CASE STUDY · RETAIL · COST DISCIPLINE & INTELLIGENT ROUTING

A retailer ran 240 concurrent pricing agents without a single runaway loop

Concurrency and recursion limits turned an experiment that had twice run away into a production system with a predictable bill.

Scenario	Published	Orcher components
Omnichannel retailer · 1,900 stores	2026-07-29	Cost Governance, Logic Scrubber, Observability

Summary

A retailer's markdown-optimisation agents had twice entered recursive loops that consumed a month's budget in a weekend. Orcher capped concurrency and recursion depth in the execution path, verified each price change against margin floors, and made the cost of a pricing cycle predictable enough to plan around.

240
concurrent agents under a single enforced ceiling

0
runaway loops since deployment
two in the preceding six months

$0.11
cost per SKU-region pricing decision
from $0.74, unbounded

11 hrs → 0
detection lag on out-of-policy prices
now refused pre-commit

The situation

- Markdown agents negotiated with demand-forecast agents, and the interaction had no depth limit.
- Two incidents in six months consumed the quarterly budget over a weekend with nothing committed.
- A pricing error committed at 02:00 on a Sunday took eleven hours to detect.

What Orcher enforced

- Concurrency and recursion depth became enforced properties of the directive class, not conventions in application code.
- Logic Scrubber verified each proposed price against margin floors, contractual price commitments and regional rules before commit.
- Cost per pricing cycle was bounded up front, so the finance question was answered before the run rather than after it.
- Halts and refusals were recorded as outcomes, giving the pricing team a feedback loop instead of a mystery.

The data

Control	Setting	Triggered in period	Effect
Concurrency ceiling	240 agents	17 times	Queued, none dropped
Recursion depth	6 hops	31 times	Halted and logged
Margin floor check	Per SKU-region	1,204 refusals	Prices never committed
Contract price lock	Per supplier agreement	88 refusals	Breach avoided
Cycle cost ceiling	Per pricing run	3 times	Run completed at ceiling

Enforced control activity over one full markdown season.

Outcomes

- Markdown optimisation moved from a supervised experiment to a scheduled production process.
- The pricing team plans agent cost as a line item with the same confidence as compute.
- Refusals surfaced 88 supplier contract terms that the legacy pricing engine had been quietly violating.
- The retailer applied the same ceilings to replenishment agents in the next season.

> “Two runaways is a story you tell once. The third one ends the programme, so the limit had to live somewhere nobody could edit in a hurry.”
>
> — Director of Pricing Science, omnichannel retailer (composite, modelled scenario)

Read further

- Multi-agent orchestration — dippai.com/orcher/multi-agent
- Cost governance — dippai.com/orcher/cost-governance
- Full case study — dippai.com/case-studies/retail-multi-agent-under-ceilings

Disclosure. Modelled reference scenario. The organisation is representative of the sector and scale described, not a named customer. Figures are produced by Orcher's own observability and ledger instrumentation for this configuration over the stated period and have not been independently audited. Market figures carry their own source line.