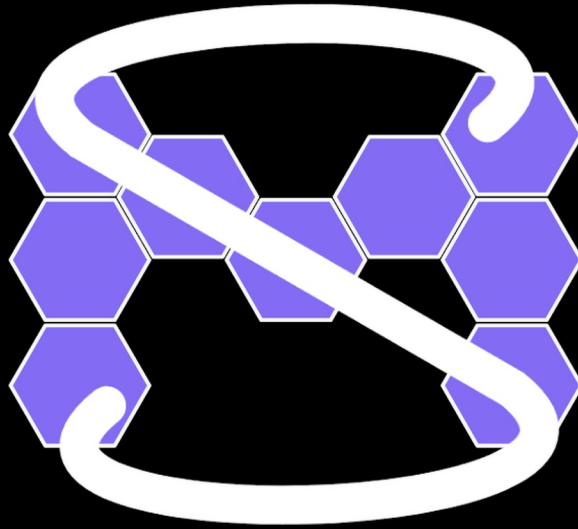




METRA PRAXIS

DISCOVERY

PHASE ONE, FROM CONVERSATION TO FOUNDATION

THE METHOD BY WHICH STANDARD IS BUILT
SEPTEMBER 2026

BY METRASLATE



What Discovery is for

Discovery is the phase before anything is built. It turns a conversation with someone who knows a domain into a written picture of the whole system, and then into a foundation decision: what infrastructure the picture needs and what stack best supports it. Nothing enters a lane until the picture is ratified. The reason is the same one that makes the fences work: a model builds exactly what it is told, so the telling has to be complete before the building starts.

Discovery produces five artifacts. Each is a file in the vault, each has a named owner, and the phase is over when the fifth is ratified.

Artifact	What it is	Written by	Ratified by
Domain Pack	The domain in the customer's own words: personas, records, workflows, standards, numbers, vocabulary, limits	Domain expert, interviewed by the operator	Domain expert
Research Brief	What the outside world says: standards, regulators, incumbents, integrations, procurement, the lead list	Operator	Domain expert
System Picture	The whole system on ten pages: subjects and lifecycles, the spine, trust, integrations, volumes, non-functionals	Operator	Ratifier, after a grill
Foundation Decision	The infrastructure the picture needs and the stack that supports it, with the reasons	Operator	Ratifier
Wedge Brief	The first product on one page, with its thirty-day scope and its acceptance audit	Operator with the domain expert	Ratifier

Three seats. The **domain expert** knows the work and owns the truth about it; they never touch code. The **operator** runs the interviews, does the research, writes the picture, and runs Praxis; they may know nothing about the domain on day one, and the process is designed so that this is fine. The **ratifier** decides; in a project with one founder the ratifier and the operator are the same person, and the seat is still named so that the decision is a distinct act.

Timebox: five to ten working days. A Discovery that runs past ten days is a sign that the wedge is too wide, not that more research is needed.

The shape of the phase

Stage	The question it answers	Instrument	Output
0 Intake	Why this, why now, and under what constraints?	Intake questionnaire	Intake sheet
1 Conversation	What is the work, in the words of the people who do it?	Eight interview worksheets	Domain Pack
2 Research	What does the outside world	Research checklist	Research Brief



Stage	The question it answers	Instrument	Output
	require, sell, and expect?		
3 The Picture	What is the whole system?	System Picture template	System Picture
4 The Foundation	What infrastructure does the picture demand?	Foundation questionnaire	Foundation Decision, part one
5 The Stack	What codebase best supports that foundation?	Stack rubric	Foundation Decision, part two
6 The Wedge	What ships first, and how will we know it worked?	Wedge Brief template	Wedge Brief
Exit	Is all of this true enough to build on?	The grill, then ratification	Filed decisions, Phase Two opens

Stages 1 and 2 overlap: the first conversation produces the questions the research answers, and the research produces the questions for the second conversation. Stages 4 and 5 are one decision written in two parts. Everything else runs in order.

Stage 0: Intake

One session with the sponsor, before the domain expert is interviewed. The answers set the constraints every later stage works inside. Record the answers verbatim; do not paraphrase a constraint.

1. Who is the sponsor, and what is their relationship to the domain? Do they work in it, sell into it, or know someone who does?
2. What is the business reason for building this now? What changes if it exists in ninety days, and what changes if it does not?
3. Who pays for the result, and what do they pay for today instead? Name the budget line it would replace.
4. Who is the domain expert, what is their role in the domain, and how many hours can they give over the next ten days?
5. Who ratifies? If the sponsor and the ratifier differ, say how disagreements are settled.
6. What are the hard constraints: money, calendar, confidentiality, jurisdictions, certifications the product itself must hold?
7. What already exists: spreadsheets, an incumbent system, paper, a previous attempt? Can the operator see it?
8. What is off limits before the conversation starts? Anything the sponsor already knows must not be built, touched, stored, or seen.
9. Is this project for revenue, for proof, or for both? The answer changes the stack decision in Stage 5.
10. What would make the sponsor stop the project at day ten? Write the kill condition down now, while nobody is attached to the work.



Stage 1: The conversation

The domain expert is interviewed across two to four sessions using the eight worksheets below. The operator's discipline matters more than the questions.

Rules for the interviewer. Use the expert's words and write them down exactly; the glossary is built from these sessions and a paraphrase poisons it. Ask for the thing, not the description of the thing: "show me one" beats "tell me about them" every time, so ask for a real record, a real form, a real report, with names redacted if needed. Never propose a solution during Stage 1; a solution offered too early becomes the expert's answer to every later question. When the expert says "it depends," ask what it depends on and write down each branch; those branches are the workflow. When the expert says "everyone knows that," write it down anyway; that sentence marks the knowledge the system will have to hold explicitly. Record the sessions if the expert agrees, and keep the transcript in the vault beside the worksheets.

1A: Personas and their days

For each kind of person who will touch the product, including the ones who only read from it:

1. What is the role called, in this domain's own words? What is it called on the org chart, if that differs?
2. Walk me through yesterday, from arrival to leaving. What did they do first, what interrupted them, what did they do last?
3. What are the three questions they need answered the moment they sit down? Which of those do they answer today from memory, and which from a system?
4. What do they produce in a day, and who receives it?
5. What do they wait on, and who is waiting on them?
6. What are they blamed for when it goes wrong? What are they praised for?
7. How many of them are there in a typical organization, and in the largest one you know?
8. What device and screen do they use, and where: a desk, a bench, a vehicle, a phone?
9. What do they hate about the current tool, and what would they refuse to give up from it?

1B: The records

The nouns. For every kind of record the domain keeps, whether on paper, in a spreadsheet, or in a system:

1. What is it called? Is there a second name for it used by customers, regulators, or vendors?
2. Who creates it, and what event causes it to exist?
3. What states does it pass through, in order, and what moves it from one to the next?
4. What makes it final? Is there a moment after which it must never change? Who has the authority to reach that moment, and how do they show it: a signature, a stamp, a login, a number?
5. Is it numbered? By what rule? What happens to a number that was issued and then voided?
6. Who reads it, and who must never read it?
7. What other records does it point at, and which of those must exist before it can?
8. How long must it be kept, and who says so?
9. Is it ever sent outside the organization? To whom, in what format, and does the recipient verify it?



10. What is the worst thing that happens if this record is wrong, missing, or altered? Answer in consequences, not adjectives: a fine, a lost accreditation, a lawsuit, a death.

Question ten is the most important question in Discovery. The record with the worst answer is the spine, and the whole system is designed outward from it.

The spine test. Every domain has one or two records where being wrong is catastrophic rather than inconvenient. In a calibration lab it is the certificate: a sealed document a customer relies on and an assessor audits. Find the equivalent before designing anything else. The spine decides what must be append-only, what must be signed, what must be numbered, what must be verifiable from outside, and what the audit trail must prove. Everything else in the system can be ordinary; the spine cannot.

1C: The workflows

The verbs. For each thing the organization does repeatedly, from the moment it starts to the moment it is done:

1. What is the trigger? A customer call, a date, a delivery, a form?
2. What are the steps, in the order they actually happen, not the order the procedure says?
3. Where are the decision points, who decides, and what do they decide with?
4. What must be approved or signed, by whom, and what does the signature attest to?
5. What is produced at the end, and who receives it?
6. How long does it take from trigger to done, on a good day and on a bad one?
7. Where does it stall, and why? Waiting on a person, a customer, a part, an approval?
8. What does "done" mean, exactly? Who can say a thing is done, and can it be undone?
9. What are the exceptions, and how are they handled today? An exception that has a name is a workflow of its own.
10. Which steps does the expert consider the real work, and which are overhead they would gladly lose?

1D: The standards that bind

Every domain worth building for is audited by someone. Find out by whom, and what they ask for.

1. Who audits or inspects this organization? Accreditation bodies, regulators, customers, insurers, the state?
2. How often, and what does an audit day look like? Who sits in the room, what do they ask to see first?
3. Which standards, clauses, or rules apply? Get the document names and the clause numbers, not the general idea.
4. For each clause, which records prove compliance? This mapping is the assessor pack, and it is the most sellable artifact the product can produce.
5. What does a finding cost? Time, money, a suspended scope, a lost contract?
6. What is the organization required to report, to whom, and how often? Get a copy of the last report.
7. Are there certifications the people must hold, with expiry dates? Who tracks them today?
8. Are there rules about where data may live, who may see it, and how long it must be kept? Name the rule.



1E: The numbers that matter

1. For each persona in 1A, what are the three numbers they look at, and how often?
2. What does the board, the owner, or the director ask for each month? Each quarter? Get the actual deck or report.
3. For each number: how is it defined, exactly? Who computes it today, from what source, and does anyone dispute it?
4. Which numbers are compared to a target, and where does the target come from?
5. Which number, if it moved by ten percent, would change a decision? That number belongs on the first dashboard.
6. What number do they wish they had and cannot get today?

1F: The vocabulary

Kept from the first session onward, as a three-column table: the domain's term, the plain-language meaning, and the words never to use because they mean something else here. Every record name from 1B, every role from 1A, every state from 1C, every standard from 1D. The product's labels, its navigation, its documents, and its shipped reports will use column one. The engineering process will use column two. If a term has two meanings in the domain, record both and let the expert pick the one the product will use.

1G: What is off limits

1. What must this product never do? A function that is regulated, licensed, or life-critical and belongs to someone else's system.
2. What data must never enter it? Regulated categories, data the organization is not permitted to hold, data that would change the product's compliance burden.
3. Who must never see what? Roles, customers, the public, the vendor.
4. What integrations are forbidden or require certification the project does not have?
5. Where does liability sit if the product is wrong, and does building it move that liability?

1H: The incumbents and the switch

1. What system, spreadsheet, or paper does the organization use today for each record in 1B?
2. What do they pay for it, what contract binds them, and when does it end?
3. What would they lose by switching that they would actually miss? Be specific; "everything is in there" is not an answer.
4. Can the current system export its data? In what format? Get a sample export, redacted.
5. Who decides to switch, who has to agree, and what does the purchase process look like: a card, a purchase order, a procurement cycle, a board vote?
6. Who else in the industry has switched systems recently, and how did it go?

Stage 2: Research

The operator's work between conversations. Every claim in the Research Brief cites a source and carries a tag: observed (read from the primary document), claimed (said by the expert or a vendor), or inferred (the



operator's conclusion). Nothing enters the brief from memory without a source; the tag is what lets the grill tell knowledge from confidence.

Outside the organization:

1. The standards named in 1D, read in full where they are available, with the clause numbers that touch the records in 1B.
2. The regulators and accreditation bodies: what they publish, their directories of accredited or licensed organizations (this is the lead list), their assessment schedules.
3. The incumbents named in 1H: their feature lists, their pricing where public, their contract terms, their export formats, their user complaints in public forums.
4. The integrations the domain expects: what systems must this product talk to, in what formats, through what interfaces, and whether those interfaces are open, licensed, or closed.
5. Procurement: how organizations in this domain buy software, what a typical sales cycle looks like, and what paperwork a buyer will demand (security questionnaires, insurance, certifications).
6. The associations and conferences where the personas in 1A actually gather, and the publications they read.
7. Adjacent domains that share the same records or workflows; a product that fits two domains at once has a larger substrate.

Inside the organization:

1. Sample data from the current system, redacted: row counts per record type, growth over the last year, the largest single organization's volumes.
2. The documents the organization produces: every template, form, report, and certificate, with the fields they carry.
3. The org chart, the actual one, with who reports to whom and who approves what.
4. The last audit report, the last board report, and the current procedure manual.

Stage 3: The System Picture

One document, ten pages, written by the operator from the Domain Pack and the Research Brief. It is the object the grill examines and the ratifier signs, and it is what Phase Two builds from. Its sections, in order:

1. **The domain in one paragraph.** What the organization does, for whom, under whose rules.
2. **Personas and their surfaces.** Each persona from 1A, their three questions, and the surface that answers them.
3. **The record map.** Every record from 1B as a subject: its lifecycle states, its numbering rule, what it references, who may read and write it, its retention. Mark the spine.
4. **The spine, in detail.** What makes the spine record final, who may make it so, how it is signed or sealed, how it is verified from outside, what must be append-only around it, and what the audit trail must be able to prove. This section is the one that cannot be wrong.
5. **The workflows.** Each workflow from 1C as a sequence of states across subjects, with its decision points, approvals, and exceptions.



6. **The trust map.** Who sees what. Tenancy: is each organization an isolated tenant, and is there any legitimate reason for data to cross between tenants (a shared catalog, a customer portal, a regulator's view)? Roles within a tenant. The public face, if any.
7. **The standards map.** Each clause from 1D against the records that prove it. This becomes the assessor pack and the audit-readiness dashboard.
8. **The integrations map.** Every external system, the direction of data, the format, and whether the interface is open.
9. **The numbers.** Each number from 1E with its definition and its source of truth, assigned to the surface that shows it.
10. **Volumes and growth.** Tenants, users per tenant, records per subject, documents produced, expected growth, and the largest plausible single tenant.
11. **Non-functionals.** Availability the domain tolerates, data residency, retention, backup and restore expectations, the audit trail, accessibility, the devices and screens from 1A.
12. **Off limits.** From 1G, restated as constraints the design must honor.
13. **Open questions.** Everything the operator could not settle, each with who can settle it.

The grill reads the picture against the worksheets and the brief, and rules on each section: holds, holds with a correction, or does not hold. A section that does not hold sends the operator back to the expert with a specific question, not to their own judgment.

Stage 4: The Foundation

The picture is mapped onto infrastructure. Each question below has a default, which is the answer Standard already embodies, and a condition under which the default is wrong. The operator answers each one from the picture, and the answer is a decision, not a preference.

Question	Default	The default is wrong when
Tenancy	Shared schema, one tenant column on every table, fail-closed row-level security, composite keys	A regulator or customer requires physical separation; then that tenant gets its own cell of the same image
Isolation proof	Leak tests per module, a fence that every tenant table ships its wall	Never; the proof is not optional
Identity	Hosted SSO with organization mapping; roles and privileges inside the tenant	The domain mandates a specific identity provider or certification, which becomes a requirement rather than a change
The spine's mechanics	Append-only tables with revoked update and delete, mint-only doors, a signed and hash-chained seal, a public verification face keyed by high-entropy token	The domain's spine has no external verifier; then verification is internal but the append-only walls stay
Documents	Server-side rendering to PDF from templates, content-addressed blob storage, rendering off the request path on a job row	The domain needs a document format other than PDF, or documents must be signed by an external authority
Background	Postgres-backed job rows with skip-locked	Throughput exceeds what one database



Question	Default	The default is wrong when
work	claims, retries with backoff, one worker per family, advisory locks per job family	comfortably queues, which is far beyond any first-year volume
Real-time	Polling and invalidation; an in-process broker for chat on a single node; a documented trigger for a relay	The domain needs sub-second fan-out across many nodes on day one
Imports and exports	Chunked imports on job rows above a threshold; a complete tenant export in open formats as a promise	The domain's incumbent exports in a format that needs a bespoke parser; budget it
Search	Register search per subject on identity handles; a lookup by number across the spine's subjects	The domain is search-first (a knowledge base); then a search index is part of the foundation
Compliance	Security page, structured logs with tenant and request id, audit trail, backups restored on a schedule	The domain mandates a framework (a criminal-justice standard, a health standard, a federal one); then the framework's controls are foundation work, not later work
Residency	One region, chosen by the first customers	A jurisdiction requires data to stay in-country; then region is per cell
Availability	A single task with fast restart, honest uptime reporting, a walked rollback	The domain cannot tolerate minutes of downtime; then multiple tasks, which requires the real-time relay first
Observability	Metrics, traces, health endpoints, alarms on the walks that must run	Never
Cost envelope	A small managed database, one container, object storage, the managed identity provider	The picture's volumes exceed the small class in year one

The output is a table of decisions with reasons, and a list of the foundation items that are new for this domain, because those are the items the substrate does not already provide.

Stage 5: The stack

Three paths, and a rubric for choosing. The choice is made against the picture, not against taste.

Path A, a pack on Standard. The domain becomes a module and a pack manifest on the existing platform: its subjects, seeds, vocabulary, numbering, hubs, decks, and checklist, gated per tenant. Fastest, because the substrate supplies tenancy, identity, the shell, the register, documents, jobs, mail, and the audit trail. Right when most of the record map is already a Standard subject or a close cousin.

Path B, a new build on the substrate. A new repository cut from the substrate template, with the domain built on top. Right when the record map barely overlaps but the foundation decisions match, when a separate team will run it, or when the product must be sold or owned separately.



Path C, a greenfield stack. New foundation, chosen for the domain. Right only when a foundation decision in Stage 4 is one the substrate cannot meet: a mandated identity provider the substrate cannot integrate, a document format it cannot render, a compliance framework whose controls contradict its shape, or a search-first product.

Score each path on the picture:

Criterion	Weight	Path A scores high when	Path C scores high when
Record overlap	High	Most subjects in the record map exist in Standard or map to one	Almost none do
Spine mechanics	High	The spine needs sealing, numbering, append-only walls, and verification the substrate already has	The spine's mechanics are foreign to the substrate
Foundation fit	High	Every Stage 4 default holds	A default is wrong in a way the substrate cannot absorb
Who runs it	Medium	The same operator and the same repository	A separate team, a separate owner, a separate sale
Purpose	Medium	Revenue: the fastest path to a paying customer	Proof: the point is to show the process builds from nothing
Compliance	Medium	The domain's compliance burden is met by the security page and the audit trail	The domain mandates a framework with controls the substrate lacks

The stack default, for Path B or C, is the one Standard proved: Kotlin and Spring Boot, hand-written SQL against Postgres with row-level security, Flyway, a React single-page application on TypeScript with a query library and a token-driven component layer. The reasons hold across domains: the choices are boring, the fences already exist for them, the isolation model is proven by audit, and the process has fifty-three days of rulings about how to build on them. Deviate only when a Stage 4 answer forces it, and write the deviation as a decision with the forcing answer cited.

Stage 6: The Wedge Brief

One page. The first product, and how we will know it worked.

1. **The persona** it serves, from 1A, and the three questions it answers above the fold.
2. **The wedge**, one sentence: the single pain this product removes first. If the sentence has an "and," it is two wedges.
3. **In scope for thirty days**: the subjects, workflows, surfaces, documents, and numbers that ship. Named, not described.
4. **Explicitly out**: what a reasonable person would expect and will not get, so nobody builds it by accident.
5. **The design partner**: the organization that will use it first, what they receive, what they owe (weekly feedback, a reference, a conversion price).



6. **The first measurable**: the one number that proves the wedge worked, how it is measured, and what it reads today without the product.
7. **The acceptance audit**: what the whole-system audit at day thirty checks, written before the build. Isolation, the spine's walls, the fences watched biting, the documents rendering, the persona's three questions answered on a real screen at the minimum viewport, the export working, the design partner's number moving.
8. **The kill condition**, carried from Stage 0.

Exit: the grill and ratification

The System Picture, the Foundation Decision, and the Wedge Brief go to the grill together. The grill is adversarial by design: its job is to find the section that does not hold, the default that was accepted without checking the condition, the record whose worst case was not asked, the vocabulary term with two meanings. Its rulings are appended to each document. The operator resolves each ruling by going back to the expert or the source, never by adjusting the text to make the ruling go away.

The ratifier signs the three documents, with a date. Ratification is a sentence, not a ceremony, and it is the only act in Discovery the process cannot perform for itself.

Filing. Each ratified decision becomes a numbered decision record. The vocabulary table becomes the glossary. The personas and the record map become the first specs, each with a status banner. The open questions become the human-hands queue. The Domain Pack and the Research Brief stay as the evidence behind all of it. Phase Two, the foundation build, opens with the Wedge Brief as its first brief.

Running it with someone else in the seat

Discovery is designed so that the operator can be anyone who follows it, and so that a domain expert with no engineering background can own the truth without touching the machinery. Three things make that hold.

The operator may not know the domain, and should not pretend to. Every question above is answerable by the expert without translation; the operator's job is to write the answers down exactly, to notice when an answer is a branch, and to ask "show me" until there is an artifact on the table. Domain knowledge the operator brings from elsewhere is a liability in Stage 1, because it fills gaps the expert should fill.

The expert never sees a brief they did not help write. The Domain Pack is theirs, in their words; the picture is read back to them section by section before the grill; the wedge is chosen with them. Their ratification of the pack and the picture is what makes the operator's later work legitimate.

Timeboxes are the guard against thoroughness becoming delay. Two to four interview sessions of ninety minutes, three days of research, two days for the picture, one for the foundation and the stack, one for the wedge, one for the grill. Ten days. A Discovery that needs more time needs a narrower wedge.



Appendix: the shape of two answers

How the calibration lab answered, and what the emergency-services answers might be. The second column is Standard's record; the third is a set of hypotheses to be confirmed or replaced by the domain expert in Stage 1, never assumed.

Question	Calibration lab (Standard)	Emergency communications (hypotheses)
The spine	The calibration certificate: sealed, numbered, revisioned, verified by token, audited by the assessor	The quality-assurance review of a call, or the personnel record that proves a telecommunicator was certified when they took it; to be confirmed
Who audits	The accreditation body, against ISO/IEC 17025, on a fixed cycle	The state 911 authority, accreditation programs, and the agencies served; against association standards for quality assurance and training; to be confirmed
Off limits	Nothing life-critical; the lab's customers' data stays inside the tenant	The call path itself: call handling, dispatch, location services, anything between a caller and a dispatcher; and any data governed by criminal-justice rules
The wedge candidates	Audit readiness; turnaround visibility with a customer portal	Quality assurance and certification currency; twenty-four-hour scheduling and overtime; to be chosen by the expert
The customer's word	Quote, not estimate; standard, not reference; cal due	To be captured; do not guess
Record overlap with Standard	Everything, by definition	People, roles, schedule, time, certifications (new), reviews (new), incidents and corrective actions, procedures, equipment, imports, reports, dashboards
Likely path	Path A is the platform itself	Path A, a pack, unless the spine's mechanics or a compliance framework forces otherwise