Industrial engineering supply-chain management: operationalizing complex planning logic

Industrial engineering and supply-chain management often rely on complex planning logic: routing heuristics, capacity rules, sequencing practices and exception thresholds developed through years of operational experience. The challenge is to make that logic repeatable without losing the context and judgment that made it work in the first place.

Move from a heuristic to a governed decision

A planner may know how to respond when one line is constrained, a supplier misses a slot or a rush order disrupts the schedule. The rule may be partly documented and partly embedded in spreadsheets or individual expertise. Automating it requires more than translating a formula into code. Teams must define when the rule applies, which constraints take precedence, who owns exceptions and how the result will be evaluated.

Start by recording the decision in operational language. What signal triggers it? Which data does the rule require? What options can be considered? Which constraints cannot be violated? What is the decision owner permitted to change? What should happen if data is missing or a constraint conflicts with another? These questions help separate reusable logic from assumptions that need review.

Preserve the engineering intent

Operations-research models and heuristics can be valuable inputs to a decision workflow. The workflow should preserve the source, version and intended use of the model output. It should also show whether an output is a recommendation, an approved action or an executed transaction. A solver’s optimal result is only optimal within the model’s assumptions and constraints.

Industrial engineers, operations owners and IT should agree how changes are tested. Include representative cases, edge conditions, model updates and policy changes. Make it possible for an authorised user to override a recommendation with a reason, and capture the outcome so the team can learn whether the policy or model needs adjustment.

Measure performance without losing control

Measures may include schedule adherence, throughput, changeover impact, recovery time, overtime, decision cycle time and overrides. Define the baseline and how each metric is calculated before introducing the workflow. A result should be attributable to the decision process only when other relevant changes have been considered.

Where Aidecis fits

Aidecis keeps one authority model and one decision record across your people, your own algorithms and every vendor’s agents. Aidecis never writes into your systems: your people, systems or bots execute the approved action, and Aidecis reads each execution back and reconciles it against the approval, so SAP stays your single source of truth. With enforced control, your own integration checks Aidecis's single-use grant before it executes. Agents and algorithms that write in their own systems are checked against their registered role and limits afterwards, and flagged within minutes if they went outside them: observed control, which does not prevent an action but shows it. A planning model or heuristic keeps its logic, and its proposals enter the same chain as a planner’s. Aidecis does not replace operations-research software or an industrial engineer’s role. A pilot can test how one defined heuristic moves from analysis into a governed operating decision.

Explore Aidecis use cases and discuss a pilot.

Copyright 2026 Aidecis