



One Issue, End to End

The Governed Knowledge Loop, demonstrated on a single ticket

Every module in this toolkit describes one station on the same assembly line. This document rides a single issue through all of them — a real trace in the toolkit's example universe (Meridian Systems, the fictional company whose articles fill Module 2's `examples/`), so you can see the modules working as one system rather than seven documents. Total human time across the issue's entire life: **about seven minutes**. Count them as we go.

Act 1 — Birth: a resolved case becomes governed knowledge

May 2, 2026. A Meridian customer on a managed Mac reports the VPN "connects for a second and then kicks me out" after a macOS update. An agent works it: the update reset the approval for the VPN client's network system extension. Fix confirmed, the agent closes case-2291 with a one-line wrap-up note — *"re-approved the system extension in Privacy & Security, restarted the client, tunnel held."* That sentence is the entire capture tax. **(Agent time: ~20 seconds.)**

The prompt (Module 3). The case is flagged KB-worthy, and `synthesis-issue-resolution.md` runs against the full thread plus the wrap-up note, with Meridian's `{{ORG_CONTEXT}}` — products, version scheme, domain taxonomy, allowed tags — filled in. Out comes one complete candidate, in Module 2's exact format:

```
candidate_id: CAND-0017
type: candidate
proposes: new
proposed_type: issue-resolution
origin: case-synthesis
source_refs: [case-2291]
demand_evidence: "recurring since the macOS 15.x rollout; 3 same-symptom
  cases this week"
generated: 2026-05-02
model: "prompts-v1 / drafting model 2026-04"

Proposed title: VPN shows "authentication succeeded" then drops with
  error 720 after macOS update
Proposed frontmatter: product: GlobalConnect VPN · product_version:
  [NEEDS-VERIFICATION: 4.x suspected] · platform: macOS 15.x ·
  domain: endpoint-support · audience: customer · volatility: high
  ("tracks macOS releases") · tags: [vpn, macos, connectivity, error-720]
```

Note what the prompt did *not* do: it did not guess the affected client version (grounding rule 1 — the thread never stated it), and it kept the customer's own words in the symptom.

The gate (Module 4). A curator opens the new-candidate gate checklist. Dedupe: nothing covers error 720 on macOS (the nearest article is the Windows variant). Grounding: every claim traces to the thread. Applicability: product, version, domain, audience — present, one flagged for verification. The curator confirms 4.x against the case's client logs, and adds the one thing the draft couldn't know: the fix does **not** apply to personally-owned, non-MDM devices. Disposition: **A2 (edit-approve)**, **edit-distance 1** — light touch, under ten percent of the words. **(Curator time: ~4 minutes.)**

The record (Modules 2 + 5). The candidate is promoted as **KO-000142**, `status: draft`, `source: case-2291`, `owner team-endpoint` per the ownership policy. A reviewer validates it the same day — `status: validated`, and only now can any delivery surface serve it. One row lands in the decision log:

```
date,actor,object,event,decision,edit_distance,trigger,note
2026-05-02,j.rivera,CAND-0017,gate-new,A2,1,case-synthesis,promoted as KO-000142
```

The full article, exactly as it ships, is Module 2's `examples/KO-000142-vpn-error-720.md`.

Act 2 — Middle age: the world changes and the article keeps up

Through the summer the article does quiet work: two light revisions (a step clarification in June; a routine validation on August 9 leaves it at `version: 3` — the snapshot shipped in Module 2). Then **August 13**: macOS 15.6 moves the settings path the resolution depends on. A product-change signal fires, and Module 3's `update-diff-proposal.md` produces a surgical before/after DIFF — not a rewrite:

```
- 1. Open **System Settings → Privacy & Security**.
+ 1. Open **System Settings → Privacy & Security**
+   (on macOS 15.6 and later: **System Settings → Security**).
```

The update-DIFF gate (Module 4) applies the meaning-change test: is truth changing, or only its address? Users on 15.5 still need the old path, so the DIFF *adds* the version-scoped path rather than overwriting — no meaning change, applicability preserved. Accepted, edit-distance 2, `version: 4`, `last_validated` updated. The decision log gets the row you'll find as an example in Module 5's decision-log template:

```
2026-08-13,t.okafor,KO-000142,gate-update,accept,2,product change,macOS 15.6 changed settings path
```

(Curator time: ~2 minutes.) No gate, and this is the other timeline: a confident answer engine citing stale instructions at every customer the day after the OS ships — the Precision Paradox collecting its fee.

Act 3 – The economics: what one governed article is worth

Numbers below are **illustrative placeholders** — replace them with your own channel economics from Module 6's Channel Inventory tab.

Before KO-000142 existed, this symptom hit the assisted queue roughly 70 times a month at a loaded cost of, say, \$18 per assisted answer: about \$1,260 a month to re-answer a known question. After validation, the article serves the same issue through the help center, the agent-assist panel, and the answer bot — self-service unit cost measured in cents. Within two months the assisted queue sees single-digit residuals (mostly the non-MDM edge case the curator fenced off in Act 1).

Where each thread lands in the **Measurement Workbook (Module 6)**:

What happened	Where it lands	What it tells you
A2 + edit-distance 1, then 2	Edit-Distance Log	Drafting quality trend — falling means the prompts are learning your standards
Reason codes on this and every disposition	KPI Catalog / gate metrics	Reject-reason mix tunes capture upstream
~70/month assisted → single digits	Monthly Volumes → Blended CPA	The blended cost per answer falls as answers shift left
Issue answered across four channels	Answer Mix / shift-left ratio	One captured solution, hundreds of deliveries
Known issue leaving the queue	Known vs New	Leak-back here would mean a findability failure
Harder residual queue	Inversion Brief	Assisted TTR rises by design — the brief preps your executives

And the sixth channel is already implied: the same validated, cited, applicability-filtered article is what a customer's own AI agent consumes through an API — at which point the metadata the curator verified in Act 1 stops being hygiene and becomes the product.

The point

Module	What it contributed to this one issue	Human minutes
2 – Templates	The typed shape and metadata the machine could draft into	0 (built once)
3 – Prompts	The draft, grounded and PII-scrubbed, refusing to guess	0
4 – Gates	Two dispositions; the non-MDM boundary; the meaning-change test	~6
5 – Governance	Ownership, lifecycle states, two append-only log rows	~1
6 – Workbook	The financial story: one issue, N answers, falling blended CPA	0 (rolls up)

Module	What it contributed to this one issue	Human minutes
1 & 7 — Assessment / Rollout	Why Meridian sequenced it this way at all	(see below)

Seven minutes of human judgment, applied only where judgment is the actual job. That is the whole model: *AI drafts, humans approve, machines and customers consume, and the system tells you what's missing.*

One caution, because this walkthrough makes it look easy: Meridian could run this loop **because it was sequenced**. Its February self-assessment (Module 1 ships the completed grid as a worked example) placed it at Level 1 with delivery running two levels ahead of governance — the classic danger configuration. Standards were ratified first (Module 5 ships that worked policy too), the gate stood up second, and only then were the prompts turned on. Run the acts in that order. Running them in reverse is how you ship confident, cited, wrong answers at scale.

Contextkeeping is a KCS-aligned operating model. KCS® is a service mark of the Consortium for Service Innovation; this material is not affiliated with or endorsed by the Consortium.