

A governed operating layer for work that crosses functions

The design of the Enterprise Technology Accelerator (ETA): how it measures cross-functional work, keeps AI agents inside written rules, and what we have not yet proven

**Jack Dinneen**

Chief Product and Technology Officer

**Andrew Nolan**

Chief Executive Officer

CONTENTS

Abstract

Key findings

1. The problem: agents arrived, the silos stayed

2. What ETA is, and what it is not

3. Five rules that do not change

4. Architecture: a work plane and a control plane

5. The harness as the unit of governed work

6. The work ledger: instrument forward

7. Governed autonomy: earned per step class

8. Build, buy or ride

9. How an engagement runs

10. Value that survives a finance review

11. Security, assurance and regulation

12. What we do not claim yet

13. Next steps for pilot partners

Figures

References

About the authors

ABSTRACT

Enterprises that have bought AI agents still run cross-functional work by having people carry it between systems, and few can say how long that work takes or whether AI spend changed it. The Enterprise Technology Accelerator (ETA) is Anitech's design for the missing layer: an additive operating layer in the client's own cloud that measures a workflow before changing it and lets agents act only as far as they have earned. This paper is the design companion to our point-of-view paper on ANI. It sets out five design rules, the architecture, the work ledger, four-tier governed autonomy and how an engagement runs. ETA is a design. No ETA pilot has run, and we are looking for first partners.

KEY FINDINGS

- 1 Work that crosses sales, finance and delivery is still carried by people, and nobody owns the whole workflow or its cycle time. Agents bought per function leave that gap where it was.
- 2 Measurement is cheapest as a by-product. If the harness runs the work, the record of who did what, when and at what cost is written as it happens. Rebuilding it later from other systems' logs is slower and, for work done in inboxes, often impossible.
- 3 Five rules are permanent: it runs in the client's cloud, data is mapped and not copied, reading is separated from acting, a person commits, and everything is recorded. The fourth limits the value case to cycle time, error reduction and released capacity.
- 4 Autonomy is earned per step class, never granted per agent, through four tiers with written promotion criteria and automatic demotion. An override rate that falls to zero can mean the reviewer stopped reading, so we treat it as an alarm.
- 5 ETA adds to what a client owns. Anitech builds the meaning content, long-tail connectors, instrumentation and ledger, delivery method and control plane, and rides the client's own platform for the rest.
- 6 No ETA pilot has run. The hardest technical question, following one case across systems that share no key, is unanswered in production. A candidate workflow must pass four tests before we start.

1 The problem: agents arrived, the silos stayed

Software has been bought for each function, and in the last two years agents have been bought for each function as well. The functions still hand work to each other through people. A deal is signed, finance sets up the customer, delivery mobilises, an invoice goes out, and at every hand-off a person reads a message and re-keys it. Each step is fast. The work as a whole is slow, and nobody can say how slow.

The maturity ladder

We think many enterprises sit on the second rung of a four-layer ladder and are buying the third.

1. Systems of record, one set per function: customer relationship management in sales, accounting in finance, a people system in HR, a work-management tool in delivery.
2. Point-to-point automation. Fields move between tools through integration platforms, scripts and low-code workflows. Data moves, and meaning does not move with it.
3. An assistant or agent per function. None of them share context, so people still connect them.
4. A governed operating layer: one shared meaning of the entities that matter, one set of rules about what may be done to them, and one record of the work.

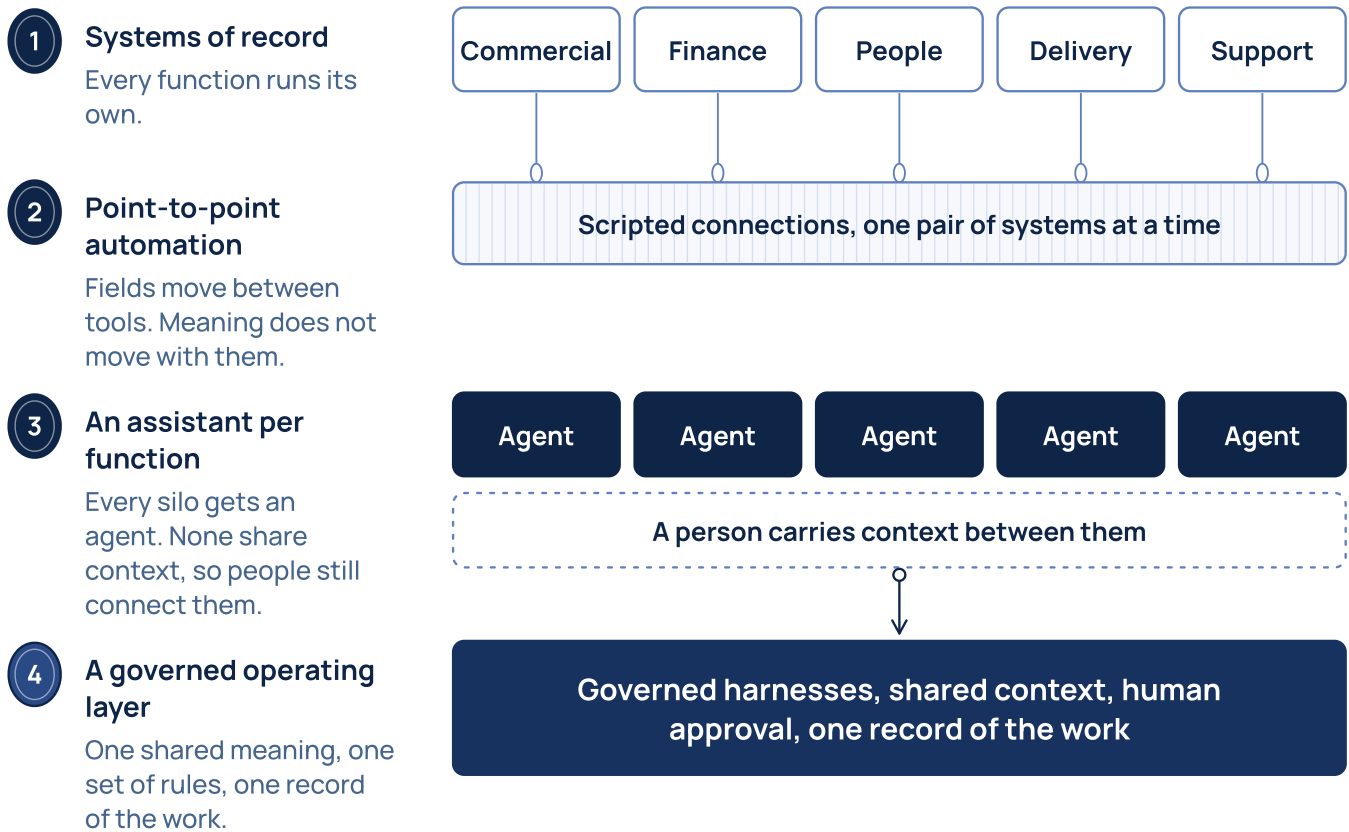


Figure 1. The maturity ladder. Layers one to three are bought function by function. The fourth, a governed operating layer with shared meaning, shared rules and one record of the work, is the one ETA is designed to supply.

Layer three is where we would expect disappointment, and not because the agents are poor. A finance agent can match invoices well and still not know that sales amended the contract last week.

One workflow, three functions, nobody owns it

Follow one piece of work. A deal is signed, and that one commitment has to become a customer record, a purchase order, a project, a team, an invoice and eventually cash. Commercial owns the signature, finance owns the set-up and the invoice, delivery owns the mobilisation. Each has good tools and, increasingly, a good agent. Every hand-off between them is a message to a person, and the cost of an error is paid later and elsewhere, which makes it hard to trace.

Ask who owns the signature-to-cash workflow end to end and, in most organisations, the answer is nobody. People own the machinery. Almost nobody owns the number.

Three deficits, none of them a model problem

- No shared meaning. Sales calls it an opportunity, finance a customer, delivery a project. The same thing has three identifiers and three spellings, and no system holds the link.

A governed operating layer for work that crosses functions

- No governed action. Most deployments have credentials and optimism. It is unclear what an agent may write unattended, what needs approval, what is logged and how a mistake is reversed.
- No evidence. Nobody can say what the agents changed, because nobody measured the position before they arrived.

Each is an operating-model problem presenting as a technology problem. Better models close none of them. A board is entitled to ask what an outcome costs in human hours and elapsed days, and whether two years of AI spend changed that number. Systems hold artefacts such as opportunities, invoices and tickets, and nothing holds the process. Elapsed time lives in the queue after one person finishes and before the next starts, where no single system sees it.

How much time can sit in the gaps

We have one measured example, from software delivery and not from a business workflow. It comes from a snapshot, taken on 24 September 2026, of one client's work-tracking history: 587 timed stories, out of 1,244 in that history. The median timed story took about 18.5 days to get from leaving Open to Client Demo. About 47% of story days were spent in statuses we classed as waiting, meaning ready for development, blocked, or finished and waiting for the next stage. Roughly two thirds of that waiting was finished work waiting for the next person. Separately, 212 of the 587 stories (36%) re-entered a status they had already left. That 36% is a share of stories, not of time.

WHAT THIS EXAMPLE DOES NOT CLAIM

It is one client's software delivery history, where the data is unusually good. It says nothing about finance, sales or HR workflows, and we have no measured baseline for any cross-functional business workflow yet. The waiting classification is ours. It shows that waiting can be large and invisible. It is not a benchmark for ETA.

2 What ETA is, and what it is not

ETA (Enterprise Technology Accelerator)

An agentic operations layer that runs inside the client's own cloud, puts a governed harness into each function, lets work move between them with people in control, and records what it changed.

For a finance reader: ETA is a phased programme that instruments how work happens across an enterprise's systems, shows what it costs, and then removes the manual steps it has measured, on the platform the client already owns. The order is the product, measurement first and automation second. A client can stop after measurement and still hold a number nobody had before.

What ETA is not

- Not a data platform or a data lake. ETA keeps no copy of the client's data. It maps onto the systems the client has and reads from them for the length of a task.
- Not a replacement for any system of record. Those systems stay, and remain where decisions are committed.
- Not an integration platform. Clients already pay for data movement, and we would use theirs.
- Not a headcount programme. Rule four, below, explains why.
- Not, today, a shipped product. It is a design we want to prove with partners.

Next to what a client already owns

Many clients own an assistant platform, a workflow tool and some process analytics, and we expect the objection that they already pay for that. If a client can already say how many people touch a deal from signature to cash, and how long it takes, with numbers they trust, they may not need us for measurement. If not, the answer is usually spread across systems that share no case identity, and that is the gap ETA targets.

The claimed gap is reachability, not technology. Large platforms are strong inside their own estate and weaker at the hand-off between one vendor's system and another's. A smaller firm can cover the long tail of connectors and do the change management that gets a pilot into production.

3 Five rules that do not change

We hold five design rules as permanent. They answer most security and architecture questions before they are asked, and we would not trade them away to win a deal.

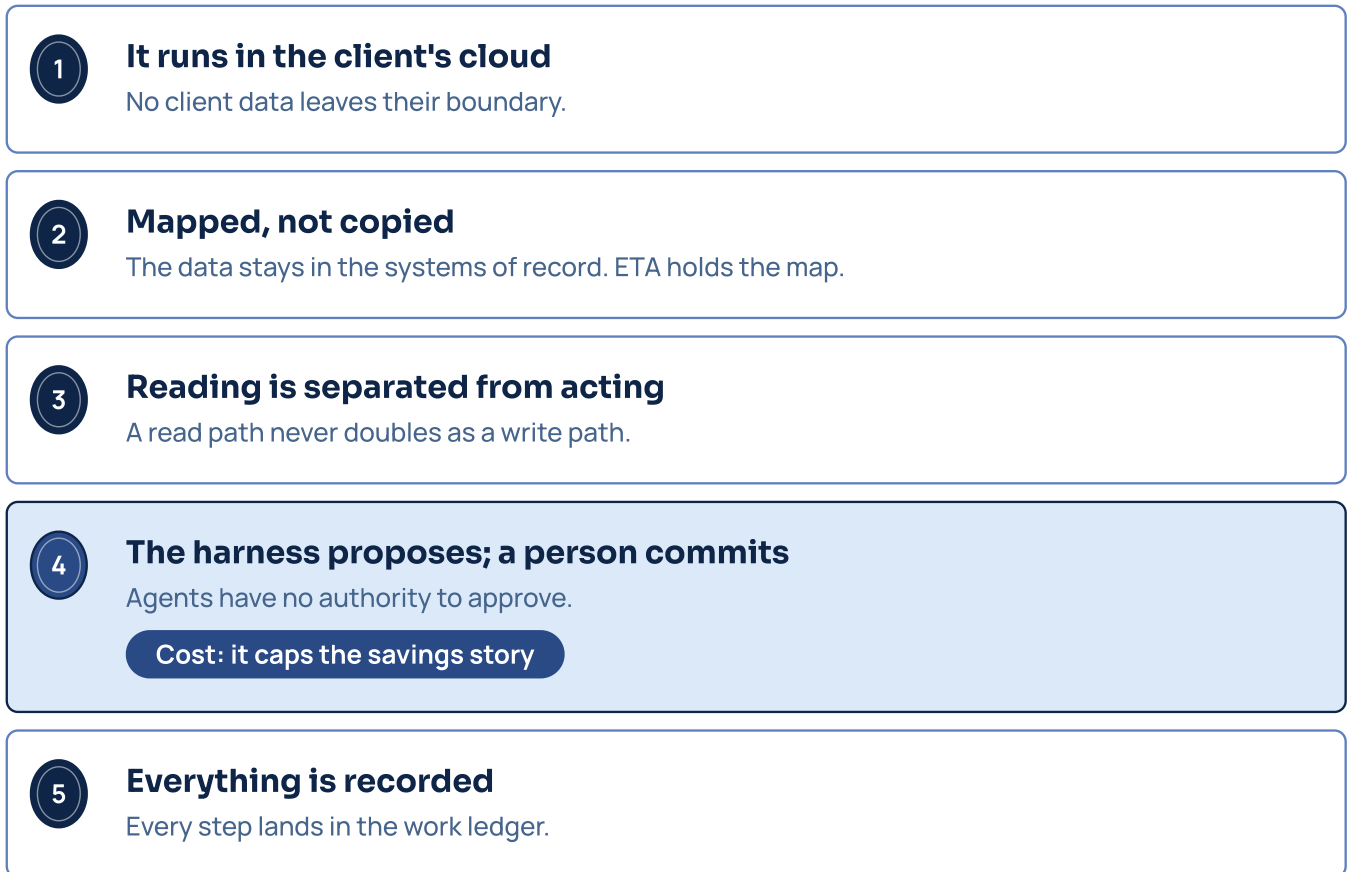


Figure 2. The five permanent design rules. Each removes a class of finding that a security, data protection or finance reviewer would otherwise raise.

Rule one: it runs in the client's cloud

The work runs in the client's own tenancy, under the client's identity and policy controls. Anitech holds metadata only: harness versions, autonomy tiers, and evidence packs the client chooses to share. No payloads, prompts or business data cross. A chief information security officer's first question is where the data goes, and the strongest answer is that it does not go anywhere.

Rule two: mapped, not copied

A common reflex is to land everything in one place and then query it. That creates a second copy to secure, keep in step and reconcile. ETA instead holds a description of the entities and links one workflow needs, pointed at the systems where the information already lives. The description is small and can be reviewed. The data stays where its owner put it.

Why meaning matters is visible in public research. In the Spider 2.0 benchmark, a code agent that solved 91.2% of the tasks in Spider 1.0 solved 21.3% of those drawn from real enterprise workflows (Lei and others, ICLR 2025, reference 1). It is a 2024 benchmark and models have improved since, so it shows how hard real enterprise schemas are, not how agents perform today. Our working belief, which we have yet to test, is that a defined model of the business helps an agent more than raw tables do.

Rule three: reading is separated from acting

The dangerous combination for an agent is untrusted input, access to sensitive data and a way to send something out, all in one session. An attacker who can plant instructions in an email the agent reads can then reach the data and carry it away.

We make that combination impossible and do not try to prompt our way around it. In ETA the agents that read have no write path to a system of record. The agents that act hold narrow, scoped credentials and see only what their step needs. We never describe prompt injection as mitigated. The design assumes an injection will succeed against a model and ensures it cannot reach anything that matters. Section 4 shows the two paths.

Rule four: the harness proposes and a person commits

In the first two autonomy tiers, every change to a system of record is committed by a named person. In later tiers the commitment can be given in advance to a boundary, for a class of step, and is recorded. Some step classes never leave the first tiers. Moving money and anything that decides something about a person are the clearest cases.

The rule has a cost, and it shapes the value case. If a person commits, the saving cannot be that the person disappears. What the harness saves is the time between steps, the errors from re-keying, and the hours spent on preparation rather than judgement. The case ETA supports is cycle time, error reduction and released capacity. We would not put a headcount number in a contract.

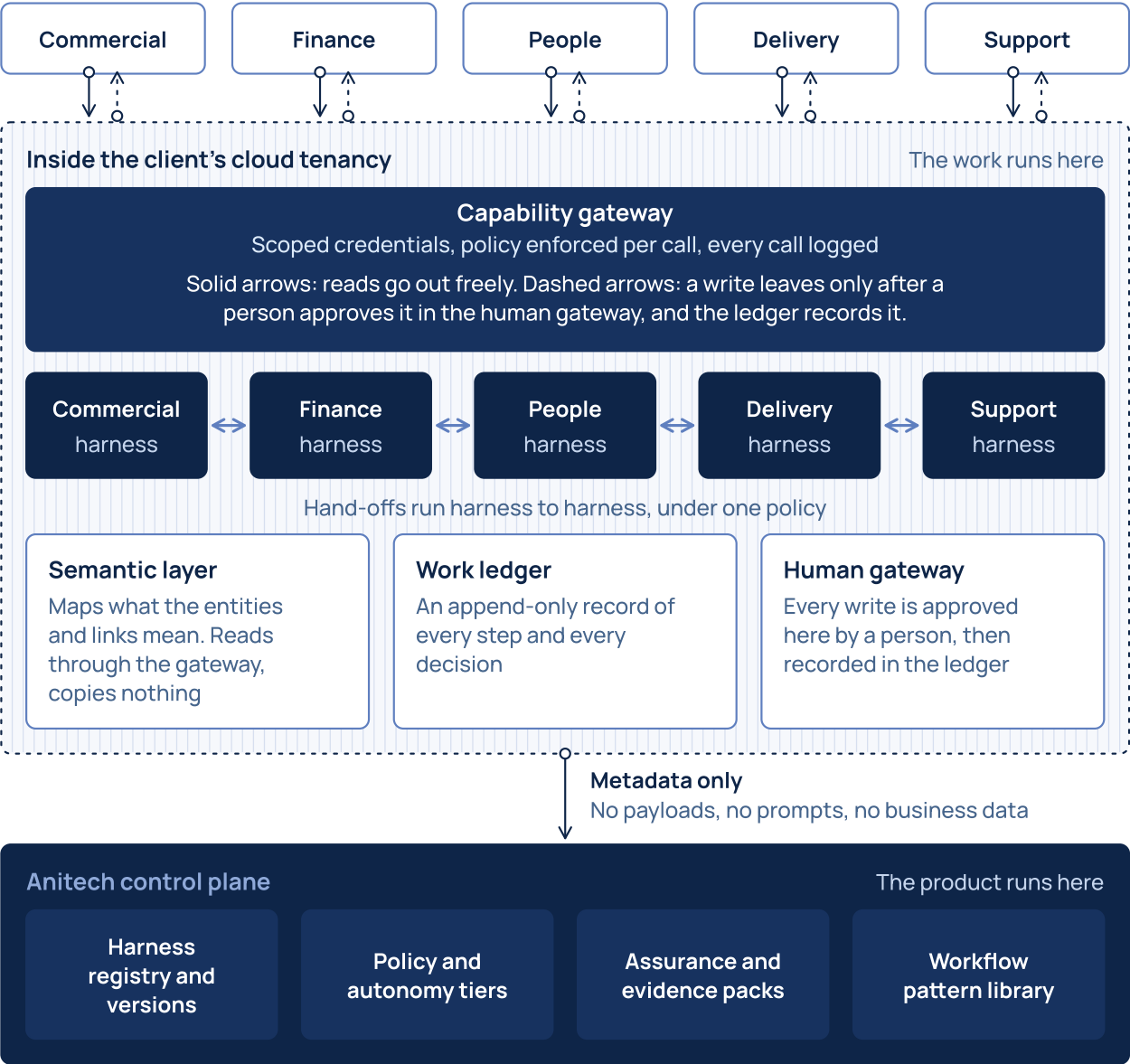
Rule five: everything is recorded

Every step an agent takes and every decision a person makes is written to an append-only record. A harness that acts without one cannot be audited, cannot be improved and cannot show a board what changed. The record is also the measurement, and section 6 describes it.

4 Architecture: a work plane and a control plane

ETA has two planes. The work plane runs in the client's cloud and does the work. The control plane runs in Anitech's cloud and governs the harnesses, holding metadata only. Inside the work plane, rule three shows up as two separate paths: a read path and an act path.

Their systems of record



The split is the product decision: deployed software on one side, what amortises across every client on the other.

Figure 3. The work plane and the control plane. Inside the client's tenancy, function harnesses read the systems of record through the capability gateway, using the meaning mapped in the semantic layer, and have no write path. Any write goes through the human gateway, where a named person or a pre-approved boundary approves it, and is recorded in the work ledger. Only metadata crosses to the Anitech control plane.

Inside the client's cloud

Function harness

The unit of deployment, one per function: a goal, tools, written rules and a way of verifying its own output, configured by the team that owns the function.

Capability gateway

The single door agents use to reach any system. It issues scoped credentials, enforces policy on each call and logs every call.

Semantic layer

The shared meaning: the entities, links and actions a workflow needs, mapped onto the client's systems and never copied from them. It covers one workflow at a time, not the whole company.

Human gateway

The approval queue, escalation path and exception desk. Every write passes through it. It is built against rubber-stamping, as section 7 explains.

Work ledger

An append-only record of every step an agent takes and every decision a person makes. Section 6 describes it.

The read path and the act path

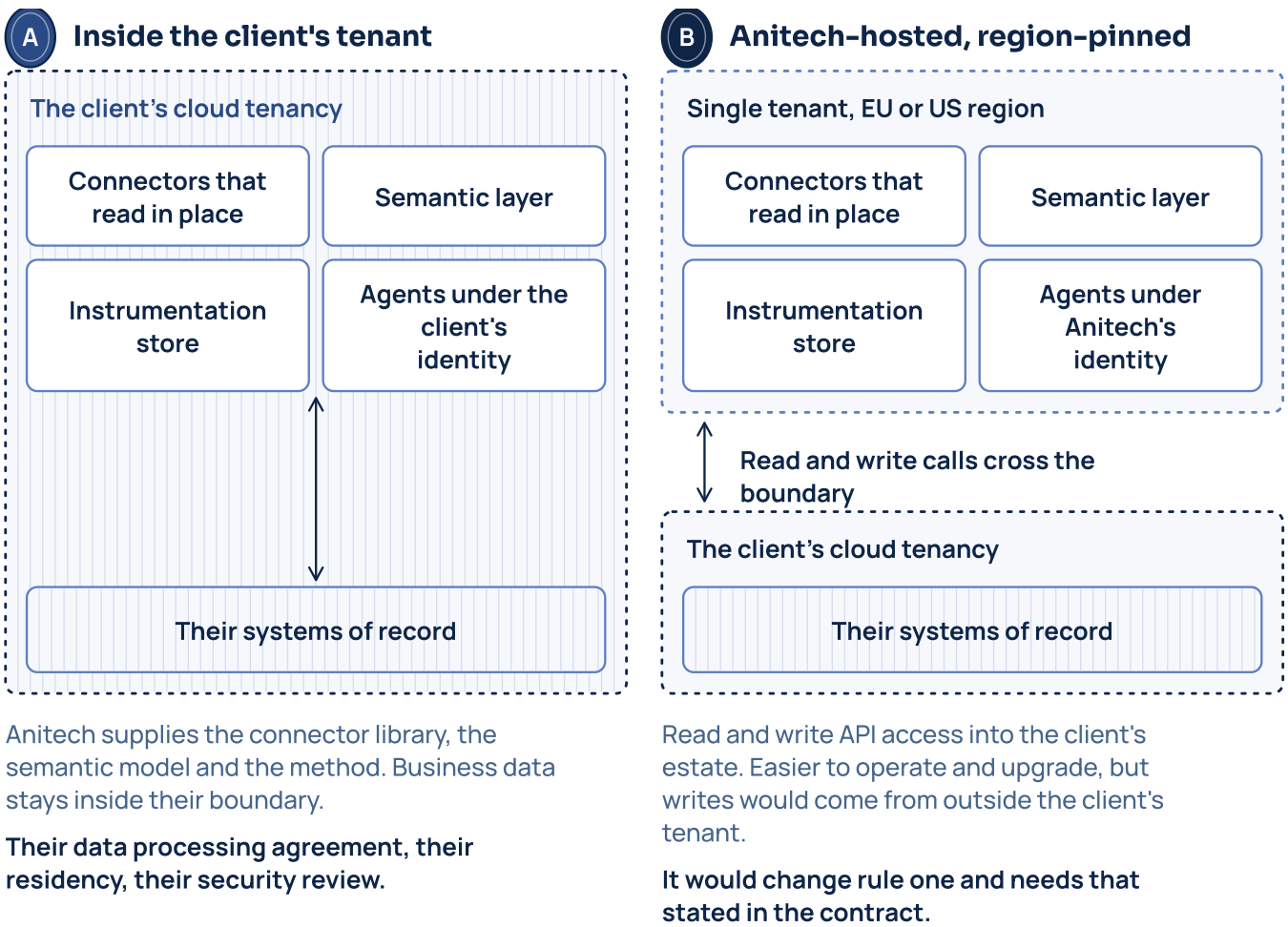
- Read path. Agents that research a case or draft a recommendation read widely through the capability gateway. They have no write path to any system of record.
- Act path. A proposal that would change something goes to the human gateway. A named person, or a boundary a named person approved beforehand, commits it, and the commit is scoped to one step and recorded in the ledger.

The control plane, and what crosses to it

The control plane holds the registry of harness versions, the policy, the autonomy tiers and the evidence packs. What crosses to it is information about how the harnesses behave: versions, policy settings, tiers, counts and rates, and evidence packs the client decides to share. The content of the work does not cross. A client should be able to confirm that by inspecting the egress from its own tenancy, and the design depends on it being true.

Hosting: our default, and an option we do not default to

Our default is the design above, with the work plane in the client's tenant. It is the only hosting arrangement consistent with rule one. We considered a second option, a region-pinned environment that Anitech hosts for the client. It is not our default, because it would break rule one: client business data would cross to us. We would offer it only with that change stated in the contract and in the next version of this paper.



A CTO decides it on four questions

- ①

Whose tenant
- ②

Whose data processing agreement and residency
- ③

Which security review
- ④

What happens at contract end

Figure 4. Two hosting topologies. Option A, our default, runs the work plane in the client's own tenant, so rule one holds. Option B, considered and not our default, runs it in a region-pinned environment that Anitech hosts for that client. It would break rule one, so we would offer it only with that change stated in the contract and in the next version of this paper.

Hosting options compared. The right choice for a client depends on its security review, residency requirements and operating capacity.		
Question	Option A: client tenant (default)	Option B: Anitech-hosted, region-pinned
Where does the work run?	In the client's cloud tenancy, under their identity and policy.	In an environment Anitech operates for that client, pinned to an agreed region.
Who is the data processor?	The client keeps its own data. Anitech handles metadata only.	Anitech becomes a processor of client business data, with a data processing agreement and residency commitments.
What does the security review cover?	The client's own controls, plus the harness software and the metadata egress.	Anitech's environment and operating practices as well, which widens the review.
How fast can it start?	Slower. The client supplies a landing zone and identity set-up first.	Faster to stand up, with less for the client to prepare.
Who runs updates and support?	Shared. Anitech needs a controlled way to ship versions without holding standing access.	Anitech, which is simpler to operate and keep consistent across clients.
What happens at contract end?	Little changes: the data was never ours, and the harnesses are removed from the tenant.	Data export and deletion become a contractual and technical exercise.
Effect on the design rules	Rule one holds as written.	Breaks rule one. Offered only with the change stated in the contract and in the next version of this paper.

5 The harness as the unit of governed work

ETA builds on something we already do for software delivery, so it helps to be precise about what a harness is and how far the comparison goes.

Harness

Everything around a model that decides what the agent may do, checks what it did and records the result. It is not a test harness, an agent framework or a prompt collection. Agents execute bounded tasks and the orchestrator owns control.

Our software delivery work has three Harnesses. The Requirements Harness is live with one client. The Engineering Harness is built and going into first client use in October 2026. The Quality Harness is in development. We cite them for how they work, and not as proof that ETA works.

A short list of rules settles every ambiguous call in the Requirements Harness, and the same shape carries over to a business function. A finance harness proposing a customer set-up would cite the contract clause it relied on, have no path to post an entry, and list a missing tax identifier as an open question rather than guess it.

- No uncited claim. Every statement in a draft carries a footnote with a locator and a verbatim excerpt, and a script checks the excerpt is at that locator.
- An agent never approves its own work. A fresh reviewing agent, in a separate read-only context, doubts the draft and cannot edit it.
- Evidence comes from the pipeline, not from the agent's account of what it did.

One delivery result shows governed work running. In a two-week pilot in August 2026 with a North American education-technology client, a two-person pod took 49 user stories to opened merge requests in about 48 elapsed hours, including the human approval gates, after seven working days of setup (reference 4). The figures are self-reported from the pilot's run logs. The merge requests were opened and had not yet been merged, and quality outcomes have not yet been measured.

We draw one cautious lesson. Most of the effort went into setup, encoding the client's standards and approval points into the harness, and we expect the same for ETA, where the semantic layer and the rules are the work.

 NOT AN ETA RESULT

The pilot is a software delivery result from the Harnesses. It says nothing about how an operations harness would perform on a finance or sales workflow. Whether the two share more than a shell is something we need to test.

6 The work ledger: instrument forward

The work ledger is the part of ETA we think is most distinctive, because it changes how measurement is done. To answer how long a cross-functional workflow takes, you need an event record: for each piece of work, which step happened, who did it, when and with what result. There are two ways to get one.

- Reconstruct it afterwards. Pull the logs from each source system, find a common case identifier, stitch the events together and infer the process. This is what process-mining tools do.

- Instrument it forward. If the harness runs the work, the harness writes the event as the work happens. The record is a by-product of doing the work and not a prerequisite for starting.

Reconstruction has a ceiling for this kind of workflow. The logs it needs are often missing: audit trails can be short-lived, limited to certain event types or lacking a case identifier, and work done in an inbox or a chat leaves no event at all. The step nobody has solved reliably is correlation, following one case across systems that share no key.

Instrumenting forward avoids much of that, because the harness assigns the case identity when work enters its scope and carries it through. The trade is that it measures only the work the harness runs. That is why measurement and automation are linked in ETA, and why it is no substitute for a retrospective diagnostic of work that never passes through it.

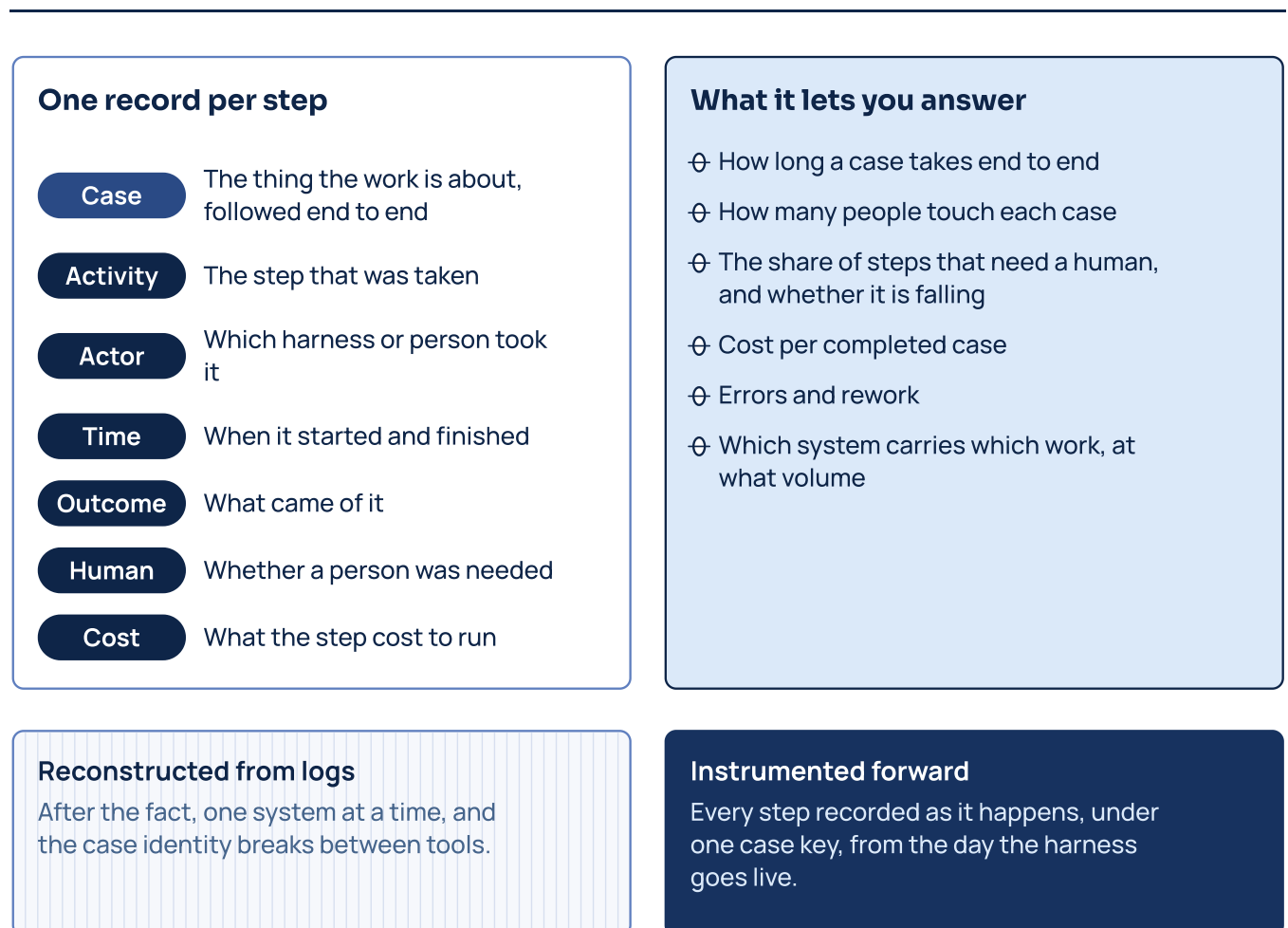


Figure 5. The work ledger. Each step writes one record of seven fields as it happens. Instrumenting forward gives a complete record of the work the harness runs. Reconstructing from other systems' logs is partial, and often misses work done in inboxes.

The event record the ledger writes for every step, and what each field makes it possible to answer.

Field	What it records	What it lets you answer
Case	The unit of work, such as one deal moving from signature to cash.	How long the whole workflow takes, end to end.
Activity	The step that took place.	Where the time goes, step by step.
Actor	Which agent or which named person acted.	How many people touch a case, and who.
Time	When the step started and finished, including waiting.	How much elapsed time is work and how much is queueing.
Outcome	What the step produced, including failures and reversals.	Where errors and rework arise.
Human	Whether a person was needed, and what they decided.	What share of steps still needs a person, and whether it is falling.
Cost	Model spend, compute and reviewer time for the step.	Cost per completed case, counting the whole stack.

What the answer looks like

We have no client data to show, so this example is invented, to show the form of the answer.

ILLUSTRATIVE EXAMPLE, NO CLIENT

A deal is signed. It passes through five systems and four hand-offs, and eleven people touch it before cash arrives. The elapsed time is 19 days. On day 6 the case identity broke: the customer appears under different identifiers in the sales, finance and delivery tools, and someone matched them by hand. On day 15 the work was chased twice. None of this is visible in any one system. All of it is in the ledger, because each hand-off is a step the harness ran.

Illustrative only. The form of table a finance director could ask for. Every figure is invented to show the shape and is not drawn from any client or measurement.

Workflow	Runs a month	Median elapsed	Worst case	People involved
Signature to cash	38	19 days	61 days	11
Requisition to hire	12	34 days	88 days	9
Ticket to resolution	1,410	2.1 days	22 days	4
Supplier to payment	96	27 days	70 days	7
Change to production	54	9 days	31 days	6

7 Governed autonomy: earned per step class

Autonomy is earned per step and not granted per agent. A particular kind of step, such as matching an invoice to a purchase order within a tolerance, earns independence from evidence about how that kind of step has gone. The same agent can be fully supervised on one step class and delegated on another.

Autonomy is earned one step class at a time

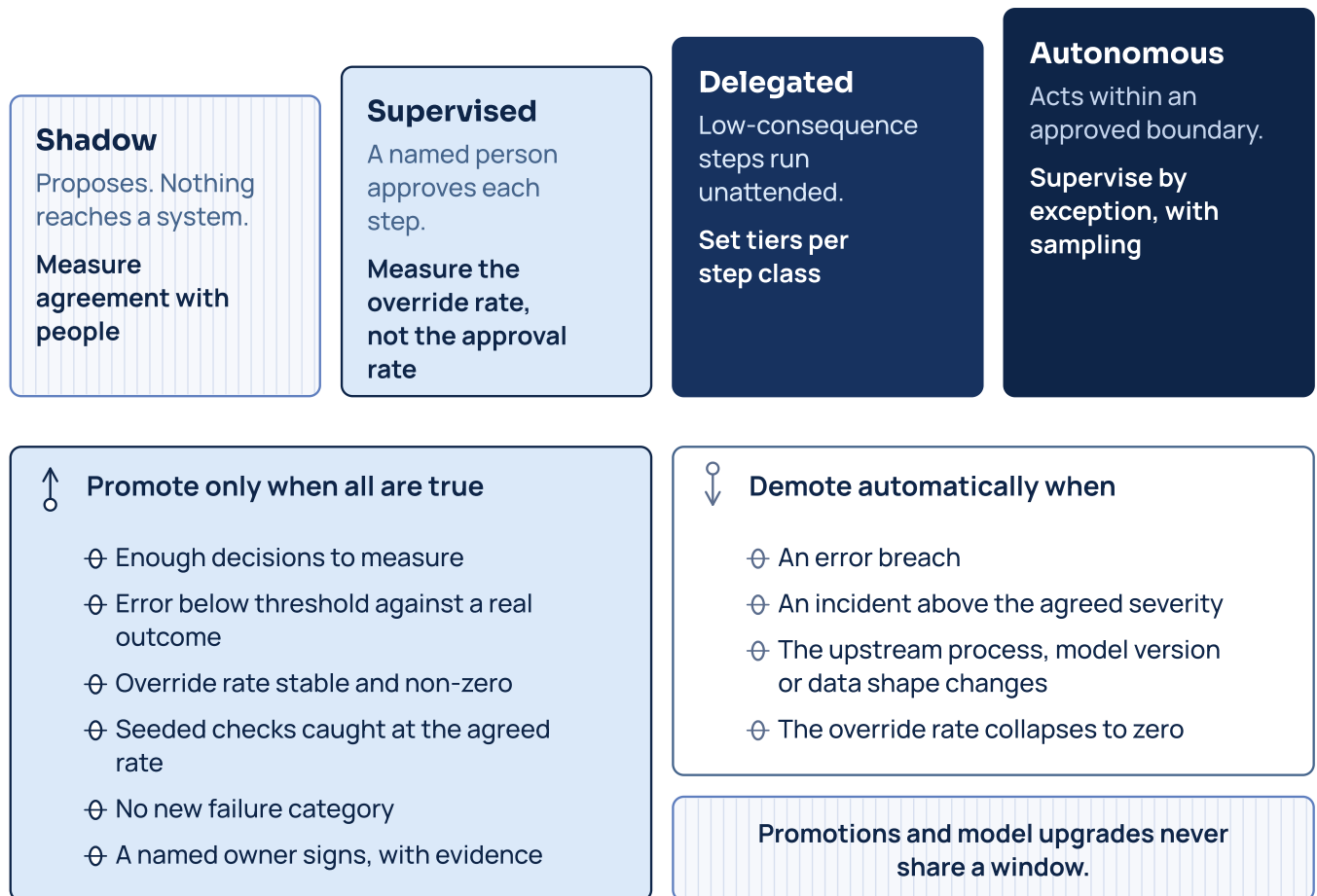


Figure 6. The four autonomy tiers. A step class moves up only when written criteria are all met, and moves down automatically when a demotion trigger fires. Promotions and model upgrades are never made in the same window.

The four autonomy tiers, what each permits, and the measure that governs it. Delegated and Autonomous differ in the width of the boundary and in how much supervision remains.

Tier	What the harness does	What is measured
Shadow	Proposes an action. Nothing reaches a system of record. A person does the work as before.	How often the proposal agrees with what the person actually did.
Supervised	Proposes, and a named person approves each action before it is committed.	The override rate: how often the person changes or rejects the proposal. The approval rate is not the measure.
Delegated	Acts without per-step approval on low-consequence steps, inside a narrow boundary a person approved, with sampled review.	Error measured against a real outcome, by step class.
Autonomous	Acts inside a wider approved boundary, with supervision by exception and sampling.	Exceptions raised, sample error rate and any drift.

Not every step class will reach every tier, and some should not. Anything that decides something about a person, such as hiring, discipline or pay, stays in the first tiers whatever the data says. So does anything that moves money, until the client and Anitech agree in writing what boundary applies.

Promotion: all criteria must hold

- There are enough decisions to measure.
- Error is below the agreed threshold, measured against a real outcome and not the agent's own opinion.
- The override rate is stable and non-zero.
- Deliberately seeded checks are caught at the agreed rate.
- No new category of failure has appeared.
- A named owner signs the promotion, with the evidence attached.

Demotion is automatic

A step class drops a tier, without waiting for a meeting, when any of these happens.

- Error breaches the threshold.
- An incident above an agreed severity occurs.
- Something upstream changes: the process, the model version or the shape of the data.
- The override rate collapses to zero.

The fourth trigger runs against intuition. In a supervised tier a person approves proposals, and a good proposal is usually right. If the person approves nearly everything, a healthy agent and a tired reviewer look identical from outside. A small, stable, non-zero override rate is evidence that the person is still reading. A rate that falls to nothing can mean the reviewer stopped reading, and a control nobody exercises is no longer a control. That is why the human gateway seeds known-bad cases into the queue, measures how many are caught, and treats a collapse in overrides as a demotion event.

We also never promote a step class and upgrade the underlying model in the same window. If both change and the error rate moves, there is no way to tell which change did it. The tiers and criteria are our design, not an industry standard, and we expect partners to push on them.

8 Build, buy or ride

A reasonable early question is why a firm of Anitech's size should build any of this. Our answer is to build as little as possible and be clear which parts are ours: the meaning content, the long-tail connectors, the instrumentation and ledger, the delivery method and the control plane. We ride the client's own platform for everything else.

Build Ours to build Ride Theirs, we ride it Buy or ride Not worth owning		
Delivery method, ontology workshops, adoption	The moat	Build
Ontology and meaning content	The model of the business, per workflow	Build
Instrumentation and the work ledger	The wedge: the gaps between systems	Build
Long-tail connectors	Where the big platforms have none	Build
Anitech control plane	Registry, policy, tiers, evidence packs	Build
Departmental agents	We own the content, not the runtime	Ride
Governance, agent identity, audit	The client platform already provides it	Ride
Rules and workflow execution	The platform's own workflow engine	Ride
Semantic and ontology engine	Their engine, our model	Ride
Integration and data movement	Building it means upkeep and licences	Buy or ride

Anitech builds the method, the meaning content, the instrumentation, the long-tail connectors and its own control plane. It rides the client's platform for the rest and replaces none of its systems of record.

Figure 7. What Anitech builds and what it rides. Anitech builds the meaning content, long-tail connectors, instrumentation and ledger, delivery method and the Anitech control plane. The client's own platform supplies the rest: the agent runtime, identity and governance, workflow execution, the semantic engine, and integration and data movement.

The ETA stack by layer, with the position we take on each and the reason.

Layer	Position	Reason
Meaning content: the entities, links and rules for each function, built in ontology workshops	Build	It is organisational work, hard to copy, and no product release commoditises it.
Long-tail connectors for sector and regional systems	Build	Large platforms cover the mainstream well and the long tail poorly. Reach matters here.
Instrumentation and the ledger, including cost per case	Build	Measuring the gaps between systems is something no single system does. A platform vendor is also poorly placed to show a client what its own agents cost against the work they did.
Delivery method: process redesign, consultation, adoption	Build	It decides whether a pilot reaches production, and it is the part least affected by tooling.
Anitech control plane: registry, policy, autonomy tiers, evidence packs	Build	It is the part Anitech runs. It governs the harnesses and holds metadata only.
Agent runtime and departmental agents	Ride	We supply the content, meaning the rules and prompts for a function, and run on the client's own agent runtime.
Identity, governance and audit for agents	Ride	The client's platform already has these and is better placed to attest them.
Workflow execution and the semantic engine	Ride	These tools exist and are improving quickly. We bring the model of the business and the platform supplies the engine.
Integration and data movement	Buy or ride	Not worth owning. Clients already pay for integration tools, and maintenance outlasts the build.

None of this is defensible intellectual property in the usual sense. It is services with a productised toolkit behind them, which is what Anitech is. Whether ETA runs as a service line or something more separate is undecided.

9 How an engagement runs

An engagement is phased, and each phase ends at a gate. The client can stop at any gate with something of value in hand.

Qualify the workflow first

The first and most consequential decision is which workflow to start with. Tasks of apparently equal difficulty fall on opposite sides of what agents can do reliably, and the failures are fast, confident and plausible. A candidate workflow must pass four tests. Fail any one and it is not a candidate.

1. A ground-truth signal exists. There is a way to know, after the fact, whether the step was done correctly: a reconciled ledger, a customer complaint, an audit finding.
2. It is reversible or bounded. A wrong action can be undone, or the damage is capped.
3. The inputs are retrievable. The information the step needs can be reached through a system and is not locked in someone's head.
4. Consultation is obtainable. The people whose work changes can be consulted, in writing, before the work starts.

Good first candidates are back-office, high volume and already have a trusted signal of what good looks like. Workflows that decide things about people are never the pilot.

The phases

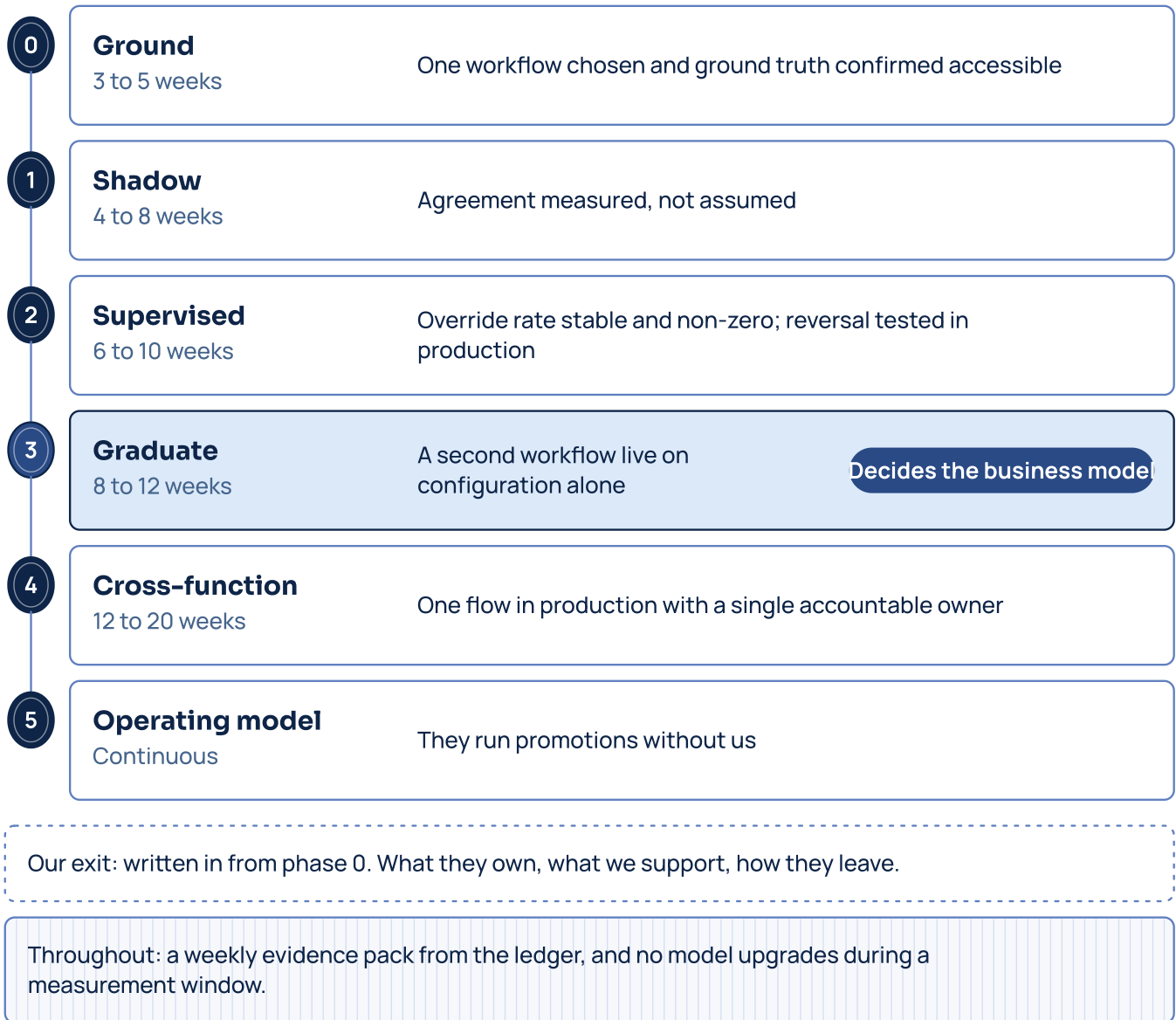


Figure 8. Phased delivery with gates. Each phase ends at an exit gate, listed in the table below. The third phase, graduating a workflow to a configured second one without new engineering, decides whether ETA is a product or a service.

The phases as we currently plan them. The durations are planning assumptions from our own design. They are not measured and are not industry data, because no ETA engagement has been delivered. Phases 0 to 4 add up to 33 to 55 weeks before the first cross-function flow is measured.

Phase	What happens	Exit gate	Planning range
0. Ground	Agree the workflow, the metrics and the baseline. Map the systems. Measure what can be measured from existing sources. Nothing writes back.	A baseline the client's finance team accepts, and a named owner for the value.	3 to 5 weeks
1. Shadow	A harness runs alongside the people doing the work and proposes. Nothing reaches a system. The ledger starts recording.	Agreement between proposals and what people did is measured and good enough to supervise.	4 to 8 weeks
2. Supervised	A named person approves each action. Override rate, not approval rate, is the measure.	Error is below threshold against real outcomes and the override rate is stable and non-zero.	6 to 10 weeks
3. Graduate	Step classes that have earned it move to delegated. A second workflow is stood up using configuration, not new engineering.	A second workflow runs on configuration alone.	8 to 12 weeks
4. Cross-function	Work moves between function harnesses under one policy. The first end-to-end flow crosses two or more functions.	End-to-end cycle time is measured against the Phase 0 baseline.	12 to 20 weeks
5. Operating model	The client's own team runs the layer. Policy, tiers and the ledger become routine governance.	Continuous. The client can run it without us if it chooses.	Ongoing

Measurement first

Phase 0 does not depend on any agent acting. It fixes the definitions: what counts as the start and end of the workflow, a hand-off, rework. It builds the baseline from whatever the source systems can provide, and records where they cannot. It makes the cost of the current process visible in people and elapsed days, and everything later is measured against it. Reconstruction from existing systems will be partial. In Phase 0 we expect to learn where a case identity breaks, which systems keep usable logs and how much work happens where nothing records it. The ledger carries the baseline forward once a harness is running.

What the client commits

- A workflow owner with the authority to change how the work is done, not only to approve a project.
- A named owner of the value, who will answer for the result in finance terms.
- Access to the ground-truth signal, so that error can be measured against something real.

- A landing zone and identity set-up in the client's cloud for the work plane to run in.
- Subject-matter expert time agreed in writing, reviewers who are named and trained, and consultation in Phase 0 with the people whose work will change.

Where the process around the work does not change, the agent speeds up the old process and the problem moves rather than shrinking.

10 Value that survives a finance review

A value case for an AI programme tends to fail because the numbers do not reconcile when someone checks them, or because the saving needs a headcount reduction the client does not intend to make. ETA is designed to avoid both, with six rules for the case.

1. Baseline before deploying. The baseline is measured in Phase 0 and accepted by the client's finance team, with metric definitions agreed up front. Without it, any later improvement is an opinion.
2. Cost the whole stack: model spend, compute, reviewer time, integration and maintenance. Model spend runs in the client's own cloud, so it appears in the client's bill and cannot be quietly excluded.
3. Lead with cycle time and cost to serve. Elapsed days and cost per completed case are numbers a board can understand and test.
4. Show the decay curve. The headline chart is the share of steps that still need a person, falling as step classes earn autonomy.
5. Name what gets defunded. If the case relies on a cost going away, such as a licence or an overlapping tool, say which, so someone is accountable for removing it.
6. Write the conditions into the contract. If the saving depends on the client redesigning the process and on an executive sponsor, those are client commitments.

Because a person commits (rule four), the saving sits in three places only: cycle time, which the ledger shows directly; error reduction, fewer re-keying mistakes and fewer chasers, which the Outcome field shows; and released capacity, the hours people no longer spend on preparation and chasing, which they can spend on judgement and exceptions. Released capacity is time, and what the organisation does with it is the organisation's decision. We would sooner lose an argument about a smaller number than win one on a number that does not reconcile.

We apply the same standard to our own claims. In delivery work we built a duration estimator and set a ship rule in advance: it would ship only if it beat the plain median by at least 15%. On 127 backtest cases the best title-based method improved on the median by between 0.0% and 1.5% (24 September 2026 run), so we did not ship it. The discipline is the one we intend to apply to ETA: set the pass mark before the test and report the result whichever way it falls.

11 Security, assurance and regulation

This section is for security, data protection and legal reviewers. It says how ETA is designed to behave and where that is not yet proved. It is not legal advice or a statement of certification.

The assurance model

- Isolation. Each client's work plane is separate, in that client's tenancy, and only metadata crosses to Anitech (rule one).
- Least privilege and containment. Credentials are scoped to a single step, and agents that read cannot write (rule three), so untrusted input, sensitive data and an outbound path never exist together in one place.
- Accountability and evidence. Every action is attributed to a named agent version and, where a person was involved, a named person (rules four and five). The ledger can be exported to auditors from the client's own tenancy.

Prompt injection is not solved. The design assumes it will happen and limits what it can touch. A reviewer can test that claim, and a client's own red team is welcome to try. Whether the mechanisms hold against a motivated attacker in production is what the first engagements are for.

Regulation, at the level of principles

The most relevant instrument for most European clients is the EU Artificial Intelligence Act (Regulation (EU) 2024/1689, reference 2), alongside data protection law. The principles are easier to state than the application dates, which have been subject to change. The table sets out, in general terms, who holds each type of obligation and what ETA does about it.

Regulatory themes and where responsibility sits. A general orientation only. Specific obligations depend on how a client uses ETA and on the date.			
Theme	Who holds the obligation	What ETA does	Status
Disclose when people are dealing with AI	Shared between client and Anitech.	Harness outputs can be labelled, and the client chooses where disclosure appears.	Applies now in principle.
Meaningful human oversight	The client, as the organisation deploying the system.	Built in: the human gateway, the commit rule and the tier model are oversight mechanisms.	Designed. Not yet tested in an engagement.
Keep logs	The client.	The ledger provides the record in the client's own tenancy.	Designed. Not yet tested in an engagement.
Assess impact on people, and conform for decisions about people	The client for use, Anitech for the product.	Decisions about people are kept out of first pilots. A data protection impact assessment is a standard deliverable.	Roadmap. Not complete.

Which obligations apply to a given deployment, including those for systems used in employment and other high-risk areas, depends on how the Act classifies the use. We assess AI Act classification for each workflow with the client's counsel before work starts, and clients should treat this section as a starting point for their own legal advice.

Where personal data is involved, a data protection impact assessment under the General Data Protection Regulation (Regulation (EU) 2016/679, reference 3) would normally be expected, and we treat it as a standard deliverable. Under option A the client keeps its own data and Anitech handles metadata only. Anitech's role under any other hosting arrangement is settled in the data processing agreement before work starts.

We do not hold a security certification or audit report for ETA today, and a security reviewer should treat that as the current position. Anything we add will be stated here, in a later version of this paper.

12 What we do not claim yet

- No ETA pilot has run. There are no ETA results, and every ETA figure in this paper is illustrative or a planning assumption.
- ETA is not available, and we give no availability date because we cannot support one.

- The correlation problem is open. Following one case across several systems that share no key, without someone mapping the joins by hand, has not been shown in production. If it cannot be done, ETA is closer to a consultancy than a product, and we will say so.
- Reuse from the Harnesses is an assumption. A harness built for software delivery and one built for a finance or sales function may share a shell and little else.
- Forward instrumentation measures only what the harness runs.
- The tiers and criteria are our design, not an industry standard.
- Our default hosting keeps client data in the client's tenant. The Anitech-hosted option would break rule one and is not our default.

 A RULE WE HOLD OURSELVES TO

We make no public claims about ETA results until there are case studies, with the client's agreement, to support them.

13 Next steps for pilot partners

We are looking for a small number of organisations to test ETA with us. The aim of a first engagement is to find out, together and with evidence, whether the design holds.

Start with the four tests in section 9. Is there a ground-truth signal? Is the workflow reversible or bounded? Can the inputs be retrieved? Can the people who do the work be consulted before anything changes?

Before any engagement, can you already say how many people touch a deal from signature to cash, and how long it takes, with numbers you trust? If you can, tell us. If you cannot, the next step is a short conversation about one workflow and an agreed definition of what we would measure.

- We would ask for a named workflow owner and value owner, access to a ground-truth signal and a landing zone in your cloud, and time from the people who do the work, agreed in advance.
- In return you would get a baseline of the cycle time and cost of one workflow that your finance team can accept, whatever happens next, and a joint assessment of whether the design holds in your estate.

If one of your workflows might qualify, start a conversation with the authors through the [contact page](#). You can also read about ETA on the [ETA product page](#), and about the wider idea in our [white paper on ANI](#).

FIGURES

[Figure 1](#). The maturity ladder. Layers one to three are bought function by function. The fourth, a governed operating layer with shared meaning, shared rules and one record of the work, is the one ETA is designed to supply.

[Figure 2](#). The five permanent design rules. Each removes a class of finding that a security, data protection or finance reviewer would otherwise raise.

[Figure 3](#). The work plane and the control plane. Inside the client's tenancy, function harnesses read the systems of record through the capability gateway, using the meaning mapped in the semantic layer, and have no write path. Any write goes through the human gateway, where a named person or a pre-approved boundary approves it, and is recorded in the work ledger. Only metadata crosses to the Anitech control plane.

[Figure 4](#). Two hosting topologies. Option A, our default, runs the work plane in the client's own tenant, so rule one holds. Option B, considered and not our default, runs it in a region-pinned environment that Anitech hosts for that client. It would break rule one, so we would offer it only with that change stated in the contract and in the next version of this paper.

[Figure 5](#). The work ledger. Each step writes one record of seven fields as it happens. Instrumenting forward gives a complete record of the work the harness runs. Reconstructing from other systems' logs is partial, and often misses work done in inboxes.

[Figure 6](#). The four autonomy tiers. A step class moves up only when written criteria are all met, and moves down automatically when a demotion trigger fires. Promotions and model upgrades are never made in the same window.

[Figure 7](#). What Anitech builds and what it rides. Anitech builds the meaning content, long-tail connectors, instrumentation and ledger, delivery method and the Anitech control plane. The client's own platform supplies the rest: the agent runtime, identity and governance, workflow execution, the semantic engine, and integration and data movement.

[Figure 8](#). Phased delivery with gates. Each phase ends at an exit gate, listed in the table below. The third phase, graduating a workflow to a configured second one without new engineering, decides whether ETA is a product or a service.

REFERENCES

- [1] Lei, F. and others. Spider 2.0: Evaluating Language Models on Real-World Enterprise Text-to-SQL Workflows. arXiv:2411.07763, published at the International Conference on Learning Representations (ICLR), 2025. <https://arxiv.org/abs/2411.07763>
- [2] Regulation (EU) 2024/1689 of the European Parliament and of the Council of 13 June 2024 laying down harmonised rules on artificial intelligence (Artificial Intelligence Act). Official Journal of the European Union, 12 July 2024.
- [3] Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data (General Data Protection Regulation), in particular Article 35.
- [4] Anitech Solutions. Pilot result: two-person pod, 49 user stories to opened merge requests. Self-reported from the pilot's run logs, August 2026. Anonymised at the client's request. Merge requests opened, not yet merged. Quality outcomes not yet measured.

ABOUT THE AUTHORS



Jack Dinneen

Chief Product and Technology Officer

Owens product and technology: the Harnesses, ETA and what actually ships.



Andrew Nolan

Chief Executive Officer

Owens the client relationship, the outcome and the standard the work is held to.

How this paper was written. Drafted with AI assistance from Anitech's internal working papers and edited by Jack Dinneen and Andrew Nolan.