OPERATIONAL INTELLIGENCE NEWSLETTER

# Closing the Loop: Building Production Systems That Think for Themselves

*How to move past dashboards and establish true, automated process control across your network of plants.*

Manufacturing Excellence · OT/IT Convergence · Process Control · Operational Risk

> The newsletter for operations teams that do not mistake data visualization for embedded capability.

## Summary: The Leap from Sight to System

There is a significant difference between seeing a problem and having a system that prevents it. Most manufacturing plants today are drowning in data but starving for control. You’ve likely invested heavily in sensors, SCADA systems, and real-time dashboards. These tools tell you exactly what happened five minutes ago, an hour ago, or yesterday. They provide "sight." But sight is not the same thing as a system that thinks for itself.

A dashboard is essentially a digital mirror; it reflects the current state of your floor. If a motor overheats and triggers a red light on a screen in a remote office, you have successfully gathered data—but you haven't solved the problem. You’ve just moved the point of observation. To move from "sight" to "system," we have to shift our focus from notifying humans that something is wrong to building processes that automatically correct deviations before they turn into scrap or downtime. We aren't looking for more data; we are looking for closed loops where the intelligence of your best operators is baked into the logic of the machine itself.

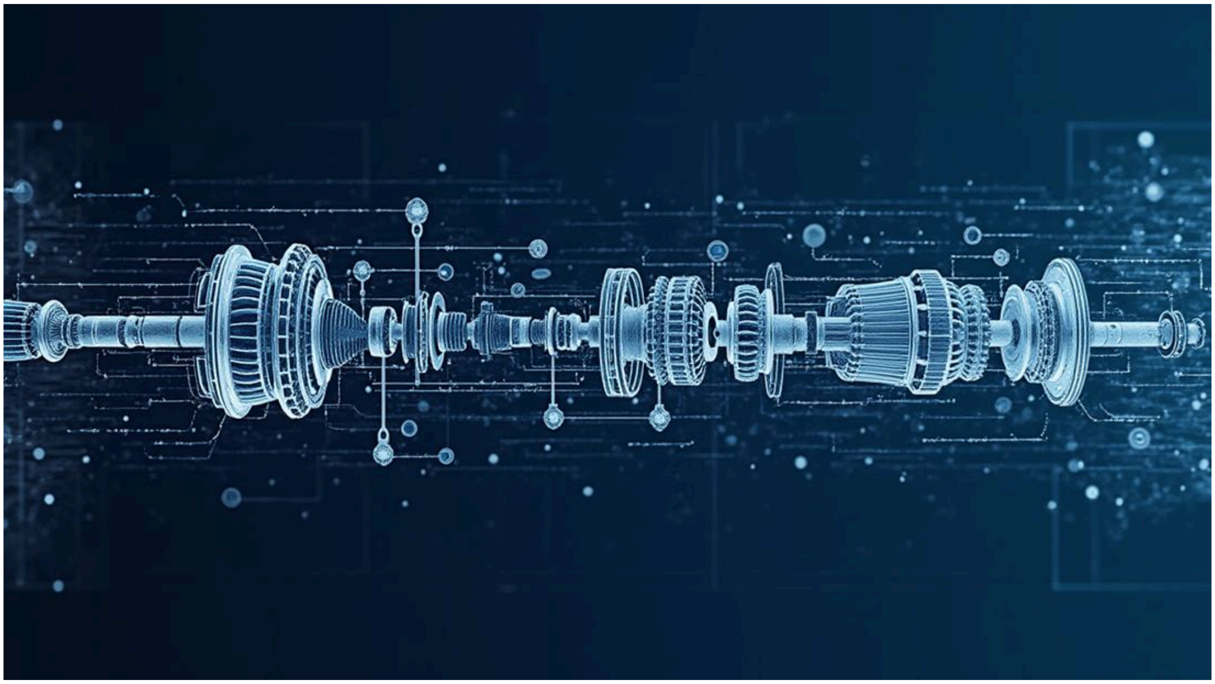
## The Dashboard Trap: Why Seeing Data Isn't Doing Work

We often fall into a trap called **The Observation Illusion**. It's easy to believe that because you can see a metric on a screen, you are "managing" it. In reality, if a human has to notice a number go out of spec and then decide what to do about it, the system is still fragile. You are relying on human vigilance—a finite resource that fails when people get tired, distracted, or overwhelmed by too many alerts.

When we rely on humans to bridge the gap between "detecting" and "correcting," we create a delay. That delay is where quality dies. If an operator has to walk across a floor, find a tool, and manually adjust a setting because they saw a notification, that intervention is reactive. You are simply managing the fallout of a failure that already happened.

| The Comfortable Rationalization | The Underlying Reality |
| --- | --- |
| "We have real-time dashboards to monitor the line." | We have an alarm system that notifies us after the process has drifted. |
| "The operators are trained to react to alerts." | The response depends on human availability, focus, and speed of movement. |
| "Data transparency improves our ability to respond." | Data without automated correction is just a record of what went wrong. |

## Why The Gap Remains (Even When We Have Sensors)



The reason most plants stay stuck in the "Dashboard Trap" isn't because they lack technology; it's because they haven't redesigned the ownership of the process. It is technically simple to install an IoT sensor that monitors humidity or temperature. It is much harder—and more valuable—to write the logic that allows that sensor to automatically trigger a change in the HVAC system or slow down a conveyor belt when those levels drift.

The gap exists because it's easier to build a "report" than it is to build a "control." A report tells someone what happened; a control ensures something happens. To close this gap, we have to move from human-mediated loops to and machine-enforced ones. We need to stop asking the operator to be the glue between two disconnected pieces of equipment. If your best technician knows exactly how much they would turn a valve if a pressure spike occurred, that knowledge belongs in the PLC logic, not just in their head or in an SOP binder on a shelf.

## The Cost of the Half-Measure: Risk, Deviation, and Downtime

When we settle for "sight" instead of "system," we are essentially betting that our people can react faster than the process can fail. This is a high-stakes gamble with three primary costs:

1. **The Cost of Scrap:** When a deviation occurs and waits for a human to notice it, you aren't just losing time; you are producing non-conforming parts. By the time someone sees the red light on the dashboard, several bins of product may already be out of spec.
2. **The Cost of Equipment Damage:** Many failures don't stay "small." A slight drift in lubricant pressure might not cause a failure today, but it will grind down a bearing over the next 48 hours. A human-monitored system only catches the problem once the machine is already broken.
3. **The Cost of Non-Compliance:** In highly regulated environments, "oops" isn't an option. If a process drifts outside of its validated window for even a few minutes while waiting for a supervisor to intervene, that entire run may need to be quarantined or scrapped.

A delayed intervention is often just as expensive as no intervention at all; it's simply harder and more public to fix after the fact.

## Designing True Intelligence: The Connect-Detect-Correct-Act Loop

To move toward a self-thinking system, we have to replace "human-in-the-loop" with a closed-loop architecture. This isn't about removing people from the floor; it's about removing them from the tedious task of monitoring and correcting trivial deviations so they can focus on higher-level improvements.

A true closed-loop system functions through four distinct stages:

1. **Connect:** The physical hardware (sensors, actuators, valves) must be integrated into a single network where data flows without manual entry.

2. **Detect:** The logic must be programmed to recognize "out of bounds" conditions instantly. This isn't just a threshold; it's the system recognizing that the process is drifting away from its intended path.
3. **Correct:** This is the critical leap. When a deviation is detected, the system automatically initiates an adjustment—adjusting flow rates, changing temperatures, or modulating speeds—without waiting for a human command.
4. **Act/Confirm:** The system confirms that the correction worked and reports the action taken to the operator as a "completed task" rather than an "emergency."

In this model, the human's role shifts from **operator of the loop** to **designer of the logic**. You are no longer asking them to watch the dial; you are asking them to define what the machine should do when the dial moves.

## Four Things You Can Build This Quarter

You don't need a massive capital overhaul to start closing your loops. You can begin by identifying "low-hanging fruit"—processes where the corrective action is clear, repetitive, and currently relies on human eyes. Start with these four areas:

1. **Automated Setpoint Adjustments:** Identify one variable—like ambient temperature or humidity in a packing area—that fluctuates daily but requires little thought to correct. Link your sensors directly to the environmental controls so the system "hunts" for the target setpoint automatically.
2. **Hard-Stop Interlocks:** Instead of an alarm that tells a person a machine is overheating, install a physical interlock that shuts down the specific motor or section immediately if it exceeds a safety threshold. This prevents a small deviation from becoming a catastrophic failure.
3. **Auto-Logging for Calibration Drift:** If your operators spend time manually logging tool offsets at every shift change, automate that data collection into your system. Use the data to trigger an "alert" only when the drift is significant enough to require human intervention, rather than having them check it every hour.
4. **Actionable Alert Filtering:** Audit your current dashboards. If a notification doesn't require a specific, pre-defined physical action from a human (e.g., "press button X," "turn valve Y"), remove the alert. Replace those alerts with internal logic that allows the machine to handle the correction itself or simply logs the event for a weekly review.

By focusing on these four areas, you move away from the **Monitoring Trap** and toward a production environment where the system is designed to stay within its bounds by default.

## Call to Action

When was the last time a process corrected itself without a human intervention? Share your most aggressive closed-loop success story with us at newsletter@sixsigmakaizen.com, and let's talk about making true operational intelligence standard practice.

Newsletter replies: newsletter@sixsigmakaizen.com · X: @kaizen_6sigma

*Stay curious, stay improving,*

***David Rodgers***
*SixSigmaKaizen.com*

**References:** Quality Digest - Flexxbotics IMTS 2026; Industry standards for closed-loop process control (ISA/IEC)

---