Case Study: Catching Material Shortages Before the Line Does

FirstPass Systems — Production-Grade AI for Manufacturing & Operations. Anthony Ford, founder. Delivered as an internal improvement project at the global manufacturer where he works as a manufacturing engineer — an extension of the line-sequencing engine already running on the plant's converting floor. Details are masked to protect the plant's process knowledge: part numbers are fictional and quantities are rounded.

The Challenge

The converting lines share their raw material: multiple lines pull rolls of the same substrates from the same racks. As volume grew, a new failure mode appeared — a heavy run on one line would quietly consume the substrate that jobs on *other* lines needed later in the week. The crew discovered it the hard way: walk to the rack, find it empty, tear down a setup that never should have been built. Every dead-end setup cost roughly a full changeover plus the disruption of re-planning on the spot.

The ERP knew the on-hand quantities the whole time. It just would not say so in any useful form: the inventory screen's export function is disabled in the plant's ERP version — a known limitation — and a custom extract meant a ticket in the IT queue.

The Approach

Instead of an integration project, FirstPass found the answer in a report the plant already owned: the ERP's standard job-shortage report, which lists — for every open work order — the material required, the quantity on hand, and the net shortage. The ERP can run it on a schedule and email it out automatically. Total new infrastructure: one folder.

The scheduling engine reads that report and joins it to the schedule **by work order number** — no estimated consumption rates, no assumed yields. The ERP's own numbers, tied to the exact jobs on the wall. Then it does what no report can:

- It walks every line's run order in projected time sequence, draining each substrate's on-hand as jobs consume it — across all lines at once, because the lines share the racks.
- The exact job that would take a substrate below zero is flagged in red on the schedule: *runs this material dry, short this many pounds by then*. The jobs queued behind it on the same substrate are flagged at-risk.
- A material summary separates *short on released work* from *short only when demand loaded ahead of release is counted* — so a real today-problem is never confused with next month's.
- The scheduler answers with two clicks that already exist in the tool: SKIP parks a job out of the run order and instantly frees its material for the jobs behind it; RUN NEXT pulls a hot job's whole setup block forward.

Material intelligence is scheduler-facing by design: operators see the same clean schedule they always have, while the sign-in view carries the shortage picture — because deciding what runs and what waits is a scheduling decision.

The Results

- **First run on live data: one in three substrates was short against open demand** — roughly fifty substrates tracked across more than two hundred open work orders, with ten specific jobs flagged as the ones that would run a substrate dry mid-schedule, and the at-risk jobs behind them named.
- **From problem description to running on the plant's real data: the same day.** The floor described the failure mode in the morning; the material-aware schedule ran that evening on the plant's own shortage report.
- **The picture moves fast enough to justify the automation:** between two report pulls fifty minutes apart, on-hand for one substrate dropped by more than 1,200 pounds. A daily scheduled report keeps the schedule honest; a stale file is flagged with its age.
- **Real shortages separated from build-ahead noise:** the first analysis showed every current shortage was driven by released work — demand loaded ahead of release added ten percent to totals but caused none of the shortages. The distinction is now a permanent column, not a one-time study.
- **Zero ERP integration.** Two standard reports the plant already produced — work-order exports and the shortage report — and nothing installed on the ERP. The savings ledger starts now: every dead-end setup this catches is a full changeover that never happens.

Why This Matters for Your Plant

Your ERP already knows what is on hand, what every job needs, and what is about to run out. That knowledge just is not standing next to the schedule when the run order gets decided. The pattern here — pull a standard report the system already produces, join it to the schedule by keys that cannot be wrong, and surface the collision before the floor finds it — applies to material, tooling, labor certifications, and outside processing alike.

No new system. No integration project. The gap between "the data exists" and "the decision sees it" is exactly the gap FirstPass builds in.

What You Keep

Every FirstPass build is delivered as yours: documented source code, a plain-English runbook, and the logs to prove what it does. It runs on your hardware, under your accounts — no license, no lock-in, no dependence on us. Most clients keep us on monthly to watch the alerts, tune as production changes, and build the next thing — but that's a choice you re-make every month, because the work earns it. Not because you're stuck.

Next step: a free 30-minute working session. Two or three specific ideas for your operation and a live demo of a working system — no pitch deck, no commitment. If one idea is worth proving, a fixed-scope two-week pilot puts a working tool on your real exports, and you keep it either way.

Anthony Ford · FirstPass Systems · Thorntown, Indiana · anthony.ford@firstpasssystems.com