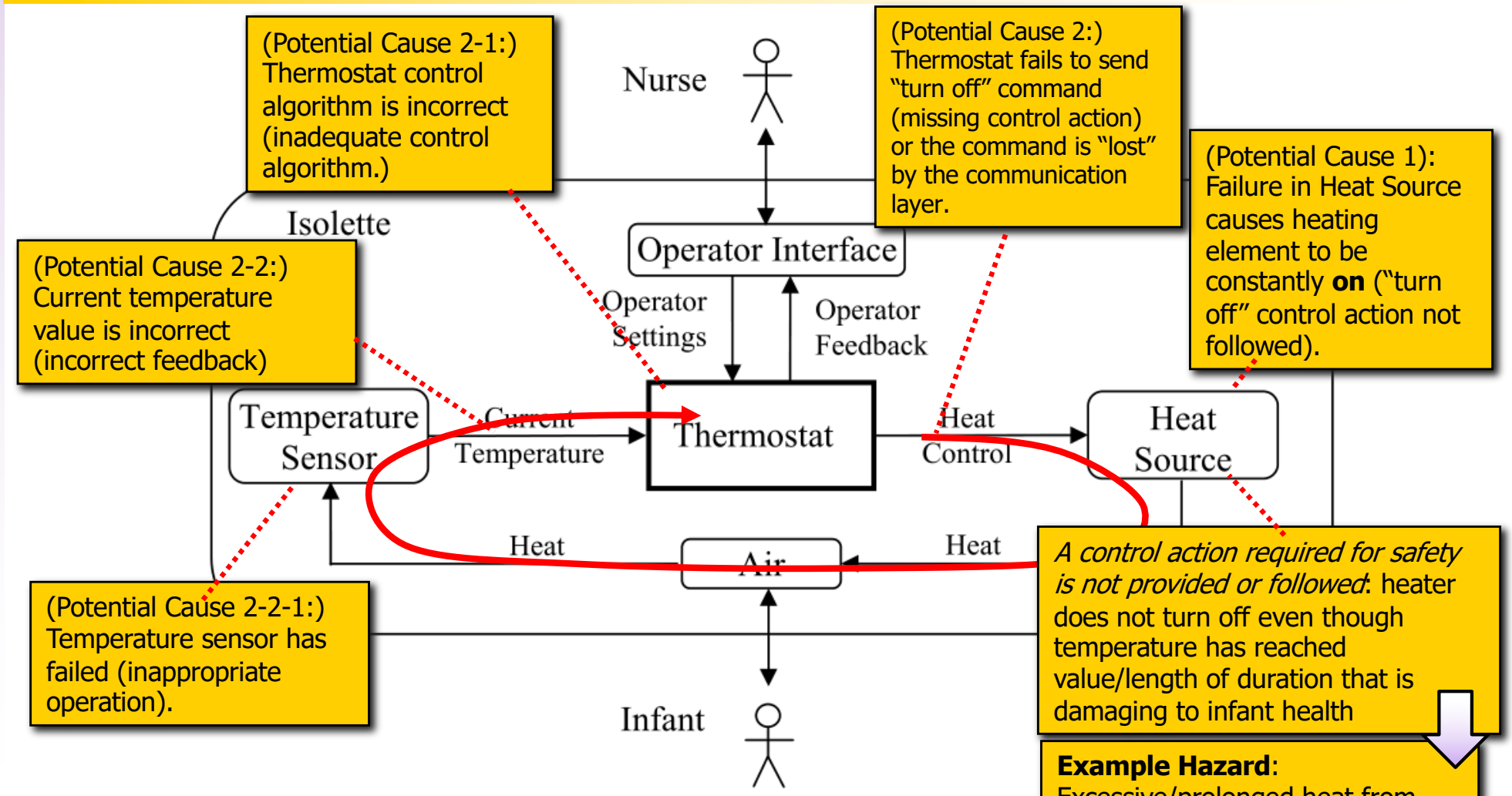


A control action required for safety is not provided or followed: heater does not turn off even though temperature has reached value/length of duration that is damaging to infant health

Example Hazard: Excessive/prolonged heat from Isolette heater harms the infant.

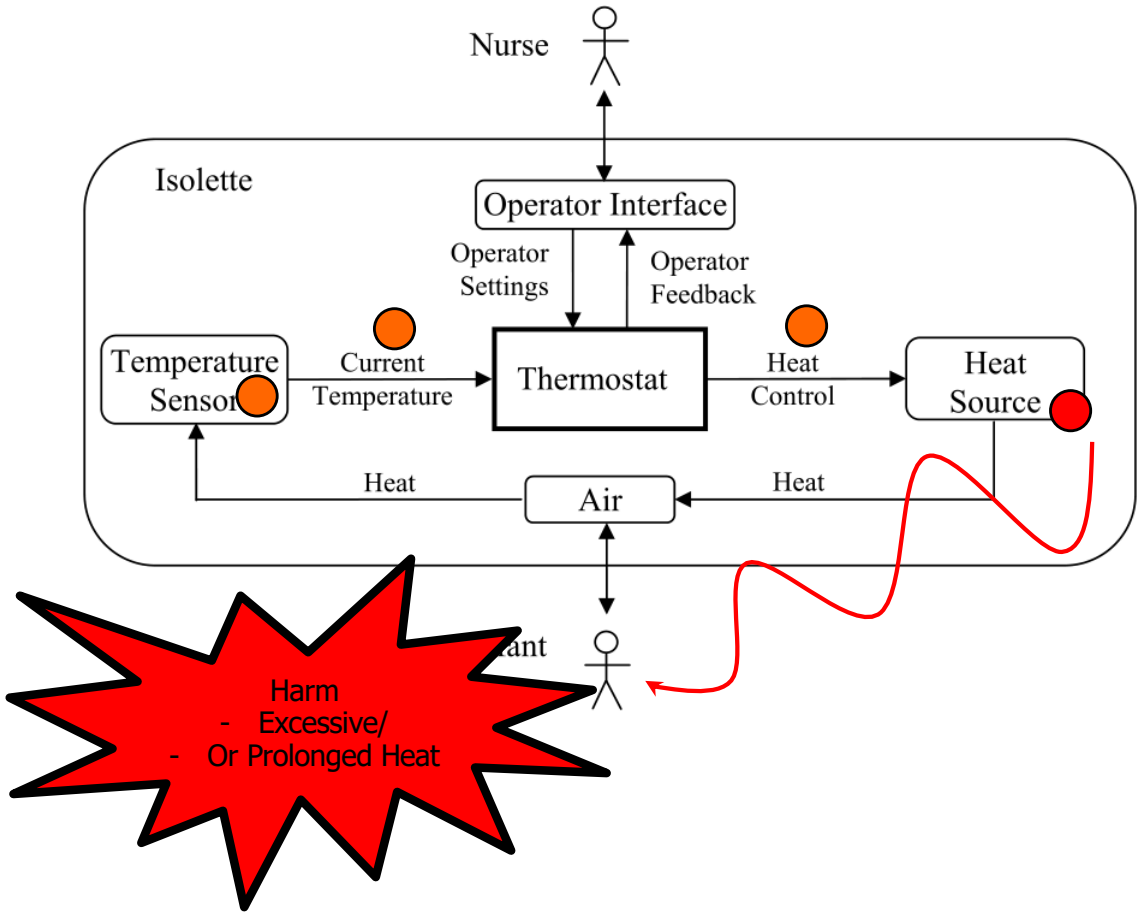
STPA in a Nutshell

Step 2: Determine how each potentially hazardous control action identified in Step 1 could occur (identify causes, causal chains)



STPA in a Nutshell

Example Causal Chain



(Potential Cause 2-2-1:) Temperature sensor has failed (inappropriate operation).

(Potential Cause 2-2:) Current temperature value is incorrect (incorrect feedback)

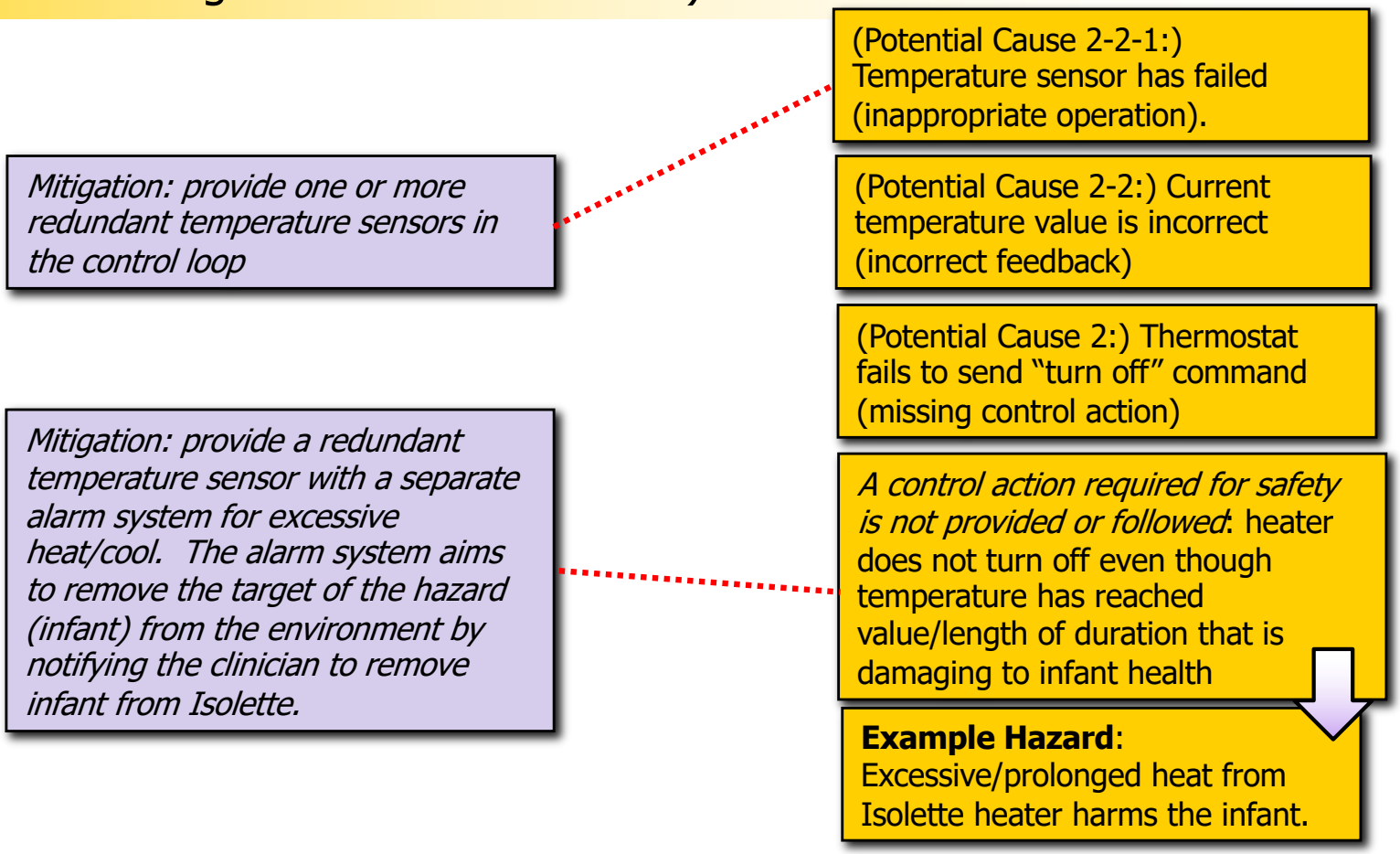
(Potential Cause 2:) Thermostat fails to send "turn off" command (missing control action)

A control action required for safety is not provided or followed: heater does not turn off even though temperature has reached value/length of duration that is damaging to infant health

Example Hazard: Excessive/prolonged heat from Isolette heater harms the infant.

STPA in a Nutshell

Study of the causality chains can help identify how the system design could be modified to avoid issues that "trigger" the problem or propagate problems. (referred to as "hazard mitigation" or "risk control")



Summary

- The notion of a control loop is a foundational concept in embedded systems. It leads to an understanding of…
  - Inputs, outputs, and system objectives
- Reasoning about control loops can also help us understand how a system can be unsafe and how failures (or security attacks) in different parts of the system can lead to safety problems
- We will see later that reasoning about control loops in terms of their inputs (monitored variables) and outputs (controlled variables) can support a methodologies for requirements design and testing