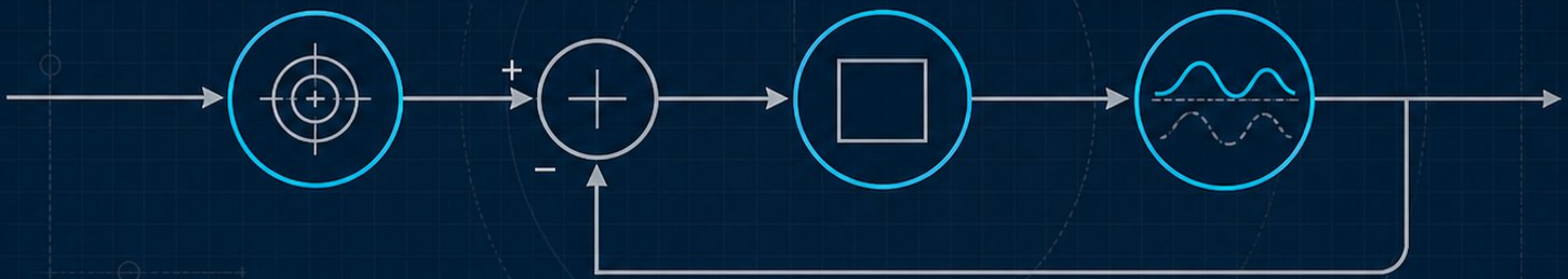
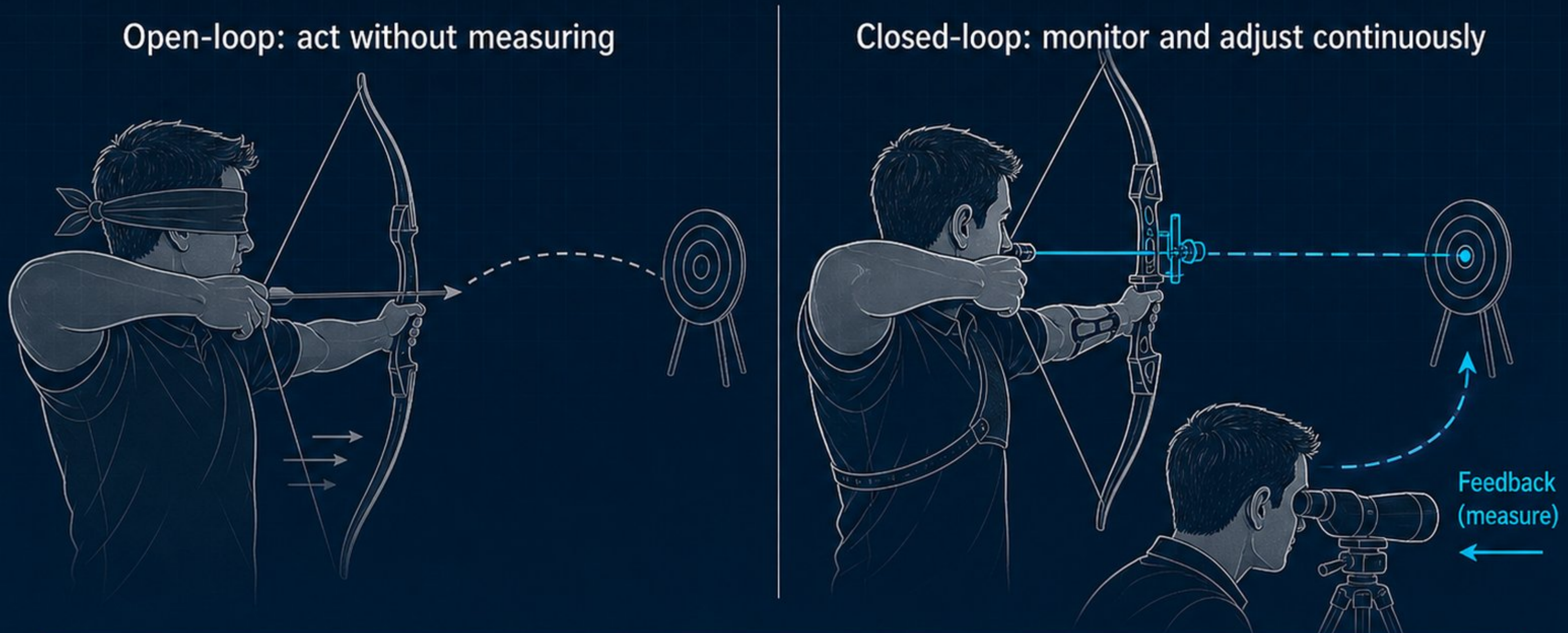
Introduction to Control Systems: Feedback and Stability



- A control system regulates a variable to achieve a desired state, from thermostats to biological processes.
- Core question: How do feedback, goals, error, and stability explain self-correction or failure?
- Roadmap: Goal → error signal → corrective action → stability/instability → real-world diagnosis.

What Does It Mean for a System to Self-Correct?

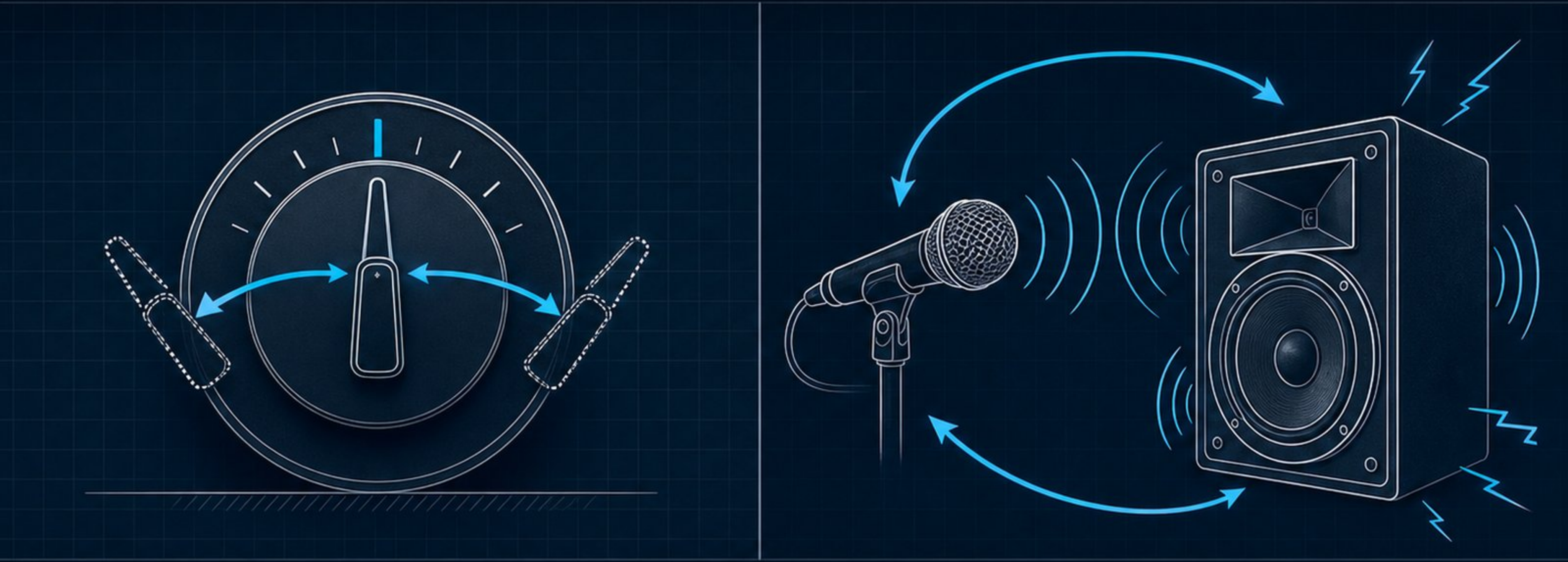


- A goal (setpoint) turns random change into meaningful deviation
- Self-correction cycle: compare → detect error → act to reduce it
- Open-loop: act without measuring; Closed-loop: monitor and adjust continuously
- Examples: thermostat, body temperature, inventory adjustment

The Feedback Loop: Four Essential Building Blocks

- Sensor measures the current state
- Comparator finds the gap (error) from the goal
- Controller decides what corrective action to take
- Actuator executes the change
- Feedback closes the loop: output is measured to adjust input

Negative Feedback:
The Engine of Self-Correction



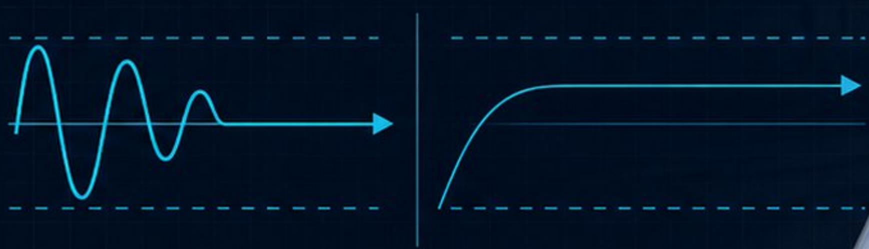
- Opposes any deviation: drives high output down, low output up
- Pulls system toward goal despite disturbances, creating stability
- Positive feedback amplifies deviation, driving system away from equilibrium
- Thermostat (negative) vs. crowd panic or microphone screech (positive)

Goals, Error, and the Correction Equation

- Error drives all correction: the gap between what is and what's desired
- P-control: response strength is proportional to the error
- P alone leaves steady-state error (close but never perfect)
- PID concept: P reacts to now, I remembers the past, D anticipates the future
- Ship captain metaphor: P sees course error, I corrects for wind, D avoids overshoot

Stability: When Correction Succeeds

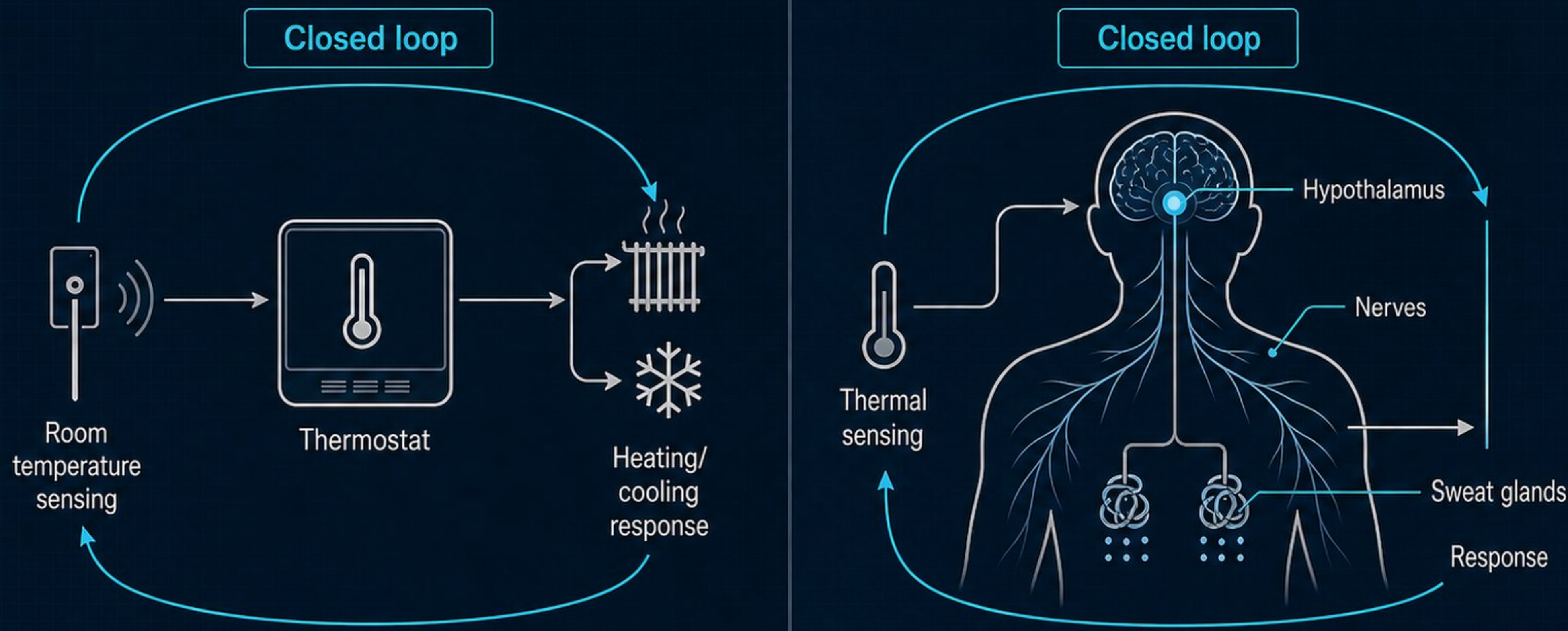
- Stability: system settles smoothly near the goal, no wild swings
- Critically damped: corrects quickly without overshooting
- Key traits: minimal oscillation, fast settling, zero steady-state error
- Modern cruise control holds speed smoothly on flat roads



Instability: When Correction Fails

- "Same feedback that corrects can cause **instability**"
- "Too **aggressive**, too **slow**, or **delayed** response"
- "Excessive gain triggers **oscillation** and **overshoot**"
- "Delayed feedback **amplifies** errors over time"
- "Sensor noise can make **jittery corrections** worse"

From Devices to Organisms: The Same Principles Apply



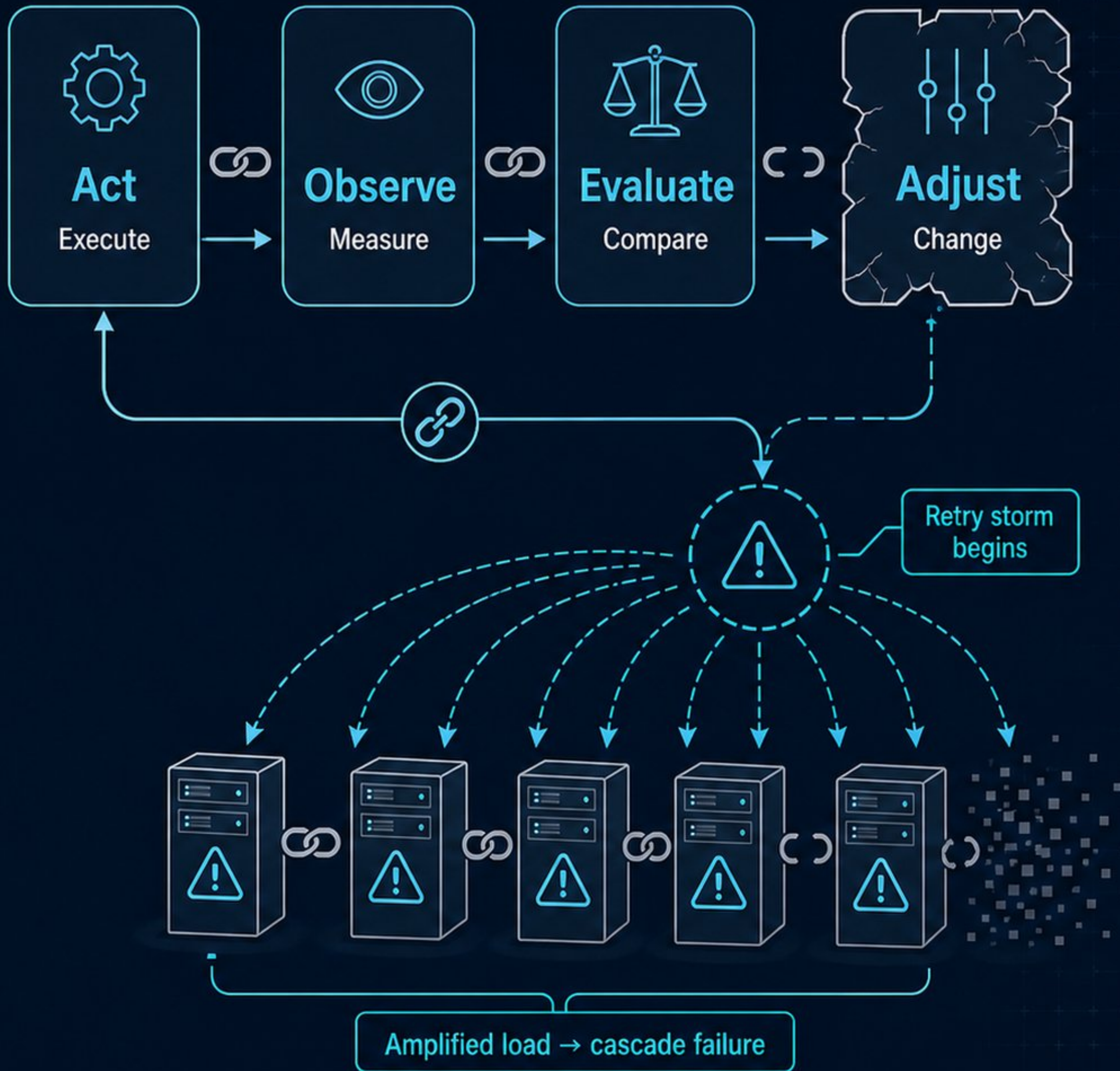
- Body temperature regulation: hypothalamus as controller
- Blood sugar: a tight negative feedback loop
- Robust despite meals, exercise, or temperature shifts
- Sensor-comparator-controller-actuator structure is universal

Feedback in Organizations and Markets

- "Inventory control adjusts orders based on sales vs. target stock levels"
- "Customer complaints trigger fixes, implementation, and follow-up checks"
- "Employee reviews set goals, measure performance, identify gaps, and correct"
- "Any goal-measurement-adjustment loop is a control system you can optimize"

Diagnosing Broken Loops: A Practical Framework

- - Loops fail at four points: Act (no execution), Observe (no measurement), Evaluate (no comparison), Adjust (no change).
- - Diagnostic checklist: Is the goal clear? Is error measured? Is feedback delayed or too extreme?
- - Use 'Act-Observe-Evaluate-Adjust' to find the first missing component.
- - The 'retry storm': a service failure triggers automatic retries, amplifying load into a cascade—runaway positive feedback.



Legend: Broken Link

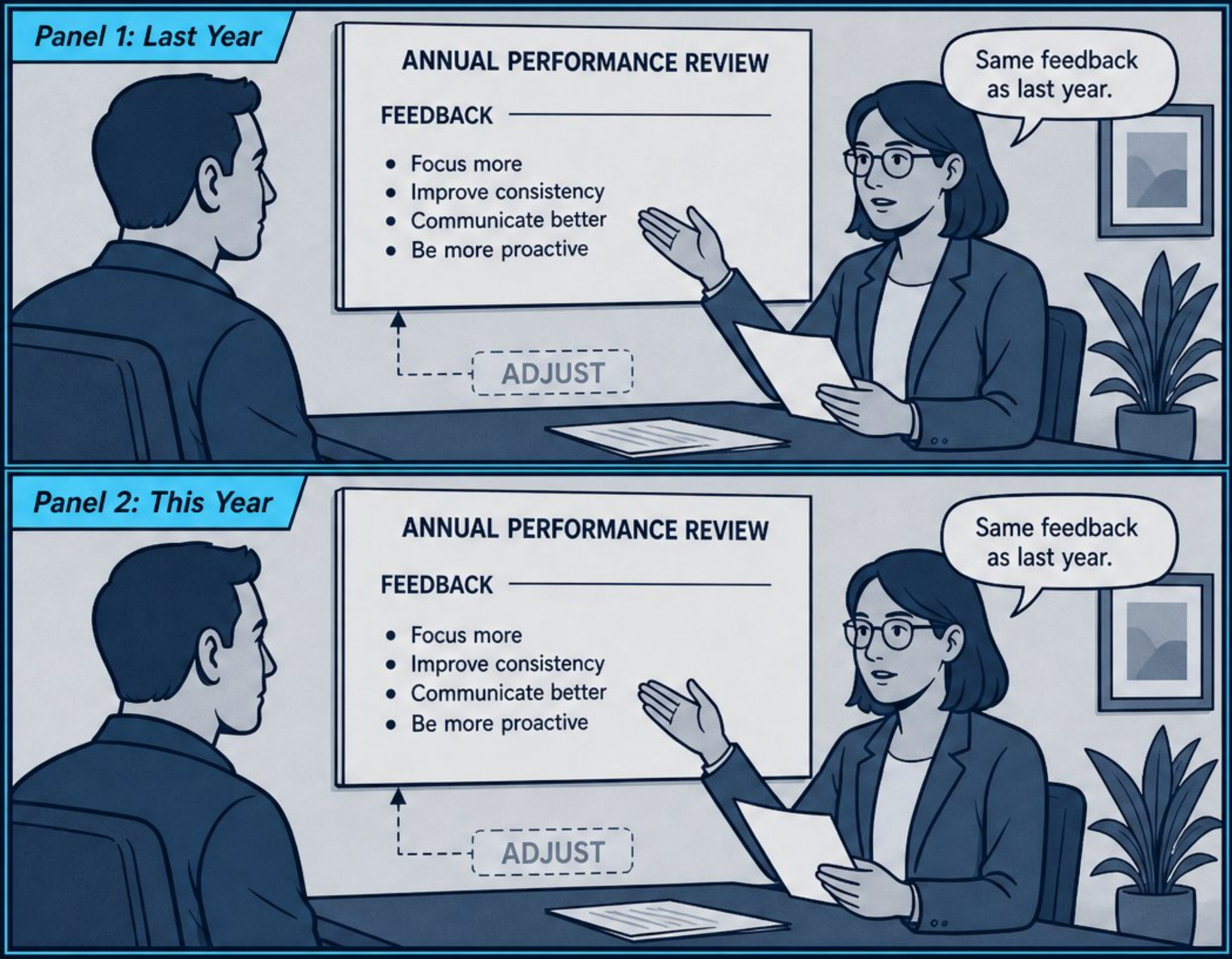
Represents a missing or failed connection that breaks the loop.

Real-World Failure Patterns (Part 1): Missing or Distorted Information

- **Blind loop:** acting without observing — "We're busy but don't know if it works."
- **Meaningless loop:** data collected, never compared to a goal — "Dashboards nobody uses."
- **Selective observation:** praising Slack feedback while ignoring a 12% adoption rate.
- **Fix:** a clear sensor, a defined success standard, and honest comparison.

Real-World Failure Patterns (Part 2): Delays and Inaction

- Delays cause oscillation: you correct too soon, then overshoot.
- Missing the Adjust step is the most painful loop failure.
- You know what's wrong but don't change behavior.
- Example: same performance gap repeats quarter after quarter.
- Rule: correction must outpace the change you're managing.



Missing the Adjust step creates a loop that repeats.
Break the loop by acting—not by waiting.

Tuning for Success: The Art of Not Overshooting

- Tuning balances three competing goals: speed, accuracy, and stability
- Aggressive (high gain) tuning: fast response but risks oscillation
- Gentle tuning: slow and stable but may never reach the target
- The derivative (D) term acts as a brake to prevent overshoot
- Always fix broken loop components before fine-tuning the response

Applying the Control System Lens: A 3-Step Approach

- **Step 1:** Identify the clear goal and what is being measured.
- **Step 2:** Map the Sensor, Comparator, Controller, and Actuator.
- **Step 3:** Analyze if the dominant feedback is negative or positive, and check for delays.

This turns any messy problem into a diagnosable broken control loop.

Summary: Feedback Turns Error Into Correction; Stability Keeps It Safe

- Feedback = measure, compare, correct in a closed loop driven by error
- Self-correction works when negative feedback dominates and is properly tuned
- Stability is the boundary between effective correction and destructive oscillation
- Apply this model to debug systems in engineering, biology, and business