

Management as Ontology

The canon · v5.1.1 · 2026-09-29 · Norman Wanto (drafted with Alfred) · release gate PASS

Name it, clock it, write it, guard it, garden it.

The map (from canon/REGISTER.yaml)

Every unit below is identified by NAME; epistemic stamps: settled = receipts cited, provisional = ratified, first receipt pending. Part numbers are reading order only.

The seven models

- The three layers P1
- The two loops P1.25
- The five verbs P2.5
- The two doors P2
- The three skills P2
- The four enforcement roles P5
- The ball and the court P5

The seven instruments

- The catechism P3
- The seven fitness tests P4
- The five presences P2.5
- The jobs admission standard P2
- The four grain tests P2
- The five dispositions P2.5
- The ten-word mantra P6

The 44 laws, seven families

Laws of the nouns

- The identity law settled P1
- One noun, one home settled P1
- The re-homing rules settled P1
- The ranking corollary settled P1
- The grouping corollary settled P1

Laws of the spec

Laws of the spine

- The reification principle settled P1.25
- Spec and implementation settled P1.25
- The spine anatomy settled P2.5
- The authored-once corollary provisional P1.25
- The workflow law provisional P1.25

Laws of the organs

- Types and bindings settled P2.75
- Type by role settled P2.75
- The register's walk set settled P2.75
- Liveness is derived, the index declares addresses settled P2.75
- Knowledge routes by role provisional P2.75

Laws of the doors

Laws of the jobs

- Jobs-first admission settled P2
- Jobs carry readings provisional P2

Laws of sizing

- The depth rule settled P2
- The split rule settled P2
- The grain law settled P2

Laws of the ground

- Construction grounds settled P1.75
- Home by construction, depth by receipt provisional P2

Laws of the running system

Laws of the walk

Two kinds of processes settled P2.5
Inspect means exceptions settled P2.5
The walk ladder provisional P2.5

Laws of the work

The induction law provisional P2.5
Deferrals carry flip conditions provisional P4
The five-step order provisional P4
The species law provisional P2

Laws of enforcement

Enforce artifacts, never conversations settled P5
The calcification law settled P5
Derivation before delegation settled P5
The OKG settled P5
The provenance law provisional P5

Laws of authority

The authority law settled P1.75
The stage curve settled P1.75

Laws of the head

Laws of the span

The cognitive-fit law settled P4
The qualifier corollary settled P4

Laws of the words

The readability law settled P4
One name per concept provisional P4

Laws of the boundary

Structure the pipeline, never the ideation settled P7
Killable by evidence settled P7

Laws of attention

The attention law provisional P1.5
The push ladder provisional P1.5

Management as Ontology

For Gotrade leads · from Norman, drafted with Alfred · v5.1.1 (2026-09-29)

*the model behind how Norman structures work, made explicit so it can be taught, practiced, and examined. A half-hour reference read; the teachable kernel is Parts 1-3, and the skill comes from the drills, not the reading. **Since v4.0.0 it is also a STANDARD with a reference implementation: the canon fixes one name per concept and its definition; a deployment inherits the names and binds only its mechanism, spoken as a qualifier; Relay is the reference implementation, and every estate Norman runs reads as one language. Opinion is formed on receipts (the first deployments' actual speech) and prescribed to the next; a renamed name is itself a major.***

Since v4.1.0 the canon has one OBJECTIVE FUNCTION, and every law below serves it: value created per unit of operator attention. Value is what the goal layer's objectives say it is (Part 2, the goal-layer clause); attention is the operator's, spent at exactly THE THREE SEATS where substance is human and cannot be delegated: setting what value means (objectives), ruling (the doors, the decisions), and escalations (the top walk). Everything else is the machine's. A system that minimizes attention alone optimizes silence, so the ratio is read on a scoreboard that carries both numbers, never attention by itself (the attention law, Part 1.5). **And the canon has two EDITIONS that differ only by player count.** The estate edition is single-player: one operator, one head, agents as the workforce, the walks agent-run, the operator at the top walk. The Relay edition is multiplayer: many heads, tenants, authority granted and recorded, ceremonies convened across people, the court between people. An edition is not a deployment: Relay is one edition with many deployments (a workspace each), the estate one deployment of the estate edition. What varies with players: the authority law (single-player, the operator is the author and the apprenticeship dissolves at once), the court's parties, who convenes ceremonies, and the cognitive-fit law's head count. What never varies: the three layers, the doors, the spine, the nine organ types, the walk ladder, the OKG, the three seats.

Part 0: Why this matters (the hook, and it is not hype)

Two facts, both checkable against your own quarter:

1. **Unstructured work does not compound.** A growth team without an experiment register repeats experiments and evaporates learnings: the same budget buys less insight every quarter. A marketing team without standardized output artifacts cannot be reviewed, handed off, or accelerated: every deliverable is bespoke, every review starts from zero. Work that is not written into named structures is work the organization forgets.
2. **Structure multiplies agent leverage.** Gotrade already runs an agent fleet. Agents can handle ad-hoc asks anywhere, but they can gate an intake, pre-brief a standup, lint a register, or author an artifact only where those nouns exist. The department with ontology inherits the fleet's compounding leverage; the department without it gets an assistant instead of an operating system, and the gap widens every month.
3. **Structure that should not exist compounds too.** A guard nobody re-derives taxes every run for every lane; a report nobody reads costs its cron forever; an addition arrives with its incident as a receipt and a deletion with none, so estates accrete by default and adding feels like hardening. The order in which structure is built and changed matters as much as building it: requirement, delete, simplify, accelerate, automate (the five-step order, Part 4).

Product, market, and distribution grow the company. This is what makes every unit of that work accumulate instead of evaporate, and it is the on-ramp for AI into your function. That combination is the highest-EV internal move available to a lead at Gotrade today.

Part 1: The three layers

Every well-managed domain has the same three layers. When work feels messy, unnamed, and untrackable, one or more of these layers is missing.

Layer	What it is	Gym example	Marketing example
1. Ontology	The nouns: what exists, its fields, its legal states, its identity scheme (how instances are named), and the declared places where it lives	Set, Rep, RPE, "Barbell Deadlift", the Daily Plan Notebook	Campaign, Experiment, Brief, Learning; the campaign register; the launch checklist
2. SOP	The verbs: what may be done to the nouns, by whom, on what clock	"Every morning, check the Notebook"	"Every Monday 10:00, standup walks the register: exceptions first"
3. Data	Today's instances, written in the layer-1 vocabulary	"Deadlift 5x8 @ RPE 7-8"	"EXP-041: IDN lookalike audience, CAC hypothesis, verdict pending"

The test of a good stack: **the daily instruction becomes one line**. "Deadlift 5x8 @ RPE 7-8" is ten words only because layer 1 defined Set, Rep, and RPE. Without the layers, every instruction is a bespoke 40-minute meeting, unrepeatable and unverifiable. With them, management's daily cost collapses to writing rows, and the team executes without the manager present.

The same layers in a second domain, to see the pattern generalize. Training for a marathon:

Layer	Marathon training
1. Ontology	Session types (long run, tempo, intervals, recovery), pace zones (Z1-Z5), weekly mileage, the training plan as the declared place
2. SOP	"Check the plan every morning; long run every Sunday; log every session"
3. Data	"Today: 16km @ Z2" and tonight's logged actual

The marathon adds what the gym example hides: **the loop over time**. Enforcement is visible (a coach or Strava sees the missed Sunday run the same day: skipping is loud), and gardening has a name in this domain, periodization: the plan amends itself on evidence (an injury, a fitness test, a bad week), and a plan that is never amended is how runners get injured. Lifting teaches the nouns; the marathon teaches the operating loop.

For engineers this is instantly familiar: **layer 1 is the schema and models, layer 2 is the functions and their scheduler, layer 3 is the rows and logs at runtime**. Management is systems engineering where the runtime is people.

The identity law

No noun without identity (the law of layer 1, in the spirit of Quine's dictum). A noun is not ontologized until its instances carry a DECLARED identity scheme, and two schemes are legal: a MINTED token where instances must be cited across time, people, or surfaces ("we track experiments" is vocabulary; "EXP-041" is an ontology), or a NATURAL KEY where the date or slot

already individuates ("Tuesday's session" needs no serial number; the timestamp IS the scheme, declared).

Where tokens are minted, three rules: the scheme is set at ontologize time; every instance mints at birth, at insert not at approval (identity precedes status, so even a dropped thing stays citable); and IDs are permanent, never reused, never renumbered, surviving soft-delete, because the code is the audit trail's spine. The quality bar is HUMAN-SPEAKABLE: machines keep UUIDs underneath; the management tier is the conversational identity, one short token that survives being said aloud in a standup.

Identity is what the rest of the machinery stands on: the one-line instruction runs on it (the ID carries the context), enforcement needs it (you cannot make skipping loud for a thing you cannot name), inspection needs it (dispositions attach to IDs), receipts need it ("done" plus the row code is a verifiable claim; "done" alone is an assertion), and gardening needs it (you can only merge or kill what you can cite).

Namespacing is scoped by travel: a code that never leaves its workspace needs no qualifier (the workspace IS the namespace); a code that crosses boundaries carries its origin, added at the boundary by whatever carries the citation: a rendering surface where one exists, the citer's own discipline where the habitat is a plain sheet (the DNS model: short name inside the zone, fully qualified across zones).

The portfolio clause (stamped 2026-08-30 on its first live registry): an operator running MULTIPLE deployments keeps a declared ZONE REGISTRY — one code per organization or logical entity — homed with the operator, the portfolio's best witness (no single deployment may own it). Worked neutrally: a founder holding an operating company, a spun-up venture, and a personal estate declares ABC, XYZ, and HOME; inside ABC's workspace a bug row is simply BUG-714, in the venture's board deck it egresses as "BUG-714 (ABC)", and HOME's prefixes never collide with either because the registry is the collision check. A zone may be RESERVED for the cost of a row while carrying no deployment (a code is cheap; walks are not). And a partition — the venture spinning out, a tenant splitting entities — is a re-homing at org scale: identities never renumber, shared registers get zone assignments, vacated namespaces alias forever. (First instance: the holder's own four-zone registry, charter-homed the day this clause was stamped; the second is pre-registered in the tenant model's entity-registry design.)

(SETTLED: codes-at-birth running estate-wide. Ratified 2026-08-26; natural-key clause and qualification generalization per review v4, findings 4-5; the portfolio clause 2026-08-30.)

The ranking corollary

Position is a field, never an identity (the identity law's corollary, position versus identity). Where ordering carries meaning — a ranked principles list — rank is an edition-scoped attribute of a stably-NAMED instance: cite by name, qualify rank by edition ("#1, Aug-2026 edition"; bare rank is legal only inside the edition), and a reorder becomes an auditable diff that forces the amend-versus-replace call a silent renumbering dodges. Where ordering is mere arrangement — this canon's Parts — position is a pointer, free to restructure because nothing load-bearing cites it. Either way the identity never renumbers.

(SETTLED: ratified 2026-08-29 from the Gotrade principles-reordering question; each half already lived in the field — the code egress grammar, edition-scoped ranks.)

The grouping corollary

Membership is a field, never a domain (the identity law's second field-corollary: ranking ruled position a field; this rules grouping one). Humans require labeled hierarchy — recall descends trees, the cognitive-fit law's own evidence — so every SURFACE renders the tree: indexes group, registers nest under families, the head gets its branches. But the tree is built from FIELDS on flat entities, so jurisdiction never nests: a group that owns nothing, walks nothing, and clocks nothing is a category (the grain law's dead form, one altitude up), and making it first-class would tax every re-grouping with a re-homing.

Fields buy what containers cannot: MANY trees from one flat estate — by family for navigation, by owner for accountability, by lifecycle for gardening — because a label is cheap and a container is exclusive. And a label still graduates honestly, node by node, when it earns work: a label needing a walk becomes a ceremony's segment list; a label needing an owner and rulings takes the grain tests as a concern, or is the meta-concern's territory. The tree deepens by receipt exactly where reality demands, and stays weightless everywhere else.

(SETTLED at naming, 2026-08-30: the canon's own register runs its laws under seven weightless family labels; the fleet ruled categories-as-grouping-never-concerns (DEC-29) and runs categorized dossiers; the estate's dossier family field is the arriving third instance, queued as a render change precisely because this corollary makes it one.)

One noun, one home

Every noun lives in exactly ONE home concern (the jurisdiction law). The identity law names an instance; this law places the noun. The home concern's spine declares it and its owner stewards it; every other concern uses it BY TYPED REFERENCE, never by copy — a duplicated entity table forks truth (two SaaS registers cannot answer "what do we run"; the delta between them is invisible exactly where it matters). The split that makes sharing safe: the home owns the ENTITY — existence, identity, the universal fields, kept deliberately thin — while each referencing concern keeps its OWN FACTS about the entity in its own register, keyed by the home identity (finance's costs and the access owner's seats never share a table; both reference one SaaS row, and the reconciliation between them becomes a computable diff instead of two drifting spreadsheets).

Ownership: MINTED by whoever needs the noun first (the pain door, no committee), OWNED by the concern that best WITNESSES the entity's lifecycle — births especially — and bears the failure cost of an incomplete set; when first-filer and best-witness diverge, that is a jurisdiction dispute, ruled where grain disputes are ruled, and proposals stay open to every concern regardless (the home ratifies, and proposing must stay cheaper than forking).

DECISIONS obey the same law: a ruling lives in the spine of the concern holding jurisdiction over it (concern-scoped sequences, qualified at egress like every code); a ruling spanning concerns belongs to the META-concern, whose territory the concern estate is, and is never fragmented into per-concern copies — affected concerns pin it and render it by reference.

(SETTLED: first receipt 2026-08-28, hours after ratification — the fleet's decision register re-homed under its meta-concern through the six-step procedure, DEC-50 standing; the per-concern decision-sequence mechanics carried their own pre-registered receipt, which landed 2026-08-29: CRN-D1, the first concern-local ruling filed in its own spine, unmanufactured, standing beside the pinned DEC rows. Ratified 2026-08-28 from the SaaS-registry derivation; standing receipts on the noun half — the fleet's shared-source register serving two readers so "what binds us" never forks, and one cross-concern ruling governing three concerns by pin.)

The re-homing rules

Stewardship moves, identity never does. RE-HOMING, when the true witness turns out to live elsewhere, is an induction-class act (on ruling, no clock) in two shapes, named by its first executor: CUSTODIAL re-homing moves stewardship AND address (the full six steps bind), while

JURISDICTIONAL re-homing changes only the governing spine — slug, codes, and custody survive, so the alias and handover steps are satisfied vacuously rather than skipped, and the executor should know that lightening is legal.

In either shape: rows keep their codes, the vacated address becomes a PERMANENT one-hop alias (append-only records cite it forever, so its reader set is never provably empty and the retirement path can never legally reach it; vacated slugs are never reused), and the incoming owner verifies the inherited set against their own witness stream before accepting, backfilled from ground truth, never reconstructed.

Deletability is decided by PROVABILITY OF ZERO READERS: a same-cycle mint error nothing ever cited may be cancelled and its slug freed; anything that ever drew breath in the record may not. Re-homing frequency is itself a health metric: a handful a year is an ontology learning, a pile is an ontology thrashing, and then the grain is the disease to treat.

(SETTLED: DEC-50, the DEC-register re-homing run through the six steps hours after ratification — the code sequence continuing unbroken THROUGH the re-homing as its own identity proof; the two shapes named by the first executor's friction note.)

Part 1.25: The two loops (where enforcement and gardening actually live)

Enforcement and gardening are not layers, and "on top" is the wrong preposition (this document's earlier phrasing). A layer is structure; these are FEEDBACK LOOPS, closing at different speeds over the three layers:

- **The fast loop (enforcement)** closes the runtime against the design: do the records conform to what the schema and SOPs intend? Its machine half is authored INTO layers 1-2 (state machines, required fields, gates); its human half is the Inspect ceremony walking the exception set. Cycle time: whatever clock the structure declared (a daily standup, a Monday walk).
- **The slow loop (gardening)** closes the design against reality: do the schema and SOPs still fit the drifted world? Cycle time: amendments, weekly to quarterly. It is the only loop allowed to reopen layers 1-2.

This is Argyris's single/double-loop learning in operational clothes (Part 5.5), and it dissolves a confusion the layer picture creates: asking "which layer does the walk belong to?" is asking which STRUCTURE a LOOP'S CARRIER lives in. Loops are behavior over the layers, carried by ceremonies like any other process (Part 2.5).

The reification principle

In a platform that implements this model, the definitions themselves are records. Layers 1 and 2 live as rows in meta-registers (the register of registers, the register of ceremonies, the charters), so the slow loop runs as ordinary register operations: an amendment is a row, a retirement is a kill-docket entry, and the gardening ceremony walks the meta-registers exactly as the standup walks the concern ones.

One discipline rides with the promotion: the meta-register is the LAST meta level; nobody ever proposes the register that manages the register that manages registers. Which platform implements this is a deployment decision, recorded where deployments are tracked, never in this canon. The meta-registers have one derived organ, the OKG (Part 5).

(SETTLED: first receipt 2026-08-28, exactly on its pre-registered condition — the fleet's meta-concern, a register of registers entered through the pain door after one night of ad-hoc estate-gardening supplied its jobs with hours-old receipts, went live, ceremony-walked, with a grain-reversal amendment chain having run through the declaration machinery as ordinary operations. Ratified 2026-08-26; de-productized per review v4; promoted 2026-08-28.)

Spec and implementation

The spec owns intent, the implementation owns mechanism, and implementations conform to the spec. A structure's definition and its mechanisms rarely live in one artifact. The SPEC organ (a dossier, a charter, this canon) owns INTENT: names, legal states, contracts, clocks-as-commitments, decisions, and why. The IMPLEMENTATION organs (code, tools, configured platforms) own MECHANISM: the exact validation, retry logic, plumbing, which the spec never restates (prose restating mechanism is the mirror that drifts).

Declare the conformance direction at ontologize time: implementations conform to the spec, so a divergence is either an implementation bug or evidence the spec needs amending, and a human ruling decides which — the fast loop fixes the first, the slow loop the second. Implementation-as-SSOT collapses both loops into "whatever runs is right": a bug becomes indistinguishable from a decision. Conformance is checked in the CONSUMER's context, never only where the author runs it: derived truth that passes on the author's machine and fails where it is read is drift of the same class as a divergent spec (settled at naming, 2026-09-09, on two deployments' receipts: the fleet's schedule introspector, green in the CLI audit while every HTTP consumer read nothing, GBUG-1123; the estate's yaml parser, present locally only through a dev package, green locally and a 422 on prod, promoted to a production requirement twice).

Where one definition must exist in both prose and machine form (an enum, a type table), compile both consumers from one machine-readable source (config-as-data) instead of maintaining the pair by hand. Prior art: desired-state reconciliation (Kubernetes, Terraform: declared spec, controllers reconcile actual toward desired, humans amend the spec) and normative-language specs (RFCs).

The AUTHORED-ONCE corollary (ratified 2026-09-07): a spine is authored once. Every datum of a concern's spine — a job, a noun and its states, a typed link, an organ, a SOP, a decision, the version — has exactly one authored source, and every surface renders from it: the dossier page, the map, and any markdown twin kept for readers and version control, which is then a generated artifact and never hand-edited. The version is minted at the write path, never typed. Two artifacts authored by hand for one spine, however disciplined the same-touch rule and its noticer, is the mirror that drifts at the scale of a whole deployment. Where the datum is prose (the why, a runbook), the law asks only that it be authored once; how the source is laid out — one declaration file per concern, with prose as fields inside it, is the binding both deployments have chosen — stays a deployment binding recorded where bindings live. Words: the DECLARATION is the authored file, the SPINE its content, the DOSSIER the rendered page. (PROVISIONAL: receipts at ratification — the fleet's one-file declaration with its dossier composed from it, running before the estate's pair; the fleet mirror's finding that a hand-kept version is the mirror that drifts (T20, 2026-09-06); the estate's prose-plus-sidecar pair, which proved the anatomy and then this hazard; the author's ruling "we should not let it diverge" (NW-ANA-43). FIRST RECEIPT 2026-09-09: the estate's migration landed, nineteen declarations, every clause of the old prose present in the render by the induction lint (6,241 atoms, three phrasing residues), the version minted at nwsos-todo's push door and proven by a second push that bumped only the eleven declarations whose bytes changed. REMAINING DUE: one amendment afterwards surviving on both deployments with the prose and the machine consumers moving together, nothing edited twice.)

(SETTLED: first receipt 2026-08-27, the email-intake concern's court read-state ruling — the pass implementation marked flagged court items read where the spec's enumeration implied they stay unread; the human ruling resolved it as an AMENDMENT, read-state redefined as "triaged by the pass" with the court pushing through the daily report line rather

than the unread badge, and the spec now states the choice explicitly with a pre-registered flip condition. The law ran exactly as written: divergence found by conformance sweep, bug-or-amendment put to a human, spec amended, drift now impossible to repeat silently. Ratified 2026-08-26, from the email-intake design session's spec-versus-implementation question.)

The workflow law

A WORKFLOW is a deployment's binding of one verb to one RUNNER: the trigger or clock, the invocation, the receipt. It is named once, in the catalog, cited by name from the verb's row in the spine and read by the organ that names its address; its concern is its verb's, by reference, never a container of its own, and a ceremony's workflow is the ceremony's. One workflow binds exactly one verb: two verbs on one trigger are two workflows sharing the trigger, whatever runs them (the estate's hotel confirmation: the claim filing on the claims concern, the pre-arrival request on its own project, one routing rule between them, NW-ANA-67). Its aspects each have one home and no other: the design (why, contract, the suite row) in the concern's declaration; the assignment in the deployment's roster; the execution config in the CATALOG, the deployment's index of its bindings, which is never a home; the invocation in the runbook the catalog names; the receipts in the ledger. An organ whose address names a workflow reads its receipts and does not home it. A workflow that runs and no verb names is an orphan of the fifth class (home by construction, Part 2), and a deployment's OKG lists its workflows beside its verbs: a name in, its verb and concern out, with the clock and the last receipt.

(PROVISIONAL: ratified 2026-09-18 on the operator's question "does a workflow belong to a mother concern?", asked of the estate's catalog and answered from both deployments' files. Receipts in hand: the estate's concern model (2026-08-26: a named workflow has five aspects, one home each, the design in the owning concern's dossier; the orphan noticer in the index build since the same night) and its workflows index (nwos-todo, 2026-09-18: thirteen workflows, six named by no verb, the lookup door answering a name with its verb, concern, clock and receipt); the fleet's catalog with its weekly audit gate and the WCR lane on a non-owner's ask (Charter sections 6 and 12), its own gloss, "a mini-app is a place, a workflow is a habit", and the reference implementation's read of 2026-09-18: a Relay SOP row cites its workflow by KEY, validated at apply against the owning agent's catalog (the console's DeclarationValidator; gtos-bug-intake's intake SOP), the same name entering the organ index as a receipt reader, the pulse deriving trigger, clock and last event from it; its lookup door answering a workflow name with the organ and the concern but not yet the verb; the Charter speaking the catalog as a curatorial place and never saying where a workflow is homed, one sentence owed; "runner" a word Relay has no equivalent for (it speaks trigger, scheduler, console_command as the runner's parts), "catalog" its own word, "inventory" retired for workflows by amendment. The reference implementation met its half of the due the same morning (T45, console v1.40.45, 2026-09-18 10:00 SGT: relay_okg_lookup, the lookup door's live name since DEC-166, answers a workflow name with a verbs group beside the organ, the SOP row that cites it by key, its concern, clock and last event; the fleet's catalog READER is the console's FleetCatalogService, its aggregator gtos-root's fleet-workflows poll, and the pulse reads the SOP's key through the graph builder's joint) and re-stamped its skill and Charter the same hour (core PR #1089: section 12 says the catalog is an index and never a home, section 19 reaches a workflow through the verb that names it). Due: one two-verbs-one-trigger case run through its two workflows unattended (the estate's hotel confirmation, NW-ANA-67: the pass emits the booking, the claim filing and the pre-arrival request consume it; the Charlotte instance of 2026-09-17 ran by hand before the workflows existed).)

Part 1.5: What success looks like

Before, a day in the life of an unstructured function: the lead's Slack is the intake, so they are interrupted hourly; every instruction is a meeting because there is no vocabulary to compress it into; the standup (if it fires at all) is a status recital with no sections, so decisions leave without owners; experiments run, learnings live in someone's head, and the next quarter re-runs them; when the lead travels, the function idles; and no agent can help, because there is nothing shaped for an agent to hold.

After: **the daily instruction is one line** ("EXP-044: run the lookalike variant, verdict Friday").

Ceremonies fire on clocks and run without the lead present (the absence test passes: two weeks

of holiday, nothing rots). **The lead's attention is spent only on exceptions**, because compliance is silent. **Learnings accrue in registers** that new hires read on day one. **Agents carry the floor**: intake is gated, the standup is pre-briefed, the register lints itself. And the lead's calendar shows it: hours moved from chasing status to making the few calls only a human can make.

Measurable markers, baselined at the workshop and read at the practicum (day 30 formative, day 60 certified): average instruction length (should shrink to a line), ceremony survival rate (fired on cadence without prompting), exception-to-noise ratio in what reaches the lead, register growth (learnings written per week), and the absence test — which the practicum's unwatched weeks 5-8 administer for real.

The attention law

Attention is the denominator, never the objective: every structure is judged by the value it returns per unit of operator attention it costs, and the operator's attention is spent only at the three seats (objectives, rulings, escalations). Compliance is silent at every altitude; a structure that reaches the operator with anything but a ruling to make, an objective to set, or an escalation to take is spending the scarcest input on the machine's work. The reading is a scoreboard (the ninth organ type, Part 2.75) on each deployment's meta-concern, and it reads both numbers: attention from the HUMAN seats only (rulings by a person, escalations handed to a person, objectives declared by a person, a person's walk sessions and stamps), never from a walker's dispositions, which are the capacity loop's drain meter and sit beside the ratio as supply; value as objectives moved (lead-measure readings by their standing) and misses prevented (the jobs' readings, Part 2). Minimized alone, attention optimizes silence: a walker that stops escalating scores perfectly, which is why the ratio is never read from one side. The markers above (instruction length, ceremony survival, exception-to-noise, register growth, the absence test) are this ratio's leading indicators, read before the scoreboard exists.

(PROVISIONAL: ratified 2026-09-14 on the operator's north-star ruling for the canon and both editions, "create value with as little human attention and effort as possible", corrected to the ratio in the same sitting. Receipts in hand: the markers of Part 1.5; the estate's attention lanes (deadline-grade only, compliance silent, 2026-08-31); the fleet's DEC-162 (2026-09-12 12:49 SGT: the concern estate's register walk agent-run, the operator receiving inspection escalation only) and walk 445 (2026-09-13 04:36 to 04:52 SGT: 33 dispositions by the agent, one escalation to the operator). Due: one deployment's scoreboard reading both numbers for one quarter, keyed to the concern; on Relay the rows exist for value (lead-measure readings with standings) and for two of the three seats (rulings, escalations) and none is keyed to a concern; the estate's session and ledger rows likewise.)

The push ladder

A structure reaches the operator by one of five rungs, each cheaper in attention than the one above, and every recurring ceremony is a candidate to descend one rung. PULL: a surface that waits to be opened; it dies (the fitness test). PUSH: the state arrives on a clock with a pre-computed docket (a pre-brief, a digest, an hourly line). CEREMONY: push with a response slot, every item ending in one of the five dispositions. AGENT-RUN WALK: the walk itself is the machine's and only exceptions arrive (the default walker, Part 2.5). NOTICE: nothing arrives until a threshold breaks, the same-day alarm of Part 5; a sensor is the mechanism that turns an always-identical walk into a Notice, never the rung's name. "A good ceremony shrinks" is this ladder read downward, and the attention law is what it optimizes: the operator's seat climbs the ladder only when a rung fails loudly, never by habit. THE REPEAT CLAUSE (2026-09-18): a push that has reached the operator returns only when a threshold breaks (the item is new, or its age crosses a declared rung), never because its clock fired again with nothing changed; a channel keeps one docket per clock (a morning brief, a day's first line) and thereafter carries only the new and the crossing, the standing set living on the

scoreboard and the ceremony's docket. A line repeated on a clock is wallpaper: attention spent with no seat in it, the pull surface's death in reverse. First instance: the estate's three descents of 2026-09-18 (the hourly hook to a daily docket plus thresholds at 3, 7, 14, 30, 60 and 90 days; the email pass silent on a clean run after the day's docket; the morning brief's exceptions line carrying only what is new since yesterday), read on the attention scoreboard's pushes row. A job declares which rung its reading reaches its hirer by, and how it returns (the push moment, Part 2); the ladder is read from those declarations, and the scoreboard's pushes row derives from them.

(PROVISIONAL: ratified 2026-09-14. Receipts: the fleet's G8 concern segment descended from a human ceremony to an agent-run walk (DEC-82 to DEC-162, ten days; the operator's concern attention from "walk 32 spines" to one line); the estate's 08:30 pre-brief and hourly hook (push), its Sunday walk (ceremony), its email pass (an agent-run floor walk with a ledger receipt), and the walk ladder's own sensor-to-Notice sentence. Due: one ceremony observed descending a rung on a deployment with its human attention reading falling; the estate's candidate is the Sunday walk's money segment.)

Part 1.75: Authors and operators, and when structure pays

A well-run company is not a company of intuitive ontologists. Ritz-Carlton's average manager never authored anything: Horst Schulze authored the service ontology once (the Gold Standards, the \$2,000 pre-authorized empowerment rule: a defined enforcement boundary), and everything since is trained OPERATORS running an inherited stack, with the daily lineup as the ceremony and QA audits as enforcement. McDonald's is the extreme: a franchise manual so completely externalized that teenagers operate the restaurant. Mature industries commoditize the authoring: telcos import eTOM, banks inherit settled banking ontology plus regulation (enforced ontology imposed from outside). A company in a NEW category has nothing to import; its leads must author, and **authoring is the rarer skill this module trains**: most corporate leadership programs only mint operators. The caution: inherited stacks rot without gardeners; big-company bureaucracy is an ungardened ontology, and the genuinely great operators institutionalized the gardening itself (kaizen is Toyota's gardening as a standing institution, and it is the half every Toyota cargo-cult skips).

The stage curve

When structure pays: the variable is coordination surface (people x functions x handoffs), not the funding round; rounds just correlate with it:

- **Founder-scale (Seed): hustle is correct, not primitive.** Everyone is in one room; the founder IS the ontology and walking enforcement. Heavy structure here is negative EV by this document's own fitness tests: the domain pivots weekly, the ontology would be gardened to death, the docket is empty.
- **Operator-scale (Series A): one tacit operator suffices,** applying structure just-in-time from intuition. Fragile: bus factor of one, and a trap: the operator's excellence DELAYS externalization, because everything works while routed through them, so nobody feels the missing structure until the operator saturates, and then it arrives as a crisis.
- **System-scale (Series B+): only externalized systems scale.** Either every executive carries the skill, or the stack is institutionalized (authored structures + trained operators + enforcement + gardening).

Two amendments: **agents shift the curve left** (enforcement no longer costs executive headcount, so a Series A company can afford Series B institutionalization early), and **a moat buys tolerance for mess** (structure is key to compounding, not to survival: a company with brutal switching costs can export its operational mess to its customers for years, paying in growth rate and customer resentment rather than death, and that resentment is the leading indicator of how moated incumbents eventually get displaced). Survivorship applies: the messy survivors are visible, the messy dead are not.

(SETTLED: founder-scale hustle correct at seed; agents-shift-left running in both estates.)

Construction grounds

Working ontologies are always EXTRACTED from evidence; failing ones are always speculated.

The top-down-versus-emergent ontology debate dissolves on inspection: the real choice is which evidence ground fits the domain. A SETTLED, EXTERNALIZED domain (a regulation, a reference model, twenty years of source systems) is imported and pruned: top-down is honest there because it is archaeology, not invention. A NOVEL, INTERNAL domain (your own operations) is extracted door by door, because its true structure lives distributed and tacit in the operators and is revealed only through use. The committee ontology grounded in neither is the form that fails from every direction.

Enforcement scales the stakes: the stronger the gates, the more receipts-first the authoring must be, because a wrong ontology under weak enforcement is ignored, and under strong enforcement is inflicted.

Top-down keeps exactly one legitimate role in an extraction shop: THE COVERAGE WALK, a periodic audit ceremony that enumerates the territory, marks which domains are committed, and consciously accepts the unmanaged remainder. The map is walked top-down; nouns are never minted there. An accepted gap is a decision; a silent one is a risk.

(SETTLED: distilled 2026-08-27 from the Palantir-versus-doors first-principles sitting; promoted 2026-08-29 on its pre-registered receipt, the first import-ground domain shipped — the fleet's ISMS certification-readiness concern, ISO/IEC 27001:2022 imported and pruned to Gotrade's scope, live at v2 with typed organ refs and its un-walked amber honest.)

The authority law

Authoring authority is GRANTED, RECORDED, and SCOPED, never assumed. Operators need no license: writing rows, running ceremonies, and walking exceptions stay open to everyone. Authoring — defining layer-1 nouns and layer-2 verbs, which enforcement then inflicts on everyone downstream — requires a recorded grant, and the grant has TWO paths: the APPRENTICESHIP — a reviewed first authoring, where the unlicensed author walks the door with a licensed one same-day and the pass mints permanent self-serve, so the license dissolves itself author by author at the pace real authors arrive (the fleet's working path, and the lighter of the two) — and CERTIFICATION, the practicum's day-60 pass minting an Author. Either way the grant lands in an authority register (who may author, in which concern's scope, granted when, on what receipt).

Two guards keep the license from becoming a priesthood: the pain door is never blocked (the unlicensed propose through agents, which enforce the admission standard mechanically; a licensed Author ratifies the apply), and the licensed path must remain the easiest path, or the org routes around it into shadow structure.

(SETTLED: the first authority register ran at a live write path 2026-08-28 — the fleet's concern-apply allowlist, ten frictionless hours on day one, with the two reviewed births producing the estate's two best declarations.)

Part 2: The three skills (what actually separates good from bad)

The layers are anatomy. The skill is physiology:

1. **Diagnosis: seeing an incident as a missing noun.** Chaos is never "people not trying hard enough"; it is a structure that does not exist yet. Interruptions plaguing the team = there is no Intake. Learnings evaporating = there is no Experiment register. The unskilled manager works harder inside the chaos; the skilled one asks "what noun is missing?" Think in classes, not instances. Since v4.3.0

the diagnosis has a step before the noun: the skilled one first asks whether anything should exist at all (the five-step order, Part 4: the requirement questioned by its hirer's name, then the deletion question), and a pain that survives that question is a missing noun. The commonest error of a capable author is to structure a thing that should not exist.

2. **Design taste: which ontology, not just an ontology.** Two managers both build a standup; one's sections mirror real decision points, the other builds status theater. The fitness tests in Part 4 are the review side of the difference; the design side has a procedure: **enumerate the jobs before naming the nouns**. List who will hire the structure and the questions they will bring to it ("what is late?", "who has the ball?", "have we already learned this?"), then derive the nouns and fields from the jobs. Ontology engineering calls these competency questions (Part 5.5); product calls it JTBD. It converts "every field changes a decision" from a filter into a generative method: fields are born from decisions instead of deleted for lacking one.

Guard, the admission standard: a job enters the enumeration only by stating four things — WHO hires it (a person or process by name, never "we"), on WHAT CADENCE it fires, WHICH DECISION its answer changes (else it is curiosity, not a job), and its FAILURE COST when unserved; speculative jobs are how the 22-field form gets built. The dialect is part of the standard (the readability law, Part 4): a job is written as the HIRER'S QUESTION, in the hirer's words — the word "answer" in the third part presumes it — and an assertion in the system's voice must be reverse-engineered back into its question before its decision can even be tested. Conversion CONSERVES: a concrete term may leave a job cell only if it lives in the spine — intent-class terms (definitions, contracts) in the spine's own ontology, mechanism-class terms anywhere the organ index reaches — relocated there in the same edit when absent. The direction of derivation is the method's spine: jobs then structures, never structures then jobs — writing a job to bless a structure you already want is post-hoc rationalization, the disease pre-registration exists to kill. And the door stays open after birth: a job discovered late enters by amendment with full honors — gardening is the JTBD door's safety net, a late job is not a failure, an UNNAMED one is, and the enumeration is never finished at birth; it is finished never. The standard is the five-step order's step one read as a challenge (Part 4): the hirer named answers for the requirement TODAY, and the requirement is questioned whoever gave it, most of all a smart person; a job an intern wrote two years ago that nobody will now answer for is not a job.

This gives Ontologize two doors (the two doors): the PAIN door (the catechism, Part 3: a live domain, entered through recurring pain) and the JTBD door (a greenfield domain, entered through job enumeration). Two sizing rules ride the doors (ratified 2026-08-27, from the first live teach): THE DEPTH RULE — the doors do not just start the loop, they bound it: ontologize only as far as needed to solve the pain or satisfy the jobs, never for ontology's sake ("should I ontologize strategy?" has a one-question answer: which door?); and THE SPLIT RULE — a concern whose enumeration passes roughly ten jobs is two concerns wearing one name; split it.

And THE DONE-TEST routes the intake (stamped 2026-09-04, the codify-first sitting's coda): any ask, from any hirer, classifies by one question — "when would this be done?" — with four answers. "NEVER; it keeps flowing" is engine-shaped: a concern, through the JTBD door NOW — its cadence is knowable at minute zero, and recurrence is only the PAIN door's threshold, never a waiting requirement on a knowable engine. A DATE or END-STATE is destination-shaped: a project on a board, closed by its definition of done, never a concern. "WHEN THE NUMBER HITS X" is the goal layer (the grain law's clause): an objective borrowing engines, not an engine. And a shape that MISLEADS — the one-off-looking chore that quietly recurs (the SP bill), the standing-sounding ask that dies after one use (the 22-field form) — files as a project with a pre-registered recurrence flip: the ambiguity fallback, cheap and self-correcting. The test is the two-doors model spoken at intake speed, and it is how a noticing agent routes a conversation without teaching the user a single term of art.

(Renamed from incident/survey at v3.0.0: the field-tested names from the first live teach, 2026-08-26, displaced the bookish pair before any cohort learned them. Admission standard, retro-fit trap, and late-entry clause ratified 2026-08-26, sharpened on the J8 case, where a summary-corpus proposal failed the guard and its real job then entered by amendment with a serving path.)

The verbs of arrival, named at v3.9.1 so the act, the event and the state stop borrowing each other's words: to DECLARE a concern is to author its declaration through a door: the jobs admitted, the nouns ontologized, the verbs processized, the organs indexed, the past inducted. Its BIRTH is the day the declaration enters (the spine's birth story records why; the orphan rule asks at birth), and a COMMITTED concern is the result: one the estate manages from then on, with a walk. The coverage walk lists the candidates; only a door admits them. "MO-ify" and "concernify" are the operator's slang for the whole program and for the outcome, useful at the desk and never canon words, since a canon that minted a verb named after itself would have stopped extracting. (Ratified 2026-09-11 on the estate's first coverage walk, where the question "shouldn't we concernify what already runs" needed the three words to answer.)

3. **The two loops: enforcement and gardening.** Structures die of non-use, and bad design is the leading cause of non-use (a 22-field form does not get "enforced into" adoption; it gets routed around). A structure is alive only if skipping it is loud (enforcement) and someone amends it when reality pushes back (gardening).

The grain law

A concern is a managed PIPELINE, not a category and not an outcome. The split rule caps size by count; this law tests it by KIND, on a ladder. A CATEGORY ("Marketing") is too big: no single owner, no closable noun-set; its spine would be an org chart, and nothing can conform to an org chart (that axis belongs to responsibility, not territory). An OUTCOME ("Retention") is not a domain either: it is a KPI axis; you do not walk an outcome, you walk the programs that move it (outcomes belong to goals and scorecards). A concern is the PIPELINE underneath: one owner, a closed vocabulary, real records moving.

Four portable tests (the four grain tests): (1) ONE accountable owner — two owners means too big; (2) a closed noun-and-verb vocabulary that fits one spine page — if you cannot close it, too big; (3) two or more admitted jobs with DIFFERENT named hirers — one job serving one hirer through one register is an ORGAN, and an organ lives inside some concern, never as one; (4) the spine test — a divergence between the spine and a running organ must yield a question a human can rule on ("Marketing diverged from reality" is unrulable; "a redemption row missing from the reconciliation" is a Tuesday).

And the CAPACITY LOOP (renamed from the capacity guard, 2026-09-04, the codify-first sitting — a guard refuses, this steers): every concern requires a live, drained walk — a conservation invariant binding the MATCH between concerns and judgment capacity, never the ambition to codify, because the alternative to a codified domain is never a managed one: the work happens either way, and uncoded it happens in the dark. Codification is therefore always preferred — it converts the silent gap into VISIBLE DEBT (the aging court, the un-walked docket), and visible debt is what staffing steers by. Under pressure, two levers by stage: shrink the count (founder-scale, the walker set fixed — lumping starves clarity, slicing starves the walkers) or GROW the capacity (system-scale: inspection is operator-tier and unlicensed under the authority law, so walkers are hireable and ownership delegable, with agents multiplying each walker's reach). The staffing case reads from two meters, both receipts-derived: the debt's aging (demand) and the walkers' recorded drain of dispositions and rulings (supply — an artifact read, never surveillance); capacity is added when both run red, a red queue over idle walkers indicts routing rather than headcount, and drain-rate stays a

capacity signal, never a performance quota. The loop's one guard residue: never read a red meter as quiet, and never build a surface that hides the meter.

The goal-layer clause (stamped 2026-09-02, on the first outsider running the vocabulary back — "how do concerns link to growth objectives?"): an outcome refused as a domain is not homeless — it lives in the GOAL LAYER above concerns. An OBJECTIVE is a closable battle: from X to Y by when, one accountable owner, ending won, missed, or abandoned — where a concern is never done. The two connect ONLY through LEAD MEASURES — predictive, influenceable, one-owner bars — and a lead measure's readings come from some concern's standing organs, never from fresh bookkeeping: a goal that mints a parallel register for what a concern already tracks is shadow structure wearing a scoreboard. Concerns own the pipelines; objectives borrow them and point them at targets. Engines are permanent, destinations are reached: when the battle closes, the machinery remains, which is the compounding claim in one sentence. The layer needs no new process vocabulary (its cadence is an ordinary ceremony) and one organ shape (the scoreboard, Part 2.75, deployment-local with its flip condition armed). Worked at the canon's own example: training is the concern, never done; sub-4 in this race is the objective, closable; weekly zone-2 volume is the lead measure the training log already feeds. Import ground: 4DX (Part 5.5).

(SETTLED: distilled 2026-08-27 from the fleet's first sizing ruling (DEC-29, the grain ladder) — a gift-card pipeline admitted, "Marketing" and "Retention" refused — and promoted 2026-08-28 on its pre-registered receipt: the second independent sizing ruling landed on the ladder unamended, its author running the exclusion analysis herself and stamping the one genuinely uncertain boundary job provisional with its own receipt-due condition, the grain machinery absorbing what a premature call would have forced. The goal-layer clause stamped 2026-09-02: first declared instance the fleet's WIGs concern — closable battles, lane-declared lead measures, readings-never-testimony, live with receipts — the second pre-registered, the estate's hyrox goal-layer declaration, made the same sitting.)

The species law

The done-test's answers name three SPECIES of managed thing, and each has its own anatomy: a CONCERN (a domain once committed: an engine, never done), a PROJECT (bounded work with a pre-registered definition of done), and an OBJECTIVE (a number by when, the goal layer). The fourth answer, the misleading shape, is not a species but a transition: it files as a project with a recurrence flip and CONVERTS into a concern when the flip fires. A concern's anatomy is the spine (Part 2.5); an objective's is the goal-layer clause above; this law gives the project its own, because two deployments ran projects for months on a noun phrase in a routing rule (an estate's item tree, a fleet's boards) and both paid for it the same way: membership kept by hand in the concerns' declarations, boards nobody could draw, and work that was concern-owned in substance but unplaced in the machine.

A project is a noun of the WORK-MANAGEMENT concern, and that is its one home (the jurisdiction law, unchanged: one noun, one home). The work-management concern is the concern that owns the work noun and its register, and it passes the grain tests on its own receipts, not as a category of "all projects": one owner; a closed vocabulary (item, kind, ball, next action, tickler, the served concern); jobs with different hirers (the operator at the daily walk: "what needs me today?"; the agent at the pre-brief: "what surfaced overnight?"; a delegate: "what is mine?"); and a rutable divergence (an item carrying a next action with no holder). It owns every project's LIFECYCLE and never a project's substance. The anatomy: identity (a minted code); a scope sentence; the DEFINITION OF DONE, pre-registered, which is what separates a project from a concern; one owner; a CLOCK of two kinds, a HARD clock (the world set the date and enforces it, a filing, an expiry: it moves only when the world's date moves, and missing it is a miss) or a TARGET clock (self-set pacing, the appetite: it moves by a recorded shift, and every move is a counted slip), the two binding to the two attention grades (hard is deadline-grade, target is walk-grade) and required only while the project is active; what it SERVES; its work on a board; decisions pinned by reference into the concern

they bind; and a close in exactly one of DONE, ABANDONED, or CONVERTED — where DONE means the pre-registered definition met; a date passing with the definition unmet closes ABANDONED, or re-scopes by a recorded ruling with the residue minted as a successor project, never DONE by the calendar. The BASELINE (the definition of done, the clock and its kind) moves only by a recorded SHIFT carrying old, new and a reason, never silently: a shifted line is information (one slip is estimation; the third slip on one project is a signal about the system, for Garden; many single slips across projects is the capacity meter reading red), an unrecorded shift is drift, and a definition that slides to match what got built is the post-hoc failure. The definition is frozen once the project is closed, because the ruling was judged against it.

The SERVES question is asked at declaration, never at capture, and takes two answers: a CONCERN (the project builds or amends one of its organs, or is its induction), or LOOSE, DECLARED (personal or exploratory work, said out loud). Loose work is not homeless: its home is the work-management concern by construction, its board that concern's own, its walk that concern's ceremony; only the silent no-answer is forbidden (the orphan rule applied to work). An objective is never an answer, because an objective owns no organs and no walk of its own beyond its scoreboard ceremony (the goal-layer clause): a project MOVES an objective by reference, through the lead measure it acts on, and the objective's scoreboard reads that lead measure from a concern's standing organs, never the project's rows. One served concern, many typed references (a project touching three concerns has one accountable walk and pointers to the rest); re-homing a project's served concern keeps its identity; a concern's board, an objective's scoreboard, and the loose docket are DERIVED views over the one work register, never kept. The triangle, for a project that serves a concern: the concern BIRTHS it, it DELIVERS INTO the concern, and an objective is MOVED BY it; a loose project delivers into nothing but its own close, which is what makes it loose. Loose projects recurring in one territory are a concern being born (the pain door's threshold).

The grain inside a project is three nouns, all homed in the work-management concern, each defined by the sentence it must be able to say. A PROJECT says "this is done when ..." (the end state, the owner, what it serves, the clock). A WORKSTREAM says "the thread on Y": a parallel thread of the project with a focus, and NO OWNER (v5.0.0). It moves by a rolling NEXT MOVE, a ball (who, what, since when) with an update each time it moves, and the holder may change move by move: whoever moves the thread this week need not move it next. Its path ahead is a plan written in its context, never a stock of tasks. A TASK says "the next move is ..." (one ball: who, what, since when); a step of a thread becomes a task only when it runs alongside another holder's move, carries a world's date, or needs a record of its own. Two invariants underneath: ONE BALL PER TASK and ONE LIVE MOVE PER WORKSTREAM; and no holder defaults down the tree: a ball is named where it is set, and one named by nobody is the operator's. THE WALK OF A PROJECT reads its threads, never its tasks: at each inspection every open workstream has MOVED since the last one, RESTS on a declared tickler, or WAITS ON a named blocker; anything else is the walk's exception. A waiter reads as waiting, never as the exception: the walk chases its blocker, which carries the exception where it lives. WAITS ON is one typed edge for a dependency and a snooze-until-something-closes alike: any of the three may wait on any other, never on what contains it or on its own part (a workstream on another workstream of its own project or of another, a project on a workstream of another project, a task on whatever it needs, or on several at once); a blocker is named only as far as the waiter's readers may read it, the audience each deployment's to set; the waiter rests on a backstop date while any blocker is open and wakes, with a note saying why, when the last one closes; a person moving the waiter by hand clears it; and the walk chases the blocker, never the waiter. The sentence test is a noticer, never a gate: a workstream that cannot say which thread it is is either the project itself or a label on tasks, and a project with one thread needs no workstream at all. Projects compose under projects (a program is a project of projects, by the BELONGS_TO structural verb). How a deployment REPRESENTS the three — one work record carrying a kind, or three records with their own states — is

a binding, earned by that deployment's receipts (the qualifier corollary), never prescribed here; the nouns and their sentences are the canon's. Import ground: PMO practice (portfolio, program, project; the stage gate) for the lifecycle; the category leaders' two work nouns and the fleet's board-workstream-task for the grain (Part 5.5); the fleet's G8 walk for the thread and its waits (DEC-281).

Any of the three may REST (v5.0.0; a project only, before it): a project, a workstream or a task, hidden until a date (a tickler) or a condition (it WAITS ON a blocker), then returning, neither open nor terminal to the walk. On an active project rest means resume (the clock stands, and the walk still calls it late); on a shelved one it means reconsider. Whether rest is a state on the record or a field beside it is a binding (the fleet's snoozed state with its reason; the estate's tickler field), and a rest never outruns a hard clock: the clock wins and the row returns.

(PROVISIONAL: ratified 2026-09-07 in the estate's project sittings (NW-ANA-44), on three rulings by the author — "the gate requires an answer, not a domain"; "a workstream is for somebody to do something"; "MO is for organizations, where several people move a project at once, so rule the structure". Receipts at ratification: the estate's work register carrying a served-concern field with the concerns' boards DERIVED from it, the hand-listed boards retired the same night; two projects closed at their date by a recorded re-scope, the residue minted as a successor rather than the deadline stretched (NWTD-ITM-247, 257); and the audit of a keyed register armed as the first project declared under this law: one served concern, three workstreams, three owners. DUE, on the second deployment and testing the claims, not an implementation: one project declared serving a concern and one declared loose; one re-homed to another concern keeping its code; one converted into a concern by the recurrence flip; one project walked at the thread level (every open workstream moved, resting or waiting at an inspection), a thread whose next move changed holder, and a wait woken by its blocker's close; one single-thread project that needed no workstream; one project composed under another; one close ruled ABANDONED against a passed date; and no project the SERVES question could not route. A project the question cannot route, or a walk that needed a fourth noun, falsifies the law as written. v3.6.1, 2026-09-07 evening: the clock's two kinds, the baseline shift and rest added inside the law. Receipts at that ratification: Relay's Board carrying target_date and hard_deadline, definition_of_done and a snoozed state with its reason, and its parked agenda workstreams returning on a date or a condition (read from gtos-comms-console's models); the estate's tickler-first register, the journey certificate's hard date beside a Sep 27 target, a due moved on a dated line (NWTD-ITM-247) and a definition changed as an undiffable blob (NWTD-ITM-258). DUE for the added clauses: the estate's shift note landing (NW-ANA-47) and Relay's boards recording a target-date change with its reason; a rest that outran a hard clock unnoticed falsifies the rest clause as written. v5.0.0, 2026-09-27: the first thread walk falsified the owner clause, as this law's due invited. The author's own ruling at ratification ("a workstream is for somebody to do something") was reversed by his practice: "the way I initially designed WS is not group by owner ... once it's done by person X, the next action to move that WS can very well be person Y"; "every WS have movement per inspection ... the parallel threads that can move at the same time"; "I don't really put many tasks in the WS, I just have an update and the next action" (Norman, 2026-09-26, NW-ANA-94). So a WORKSTREAM is a thread with no owner, its next move's holder changing move by move; no holder defaults down the tree; the walk reads threads by moved, resting and waiting; WAITS ON joined the grain as one edge, any grain waiting on any other across projects ("Proj and W can depend on W on this proj or even other projects, which models real life projects", Norman, 2026-09-27); and REST, a project's only until now, became every grain's ("Proj, W and T each can be snoozed"). Receipts: the estate built it first (nwos-todo 02ffc11: no ball inherited, the walk on the project page, waits on with a backstop and a wake; personal-work-management D-threads-and-waits) and walked the Mom's-birthday trip as five threads, three of them waiting; the fleet's DEC-281 (the G8 walk of 2026-09-23) had already made dependency and snooze-on-condition one edge, and DEC-312 (the G8 walk of 2026-09-24) whole boards as blockers and several blockers to one waiter, the second deployment's receipt for waits on; the fleet already refuses a row waiting on its own board, a task on its own workstream, a workstream on its own tasks, and any loop. The fleet's boards have rested all three grains since T61 (2026-07-22), the second deployment's receipt for REST at every grain. The fleet's DEC-144 (2026-09-09) is its species-law binding, and only its workstream clause stood on the old sentence: an owner column on the board workstream, carried by 19 of 428 workstreams with 7 open boards walked by owner (read 2026-09-27 16:30 SGT); Norman re-ruled it the same day in the fleet's lane: the owner column retired, a workstream's live move derived from its first ball-holding task, and a whole board allowed to wait. v5.0.1, 2026-09-27: the fleet lane's read of 89ac448 (Dario[RefactorWorkstreams]), five wording corrections landed: DEC-144 named as the whole binding with only its workstream clause re-ruled; DEC-312 cited; a project waiting on its own workstream removed, since the same sentence forbids waiting on one's own part; the writer's reach replaced by the waiter's readers (a blocker named only as far as they may read it); the glossary's Waits on given its never clause.)

Home by construction, depth by receipt

Everything that RUNS is homed at birth: every noun, organ and process is a concern's own, a ceremony's (owned at the level spanning what it convenes), or a project's on the work-

management concern; only the silent no-answer and the standing orphan are forbidden. The home is given by construction, one row through a door; DEPTH is earned by receipt, the depth rule unchanged: ontologize only as far as the jobs or the pain require, and let the walk find the rest. The doors are therefore AUTHORIZING ACTS, never waits: the JTBD door is the first page of a declaration, drafted from ground and prior art (a settled, externalized domain imports and prunes, the construction-grounds law) and ruled by the operator by number; the pain door's recurrence threshold governs amendment, never birth. The orphan rule's third answer, the workshop, folds into the project species: a workshop is a loose declared project, explicitly experimental, and its conversion re-homes its organs into the born concern by custodial re-homing (the re-homing rules, Part 1); the orphan rule's legal answers are two. The coverage walk keeps its ACCEPTED state for territory where nothing runs, because the law binds things, never territory. Three guards. The only residual home is the work-management concern, and only through a project with a definition of done: a "General Management" concern minted to satisfy the rule is the junk drawer the grain law refuses. The law is affordable only under the default walker (the walk ladder, Part 2.5): a home whose walk is human spends the attention law's denominator, so homing everything without agent-run walks mints dead walks, and a home with no walk receipt in one cycle routes to Garden (merge, or convert back to a project). And the fitness tests apply at the first walk, never at birth, so a one-line spine cannot fake a home. Enforcement climbs the ladder: the OKG's orphan question is the Notice, across six classes (registers no concern claims, surfaces no concern names, workflows no concern names, notices no job names, open projects with SERVES unanswered, implied nouns a link names that no home declares); a gate at the declaration door only when coverage makes it honest.

(PROVISIONAL: ratified 2026-09-14 on the operator's ruling that anything that happens must belong to a concern, with the jobs estimated from prior art, and projects in infancy homed on the work-management concern until they graduate. Receipts in hand: the estate's 23 declarations and its coverage register (NW-COV, 2026-09-11); the estate's credential ruling of 2026-09-13 (an ask for a password concern answered by homing four jobs in its security concern with the flip written: the law's shape before its name); the fleet's 58 declarations and its Coverage tab (one orphan class wired: 17 of 35 engine registers unclaimed on 2026-09-14; 177 of 188 open boards with SERVES unanswered, the legacy stock before SERVES became required at birth, DEC-144; the accept-the-gap and nudge verbs). The fleet's cost, named: its Charter speaks "workshop" in a second sense, the agent's own estate as the build lane (T8, DEC-83), naming some forty homed organs; that sense retires for "estate lane" at the fleet's next vocabulary cut. Due: a coverage walk on each deployment reading zero orphans across the five classes, and one project converted into a concern with its organs re-homed by alias.)

Jobs carry readings

A job carries, beside its four admission parts, a KIND, a READING, a TARGET and a GRADE, so the J-table declares the concern's STATE VECTOR: each reading a state variable, its target the constraint, an objective's lead measure the reward. The KIND is the role the consuming hirer's question plays, extracted from the living declarations and never designed: an OBLIGATION job (something must happen by a clock or on an event, and the miss is a rate read from a register or ledger: late, unrouted, aging, dark, missed; a machine hirer checked per cycle is an obligation too), a LOOKUP job (an answer on demand; the miss is a hunt or a wrong answer, readable only where the answering door logs its reads), an ASSURANCE job (is the engine itself running and how much does it cover, read as liveness and coverage), and a RECOMMENDATION job (the answer is a recommendation the consuming hirer rules on; the reading is rulings pending and their latency, read from the decisions register). A capture job, where a hirer hands a thing in through a door so it survives, is an obligation read downstream (the captured row aging unpicked); the un-captured is unreadable by construction and is the Scribe role's work, not a kind (three receipts, exactly at the threshold, on 2026-09-14; the flip pre-registered: capture earns a row the day a door's write rate is read by a walk). A job with a recording hirer and a ruling hirer carries the ruler's kind, never two. The READING names the organ it is read from (a handle in the concern's own organ index) and the derivation; it is derived by the OKG on read and born VERIFIED, or recorded by a person and born

CLAIMED until its declared verifier ratifies it; it is never typed into the declaration, and the standing is what a scoreboard shows. The TARGET is the constraint (zero misses, full coverage, a latency). The GRADE is the ATTENTION GRADE the canon already owns (deadline-grade or walk-grade, the species law's two clocks); money enters only where a ledger already reads it. A job whose reading is none stays admitted as a loud gap the coverage walk sees, the same shape as a register declaring no closure; a job that cannot be read under any kind is information: a why-sentence for the birth story, or a policy for a register. A lookup job's reading is null wherever the answering door does not log its reads, and the law says so rather than promising it. Engine readings belong to the concern; value readings stay in the goal layer; the two meet only in the attention law's scoreboard. Machine form: the jobs component of canon/SPINE-SCHEMA.yaml carries kind, reading, target, grade and push; the derivation's encoding and the standing's spelling are bindings. THE PUSH MOMENT (v4.5.0): a job carries, beside its reading, its PUSH MOMENT: the RUNG of the push ladder (Part 1.5) by which the reading reaches the hirer (pull, push, ceremony, agent-run walk, notice) and its RETURN, the rule by which a reached hirer sees it again (new only, or the declared age rungs it returns on; the repeat clause's two thresholds); the clock stays the verb's, authored once in the runner. Absent, a job's push moment reads PULL, which is legal for a lookup job and a loud gap for an obligation or an assurance, so an undeclared push is never a silent zero on the scoreboard and the coverage walk can list the obligations nobody is pushed. An alert is a job's notice, never a thing of its own: its consumer is the job's hirer, the decision it changes the job's, its threshold the job's target, its meter and its retiring reading the job's reading, and the guard it is retires as any guard does (the five-step order, Part 4). A deployment's catalog of alert kinds (the kinds routed to rooms, the workflows that page) is a BINDING, an index and never a home: a job cites its kind exactly as a verb cites its workflow, and a kind that fires which no job names is an orphan of the SIXTH CLASS on the coverage walk (Part 2). The attention law's scoreboard derives its pushes row from the declared push moments (a line reached a human seat iff the job's rung is a push, a ceremony or a notice to a person), never from a hand-kept list of workflows that send; and a job whose push is read at a rate below the deployment's bar for two cycles is a Garden question on the concern's next walk, never a louder page. A read meter is a FLOOR: a read outside the app (a banner, a lock screen) leaves no mark, so below the bar means rarely opened, never unseen.

(PROVISIONAL: ratified 2026-09-14 on the operator's question whether concerns should carry measurable metrics rather than prose jobs, ruled "and, not rather than": the metric is the job's failure cost made observable. Receipts in hand: the extraction over the estate's 124 jobs and 47 fleet jobs in ten declarations (2026-09-14): obligation 61 and 20, lookup 38 and 10, assurance 9 and 6, recommendation 11 and 8, two policies and three captures on the estate, no why-sentence; 113 of 124 estate jobs and 44 of 47 fleet jobs name an organ in their own spine that their reading is read from; eleven estate jobs unread (five with no organ, six whose door keeps no log), the loud gaps the law admits; the estate's attention lanes, whose deadline-grade readings already exist as the walk sets' open counts; Relay's lead-measure readings (1,720 rows on 2026-09-14, 907 verified, 812 claimed, one refuted) as the standing's receipt. Due: one deployment's OKG deriving readings for every obligation job of one concern, and a walk reading them; on Relay, the validator resolving reading.organ, the pulse picking it per job, the builder projecting it. The push-moment clause, v4.5.0, ratified 2026-09-22 on two receipts. The fleet's: a hand-kept map of fourteen alert kinds routed to rooms (DEC-209, 2026-09-18) with no job naming any of them, and a push inventory that had to be read from the code by hand (T46: 1,863 notifications delivered to the operator in fourteen days, 616 from one agent, the two clock chases 425 of them; then descended one push at a time to age rungs, the chases on 2026-09-18 falling from 91 a day to single digits, the runtime-drift page on 2026-09-22, core #1128, v1.8.52, the first rung page due the next noon on the box that had been paged five mornings straight). The read rates first quoted from T46 (the per-event pages at 90 to 100 percent, the chases at 0 and 9) came from a badge column that is not read state; re-derived on 2026-09-23 from the readers' thread markers they are 0 to 20 percent and 15 and 43: the operator opened almost none of either, and the ruling never rested on a read rate (v4.5.1; DEC-265's corrected receipts, console v1.40.214). The floor (v4.5.1) on both deployments: the fleet's thread markers miss a notification read outside the app, and the estate's WhatsApp read receipts miss a read in the banner (the estate's push meter, NW-ANA-91, its first read 2026-09-26). The estate's: its scoreboard reading the pushes row from a hand-kept list of three workflows that send, the mirror this field retires. Due: on one deployment every alert kind that fires named by a job's push moment, the scoreboard's pushes row derived from the declared rungs, and one push descended or retired on its read rate at a walk; on the other, the induction begun.)

Part 2.5: The five verbs (the practice)

The layers, projected into what an author-operator actually DOES to a concern. This is the sequence students should carry in their heads:

#	Verb	What it does	What it projects
1	Ontologize	Name the nouns: entities, fields, legal states, declared places	Layer 1 (authoring)
2	Processize	Define the verbs: SOPs, ceremonies, cadences, owners (including the two meta-ceremonies below)	Layer 2 (authoring)
3	Log	Every event lands as a row, as it happens	Layer 3 (runtime)
4	Inspect	Walk the exception set the machine surfaces; disposition every item	The fast loop's human half (runtime)
5	Garden	Amend the ontology and SOPs as reality pushes back	The slow loop (runtime)

Two teaching points, each load-bearing:

The list is two kinds of things, not one pipeline. Verbs 1-2 are authoring acts: one-time transformations of the concern, revisited only through Garden. Verbs 3-5 are runtime FUNCTIONS the running system must perform, each carried by processes authored in step 2: Log is carried by the "write the row" clauses inside every SOP, Inspect by the walk ceremonies, Garden by the gardening cadence. The classic objection ("the walk's SOP is defined in Processize, so isn't Inspect just part of step 2?") applies equally to Log, and that is the tell: 3-5 name duties whose PRESENCE the deployment is audited on, not additional things to author. The suffix asymmetry is deliberate: you -ize twice, then you run. ("Processize" is a deliberate coinage mirroring Ontologize; the dictionary neighbors are proceduralize and operationalize, and the symmetry was ruled worth the coinage, 2026-08-25.)

Inspect and Garden are meta-processes. Log operates on the domain (deals, sessions, campaigns); Inspect and Garden operate on the system itself: Inspect carries the fast loop's human half (runtime state versus intent), Garden carries the slow loop (schema and SOPs versus drifted reality; the loop model is Part 1.25), and Garden is the only runtime verb allowed to reopen layers 1-2. The system manages the work; two of its processes manage the system. A deployment that processized only domain verbs, and forgot to processize its own inspection and gardening ceremonies, is exactly the deployment that rots.

(Ratified 2026-08-25. Born from the Inspect-versus-walk question; the observation that the walk's SOP lives in Processize is what forced the authoring-acts versus runtime-functions distinction.)

Two kinds of processes

A process is either a concern-owned SOP or a ceremony, and forcing a ceremony into one concern mints a junk drawer. A CONCERN-OWNED SOP belongs to exactly one concern: authored in that concern's SPINE (the term of art for a concern's governing spec document, whatever its local name: a dossier, a charter, a concern page), one owner, the ownership invariant strict. A CEREMONY is a cross-concern CONVENOR: it walks segments that several concerns each own (a standup, a weekly walk), so it is owned at the level that spans what it convenes and is never forced into a single concern (forcing it produces the junk-drawer concern). Ceremonies are how Inspect runs ACROSS concerns: one sitting reads each concern's exception segment.

The diagnostic is itself useful: a process nobody can assign to one concern without inventing a fake one is a ceremony, and that is information about the ontology, not a defect in it.

(SETTLED: first receipt 2026-08-26, the standup-prebrief diagnosis; the second, independent, on the fleet 2026-09-10: relay-concerns declares its coverage job walked by relay-eng-daily's walker on the Coverage tab, a ceremony reading one concern's segment. Ratified 2026-08-26 from the estate's concern-model sitting, where a ceremony's "missing owner" turned out to be the model missing this category; promoted 2026-09-11, v3.9.0.)

The spine anatomy

A concern's spine composes, whatever its local format: the SCOPE sentence (what this concern is and when to open it), the BIRTH STORY (why the concern exists and what forced it into being — the why that pinned decisions cannot carry), the REIFIED JOBS (the admission-standard table), the ONTOLOGY declarations (nouns, legal states, identity schemes, and their typed LINKS — the noun GRAPH, never a noun list, because a concern is a pipeline and pipeline-ness is edges: each link a DOMAIN VERB in the hirer's vocabulary (a Payslip PAYS a Report; the ORM's belongsTo names an edge's shape, the domain verb names its meaning), cross-concern links naming the foreign home per the jurisdiction law, whose typed-reference mandate this completes on the intra-concern side; import ground: fifty years of relational modeling, ER diagrams to RDF triples — the links clause ratified 2026-09-04 on first principles plus import ground, first instance the estate's sidecar dossiers, the link ENCODING a deployment binding per types-and-bindings; the fleet's OKG mirror confirmed it 2026-09-06 (T20: nouns and links taken UNAMENDED into a second consumer, the encoding now the candidate key set for a spine-schema instrument)), the SOP SUITE (each verb named, clocked, owned), the ORGAN INDEX (typed pointers to the registers, tools, and workflows that implement — mechanism never restated), the PINNED DECISIONS (rendered by reference, per the jurisdiction law), and a VERSION.

Machine form (v3.7.0): canon/SPINE-SCHEMA.yaml is the config-as-data source of the spine's eight components and, per component, the fields whose MEANING the canon owns: a job's number, question, hirer, cadence, decision changed and failure cost; a noun's name, identity and states, each state carrying two author-declared bits, terminal and resting (the register's walk set, Part 2.75); a link's from, verb and to, with the foreign home where an edge crosses concerns; an organ's name and type; a verb's name, kind (SOP, ceremony, induction: the two-kinds law and the induction law encoded), clock and owner; a decision's date, ruling and status; the scope sentence, the birth story, and the minted version. Everything else a deployment writes on a row (a job's admission date, a noun's decision fields, an organ's address and binding, a verb's door or receipt name, a concern's code or visibility) is a BINDING recorded in the deployment's own contract, never here, by the types-and-bindings law. The schema was EXTRACTED from the two living deployments' declarations, like the organ types; the release gate checks its component count against this anatomy. (Ratified 2026-09-09 on the T20 receipt, nouns and links taken unamended into a second consumer, and on the two contracts read side by side, the estate's declaration contract and the fleet's validator, which agree on every field above and differ only on bindings. A jobs stamp, settled or provisional with a receipt due, was refused at the door: no deployment carries one. The verb's kind entered on the fleet's instance, nineteen SOPs, one ceremony, three inductions; the estate's binding is armed as its confirmation.)

Corollary, the orphan rule: an organ without a spine home is an orphan. At build time the question "which concern does this organ serve?" is MANDATORY, with two legal answers since v4.1.0 — an existing concern covers it (build under it, back-link), or the domain is real but undeclared (declare it through a door as the FRONT HALF of the build, jobs-first). The former third answer, WORKSHOP (a build explicitly experimental and uncataloged, earning structure by receipt), is a loose declared project on the work-management concern since home by construction (Part 2): still experimental, no longer homeless. Only the silent path is forbidden: a standing, cataloged organ with no answer at all.

Enforcement climbs the ladder on evidence — ask at birth, notice after, gate only when coverage makes gating honest rather than performative, because a day-one no-concern-no-build gate inverts

the method into permission-first bureaucracy and mints junk concerns written backwards to bless wanted builds.

(SETTLED at birth: two implementations converged on it independently — the estate's thirteen dossiers and the fleet's eight declared concerns, across four different authors. Ratified 2026-08-28. Provenance, per the double loop: this REVERSES the 2026-08-26 ruling that the composed pattern should live as examples only; that ruling's blocker was the then-ungated authority dimension, which has since earned its receipts and its law. The birth story joined 2026-08-30 on the same second-deployment evidence that promoted types-and-bindings: every estate dossier opened with one and every fleet declaration carried a birth_story field, independently, from their first spines — the component was running everywhere and declared nowhere.)

The induction law

Between authoring a concern and running it lies the INDUCTION: the one-time act carrying the concern's unstructured past into the standing process's steady state. The two process kinds do not name it. It has an owner and a declared end state ("caught up", defined in advance) but NO clock: it runs once.

Three legal treatments of the past, mixable per concern: BACKFILL (enrich from retained ground truth — statements, archives — so the past becomes rows under the new ontology), FREEZE (a dated legacy line: the past stays readable but unconverted, and the new capability stays quiet behind the line, which is correct, not a gap), and BANKRUPTCY (the past declared handled as a class without item-level processing — legal only when reversible by construction: archived, never deleted, the corpus retained). The governing constraint on all three: records never mutate, derived views may recompute everywhere, and the past is enriched only from real ground truth, never reconstructed.

A second induction on the same concern is not an induction; it is evidence the standing process's enforcement failed and backlog re-accumulated: diagnose the fast loop, then sweep.

(PROVISIONAL: three receipts on record — an email bankruptcy sweep, a brokerage backfill, a day-ledger's frozen legacy days; promotion when the next induction fits without amending the taxonomy. Ratified 2026-08-26; the vocabulary is the author's own "onboarding to the process", born from the EMLSOP-1 question.)

Inspect means exceptions

Inspect means exceptions only (the Deming note). Deming's point 3 condemned depending on END-OF-LINE mass inspection, and his remedy (improve the process itself) is this model's Garden; Inspect survives his critique precisely because enforcement (machine-run invariants, authored in steps 1-2) builds the quality in upstream, so what reaches the human is only the exception set: overdue, blocked, stale ball, no next action, failed assertion.

Walked on cadence, as a tick-down, every item ending in exactly one recorded disposition (the five dispositions): do, delegate, defer, drop, or decide. A manager inspecting everything is the enforcement layer failing (the two-hour, 150-item standup is this failure mode with a calendar slot). And a recurring exception is information about the system, not about the work: route it to Garden instead of re-handling it heroically each week.

(SETTLED: the 150-item-standup failure mode named and avoided; the email court's disposition set completed by defer.)

The walk ladder

A concern normally has several walks, nested by altitude, and an exception climbs exactly one level at a time. A WALK is the Inspect verb over one exception set, on one clock, by one walker; the register's walk set (Part 2.75) says what is open, the ladder says who reads it and when. Three altitudes recur. The FLOOR WALK is the walker closest to the work on the fastest clock: it reads the

raw open rows and discharges what it can (do, delegate). The REGISTER WALK, one level up on a slower clock, reads only what the floor could not discharge, and the ladder defines that mechanically: a row that AGED past the floor walk's clock, or RECURRED. The TOP WALK is a line in a ceremony (the two-kinds law), derived from the registers beneath it and never hand-kept; its exception set is what escalated plus the LIVENESS of the walks below, because a register that has not moved for its walk's clock is itself an exception (the dead walk). So the executive walks the walks, never the work. Compliance is silent at every altitude, and each level sees only the narrower set the level below discharged.

Two rules size the ladder. An exception climbs one level when it ages past the lower walk's clock or recurs, and RECURRENCE routes to Garden (the process), never back to the work and never further up. The walker at each level is sized to the exception rate below it (the capacity loop); a walk that is always empty merges away, and a walk that is always identical is an automation candidate: a sensor turns a floor walk into a Notice, and the walk above it shrinks. Altitude is the clock and the exception set, never seniority: one person may hold two rungs, and a machine may hold the lowest. THE DEFAULT WALKER (v4.1.0): at every altitude below the top the walker is DECLARED ONCE by a person and the walk is agent-run thereafter with no per-walk direction (agent-run as the default is a policy an edition may set; the mechanism is the declaration); the human walks only the top walk, whose docket is derived and whose exceptions are what escalated plus the liveness of the walks beneath; an agent's walk logs its dispositions, so the human can audit the walker, the absence test made mechanical.

The worked case, a mall's toilets. Jobs: are they clean and stocked now (the supervisor, hourly); which fail repeatedly (the facilities manager, weekly); is the contractor meeting the SLA (the contract owner, monthly). The checklist sheet on each door is the inspection register, the complaint line a court, the contractor's monthly report a ledger. The floor walk: the supervisor, hourly, reads failed checks and sends the cleaner. The register walk: the manager, daily, reads only what aged or repeated (a stall failing three days running, a complaint past thirty minutes, a contractor no-show) and fixes the fixture or routes to the contractor. The top walk: one line in the mall executive's monthly review beside security and parking, derived from the sheets: SLA breaches, cost, the accepted gap of a stall closed for renovation, and whether the supervisor's walk is alive. A sheet unsigned for three hours is the exception the executive sees, never the dirty stall behind it. A dispenser sensor later turns the hourly floor walk into an exception-only notice, and the supervisor's walk shrinks.

(PROVISIONAL: ratified 2026-09-11 on the two deployments' nested shapes: the estate's email pass daily, Sunday walk weekly, and the standup's exceptions line above both, with deadline-grade material alone in the nag lane; the fleet's walked-by ceremony per concern, the Coverage tab's dead walks, and the charter's escalation clause. The estate is the first instance, the fleet armed as its confirmation. Due: one exception observed climbing exactly one level by the aging rule on a deployment, and a top walk reading a dead walk beneath it. The default-walker clause joined v4.1.0 (2026-09-14) on the fleet's receipt: DEC-162 (2026-09-12), the walker declared on production the same evening (FIO-130), the first agent-run walk on 2026-09-13 (walk 445: 33 pins, 33 dispositions, one escalation to the operator), the Charter's walker exception (core PR #1038); the estate's email pass (a headless floor walk with a ledger receipt) and pre-brief (the register walk's docket) the second deployment's instance.)

The five presences

A corridor screen, never a gate. The catechism (Part 3) authors a structure; this is the quick screen for a live one, run in a hallway minute. The GATE instruments remain Part 6's (the scorecard at design time, the practicum's receipts checklist at day 30/60; where they overlap, the gate wins).

For any register, five questions: (1) Ontologized? Show the schema, and does any field still change a decision? (2) Processized? Show the SOPs, including the meta-ceremonies. (3) Logging? Show rows landing. (4) Inspected? Name the ceremony that walks its exceptions, and its cadence. (5) Gardened?

Name the last amendment and its evidence, or the honest null (nothing creaked, and the monitoring that would catch it is armed).

Five yeses is a screen passed, not a certificate: a backfilled register can fake presences, and only the practicum's decision-with-artifact test proves life.

Part 2.75: The organ types (one language across deployments)

A concern's spine points at its organs (the organ index, Part 2.5), and until now every deployment named those organs in its own dialect: the estate says dossier-table, the fleet says tracker, and a conversation between the two needs translation. This Part fixes the vocabulary: a closed set of organ TYPES, each named by the one JOB it does for a concern. The set was EXTRACTED, never designed (the construction-grounds law): a coverage walk over the two living deployments' actual inventories — the personal estate's substrate and know-how dials, its registers, mini-apps, and ledgers; the fleet's concern declarations, tracker engine, boards, and receipts lanes — kept the common types and marked the divergences.

What an organ IS (the definition, ratified 2026-08-29): an organ is a DECLARED PLACE. A standing artifact that one concern declares for exactly one job, that the concern's processes operate on their clocks, and that the spine's organ index names. The concept was latent in layer 1 all along: "the declared places where it lives" is the organ slot, and the type table below enumerates the jobs a declared place can do — even the type names are spatial (the door, the court, the board). ENTITIES are what live IN the places; a concern manages its entities THROUGH its organs. Four membership tests: (1) a made, standing thing, never a verb; (2) exactly one job — the mixed-type feeling is the substrate tell; (3) operated — some process writes it or walks it on a clock; (4) declared — the spine points at it (the orphan rule as a membership test).

Three boundaries before the table. The SPINE is not an organ: it is the declaration itself, the organ-HOLDER (the spine anatomy, Part 2.5). A PROCESS is not an organ: ceremonies and SOPs are layer-2 verbs that operate places (a ceremony's derived docket binds as an index, its walk record as a ledger), and a notification route ("on this exception, ping the owner in space X") is a delivery clause of a process — trigger on an organ's state, ball to an owner, addressed to a channel — never an organ. A CHANNEL is not an organ: a communication container is SUBSTRATE, type-promiscuous by nature, and the container never takes a type — though an organ may bind INTO one (the estate's court binds to an inbox; a standing notices thread is a ledger binding), and a space is also itself an ENTITY with a home concern (one noun, one home), referenced by other concerns as an address field, never listed among their organs. (The channel boundary ruled 2026-08-29 on the fleet's organs.spaces question, the types-and-bindings law's first boundary dispute; the mixed-type-means-substrate tell was the deciding argument.)

The register's walk set (v3.7.0): a register declares which of its states are TERMINAL (the row is closed) and which are RESTING (settled but not closed: a standing ruling, a recorded receipt); its OPEN rows, neither, are the exceptions its walk exists for. Two author-declared bits on a state, never a colour, never a per-organ column binding. A register that declares no terminal state testifies nothing about what is open, and its consumer shows that as a loud gap, never a silent zero. (Ruled on the fleet 2026-09-07 when the decisions register read 104 of 105 rows open until resting was one declared bit on a state; the fleet's pulse reads open as neither terminal nor resting and counts a register as walk-grade only when it declares closure. The estate diverged the same week, naming a status COLUMN per organ and letting chip colours split it; armed: the estate binding the two bits on its status options and retiring that column key, unamended.)

#	Type	The organ's one job	Estate binding today	Fleet binding today
1	The register	The concern's records: identified rows, mutable cells, exceptions walked	dossier-table, Notion database, Eloquent models	engine tracker, workshop-lane register
2	The ledger	Append-only receipts: proof that events ran, never mutated	the Alfred ledger, the email manifests	the access-change ledger, workflow run receipts
3	The court	Where balls sit, visible at a glance	inbox-as-court, with snooze and resurface	the Relay court
4	The board	A bounded body of work in flight: tasks, definition of done, closable	the nwos-todo item tree	the Relay board
5	The surface	The screen a human opens; view-only is a qualifier, never a noun	the mini-app	the tracker surface, the tier-3 dashboard
6	The door	The typed write path: Gate-mode enforcement embodied	MCP tools, the email actuators	the concern-apply door, engine row writes
7	The index	Derived views recomputed on events, never kept by hand	the dossier index	the concern index, ceremony dockets, coverage views
8	The corpus	Retained raw ground truth for recompute and backfill	the email corpus, statement PDFs	transcripts, harvest corpora
9	The scoreboard	The goal layer's readings with standings, derived from registers, never testimony	hyrox's dashboard strip (the bedtime ratchet, Race Readiness)	the WIGs strip

The binding columns are ILLUSTRATION, not law: they record where each type ran on the day of extraction, and they will drift as deployments evolve, which is legal — that is the point of the law below. One organ shape stays deliberately deployment-local: the FORM (the estate's printed and iPad fill-in sheets; agents print nothing), a legal local noun with a standing flip condition: it enters this table the day a second deployment earns it by receipt. The SCOREBOARD took exactly that path: local on the fleet's WIGs concern, armed on the estate's hyrox goal layer (declared 2026-09-02), it entered the table as the ninth type at v4.0.0 when hyrox declared its dashboard strip (health-and-training-management D14, 2026-09-12), the second deployment's receipt.

Types and bindings

The canon owns organ TYPES and their NAMES; a deployment owns only BINDINGS: the mechanism that implements a type, named as a qualifier on the canon's word (the naming half amended at v4.0.0 by one name per concept, Part 4). A type names the organ's job, exactly one per organ, and lives here (machine form: canon/ORGAN-TYPES.yaml, the config-as-data source deployments bind against — deliberately carrying types only, since a binding listed in the canon would violate this very law). A binding names the type's substrate in ONE deployment — the register binds to a dossier-table on the estate and to an engine tracker on the fleet — and lives in that deployment's own declarations, amended freely on its own clock. The binding is spoken as a qualifier, never as the name: a dossier-table and a tracker are two mechanisms of one thing, and that thing is called a register in every declaration, every kernel and every agent's speech (the speech test, Part 4); the purpose sentence stands, a conversation about any estate Norman runs needs no translation, and at v4.0.0 the canon stopped tolerating the translation layer it had allowed at the binding.

New vocabulary enters a deployment as a binding of an existing type; a shape no type fits stays a deployment-local noun with a pre-registered flip condition, and is canonized as a new type only when

a second deployment earns it by receipt (the form is the standing example; the scoreboard entered at v4.0.0 on its second deployment's receipt). This is the spec-and-implementation law applied to vocabulary itself: the canon is the spec of the language, deployments are its implementations, and a conversation about any estate Norman runs needs no translation.

Type by role (v3.7.0): an organ's type is its ROLE, the job it does for the concern, never the mechanism that implements it. An engine table used as an append-only receipt lane is a LEDGER whatever engine holds it; DERIVED and FOREIGN are qualifiers on a typed organ, never types. A deployment's organ index is a FREE TYPED ENTRY (name, type, binding, address), never a closed list of engine kinds: a closed list mis-types the organs the engine did not foresee and renders the concerns the engine does not host organ-less. (The T20 mirror's finding, 2026-09-06: the fleet's closed engine kinds mis-typed role-ledgers as registers and left first-party product concerns organ-less, while the estate's free typed entry took the fleet's whole inventory unamended; armed: the fleet amending its validator's closed organ mapping toward the free typed entry.)

Liveness is derived; the index declares addresses (v3.7.0): an organ index declares WHERE an organ is and WHAT it is, never how it is doing. Its entry carries the organ's name, type, binding and ADDRESS (a table's slug, a URL, a workflow's name); its LIVENESS (the last receipt, the open rows, a clock gone stale) is DERIVED by the consumer from the organ itself at that address, recomputed on read: derivation before delegation applied to the pulse. The one exception is the estate boundary: an organ the consumer cannot read (another estate's register, a foreign system) carries its liveness in declared form, and that declaration IS the receipt door for foreign organs. A verb's clock is authored once, in the spine or in the scheduler the spine names by the workflow's name, never in both; staleness is derived from that one source. (The T25 verdict, 2026-09-07: the fleet's pulse derives every organ's liveness with zero binding, and only the doors on other agents' estates stay unlit until a receipt door exists, which is exactly where the declared shape belongs; the estate had declared liveness on thirty-one organs and a cadence on seventeen verbs two days earlier, and this clause re-reads its declaration as the ADDRESS binding it already is and retires the duplicated cadence wherever a workflow is named. Both landed: the estate derived liveness by address for its engine-native organs on 2026-09-09 (NW-ANA-49), and the BOUNDARY declaration landed on both deployments in two spellings: the estate's on 2026-09-10, a feed inside the address ({source: nwos-finance, feed: due-bills}), the map owning the feed's contract, the due-bills docket lit on the Park, NW-BUG-59), and the fleet's on 2026-09-11, a receipt: {connection, feed} key beside the address (the connection module owning the feed's typed contract, the map calling the feed as the concern's owner agent, cached one window, a feed declaring no clock; DEC-158, console v1.39.38, the fleet roster the first foreign organ lit through a declared door). The canon owns the meaning; the spelling is each deployment's binding, per types and bindings.)

(SETTLED: extracted 2026-08-29 from the two living deployments' inventories and ratified the same day on Norman's one-language directive, NW-ANA-22; promoted the same night on its pre-registered receipt — the fleet's binding landed on the type table UNAMENDED and live at the write path, hours after the estate's own declaration shipped: the bindings served as data on every dossier payload, the channel boundary enforced by a teaching refusal on organs.spaces with the home_space address field replacing it, zero live declarations needing migration. The eight types absorbed both full inventories as extracted.)

Knowledge routes by role

Knowledge is a role, never a type, a spine component, or a noun; it routes by one question, what decision does it change and does it have an identity, into the types the canon already has. The why lives in the spine's own prose (the birth story, the decision rows). A rule with an identity that a verb applies is a register, the concern's policy library. A source cited or recomputed from is a corpus. A dated lesson or attempt is a ledger. How a verb is done lives in the door the SOP row names, never as prose in the spine. A quirk no concern owns lives in the agent's own memory, the

agent's register, never a concern's. An idea with no hirer stays outside, a workshop project (home by construction, Part 2). Every artifact an agent or a human works from is an organ of exactly one concern, typed by role and named in that spine, or a workshop project: a knowledge directory is a drawer, never a home. Promotion runs one way, from ledger to register or door, so a recorded attempt never becomes a rule by sitting there.

(PROVISIONAL: ratified 2026-09-12 on two deployments' receipts in hand, the estate's know-how dial (charter, 2026-08-16) and its Codex runbook ruling (2026-09-12), the fleet Charter's reference-material taxonomy, its orphan red flag (a knowledge file no skill or CLAUDE.md references) and its Learning PR path; the fleet routes by reachability from a verb today, never by membership in a concern, and its coverage walk lists no drawer files yet; promotion when each deployment's coverage walk lists its knowledge drawers and every file lands in a spine or a declared workshop.)

Part 3: The catechism

Run this every time a pain recurs twice. Write the answers down; they ARE the design, and their order is the five-step order's (Part 4): the first two answers may end the catechism with nothing built, and that is a success. (This is the PAIN door into Ontologize, for domains already running and hurting; a greenfield domain enters through the JTBD door instead, the job enumeration in Part 2, skill 2.)

1. What is the recurring pain, and WHO has it, by name? (step one: a requirement carries the name of the person who answers for it today, never a department; question it whoever gave it)
2. Would anything be missed if nothing were built, and by whom? (step two: the deletion question, asked before any noun; the honest answers include "nothing", "a rest", or "delete the thing that causes the pain")
3. What noun or place is missing? (layer 1: define its fields, its legal states, its ID scheme, where it lives)
4. What is the one verb, on whose clock? (layer 2: a cadence with an owner; "sometime" is not a clock. One DOMAIN verb: the meta-ceremonies ride your existing walks until the structure earns its own)
5. What does one row look like, who writes it and when, and what does empty look like? (layer 3: the row-writing moment is Log's home; empty must degrade gracefully; "nothing due, closed in 30 seconds" is a success)
6. What makes skipping it loud? (enforcement; the guard born here carries its incident, one meter, and its retiring reading)
7. When it creaks, who amends it, and how? (gardening)

Part 4: The fitness tests (the review rubric)

A proposed structure passes review only if:

- **Every field changes a decision.** If no decision differs based on a field's value, the field is decoration. Delete it.
- **Every instance has a declared identity.** A minted, permanent, speakable ID where instances are cited across time, people, or surfaces; a declared natural key where the date or slot individuates (the identity law, Part 1). A structure whose instances can only be referenced by description ("the campaign Budi ran in March") fails: claims cannot carry receipts and dispositions have nothing to attach to.
- **Compliance is silent, exceptions are loud.** The structure surfaces only what needs a human; green rows make no noise.
- **Pull surfaces die.** Anything that waits to be opened will not be opened. Every structure needs a push moment: a clock, a ceremony, an alert.
- **A good ceremony shrinks.** Anything appearing identically every cycle is an automation candidate. A structure whose docket trends empty for a quarter merges into something larger. A structure that

can never end is a treadmill.

- **The instruction test:** after the structure exists, can the manager direct a day's work in one line? If not, layer 1 is too coarse.
- **The absence test:** does the structure run correctly for two weeks with the manager on holiday? If not, layer 2 has no real clock or no real owner.

The cognitive-fit law

Structures are sized to the OPERATING HEAD, never to the concern's ambition. Working memory holds roughly four to seven chunks, and recall descends labeled hierarchies (organized lists recall at multiples of flat ones — Bower), so every surface a human must hold AT ONCE is capped at a human span: the instruction fits one line, the vocabulary fits one spine page, the unsure list fits five, the routing sentence fits forty words, a hierarchy node fits about seven branches, a concern's jobs fit about ten (the split rule is this law at the jobs tier), the concern count matches the walkers' drained capacity (the capacity loop, Part 2), and the whole discipline fits a ten-word mantra.

When reality outgrows a surface, SPLIT or DESCEND A LEVEL — never stretch the surface. This is the one law beneath half the tests above, and the canon's own register obeys it (seven models, seven instruments, seven families of at most five laws).

Prior art: Miller's and Cowan's spans, Bower's hierarchical recall, Krug — whose don't-make-me-think is this law's surface-side twin in the design canon — and org theory's century of independent convergence: span of control at 5-9, squads at 8-12, two-pizza teams. The convergence is the tell: every discipline requiring one human to hold a set of accountable things lands on the same band, because the constraint was never the domain. It was always the head.

(SETTLED at naming: each of its instances already carries receipts; named 2026-08-29, when the ~10-job split line was reverse-derived from working memory.)

The readability law

Every artifact a head must read is written in that head's working vocabulary; machine encodings live underneath, never on the surface. The cognitive-fit law's encoding sibling: cognitive-fit sizes what the head holds, this law chooses the language it holds it in — and since the reading heads now include agents, one vocabulary serves both. Where every head, human and agent, loads the same kernel, the canon's names ARE the working vocabulary (one name per concept, below); the law's second half stands unchanged: a job in the hirer's own words, an instance under its own name, mechanism visible where mechanism matters.

Standing instances, each ratified separately before the law was named: identities are HUMAN-SPEAKABLE (the identity law's quality bar: machines keep UUIDs underneath); ontology states are display words, never machine keys (the fleet's standing ruling, boilerplate in every concern declaration); a JOB is written as the hirer's question, in the hirer's words (the question dialect: "did my money actually come back?" reads in one pass, where a system-voice assertion must be reverse-engineered back into its question); a concern's NAME is its territory plus MANAGEMENT (the uniform-suffix rule): a concern IS a domain plus the commitment to manage it, so the suffix restates the definition instead of choosing among commitment words; pipeline-KIND words (intake, close, operations) live in the TERRITORY half, where their information keeps (bug-intake-MANAGEMENT); a name that RESISTS the suffix is a species diagnostic — evidence the thing is a ceremony, an objective, or an organ wearing the wrong noun, never a case for a different suffix; slugs and codes never move (display names only, the identity law), and a concerns-only index may render names bare, the suffix being uniform there. (Stamped 2026-09-03, superseding the same-day default-

commitment clause: that clause lived three hours before the fleet's audit produced an 11-to-5 "default" — the smell that exposed it — while the estate's 17-for-17, already uniform and including the kind-in-territory cases like email-intake-management, was the deciding receipt for the definitional form.) And code speaks layer 1 where a concern operates through it (the estate's ORM-over-raw-SQL doctrine: the query written in the domain's nouns and named scopes, the storage encoding underneath).

The hirer's-voice clause (stamped at v3.5.1 on the fleet sweep's receipt): the dialect follows the hirer, and THE HIRER IS WHOEVER CONSUMES THE ANSWER to change a decision — supply never wears the hirer cell, however process-shaped its name. Condition shape is legal only for a genuine machine consumer, a gate that reads the answer and acts per event; an apparently-machine hirer is usually the supply defect one cell over. (The receipt: the fleet's full-conformance sweep of its seven jobs tables, thirty rows in one sitting — twenty-nine converted to question dialect; exactly one genuine machine consumer kept condition shape, the board-close DoD gate; and four of the five apparently-machine hirers turned out to be the producing machinery wearing the hirer cell.)

The boundary: readability is NOT abstraction. Explicit-over-magic still binds, and an encoding that hides mechanism from the reader who needs it violates this law from the other side — obscured mechanism is in nobody's working vocabulary. The claim is precisely: human-native vocabulary at the surfaces humans read, mechanism visible where mechanism matters.

(SETTLED at naming, 2026-08-30, when a cold comprehension read across both deployments' 26 concern declarations judged the question-dialect jobs tables clear on first pass and the system-voice tables not — the same missing-law tell that named cognitive-fit: instances ratified for weeks, unnoticed as instances.)

One name per concept

Every concept the canon owns has exactly one name and one definition, and a deployment never mints a synonym for it. The canon names the grammar, never the sentences: what a register IS is the canon's word; what a concern calls its registers is the concern's. Three boundaries keep the law honest. Concepts, never instances: the rule names what a walk is, never your walks (NW-ANA, GBUG, the weekly walk, the email pass stay free, as the identity law's identities). Names, never mechanism: a mechanism word (a Notion table, a tracker, a dossier-table, a mini-app, a dashboard, an item) is a QUALIFIER on the canon's name, never the name ("the claims register, a Notion table"), and the substrate stays the deployment's binding; the boundary is the SPEECH TEST: where a word appears in something an agent or a human reads or says while working (a tool name, a declaration key, a sentence, a screen label) it is a name and must be the canon's; where only a developer reads it in code (a namespace, a table, an enum value, a URL segment, a kind field's value) it is mechanism and may stay. Grammar, never sentences: a concern's own nouns, jobs and verbs are its own. A product that renders a canon concept to a customer may carry a display label, declared in its edition as a binding of the canon name, one per product, never a second name in the declarations or the agents' speech (Relay's Apps, for registers and surfaces opened as apps). The machine form is canon/GLOSSARY.yaml: every concept, its one name, its definition, the Part that defines it, and its deprecated synonyms; the release gate refuses a deprecated synonym anywhere but the historical records, the kernels and the fleet skill carry the names, and the OKG's handles are spelled in them. A name is fixed on receipts, never by scenario: extracted from the deployments that already speak it, prescribed to the deployments born after. Renaming a fixed name is a major: a one-hop alias forever for identities (slugs, codes, URLs), a clean cut for vocabulary (an alias for a tool name is a second name in the agent's catalog, the very leak).

(PROVISIONAL: ratified 2026-09-12 on Norman's ruling that every estate he runs speaks one language, with The Tee Inkers moving into Relay and two or three MO-powered services businesses to be born on the canon's words; receipts in hand: the OKG's name fixed on two deployments at v3.8.0, both deployments and Norman's own speech saying concern,

dossier, walk and sitting, and Dario's inventory of the fleet's synonyms the same day; promotion when the first deployment born on the glossary declares its first concern with no synonym minted and both existing deployments land their cuts.)

The qualifier corollary

A variant is a QUALIFIER on an existing noun, never a new noun, until receipts earn the split (the cognitive-fit law applied to vocabulary growth). A screen-shaped organ that owns no tables is a "view-only mini-app", not a "view": the vocabulary stays one noun wide, the qualifier carries the variance, and a first-class noun is minted only when accumulated qualified instances demonstrate jobs of their own (the split rule, applied to the dictionary instead of the concern). The organ type table (Part 2.75) grows the same way: bindings and qualifiers first, new types only by second-deployment receipt.

(SETTLED: first receipt the view-only mini-app ruling, 2026-08-29, recorded the same sitting in the estate charter, the doctrine repo, and the claims dossier; the type table's own admission rule is this corollary in standing practice.)

Deferrals carry flip conditions

Kill conditions govern live structures; their constructive mirror governs structures NOT built.

When review rejects or defers a proposal whose job is real but already served more cheaply, the rejection is pre-registered as a FLIP CONDITION: the observable event that reopens the build ("the day a real query fails the live serving path"), with the build then designed for the failing case specifically and sized by its evidence.

Structure is earned by receipt, never by scenario; a deferral without a flip condition is a door slammed, and the flip condition is what makes saying no cheap.

(PROVISIONAL: receipts, the J8 search-corpus deferral and the estate's borrowed-substrate rulings; promotion on the first deferral that returns through its condition.)

The five-step order

Every structure is born and changed in one order, the order of MUSK'S FIVE-STEP ENGINEERING PROCESS, written out here because the order is the law:

1. **Make the requirement less dumb.** Clear the job first. Every requirement is questioned whoever gave it, most of all when a smart person did, and it carries the NAME of the hirer who answers for it today, never a department's: the jobs-first admission standard (Part 2) read as a challenge, never as a form. The name is the address for the challenge and the owner of its consequences, never proof of sense (the source's own words: your requirements are definitely dumb, whoever gave them to you): a requirement is less dumb when it CHANGES A DECISION, COSTS SOMETHING UNSERVED, and MOVES AN OBJECTIVE through a lead measure, the standard's other parts run as tests, with an OKG lookup for what already serves it. A hirer who insists past a failed test owns the requirement on record, with the challenge beside it, and it is metered like a guard: a requirement that changed no decision in a cycle retires at the concern's next touch, whatever the hirer wanted. A challenge is never a veto; who rules on it is the edition's authority (the concern's owner on a multiplayer deployment, the survival classes their own gate, the operator alone on an estate), and the agent's part is to say it once, with the evidence, then respect the call.
2. **Delete the part or process.** The deletion question: would anything be missed if this did not exist, and by whom?
3. **Simplify or optimize,** only now. The commonest error of a capable author is optimizing a thing that should not exist; the depth rule and cognitive fit govern what survived step two.
4. **Accelerate cycle time,** only after the first three: a grave dug faster is still a grave.

5. **Automate, last.** The enforcement ladder (ask at birth, notice after, gate only when coverage makes gating honest) widened from gates to every automation: a cron, a report, a sensor, an agent-run walk. The push ladder's descent is step five too: a ceremony is put to step two before it descends a rung, and a docket empty for a quarter merges away, never automated. An automation of a thing nobody reads is step five run before step two.

Each step waits on the ones before it, and the named failures are the steps run backwards (the source's own confession, on one production line: automated, accelerated, simplified, and only then deleted).

The deletion question has two clocks: birth, and every change of the mechanism around the structure. A guard right in its birth scope becomes a tax when a new mechanism reads it, and nobody re-derives it because nothing asks. The bias runs one way and it is structural, never a character flaw: an addition arrives with its incident as a receipt while a deletion produces none until the failure returns; the cost of keeping is diffuse (a little on every run, for every lane) and the cost of losing is one vivid incident; the in-case argument can be made for anything. Adding feels like hardening. So the question is put on clocks and answered from readings, never from a mood and never from a principle.

A guard carries its incident, its meter and its retiring reading. A guard is one piece of the fast loop attached to a structure (a check, a gate, a rule, a test: a guard refuses). Born of an incident, it carries on its own line, where it lives, what a deferral carries: the incident that bore it and the condition that retires it; and it is METERED on one number read on a clock (minutes per run, touches per ship, pages that were wrong, reports read), so its deletion follows a reading.

A guard retires only against a named END-OF-LINE READING that covers what it guarded: a job's reading (Jobs carry readings, Part 2), taken where the work leaves the line, on its clock, at its target. Where no such reading exists it is built first, and the guard stays until it reads green; a guard whose failure the end of the line cannot see stays in the line. This is the source's own rule (in-process tests are the scaffolding of a line being debugged; once units pass at the end of the line they come out, because left in they choke the cycle time and reject good parts) and it is the other half of the Deming note (Part 2.5), never its contradiction: inspection builds no quality in, the process does; a guard is scaffolding around a process not yet proven capable, and the end-of-line reading is how capability is known. For money the end-of-line reading is reconciliation (the ledger against the custodian's positions and cash, on its clock): where it is strong and passing, in-process checks are deletion candidates; where it is weak or missing, they are all the protection there is.

The add-back rate calibrates a pass: the share of what a deletion pass removed that had to return within a cycle. The source's number is ten percent; a pass after which nothing returns was too timid. The rate is an admission that deletion is sometimes wrong, affordable only where the end of the line catches the miss and the miss can be taken back. Where it cannot (a life, customer funds, records that cannot be restored, a licence, a regulator's or a customer's trust; each deployment names its own classes), the control stays whatever its meter reads: the order still questions, simplifies and speeds the control's design, and step two never deletes the control itself.

Deletion is retirement, never erasure: a recorded reason, a reversible path, zero readers proven (deletability is the provability of zero readers, Part 1), so a control never leaves an audit silently and an add-back is one revert. The law adds no verb and no ceremony: it orders the five verbs whenever a structure is authored or gardened, and it binds its own application first, since the deletion question is put to every proposed structure, the next amendment to this canon included. Import ground: Part 5.5, the order of change; imported whole and BOUND, never re-legislated: steps one, three, four and five land on laws the canon already holds, and what the canon adds is the two clocks on step two, the guard's three fields, the end-of-line precondition and the classes that keep their controls.

(PROVISIONAL: ratified 2026-09-21 on the operator's reading across two days. After the fleet's rig post-mortem: "I am so used to adding and not deleting... Feels like hardening when you add". The next morning, the order in his own words: "First make sure the JTBD is clear. Then we see if this process is even required or not. then we optimize". And the remark that bore the end-of-line clause: the source engineers rockets and cars, where a mistake kills people; "At least for us its money, which is also bad if mishandled". Receipts in hand on both deployments, every deletion following a METER and none a principle. The fleet: the rig, two specs red for five days because a guard born for the skip path (a red run deletes the green marker, 2026-08-02) also voided the scoped path that arrived four weeks later (2026-08-29): no deploy skipped or scoped, 45 consecutive full runs and none green, and one live product crash hidden behind a spec's first failing line; the harvest cron retired after seven reports and zero approvals, the five-a-day chases read at 15 and 43 percent by the readers' thread markers (0 to 9 by the badge column first quoted, corrected 2026-09-23), the plan card over reversible work after one counted thread (19 cards and 13 taps over 13 increments, about forty of 162 messages), the run-state primitive at zero adoption; and the rig's own repair, proposed automation-first and re-sequenced deletion-first. The estate: the VPN-abroad tunnel retired after zero uses in thirteen days, by ruling, its restore one revert and its posture covered by the roaming hotspot (estate-hosts-management, 2026-09-21); the standup's daily red on every unmoved item deleted on a reading, 161 on the docket and none ever reviewed, 23 after, the rest named once on the Sunday line (personal-work-management, 2026-09-18); the three descents under the push ladder's repeat clause the same day. The end-of-line clause's own receipt is the source's: the 2021 interview at 24:49, too much in-process testing. Due: one guard pass over one deployment's guards, each guard's incident, meter and retiring reading recorded on its line; one guard retired against a named end-of-line reading read green first; the add-back rate read one cycle later; and, on the other deployment, one automation refused or re-sequenced at birth by the order.)

Part 5: Agents as enforcement

In code, enforcement is free: the compiler and the database sit in the write path and make illegal states unrepresentable. Organizations never had this; the only checkers were humans, too expensive to post at every write path. **Agents change the economics.** Four strengths, in order (vocabulary unified 2026-08-22 across the canon, practicum, and product thesis; the fleet charter independently named "the noticing layer" as its own organ, and same-day alerting is not linting):

The four enforcement roles, in rising strength:

- **Observe (CI):** an agent attends the ceremony or reads the channel and lints outputs against the ontology after the fact: did the standup cover exceptions? did the decision leave with an owner and a date? Violations still happen but become loud, eventually.
- **Notice (the same-day alarm):** an agent watches the clock and the write path live: a ceremony that did not fire, a row gone illegal, a due date crossed — surfaced the same day, to the owner, once, loudly. Detection at the speed that prevents the 29-day silent death.
- **Gate (compiler):** the agent IS the write path. Work enters through a typed door that stamps the required fields and cannot produce an illegal row. Nothing can vanish silently because the only way in is through the door.
- **Scribe:** the agent produces the artifact natively in the right shape (pre-briefs the standup, files the register row from the conversation). Compliance becomes moot. (Renamed from "Author" 2026-08-26: that word belongs to the authoring acts and the certified Author. Under the five-verb taxonomy this mode is the Log function performed by an agent; it stays in this list because making compliance moot is the strongest enforcement there is.)

The accountability primitive: the ball and the court (nouns adopted 2026-08-22 from Relay, where they run in production with a ledger). "Decisions leave with owners and dates" needs a noun to be enforceable: the **ball** is who owes WHAT, since WHEN, held by WHOM, verified against ground truth before anyone chases; the **court** is where every ball currently sits, visible at a glance. A decision without a ball is a wish with a witness; the noticing layer's core job is balls going stale.

Three cautions, all load-bearing:

1. **Enforce artifacts and write paths, never conversations.** Police how people talk and they will route around you into shadow channels. Enforcement holds when the enforced path is also the easiest

path: the typed door must be more convenient than the workaround.

2. **Enforcement strength must be matched by gardening capacity, and calcification risk must be located precisely.** Calcification does not live in the mere existence of automation; it lives in three places: IRREVERSIBLE ACTIONS (a machine verb that touched the world cannot be re-run differently), NON-RECOMPUTABLE DERIVATIONS (stamps whose inputs were not retained), and CONSUMER COUPLING (rules, habits, and downstream structures written against the current vocabulary). Where the corpus is retained and derivations are recomputable, iterating the ontology stays cheap after mechanization: version it and re-derive wholesale (derived views may improve everywhere; records never mutate).

The corollary: **mechanize observation before action.** Stamp and classify at full volume immediately, because it accelerates the slow loop (the whole distribution becomes visible instead of a hand-walked sample), but each acting verb graduates on its own ladder: suggest, then act-reversibly, then act-irreversibly, each step earned on measured precision. One irreversible mistake during ontology churn burns the operator's trust, and distrust kills the system faster than calcification ever would.

One scoping caveat that binds hardest in this course: the observation corollary was derived from a corpus of MESSAGES; when the observed are PEOPLE (this course's runtime), observation graduates on its own ladder too — artifacts before behavior, aggregate before individual, disclosed before silent — or the full-volume stamping IS the surveillance that caution 1, the performance firewall, and the power question all warn about. (Refined 2026-08-26, the email-intake case; human-runtime caveat per review v4, finding 3.)

3. **Derivation before delegation.** Never seat a model-driven keeper over what pure derivation computes: an index, graph, or scoreboard derivable from the declarations should be recomputed on write events — never stale, nothing to babysit, nothing hallucinated — where an agent doing the same job can lag, die, or invent. Agent personhood is earned only by the JUDGMENT layer (semantic collisions exact-match cannot see, drift noticing, gardening proposals), and a judgment agent PROPOSES to the owning authors, never mutates a declaration: an agent that rewrites ontologies is an unlicensed author by the authority law. (Ratified 2026-08-28, from the fleet dissolving its proposed ontology-keeper service account into console-side derivation plus a proposals-only noticer.)

The provenance law

A claim received from elsewhere is held through its PROVENANCE, its evidence and its source, and the link runs both ways: whoever verifies it reads the evidence before the conclusion, and a correction at the source reaches everyone who cites it. A conclusion handed over (in a report, a progress note, a handoff, a decision's receipts) is born CLAIMED, the readings law's word, until someone re-derives it from the evidence; the ball's "verified against ground truth before anyone chases" is this law applied to a chase.

FORWARD, EVIDENCE BEFORE THE CONCLUSION: whoever's job is to confirm or refute another's work (a reviewer, an auditor, a close, a walk disposing a ball as done, a conformance check) reads the observations and their ground truth first, forms a read, and only then compares it with the conclusion under review. The conclusion is kept, never stripped: an upstream that is usually right is worth reading, so the rule is the ORDER of the read. A handoff therefore carries its observations apart from its conclusion, and a close answers its definition of done clause by clause from ground truth, never from the row's own notes. Orientation is exempt: reading a handoff to learn where to look is not verification.

BACKWARD, A CORRECTION REACHES ITS CITERS: a correction is appended (a record of its own, or a dated paragraph added to the record whose prior text its append-only history keeps), never a silent overwrite; it names the claim it supersedes, and whoever makes it looks up that claim's CITERS through their references (inside a deployment, the OKG's who-reads-this, as far as it reads the pins and the rows that cite; across the boundary, the other side's published text or a note into its lane) and tells each one. Each citer asks whether the ONE claim it relied on still holds, never whether the source changed; only the claim's actual dependents are woken. A citation that copies a number without its reference cannot be found, which makes the jurisdiction law's pin by reference the precondition of this half.

(PROVISIONAL: ratified 2026-09-27 from two outside findings (NW-MOF-1, Audit Without Verification; NW-MOF-4, Impact Is Not Invalidation; the estate's register of outside findings) on three failures across both deployments the same week: the fleet's badge-column read rates carried as read state into DEC-265's receipts, then the canon's two citations, then an agent's answers to the operator, the chain broken only by a return to the readers' thread markers (2026-09-23); the canon's citations stale three days after DEC-265 was corrected, found by hand (v4.5.1); an agenda row closed on its own progress note with two parts unfinished (NW-ANA-61). The estate's done check is the first standing instance (its first use caught a third miss). Import grounds: auditing (sufficient appropriate evidence; a belief that management is honest never relieves the auditor of skepticism) and truth maintenance (Doyle, 1979: every belief keeps its justifications, and a retracted premise re-evaluates only what depends on it). Read by the fleet 2026-09-28, three leans amended on its receipts: DEC-265 was corrected by a dated paragraph inside a patched row, its born text kept in the event trail; the fleet's lookup finds a code's mentions in declarations but not its pins or the rows citing it, and the estate's finds only organ notes. Promotion when a second deployment runs evidence first at a standing review surface and records a catch (the fleet's candidate: its PR review gate, handed the author's description after the diff and labelled as the author's claim), and a correction reaches a citer through its reference, on either deployment, without a person stumbling on it (the work: each deployment's OKG lookup listing a code's pins and the rows that cite it; the canon needs nothing, its published text is searchable).)

The OKG (the Operating Knowledge Graph)

Every deployment that reifies its spines has exactly one OKG: the derived view of everything it manages, condensed out of the declared spines on every read, never authored, never stored as truth. Its nodes are the concerns and the parts of their spines (a job, a noun with its states and home, an organ with its address, a verb with its clock, a decision, a board in flight); its edges are declared facts (typed links, organ pointers, what a project serves, who walks what, which ruling governs), never inferred. It is the meta-concern's index organ (type 7, Part 2.75), and its human window is the OKG MAP, a surface; the Park is a lens on the Map. It answers four questions, each with a hirer: what does this organization manage, who owns it, where does it live, and how is it doing (any agent or session, per question, BEFORE grepping or asking a human); does this noun already have a home (any author, before minting); what is managed and what is consciously unmanaged (the operator, on the coverage walk); who reads this, and what is colliding, orphaned, dangling, or drifting (authors and the gardener). It has a LOOKUP DOOR: a word in, every citable handle out, with its pulse. Every citable handle means the door reads THROUGH the spines to what they govern: the nodes are the spines' parts, the handles are also every coded instance their organs hold, and a door that stops at the declarations turns a question about the work into a false negative. A register's FIELDS belong to its organ: a hit on a register carries its columns (each column's name and type, a status column's states with their terminal and resting bits), and a field's name finds the register. A coded instance (a row, a project, a task) is answered by its code and found by its words, as a handle with its state (its standing on the register's walk set, where it lives), never as the store: reading a register whole stays with the register's own read door. How much of an instance a reader sees is the edition's AUTHORITY, never the door's: on the estate edition the one reader sees the row whole; on the relay edition a row is cut to the columns the viewer may read, a private register stays private, an owner's withheld rows are said, and a call sees what everyone in the room may see. A code whose prefix the graph does not carry (minted in another app or another deployment) reads UNKNOWN, never a negative about the thing: the door's honesty reaches only as far as its reach. An agent's field of view

is its tool catalog, so the door is what makes a negative about the estate honest: "it isn't registered" is a lookup, never a guess. Judgment stays human: a semantic collision is proposed to the owning authors, never resolved by the graph (caution 3). The substrate is a binding (tables today; a cache keyed by the declarations' fingerprints when read cost demands it; a graph store only ever as a derived index, never the truth). The NAME is the one binding the canon fixes: the word exists so that every estate the operator runs, and every agent in them, says the same thing; "operating" is the operating of the organization, never the data-lineage sense the market attached to "operational knowledge graph".

(SETTLED at birth, 2026-09-10, on two deployments that built it before it was named here: the fleet's OKG, engine-derived by DEC-38 since 2026-08-29, its lookup door and the standing instruction to look up before citing since 2026-09-06 (T20), its pulse and Park lens since 2026-09-07 (T25); the estate's map since 2026-09-05 (NW-ANA-32), liveness derived by address since 2026-09-09 (NW-ANA-49). The field-of-view receipt, a foreign read: GBUG-1167 root 2, a read door with no tool invisible to an agent at xhigh, "I don't have access" three times over nine days, fixed by six typed tools. The naming receipt: a session on 2026-09-09 could not resolve the word from any file it loads. Both armed confirmations landed the same day, 2026-09-10: the estate's lookup door on prod (nwos-todo v0.1.119 at 04:15 SGT, lookup_okg and get_okg, the meta-concern estate-concerns-management minted v1 with the OKG, its Map and the door homed), and the fleet's lookup instruction widened in the shared MO skill from "before you cite" to "before you answer what the company manages, who owns it, where it lives, how it is doing" (gtos-agent-core ff7b80d; core v1.8.20 rolled and read on a box at 16:05 SGT). v3.8.1. The reach clause joined at v5.1.1 (2026-09-29), on the operator's question whether an easy lookup of concerns, registers, jobs, fields and rows was not the OKG's whole point. Both deployments had read the node sentence as the door's limit and stopped short in different places: the estate at the declarations (a session hunting a host's reach that its register row held all along, 2026-09-23), the fleet at a code's row state (gtos-comms-console 184f5c43, 2026-09-19, and f1b829fd, 2026-09-28: a code answers its row's state, its columns and its read tools, while words reach no row; the organ matcher reads name, ref, code and note, ConcernGraphBuilder line 508 at ce81e0e6). The estate met the clause the same day (nwos-todo acba2c2 and 20107af: fields, a row by its code or its words, UNKNOWN for a foreign prefix; the first reads answered ANA-90 whole and ranked the host row first for its reach). The authority half is the fleet's receipt (column classes, private registers, the withheld-rows note and the room's cut; 406bc3b2 and edc9456d, 2026-09-29). The fleet's due: a row found by its words, cut to the viewer's authority.)

Honest scope: agents enforce **shape** essentially perfectly (fields present, receipt attached, disposition recorded). **Substance** (was the decision good) stays human review. Do not oversell.

Part 5.5: Prior art (and what is actually new)

This model has a 40-year pedigree, which is a feature: it is engineering with a literature, not a founder's invention.

- **The layers:** Zachman Framework (1987, explicitly an "enterprise ontology"), TOGAF, and the BPM guild (BPMN, the BPM CBOK). Whole industries have shared, consortium-authored ontologies: eTOM for telcos (TM Forum), SCOR for supply chains, FIBO for finance, HL7 for healthcare. A mature industry is one where layers 1-2 are commoditized and importable by hiring; a company in a new category has nothing to import and must author.
- **The operating loop:** ISO 9001 (process + external audit = enforcement), Toyota (kaizen = institutionalized gardening), Grove's *High Output Management*, Bossidy & Charan's *Execution*. The closest single ancestor of this document's full stack is Amazon's **mechanisms** culture: "a tool plus adoption plus inspection plus iteration" maps one-to-one onto ontology+SOP, enforcement, and gardening.
- **The intellectual umbrella:** Herbert Simon's sciences of the artificial (design science), Stafford Beer's management cybernetics, Nonaka's externalization of tacit knowledge — and Argyris's single/double-loop learning, which is exactly the enforcement/gardening pair: enforcement is the fast loop (per event: does reality conform to the structure?), gardening the slow loop (per epoch: does the structure conform to reality?). The loop model (Part 1.25) has siblings in control theory's inner/outer loops and IBM's MAPE-K autonomic manager (monitor-analyze-plan-execute over a

knowledge base), the closest software ancestor of agents running both loops over reified definitions (its Plan/Execute is adaptation, the slow loop's work, not only checking).

- **The design methods:** Grüninger & Fox and Uschold's competency questions (1990s ontology engineering: an ontology is adequate ONLY IF it answers the questions enumerated for it — a requirements test, not a sufficiency proof), which the JTBD door (Part 2, skill 2) rhymes with from the product side in Christensen and Ulwick's jobs-to-be-done (they theorize purchase motivation, not schema adequacy; the rhyme is real, identity is not claimed). Behind the identity law sits the spirit of Quine's "no entity without identity" (his demand is identity CRITERIA — when is x the same entity as y — which minted codes implement but do not exhaust) and a practice lineage as old as coordination itself: folio and voucher numbers in double-entry bookkeeping, part numbers in interchangeable-parts manufacturing, patient MRNs, ticket and RFC numbers.
- **The platform-scale ancestor:** Palantir Foundry's Ontology (typed object and link types plus a KINETIC layer of governed action types as the only write path, with AI agents grounded on the ontology so an untyped write is impossible by construction) independently validates the layer model and Gate-mode enforcement at enterprise scale. The honest split: Foundry's construction is archaeology of settled source systems, correct for its customer per the construction-grounds law (Part 1.75); its ontology MIRRORS operations that happen elsewhere, where this model's ontology RUNS them; and it carries no coordination primitives (no ball, no ceremony, no walked exception), which is precisely the layer this discipline adds.
- **The goal layer:** McChesney, Covey & Huling's *The 4 Disciplines of Execution* is the goal-layer clause's import ground (the grain law, Part 2): WIGs as closable battles, lead versus lag measures, the compelling scoreboard, and the cadence of accountability — imported and pruned per construction-grounds (the cadence half is an ordinary ceremony here, the scoreboard an organ shape, Part 2.75). What 4DX lacks is a concern concept, so the clause's link rule — lead measures read from concerns' standing organs, never fresh bookkeeping — is this model's own addition.
- **The order of change:** MUSK'S FIVE-STEP ENGINEERING PROCESS is the import ground of the five-step order (Part 4), imported whole for its order and its three tests (a name on every requirement, the ten-percent add-back, end-of-line before in-process) and bound to laws the canon already holds. Primary source: Everyday Astronaut's Starbase tour, part 1 (published 2021-08-03), 13:29 to 26:40; his own words are "the sort of five step process" and, later in the tour, "that algorithm"; "engineering process" is the interviewer's label and the one name this canon fixes. Quote per version: the 2021 interview says "simplify or optimize" and states the add-back as a frequency ("adding things back in 10% of the time"); Isaacson's 2023 biography (chapter 46) restates it as "the algorithm", five commandments headed "Question every requirement" with the names rule folded into step one, "Simplify and optimize", and the ten percent as a share of what was deleted. The guard-retirement clause is his too (24:49: in-process tests come out once units pass at the end of the line, save the one or two the end of the line cannot see). A second sighting, of STEP TWO ONLY: Bending Spoons' "relentless simplification" (Ferrari to David Senra, 2026-09-13: "the burden of proof is on those who want to add complexity", and the longer a thing has stood the more it is questioned) and its 2026 prospectus ("eliminate as many rules and processes as is feasible"): a standing value with no ordered steps, no add-back test and no names rule, and no descent is claimed (asked whether he follows how Musk talks about it, Ferrari answered "Maybe maybe not").

Where this sits (the discipline coordinates)

Locate any discipline by three coordinates: object, method, unit of intervention. Ours: the **object** is the coordination substrate (the shared symbolic layer an organization steers itself by: registers, states, ceremonies, artifacts, and their enforcement — register-first, not process-first, which is why the nearest literature cousin is the artifact-centric process school, business artifacts with lifecycles, and the deepest cousin is Ostrom's institutional design principles: monitoring, graduated sanctions, cheap conflict resolution, nested amendment rights). The **method** is design science in Simon's sense, with receipts as the epistemology: build the artifact, certify it by unwatched operation, state

falsification conditions, iterate. The **unit of intervention** is the line-manager-authored micro-structure (one register, one ceremony), against Enterprise Engineering's and TOGAF's top-down architect-borne interventions on the same object. As a thought experiment, the object-level name would be **coordination engineering**; this course does not claim the name, for the reason below.

The honest self-test: a discipline needs a distinct object (arguably yes), distinct methods (partially: receipts + agentic enforcement), a community with shared vocabulary (no: one estate, one company), and cumulative results others build on (not yet). Two of four. Today this is a PRACTICE with a thesis; "discipline" is a title only outsiders confer (Toyota never named lean; MIT did, forty years later). The falsifiable milestones: the school exists the day someone never trained by the authors runs this vocabulary back at them; the discipline conversation begins the day an organization the authors do not own publishes receipts using it.

What is actually new here, in honest order of size:

1. **Agents as enforcement (the AI-era delta, and the big one).** Every prior framework's enforcement required expensive human audit bureaucracy, which is why only mature giants could afford compiled-grade structure, and why the frameworks so often decayed into shelfware: companies bought the ontology, could not afford the enforcement, and skipped the gardening. Beer tried to build cheap cybernetic enforcement in 1971 (Project Cybersyn, on telex machines) and failed on the technology. Agents make the checker nearly free at every write path; a Series A company can now afford what previously only a Ritz-Carlton or Toyota could institutionalize.
2. **Compression for teaching.** eTOM is hundreds of pages; TOGAF needs a certification course; nobody hands those to a marketing lead. The 3-layer kernel + catechism + rubric is a 45-minute teachable distillation.
3. **Exam-by-receipts.** The guilds certify recall; this program certifies a structure surviving 30 live days.

Part 6: The training program

Four elements. Two are assessment gates, precisely named: the exam checks you UNDERSTOOD; the practicum checks it WORKS (Norman's own ontology correction, 2026-08-20: a receipts review is a practicum, not an exam).

Element	Format	Content
Lecture	45 min, once	Parts 0-5 and 7 of this document plus the Part 8 close, and one real worked example shown failing and then fixed
Workshop	90 min, same day	Each lead brings their OWN recurring pain and authors the structure for it using the catechism, reviewed live against the rubric (drill #1). No toy cases: the domain knowledge is the point, and authoring is the skill being trained
Diagnostic exam (Gate 1)	~20 min, one synchronized sitting, end of workshop day	Tests the trainable skill, not recall: unscored warmups + fresh-bank diagnosis vignettes + the cold transfer probe. Pass rule, re-sit rule, and item pools live in ONE home: exam/DIAGNOSTIC-EXAM.md (restating gate parameters here is the E3 defect the course warns about — v3, C2). A fail is remediated same-day and re-sat once; a second fail defers the practicum to the next cohort
Pulses	Day 7 / 14 / 21, ~15 min each	The practicum's own enforcement layer (added after the v1 adversarial review found the 30-day silent run was enforcement-by-hope): agent-collected fired-counts where wired, async check where not, same-day nudge on any missed cycle, plus one vignette drill per pulse for spaced practice. Declared in the scheduling contract before anyone leaves the workshop, so nothing at day 30 is a surprise
Practicum (Gate 2)	Formative at day 30, certified at day 60	Days 1-30 are supported (pulses); day 30 reviews receipts formatively. Weeks 5-8 run completely UNWATCHED, and certification requires the structure to have survived them cold (v2 review: a structure kept alive by pulse pressure and a load-bearing one are indistinguishable under monitoring — so the course administers its own absence test to what it certifies). Pass = alive unwatched + gardened on evidence (amendment trail or honest null) + the instrumented team-side median not net-negative . Day-60 certification: a clean-room AI grader runs the committed, participant-readable protocol (exam/CERTIFICATION-GRADING-PROMPT.md) — context isolation removes authorship bias — and a human WITNESS outside the participants' reporting lines attests prompt-verbatim / one-run / verdict-unedited and delivers (operator control is the part isolation cannot fix; the witness is the firewall — v3 C3, resolved). A fail is a diagnosis, one re-review maximum; then the structure retires honestly and the participant restarts with a new pain

Format (Norman's ruling after the v1 review): **parallel 1:1 tracks, ~6 leads at a time**. All reviews, remediations, and pulses are 1:1, never in front of the room (the v1 public-review design predicted theatrical compliance when the reviewer is the CEO). Cross-pollination moves to the sanitized case library (where an anonymized FAILED structure earns a Distinction credit, so the library is not survivorship-biased by construction). Two preconditions gate scheduling: the sponsor's written **performance firewall** (course outcomes never enter performance evaluation) and **named, numbered fleet-gate capacity** (cohort 1 default: two gates). Disagreement protocol, stated at the workshop: the participant owns every domain call; the reviewer may only cite the rubric. Honest dose accounting (v2, recounted at v3): this module installs the vocabulary, TWO supervised authoring reps (the worksheet and Q11's cold transfer probe), and one 60-day operating-and-gardening run, plus recognition drills at the pulses; taste arrives with the second and third structures, which is what Distinction measures.

Note the self-hosting property, which is also the credibility argument: the lecture installs layer 1, the workshop runs layer 2, the practicum is layer 3, the diagnostic exam is the enforcement gate on the write path into the practicum, and the day-30 review is gardening. The curriculum is built from the model it teaches.

The module package (the definition of done)

The training is DONE when the module runs without its author, which means five artifacts exist:

1. **The reference doc** (this document): the model, skills, catechism, rubric.

2. **The lecture script** (45 min): Parts 0–5 and 7 plus the Part 8 close, with the worked failing-then-fixed example, openable by any trained reviewer, not only Norman. Metaphor kit for the script (chosen against the rule that a lecture metaphor's failure must be INTRINSIC to its domain, which killed the dance/choreography candidate: ensemble freestyle on a stage is merely an aesthetic choice): primary = **the kitchen brigade** (Escoffier's stations as nouns, the ticket as the row, service as the clock, the pass as a literal write-path gate, menu revisions as gardening; six talented cooks with no brigade on a busy night fail on coordination, not skill, and the food-truck-to-restaurant arc carries the stage curve); one-liner = **the rowing eight** (eight strong rowers without a stroke rhythm go in circles: strength is not the variable); evidence anchor = **the checklist studies**, WITH the replication caveat spoken aloud (the 2009 WHO trial's headline effect was strongest at lower-income sites and a large Ontario implementation study found no significant mortality change; the honest claim is "checklists help where implementation is real, and the population studied was elite" — saying this out loud costs 15 seconds and models the epistemic standard the course preaches); history slide = Part 8's lineage, presented explicitly as ILLUSTRATION, not evidence (survivorship cuts both ways: the Mongol succession collapsed within decades), with Rome/Genghis for "the system survived bad generals" and "speed FROM structure".
3. **The workshop kit**: the catechism as a fill-in worksheet, the rubric as a scorecard, and one page of facilitator notes (common failure patterns to catch live).
4. **The assessment set**: the diagnostic exam (Gate 1: fresh-item vignette bank + keys, agent-gradable, with re-sit spares held back) and the practicum protocol (Gate 2: day-30 formative receipts review + the day-60 cold certification check).
5. **The worked examples**: lifting + marathon (the domain-neutral pair) and one real Gotrade before/after once the first cohort produces it.

The module passing its own absence test (someone other than its author runs a cohort successfully) is the graduation of the training program itself from layer 3 to a standing institution.

The whole discipline, for the closing slide, in two lines. The order of every build and every change, five steps: **make it less dumb, delete it, simplify it, accelerate it, automate it**. The shape of what survives, ten words: **name it, clock it, write it, guard it, garden it**.

The two mantras track two different things and neither replaces the other: the five steps are the ORDER (the five-step order, Part 4: how a structure is born and changed, and the door's first two answers may be "nothing"), and the ten words (the ten-word mantra) are the MODEL, not the verb list: name/clock/write are the three layers, guard and garden are the two loops. "Guard it" deliberately commands the whole fast loop, both halves (author the gates, then walk only what escapes them); Inspect is the half that stays yours. A mantra that said "inspect it" would command the mass-inspection failure mode instead of the design that prevents it (the Deming note, Part 2.5).

Part 7: The honest scope

This is not a growth strategy; product, market, and distribution are (Part 0 makes the real claim: accumulation and agentization). Two more honest boundaries, stated up front because they build trust with a skeptical room:

- **Structure is not bureaucracy, and the fitness tests are the difference.** Every field must change a decision, ceremonies must shrink, and a docket empty for a quarter merges away. Anyone in the room can kill a field or a ceremony by showing it fails the rubric. Process that cannot be killed by evidence IS bureaucracy.
- **Agents enforce shape, humans judge substance.** The fleet can guarantee the experiment row exists with a hypothesis and a verdict; whether the hypothesis was worth testing stays a human call. This program makes judgment go further; it does not replace it.

- **The model is a vocabulary and a design discipline, not a predictive theory.** Any failure can be narrated post-hoc as "some layer died," which explains everything and therefore predicts nothing; own that. The falsifiable claims live at the INSTANCE level, and here is the standing one: a structure that passes all seven fitness tests (the seven fitness tests) and still changes no decisions after 60 live days indicts the model's fit for that domain, not the participant. When that happens, say so and retire the structure.
- **Structure the pipeline, never the ideation.** Brief, review, launch, learning: structure them. The thinking that produces the campaign concept, the product idea, the design: that is exploratory work, and Norman's own doctrine says formal process kills it. If the noun you are about to define is an idea, stop.

And the restated aphorism, for the room that believes "execution eats strategy for breakfast": three things get conflated in that phrase: strategy (what to do), execution (the reps performed: layer 3), and the **execution system** (layers 1-2 plus enforcement and gardening: the machine that makes the reps reliable and compounding). The aphorism is true because strategy is a one-sentence layer-3 utterance, cheap to produce, while the machine is expensive, slow, and rare: the machine is the moat. Better: a good machine repairs a mediocre strategy (the register turns wrong bets into accumulated learning, and learning iterates the strategy), while a brilliant strategy inside no machine never becomes rows. Precisely restated: **the execution system eats strategy, because it converts strategy into data and data into learning.**

Part 8: The lineage (ten thousand years of the three layers)

The model is not modern. Each historical crucible had a maximal product of coordination surface x cost of failure, and each added one element to the stack:

Granary -> Army -> Monastery -> Church -> Ledger -> Railroad -> Factory -> Software -> Agents

- **The granary: writing itself was invented for layer 3.** The lineage's true start is pre-cuneiform clay-token accounting (~8000 BCE, documented by Schmandt-Besserat: tokens standing for grain and livestock, sealed in clay envelopes), which evolved into cuneiform temple registers by ~3000 BCE. The oldest documents in existence are not poetry: they are registers of grain, livestock, and labor. Accounting predates literature by centuries. The state and the harvest (famine EV) produced the first registers.
- **The army: enforcement under adversarial selection.** War added what the granary lacked: your structure competes against another structure, losers are annihilated, winners get copied (a Red Queen race, which is why military ontology evolved fastest). Rome: the legion's full stack, from the contubernium to the marching camp rebuilt to an identical layout every night, drills that Josephus called bloodless battles, capital enforcement, and documented schema migrations (the Marian reforms). The payoff: average farm boys became the world's best infantry for five centuries, and **the system survived bad generals**: Rome could lose an entire army at Cannae and regenerate, because the ontology was reproducible and talent was not the load-bearing part. Genghis Khan: the decimal system (arban 10 / zuun 100 / mingghan 1,000 / tumen 10,000) used as a POLITICAL weapon: he broke tribal units and reassigned men across tribes, re-pointing loyalty from clan to structure: re-authoring the nouns to change what people are. Plus the yam relay network (declared infrastructure), merit promotion, collective-responsibility enforcement wired into the unit, and the nerge: the annual great hunt, a scheduled ceremony whose real function was drilling tumen-scale coordination. Mongol armies converged on schedule from continental distances and operated for years thousands of kilometers from Genghis: the absence test at its historical maximum. **The Mongols were the fastest force on earth BECAUSE of the structure**: the standing rebuttal to "process kills speed." The Prussian General Staff later added institutionalized gardening (after-

action learning, war games) and Auftragstaktik: commander's intent plus local improvisation, structured improvisation as formal doctrine. The caution is in the same lineage: late Rome is what an ungardened ontology becomes.

- **The monastery (Rule of St. Benedict, ~530 CE): the clocked ceremony.** Canonical hours as scheduled ceremonies, defined roles, and a weekly enforcement ritual (the chapter of faults: violations surfaced aloud). It passed the absence test for fifteen hundred years across thousands of houses: history's longest-running organizational system. Mechanical clock discipline itself came out of it.
- **The church: the amendment sitting, and the record's largest gardening failure.** The monastery is one house's rule; the Church as an institution ran the full stack at civilizational scale, and what it adds to the lineage is the slow loop as a standing institution. Layer 1 is literally called canon law (major versions shipped in 1917 and 1983); sacraments are typed state transitions on a soul's lifecycle; sainthood is a register with an admission process whose examining officer's actual title was **advocatus diaboli**: adversarial review institutionalized four centuries before software engineering named it. Layer 2: the Mass on the liturgical clock, with procedural instructions literally called rubrics. Layer 3: parish registers of baptisms, marriages, and burials, among the oldest continuous records in Europe. The ecumenical councils, Nicaea to Vatican II, are amendment sittings; ordination is a recorded, scoped authoring grant (apostolic succession as a two-millennium authority register) while the laity operate freely; and the parish is a franchise unit externalized enough that an average priest can run one. The caution is what earns the entry its place: **the Reformation is the largest gardening failure on record.** Indulgences drifted over three centuries from enforcement mechanism into a billing system, and the drift's owner was the house itself; Luther filed the 95 Theses as a licensed author through the system's own doors (disputation points, in Latin, sent to his archbishop the same day); and the institution processed the amendment proposal as a discipline case. Enforcement ran where gardening should have, a practice question was escalated into an authority crisis, and the organization forked. Causality stated honestly, because survivorship cuts both ways (this entry, like the whole lineage, is illustration rather than evidence): no institution survives twenty centuries *because* it adopted this machinery; the survivors are the ones that kept evolving it, and the machinery is why their structure compounds rather than merely grows.
- **The ledger (Venice, codified by Pacioli 1494): verifiable data.** Double-entry bookkeeping is arguably the most durable ontology ever authored: unchanged in its core for five centuries and still the substrate of every finance system running today.
- **Railroad and factory: scale coordination and measurement** (the first modern org charts, timetable discipline, Taylor). **Software: executable SOPs** (the process runs itself). **Agents: free enforcement** (the checker at every write path, which history could never afford: see Part 5.5).

Ten thousand years, same three layers; the module is the newest entry in a very old lineage, and the lineage is the lecture's credibility in one slide.

Appendix: governance

The steward's block: version history, epistemic-stamp governance, and the unit-register declaration. Folded out of the reading head in every render; the source keeps it at the top for the gate.

Version: 5.1.1 (2026-09-29). Semver, one line, bumped in each amendment commit: major = vocabulary or model breaks (anything derived artifacts or trained cohorts must migrate to), minor = additive laws and sections, patch = wording. v1.0.0 is the 2026-08-20 "v1 complete" baseline. v1.1.0 (2026-08-26, mislabeled minor by its own scheme: it retired the "meta-activities" framing and rewrote the calcification caution, both breaking) added the identity law, the two loops and reification (Part 1.25), the survey door (Part 2, skill 2), the five verbs (Part 2.5), and the calcification relocation (Part 5). v2.0.0 is the adversarial-review-v4 fix pass: the new taxonomy migrated into every derived artifact, each stamped with the canon version it builds against, enforced from here on by the release gate (`scripts/release-gate.sh`; no version-bump commit without a passing run cited in its message). v2.1.0 adds the epistemic-stamp governance below and the first provisional law (spec and implementation, Part 1.25). v2.2.0 adds the two-kinds-of-processes law (Part 2.5: domain-owned SOPs versus ceremonies as cross-domain convenors, provisional with its first receipt logged). v2.3.0 (from the email build's first operating night): the survey-door guard sharpened into the four-part jobs admission standard with the retro-fit trap and late-entry clause; the flip-conditions law (Part 4); the induction law (Part 2.5). v3.0.0 (2026-08-27): the doors RENAMED to the pain door and the JTBD door (breaking; the field-tested names from the first live teach displaced incident/survey before any cohort learned them, swept everywhere); the depth and split rules on the doors (Part 2, skill 2); the construction-grounds law with the coverage walk, and the authority law (both Part 1.75, provisional); the spine defined as a term of art (Part 2.5); Palantir Foundry added to prior art (Part 5.5); the KYC compliance audit joins the worked examples. v3.1.0 (2026-08-27): the grain law (Part 2) — a committed domain is a pipeline, never a category or an outcome; the four sizing tests, the organ corollary, and the walk-capacity guard; distilled from the fleet's first sizing ruling. v3.1.1 (2026-08-27): spec-and-implementation promoted to SETTLED on its first receipt (the email-intake concern's court read-state divergence, ruled an amendment; the first full audit of a JTBD-door structure against the v3.1.0 instrument set ran the same sitting). v3.1.2 (2026-08-27): the mantra-to-model note (Part 6): guard names the fast loop whole, Inspect its human half. v3.2.0 (2026-08-28), the field harvest from Relay's first two concern-authoring days: reification, the authority law (gaining the apprenticeship grant path), and the grain law promoted SETTLED on their pre-registered receipts (the meta-concern; the live apply-door allowlist; the second independent sizing ruling); the spine-anatomy law with the orphan rule (Part 2.5); the jurisdiction-and-home law (Part 1, provisional); derivation-before-delegation (Part 5, caution 3). v3.2.1 (2026-08-28, same night): the jurisdiction law promoted SETTLED on its pre-registered receipt (the DEC-register re-homing, DEC-50, run through the six-step procedure hours after ratification), and its executor's friction note folded in: re-homing named in two shapes, custodial (stewardship and address move) and jurisdictional (only the governing spine changes; alias and handover satisfied vacuously, legally). v3.3.0 (2026-08-29): the canon reified into its own ontology — unit taxonomy (MODEL/LAW/INSTRUMENT/EXAMPLE/NOTE) + REGISTER.yaml (24 laws in 7 named families, 7 models, 7 instruments; NAMES as identity, slugs for machines, Parts demoted to reading order, prose-from-rows refused with a flip condition); the cognitive-fit law (Part 4: structures sized to the operating head — Norman's working-memory rationale, reverse-deriving the split line); the ranking corollary of the identity law (position is a field, never an identity; from the Gotrade principles-reordering question). v3.3.1 (2026-08-29): the Church joins the lineage (Part 8, between the monastery and the ledger: canonization's advocatus diaboli as institutionalized adversarial review, councils as amendment sittings, ordination as the recorded authoring grant, the Reformation as the record's largest gardening failure; causality stated as illustration, not evidence); the grain law's provenance cites its ruling code (DEC-29); the print render folds the governance header into an appendix, matching the site (derived views only, source untouched). v3.3.2 (2026-08-29): the readability edition — sixteen laws promoted to headed subsections (register name as the heading, statement lead, short paragraphs, provenance and receipts demoted to a trailing stamped line), the two remaining walls split in place (skill 2, the calcification caution); prose structure only, no vocabulary or unit change (the ranking corollary applied to the canon's own reading order); renders gain law-level anchors, stamp chips on law headings, hover anchor links, and styled provenance lines. v3.3.3 (2026-08-29): Part 7 retitled to "The honest scope" — the "(say this out loud in the lecture)" parenthetical was a stage direction to the lecturer mis-homed in a reader-addressed title; delivery's home already carries it (Part 6's lecture row names the spoken set, the body's "stated up front" line keeps the honesty reader-facing). NW-ANA-23. v3.4.0 (2026-08-29): the organ type system (Part 2.75) — eight organ types EXTRACTED from the two living deployments' inventories (register, ledger, court, board, surface, door, index, corpus; the spine excluded as the organ-holder, ceremonies as processes; scoreboard and form marked deployment-local with flip conditions); the types-and-bindings law (Laws of the spec, provisional: the canon owns types, deployments own only bindings; config-as-data source `canon/ORGAN-TYPES.yaml` carrying types only; gate check 6 alarms on count drift); the qualifier corollary of cognitive-fit (Laws of the head, settled on the view-only mini-app ruling). One

language across every deployment Norman runs; NW-ANA-22. v3.4.1 (2026-08-29): the organ DEFINITION ratified (Part 2.75: an organ is a DECLARED PLACE, the latent layer-1 slot named; four membership tests) with three boundaries — spine (the declaration), process (the verbs that operate places, notification routes included as delivery clauses), channel (SUBSTRATE, which may host organ bindings but never takes a type; ruled on the fleet's organs.spaces question — spaces exit organs blocks as reference fields, the Spaces-concern question routed to the fleet's own door); the jurisdiction law's second pre-registered receipt cited (CRN-D1, the first domain-local ruling filed in its own spine); construction-grounds promoted SETTLED on its pre-registered receipt (the fleet's ISMS concern: ISO/IEC 27001:2022 imported and pruned); the flip-conditions due condition armed with its named candidate (the 28-day gap-acceptance clock, ~late Sep). v3.4.2 (2026-08-29, the same night): types-and-bindings promoted SETTLED on its pre-registered receipt — the fleet's organ-type binding live at the write path and landed UNAMENDED (bindings as data on every dossier payload; the spaces refusal enforcing the channel boundary, negative-tested), hours after the estate's declaration; the eight types absorbed both deployments' full inventories; the provisional set drops to three, each with an armed or named receipt condition. v3.5.0 (2026-08-30), the Sunday readability sitting: THE READABILITY LAW (Part 4, Laws of the head — every artifact a head reads is written in the head's working vocabulary, machine encodings underneath; settled at naming on standing instances: speakable identities, display-word states, the question dialect for jobs, commitment-suffix concern names, ORM-over-SQL; the deciding receipt a cold comprehension read across both deployments' 26 declarations) and THE GROUPING COROLLARY (Part 1, Laws of the nouns, ranking's sibling — membership is a field, never a domain: trees render on surfaces from fields, jurisdiction stays flat, labels graduate by receipt); the jobs admission standard gains its dialect clause (a job is the hirer's question, in the hirer's words); the register at 28 laws with both touched families at the five-law cap. v3.5.1 (2026-08-30, hours later): the HIRER'S-VOICE CLAUSE stamped into the readability law on the fleet sweep's receipt (thirty rows, full conformance in one sitting: the hirer is whoever CONSUMES the answer, supply never wears the hirer cell; condition shape legal only for a genuine machine consumer — exactly one of thirty, the DoD gate; four of five apparently-machine hirers were the supply defect one cell over); the admission standard gains the CONSERVATION rule (intent-class terms relocate into the spine's ontology, mechanism-class terms anywhere the organ index reaches, moved in the same edit); the workshop's facilitator notes gain failure pattern 8, the system-voice job, from the sweep's live conversions. v3.5.2 (2026-08-30, the sections sitting): the BIRTH STORY joins the spine anatomy — both deployments carried it independently from their first spines (every estate dossier opens with one, every fleet declaration has a birth_story field), the second-deployment receipt pattern; the estate's binding shipped the same night (the eight-section dossier template, two specimens converted, the index compiler's Observe-mode sections noticer). v3.5.3 (2026-08-30, minutes later): the PORTFOLIO clause joins the identity law's namespacing — a multi-org operator keeps a declared zone registry homed with the holder (no single deployment may own it), codes reserved cheaply, partitions as org-scale re-homings; taught by the neutral ABC/XYZ/HOME worked example; first instance the holder's four-zone registry, the second pre-registered in the tenant model's entity-registry design (the audience claim: power users are multi-org users). v3.5.4 (2026-09-02): the GOAL-LAYER clause joins the grain law (NW-ANA-33, from the first outsider running the vocabulary back — "how do concerns link to growth objectives?"): an outcome refused as a domain lives in the GOAL LAYER above domains — objectives as closable battles (from X to Y by when, one owner, won/missed/abandoned), linked to domains ONLY through lead measures whose readings come from the domains' standing organs, never fresh bookkeeping; engines permanent, destinations reached; 4DX cited as the import ground (Part 5.5) with the link rule named as this model's own addition; the scoreboard's flip condition armed with its declared second candidate (the estate's hyrox goal layer, declared the same sitting); the lecture objection map gains "how do goals fit?" beside the anatomy and grain questions. v3.5.5 (2026-09-03): the readability law's concern-name instance gains the DEFAULT-COMMITMENT clause (Norman's ruling, prompted by the FRO mint's naming question): management is the default commitment word, displaced only by a sharper one that names the pipeline's kind — a default, never a mandate, because intake/operations/readiness carry information a uniform suffix would flatten, and surface economy stays legal (concerns-only indexes render bare). Migration receipts: the estate audited 17/17 conformant unprompted; the fleet directed to audit bare-territory names at gardening touch, display names moving, slugs never (the identity law). v3.5.6 (2026-09-03, three hours later): the default-commitment clause SUPERSEDED by the UNIFORM-SUFFIX rule on Norman's definitional ruling — a concern IS a domain plus the commitment to manage, so its name is territory + management, always; kind-words live in the territory half; a name resisting the suffix is a species diagnostic (not-a-concern), never a case for a different suffix. The deciding evidence: the fleet's same-day audit under the default clause produced an 11-to-5 "default" (the smell), while the estate's 17-for-17 — including email-intake-management and finance-close-management, the kind-in-territory form — showed the uniform rule already running. The double loop at full speed: clause stamped, receipt gathered, smell caught, rule corrected, one afternoon. v3.5.7 (2026-09-04, the codify-first sitting): the capacity guard RENAMED and RECAST as the CAPACITY LOOP (a guard refuses; this steers) — a conservation invariant matching domains to judgment capacity, never rationing codification: "the alternative to a codified domain is never a managed one," so codification is always preferred, converting silent gaps into visible debt; two levers by stage (shrink count at founder-scale, grow capacity at system-scale — inspection

being operator-tier and unlicensed, walkers are hireable, agents multiplying reach); staffing steered by two receipts-derived meters (debt aging = demand, recorded disposition drain = supply — an artifact read, never surveillance; both red = hire, red queue over idle walkers = fix routing; drain-rate a signal, never a quota); the loop's one guard residue: never read a red meter as quiet, never hide the meter. Norman's ruling across the 2026-09-03/04 midnight sitting, provoked by the enterprise thesis (company-as-code, Relay as runtime, agents as the operator mass); the estate's aging email court cited live as the demand meter working. v3.5.8 (2026-09-04, minutes later): THE DONE-TEST joins the two-doors model (prose, no register movement; the caps hold at 7/7/28) — any intake ask classifies by "when would this be done?": never/keeps-flowing = a domain through the JTBD door now (recurrence is only the pain door's threshold, never a wait on a knowable engine); a date or end-state = a project on a board; when-the-number-hits-X = the goal layer; a misleading shape = a project with a pre-registered recurrence flip. Distilled live in the same sitting when Norman asked whether an agent must wait for a pain to recur twice when the shape is knowable upfront; the answer is the two-doors model spoken at intake speed, and it is the routing core of the pre-registered door-noticing clause (agents harvest domains from conversation, never evangelize vocabulary). v3.5.9 (2026-09-04, the same sitting): TYPED LINKS join the spine anatomy's ontology component — the noun GRAPH, never a noun list, each edge a domain verb in the hirer's vocabulary (Payslip PAYS Report; the ORM relation names an edge's shape, the domain verb its meaning), cross-domain edges naming the foreign home. Ratified on Norman's first-principles push, and the derivation held three ways without a second deployment: the grain law's own definition (a pipeline is records MOVING, and movement is edges — an ontology without links described inventory, not a pipeline); the jurisdiction law's typed-reference mandate already covering the harder cross-domain case (this closes the intra-domain incoherence); and the import ground (fifty years of relational modeling, Chen's ER diagrams to RDF triples — archaeology, not invention, the ISO/4DX door). The canon owns the PRINCIPLE; the link encoding stays a deployment binding (types-and-bindings applied to links themselves), with the estate's two sidecar dossiers the first instance and the fleet's OKG mirror armed as the encoding's confirmation. v3.6.0 (2026-09-07) added the species law (Part 2: the three species, the project anatomy and its home question, the project-workstream-task grain, PROVISIONAL), the authored-once corollary under Spec and implementation (Part 1.25), and the T20 receipt on the spine anatomy's links clause. v3.6.1 (2026-09-07 evening, patch): three clauses inside the species law: the clock's two kinds (hard, target) bound to the attention grades and required only while active; the baseline moves only by a recorded shift; a project may rest, its encoding a binding (both deployments read: Relay's Board fields, the estate's tickler). v3.7.0 (2026-09-09, the T25 sitting): THE SPINE SCHEMA as machine form (canon/SPINE-SCHEMA.yaml, the config-as-data source of the eight components and the fields whose meaning the canon owns, extracted from both deployments' contracts; gate check 8; a jobs stamp refused at the door, the verb's kind admitted on the fleet's instance) inside the spine anatomy; three clauses inside Part 2.75: THE REGISTER'S WALK SET (terminal and resting bits on a state; open = neither; ruled on the fleet 2026-09-07, the estate's column binding retiring), LIVENESS IS DERIVED AND THE INDEX DECLARES ADDRESSES (the pulse derived by the consumer from the organ at its address; the estate boundary the one exception, where a declared binding is the receipt door for foreign organs; a verb's clock authored once, in the spine or in the scheduler it names), TYPE BY ROLE (an organ's type is its job never its mechanism, derived and foreign are qualifiers, the organ index a free typed entry); the conformance-in-the-consumer's-context sentence settled inside Spec and implementation on two receipts (GBUG-1123; the estate's dev-only yaml parser); the authored-once corollary's first receipt (the estate's migration, 2026-09-09). Register counts unchanged (30 laws, 5 provisional; both caps binding, so the sitting entered as clauses and a machine form, never rows). v3.8.0 (2026-09-10) added THE OKG (Part 5, Laws of enforcement, the 31st law, SETTLED at birth on two deployments): the Operating Knowledge Graph, a deployment's one derived view of everything it manages, condensed out of the declared spines and never authored; its Map, its lookup door, and its name fixed as the one binding the canon owns. Register counts: 31 laws, 5 provisional (Laws of enforcement had room, so the sitting entered as a row). v3.8.1 (2026-09-10) recorded the OKG law's two armed confirmations as receipts, both landed the same day (the estate's lookup door on prod; the fleet's widened lookup instruction on the boxes at core v1.8.20). v3.9.0 (2026-09-11) added THE WALK LADDER (Part 2.5, Laws of the running system, the walk group, PROVISIONAL): a concern's walks nested by altitude, an exception climbing exactly one level by aging or recurrence, the executive walking the walks, the mall's toilets as the worked case; promoted Two kinds of processes to SETTLED on the fleet's second independent case (CRN's coverage job walked by relay-eng-daily); the register descended a level (a group field on rows, two families grouped, identities untouched; the three v3.7.0 clauses of Types and bindings registered as their own rows, text unmoved); the release gate gained the branching check. Register counts: 35 laws, 5 provisional, 7 families, 4 groups. v3.9.1 (2026-09-11) added the verbs of arrival (Part 2, under the doors): to DECLARE a domain is to author its declaration through a door, its BIRTH is the day the declaration enters, a COMMITTED domain is the result; MO-ify and concernify named as slang (wording; no register change). v3.9.2 (2026-09-11) recorded the liveness clause's armed pair as receipts: the boundary declaration landed on both deployments in two spellings (the estate's feed inside the address, the fleet's receipt key beside it), the canon owning the meaning and each deployment its spelling. v4.0.0 (2026-09-12, the first major): the opinionated turn. One name per concept (a law, provisional) with canon/GLOSSARY.yaml as its machine form and the speech test as its

boundary; the canon's own word for a committed domain is CONCERN (domain stays territory), a text pass across the canon, the kernel and the derived artifacts; the types-and-bindings law's naming half amended (mechanism words are qualifiers, never names); the readability law sharpened; knowledge routes by role (a corollary, provisional); the scoreboard the ninth organ type on two deployments' receipts; Laws of the head descended into three groups; the release gate reading the glossary. v4.1.0 (2026-09-14): the attention-efficient operating system. The objective function in Part 1 (value created per unit of operator attention; the three seats) and the two editions by player count; the attention law and the push ladder (Part 1.5, a new group, Laws of attention); home by construction, depth by receipt, and jobs carry readings (Part 2; Laws of the doors descended into three groups; the spine schema's jobs component gains kind, reading, target and grade); the default-walker clause on the walk ladder; the fleet's read shaped five rulings (the human seats as the denominator, Notice as the fifth rung, the workshop's second sense on the fleet, the attention grade over a new name, the reading's standing). v4.2.0 (2026-09-18): the workflow law (Part 1.25, the fifth row of Laws of the spine): a workflow is a deployment's binding of one verb to one runner, named once and cited by name from the verb's row, its concern its verb's by reference and never a container, one workflow to one verb, the catalog an index and never a home, an organ's address a reader of receipts and never a home, the orphan of the fifth class named; the glossary gains workflow, runner and catalog; the estate's workflows index and lookup door the first half of the due, met at birth. v4.2.1 (2026-09-18): the repeat clause on the push ladder (a push returns only when a threshold breaks, never because its clock fired again; one docket per clock, then the new and the crossing), on the estate's three descents of the same day. v4.3.0 (2026-09-21): the five-step order (Part 4, the fourth law of Laws of the work): every structure is born and changed in the order of Musk's five-step engineering process, written out because the order is the law (make the requirement less dumb, delete the part or process, simplify or optimize, accelerate cycle time, automate); the deletion question on two clocks (birth, and every change of the mechanism around a structure); a guard carries its incident, its meter and its retiring reading, and retires only against a named end-of-line reading, built first where none exists; the add-back rate calibrates a pass; the classes where a miss cannot be taken back keep their controls; deletion is retirement; the glossary gains three names (the law, the process, the end-of-line reading); the import ground joins Part 5.5, with Bending Spoons a sighting of step two only. v4.4.0 (2026-09-22): the five-step order as the operating mantra: Part 0 gains its third fact (structure that should not exist compounds too), skill 1 asks whether a noun before which, skill 2 reads the admission standard as step one's challenge, the catechism runs the order at seven questions (the hirer's name, then the deletion question, before any noun; the first two answers may end it with nothing built), and the closing slide carries two mantras, the five steps first and the ten words as the shape of what survives; no new law, no new name. v4.4.1 (2026-09-22): step one's test, on the operator's question "what if the hirer wants dumb requirements?": the name is the address for the challenge and the owner of its consequences, never proof of sense; a requirement is less dumb when it changes a decision, costs something unserved and moves an objective; a hirer who insists past a failed test owns it on record and it is metered like a guard; a challenge is never a veto. v4.5.0 (2026-09-22): the push moment, a clause on Jobs carry readings: a job declares the rung by which its reading reaches its hirer and its return (new only, or the declared age rungs), the clock staying the verb's, absent reading pull; an alert is a job's notice; a deployment's catalog of alert kinds is a binding, cited from the job as a verb cites its workflow, and a notice no job names is the sixth orphan class; the scoreboard's pushes row derives from the declarations, never a hand-kept list; the spine schema's jobs component gains push; the glossary gains Push moment. v4.5.1 (2026-09-26): a patch, receipts and one sentence: the fleet's read rates behind the push moment and the five-step order re-derived from the readers' thread markers (the badge column first quoted is not read state; the operator opened almost none of either class), and a read meter is a floor (a read outside the app leaves no mark, so below the bar means rarely opened, never unseen). v5.0.0 (2026-09-27): the second major, a model break in the species law's grain: a WORKSTREAM is a parallel thread of its project with no owner, moved by a rolling next move whose holder may change move by move, its path a plan and never a stock of tasks; a step becomes a task only alongside another holder, on a world's date, or needing its own record; no holder defaults down the tree; the walk of a project reads threads as moved, resting on a tickler, or waiting on a named blocker that is itself moving, anything else the exception; WAITS ON enters the grain as one typed edge for dependency and snooze-on-condition alike (any grain on any other, on its own project or another; backstop, wake, the hand clears, the walk chases the blocker); REST extends from projects to all three grains, on a date or on a condition, the condition being a wait. Major by the scheme's own rule: the fleet's board workstream carries an owner column built on the old sentence (DEC-144), and trained cohorts learned "for X to do Y". The glossary re-defines Workstream and gains Waits on. v5.0.1 (2026-09-27): a patch, the fleet lane's read of the major: five wording corrections (DEC-144 the whole binding with only its workstream clause re-ruled; DEC-312 cited for whole-board and several blockers; no project waits on its own workstream, its own part; a blocker is named only as far as the waiter's readers may read it; the glossary's Waits on carries its never clause) and the fleet's receipts (its refusals, REST at every grain since T61, 19 of 428 workstreams with an owner, re-ruled away the same day). v5.0.2 (2026-09-27): a patch, the fleet lane's second read of the major: a waiter reads as waiting, never as the exception; the walk chases its blocker, which carries the exception where it lives (read literally, 'a named blocker that is itself moving' made a waiter the exception whenever its blocker stalled, against the same paragraph's 'the walk

chases the blocker, never the waiter'; Relay took the literal sentence first and flagged 14 waiters as stuck on the live G8, fixed in console 2190af7e, live through v1.41.31 and in the fleet MO skill at core v1.8.80; the estate's ReadThreadWalk, nws-todo 02ffc11, already read the waiter as waiting). v5.1.0 (2026-09-28): the provenance law (Part 5), the first amendment from the estate's register of outside findings (NW-MOF-1, NW-MOF-4), on three failures across both deployments in one week: a verifier reads the evidence before the conclusion and keeps the conclusion, the order of the read being the rule and orientation exempt; a handoff carries its observations apart from its conclusion; a correction is appended (a record of its own or a dated paragraph whose prior text the record's history keeps), never a silent overwrite, names the claim it supersedes and reaches that claim's citers through their references, each asking whether its one claim still holds, never whether the source changed. Import grounds: auditing's evidence standard and truth maintenance. Read by the fleet before landing, three leans amended on its receipts (the correction's form, the lookup's reach, the due's candidates). The glossary gains Provenance. v5.1.1 (2026-09-29): a patch, the OKG law's reach clause (Part 5), on the operator's question whether an easy lookup of concerns, registers, jobs, fields and rows was not the OKG's whole point: every citable handle means the lookup door reads through the spines to what they govern; a register's fields belong to its organ and a field's name finds it; a coded instance is answered by its code and found by its words, as a handle with its state, never the store; how much of it a reader sees is the edition's authority; a foreign prefix reads unknown, never a negative. Both deployments had read the node sentence as the door's limit; the estate met the clause the same day, and the fleet's due is a row found by its words. The provenance law's due corrects its own lookup sentence (the estate's lookup lists the register rows citing a code since the same day). The glossary's Lookup door carries the reach.

Epistemic stamps (ratified 2026-08-26): every law is either SETTLED (receipts cited) or PROVISIONAL (ratified, no receipts yet, its first receipt's due condition stated inline). Laws enter the canon the day they are ratified; the stamp is what keeps eager canonization honest while the philosophy actively develops. Provisional laws are taught, drilled, and examinable everywhere EXCEPT the day-60 certification verdict, which may not hang on a provisional law alone (the reversibility ladder applied to the course's own instruments: the certificate is the irreversible act). A provisional law whose receipt window lapses without a receipt is a kill-docket candidate; promotion to settled is a patch-level edit citing the receipt.

The unit register (declared 2026-08-29, v3.3.0): the canon's own ontology, applied to itself. Every unit is typed — MODEL (descriptive structure), LAW (normative, stamped), INSTRUMENT (runnable), EXAMPLE, NOTE — and registered in canon/REGISTER.yaml: laws by NAME (the primary identity; human memory runs on labels) and slug (the machine identity), grouped into at most seven named families, each of at most seven groups of at most five laws (the cognitive-fit law obeyed by its own canon: 7 models, 7 instruments, 7 families; a family that fills descends a level, never stretches, since v3.9.0). Part numbers are READING ORDER only, free to restructure at editions; nothing cites them as identity. The register is metadata, never prose — compiling the canon's text from rows is REFUSED with a pre-registered flip condition (a second edition needing a different ordering or subset of the same laws rendered from one source): the half-hour read is authored, not rendered.