Dartmouth Advisory Partners · A Working Paper
APPLIED AI · GOVERNED JUDGMENT

Discovery Is the Bottleneck

Why the scarce skill in applied AI is no longer answering questions but knowing which ones to ask, and the governance architecture that turns a good question into a decision an institution can defend.

Brendan Sibeth, Founder & Managing Partner · v2.0 · 2026

ABSTRACT

The market is racing to automate answers. We argue the binding constraint sits one step earlier, in deciding **what is worth asking**, and that the durable advantage in applied AI accrues to whoever can do three things in sequence: find the questions worth asking, encode how experts actually reason, and prove every resulting decision back to its source. This paper gives that intuition its name and its architecture. We describe the system as a loop rather than a pipeline, show why it only compounds once it observes its own outcomes, and then name the two structures that make it ownable: Living Policy Architecture, in which an institution's governing documents become the configuration surface its systems run on, and Question as a Service, the class of work that sits above the answer. Institutional finance is where such a system is worth the most and, counter-intuitively, the last place you should try to teach it.

01

The inversion

Almost every tool in the current wave begins at the same place: it assumes the question is given and races to produce the answer. That is a reasonable bet, because answering has become cheap. Large models have commoditised the step the last decade treated as hard. But the commoditisation of answering exposes what was always the deeper problem, and it is not a problem of answers at all.

The real bottleneck is that, inside most valuable domains, nobody knows which questions to ask. A problem that is never named never gets the conversation that would solve it. Expertise sits on top of fragmented, unstructured knowledge that the people closest to it have stopped noticing, because they have accepted the chaos as normal. The industry solves *given a question, find the answer*. Almost no one solves *given a pile of messy domain knowledge, what are the right questions in the first place?*

This is not a rhetorical flourish. It is the load-bearing claim of this paper, because it is also the most leveraged one. Get the questions right and everything downstream, the knowledge graphs, the reasoning chains, the decision architecture, becomes buildable on top. Get them wrong and no amount of model quality rescues the result. The advantage belongs to whoever owns the step before the answer.

> **Get the questions right and the knowledge graphs, the reasoning chains, and the decision architecture all become buildable on top of them.**

The observation is not new, only newly urgent. Einstein and Infeld wrote in 1938 that the formulation of a problem is often more essential than its solution, that to raise new questions and regard old problems from a new angle requires the imagination that marks real advance.[1] A generation of research agreed. Getzels and Csikszentmihalyi, studying creative work, found that the people who discover and frame problems, rather than merely solving the ones handed to them, produce the more original and more durable results.[2] Donald Schon gave the act a name, “problem setting,” and made the sharp point that it has no place in a body of professional knowledge built only to solve problems.[3] Peter Drucker put the warning in operating terms: the dangerous mistake is not the wrong answer, it is the right answer to the wrong question.[4] What has changed is that the cost of answers has finally fallen far enough to expose how little of the value was ever in the answer.

Recent work on generative AI points the same way. Studies from Harvard Business School find these systems help most with conceptualisation, with generating ideas and framing problems, and least with the execution that carries a frame to a result.[5] The map has been commoditised. The terrain has not.

02

Why now, and why this failed before

A careful reader will object that this idea is old. They are right. “Externalise expert reasoning into a reusable, machine-readable form” is the expert-systems dream of the 1980s, the knowledge-management programs of the 1990s, and the ontology engineering that followed.[6] Each died on the same two rocks. First, tacit knowledge resists being told: experts know more than they can articulate, so asking them to write down how they decide produces a thin, lossy caricature.[7] Second, elicitation did not scale. The field literally named this the *knowledge-acquisition bottleneck*, which is our “what to ask” problem wearing a lab coat thirty years early.

So the honest question is not whether the idea is good. It has been good, and fatal, for decades. The question is what has actually changed. The answer is precise: the two failure modes that killed every prior attempt are exactly the two that recent capability shifts have dissolved. Elicitation can now be conversational rather than a consultant with a whiteboard; a model can interview an expert, follow the hesitations, and surface the questions neither party knew to ask. And the encoded structure no longer has to be a brittle, hand-built ontology; it can be fuzzy, partial, and still useful, because the reasoning layer tolerates ambiguity. Provenance and audit, the third leg, can finally be automated rather than reconstructed after the fact.

That is the whole of the “why now.” We are re-entering a famous graveyard, but with the two specific tools that were missing every previous time. It is also a warning label: the failure modes have not vanished, they have moved. Encoded judgment still goes stale, and elicitation can still degrade into mimicry. A serious system has to be built against both.

03

A loop, not a pipeline

The system has four moves over one shared spine. **Discovery** finds which problems are worth solving. **Elicitation** finds which questions actually define a problem once you are inside it. **Encoding** captures how an expert reasons, not merely what they conclude, in a durable form the institution owns and reuses. **Provenance** gives every decision its DNA: the source it drew on, the reasoning it followed, the confidence it carried, the decision it reached, and the outcome that resulted. Most platforms bolt explainability on at the end. Here it is the foundation, because in regulated domains a decision you cannot defend is a decision you cannot make.

Drawn as a loop, a property appears that a pipeline never has: the provenance layer’s own low-confidence decisions are a live map of where the right questions have not yet been asked. Uncertainty, recorded honestly, becomes the discovery engine’s next input. The output of the back end is the fuel of the front end. That circulation is the difference between a system that runs and a system that compounds.

EXHIBIT 1 · THE LEARNING FLYWHEEL

The engine as a loop

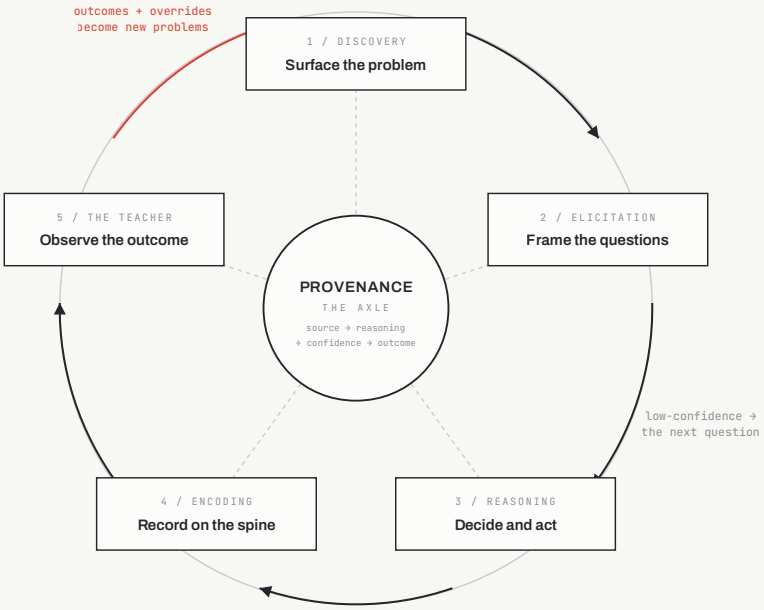

Exhibit 1. The discovery layer owns the intake and the return; the provenance spine is the axle every decision is written to. The system's own low-confidence decisions become the next questions it asks. Solid arc is the circulating loop; dashed spokes are each decision written to the spine.

04

The flywheel needs a teacher

There is a subtle failure waiting inside that loop, and naming it is the most useful thing this paper can do. A loop that records confidence is not yet a loop that learns. Confidence is the system's self-assessment. Correctness is what reality reports. If the only signal that circulates is the model's own confidence, the system can grow more and more certain of reasoning that is wrong, and nothing ever corrects it. That is an echo chamber with excellent provenance.

Closing the loop honestly requires a teacher: the real-world outcome of each decision, observed over time, and the judgment of the experienced people who stay accountable for the call. Where a trusted human and the system part ways is the boundary worth attending to. Treated that way, the trust boundary stops being a place where feedback leaks out and becomes the place where it comes in.

> A loop that records confidence is not yet a loop that learns. Confidence is a self-assessment; correctness requires outcomes.

This has a hard architectural consequence. To learn, the system must sit where decisions and their consequences both pass through it; it must be the system of record, not an adviser on the side. An advisory layer that hands over a recommendation and never sees what happened is structurally cut off from its own report card. It can be useful; it cannot improve. Calibration, the property of confidence that actually tracks correctness, is only earned by a system positioned to watch the outcomes it predicted.

05

Learn where it is cheap; deploy where it is dear

This is where the thesis meets institutional reality. The domain in which auditable judgment is worth the most, institutional finance, is the domain in which the learning loop turns slowest. A compliance or allocation decision is not adjudicated by reality for quarters, sometimes only at an audit, and sometimes never cleanly; when returns do arrive they are so confounded by market noise that separating a good judgment from a lucky one is its own hard problem. You cannot debug a learning loop on a multi-quarter, noise-soaked delay.

The resolution is to separate where you *learn* from where you *earn*. Calibrate the engine in a domain where ground truth returns in days, an operational setting that confirms or refutes a call almost immediately, such as a logistics, maintenance, or fulfilment workflow where the result is visible within the week, and carry the proven machinery into finance, where each decision is worth far more and must be provable. In finance the value was never autonomy in the first place. It is a provenance spine that defends every decision to a board or a regulator, and a policy layer that re-runs the same business day a rule changes. The trust boundary sits, correctly, toward the human: most consequential calls surface for judgment, and only narrow, in-policy actions execute on their own. Both still write to the same auditable spine.

EXHIBIT 2 · THE FINANCE INSTANCE

The wheel turns here, just slowly

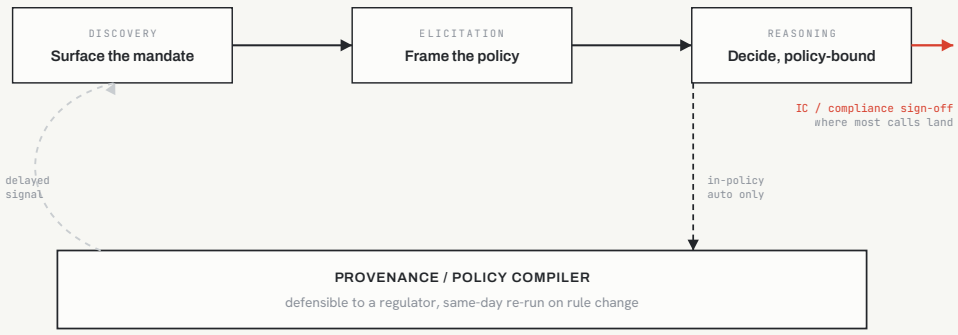

Exhibit 2. The finance instance. The return arc is attenuated (ground truth is slow and noisy) and the trust boundary is weighted toward human sign-off. Finance is the destination, not the proving ground: calibrate where outcomes return in days, deploy here where each decision is worth the most and must be provable.

We have now described two capabilities and named neither. One is a policy layer that compiles a rule change into live behaviour. The other is the disciplined production of the right questions, delivered as an ongoing service rather than a report. They deserve names, because what an institution can name it can own. The next two sections give them.

06

Living Policy Architecture

Treat the claim literally: an institution's governing documents are not descriptions of how it behaves. Treated correctly, they are the source it runs on. An investment policy statement is usually a PDF in a drawer, consulted at onboarding and during disputes, otherwise inert. **Living Policy Architecture** makes it the configuration surface instead. The document the investment committee debates and signs becomes the document the system executes and audits against. Change the policy, recompile, and every downstream system reflects the change that business day rather than that fiscal quarter.

We did not invent this discipline from nothing, and the paper is stronger for saying so. Software has spent more than a decade moving human intent into machine-executable form. Infrastructure as code turned server configuration into versioned, testable files. Policy as code, through engines such as the Open Policy Agent, did the same for security and compliance rules; a large majority of technical leaders now treat it as essential to governing systems at scale.[8] Governments have gone further still with Rules as Code, the proposition advanced by the OECD and pioneered by New Zealand's Better Rules work that legislation should ship in a machine-consumable form alongside its prose.[9] Financial

institutions already run policy as code for data access and entitlement.[10] The lineage is real and it is mature.

What none of it has done is apply the discipline to investment governance itself. That is the open ground, and it is where Living Policy Architecture differs from everything upstream of it. The compiled artifact here is not a firewall rule or a tax calculation. It is the institution's judgment. Policy authorship becomes the customisation surface: the firm expresses how it wants to think once, in the document it already owns and already governs, and the system inherits that intent everywhere. We build this as a policy compiler, and our own expression of it is Overture. The principle is what matters and it is portable: the institution owns the policy, the policy is the program, and the program is provable.

07

Question as a Service

We have a name for the layer that delivers software, one for the layer that delivers platforms, one for the layer that delivers raw infrastructure. We have no name for the layer that delivers the question. Yet that is precisely the layer the inversion makes most valuable. We call it **Question as a Service**: the disciplined production of the right questions, the encoding of the reasoning that answers them, and the provenance that defends the result, delivered as an ongoing capability rather than a one-time report.

A note on the name, in the open, because category claims made carelessly do not survive contact with a sophisticated reader. The phrase is not entirely unused. In engineering circles "query as a service" has a narrow technical meaning, an endpoint that runs a saved database query, and the bare acronym is already crowded, claimed in places by quantum computing and by quality assurance.[11] We are not describing any of those. We are naming the thing the as-a-service vocabulary has so far skipped: the institutional service whose product is the framing, not the fetch. We claim the institutional meaning deliberately, with the collisions in plain view, because the idea has earned a name and the category is otherwise empty.

The point is sharpest against the thing it replaces. Traditional advice sells answers. A consultant studies the problem, delivers a recommendation, and leaves; the report begins ageing the day it is bound, and the reasoning behind it walks out of the building with the people who wrote it. Question as a Service inverts the deliverable. What persists is not the answer but the apparatus that produces answers: the encoded reasoning, the live and growing set of questions, the provenance spine that sharpens as it observes outcomes. The institution does not rent a conclusion. It owns the engine that reaches conclusions and can defend each one. This is the premise behind Resolve Exchange, an institutional marketplace built so that the unit of exchange is the question and the governed reasoning around it, not merely the answer.

EXHIBIT 3 · THE SERVICE STACK

Where the question sits

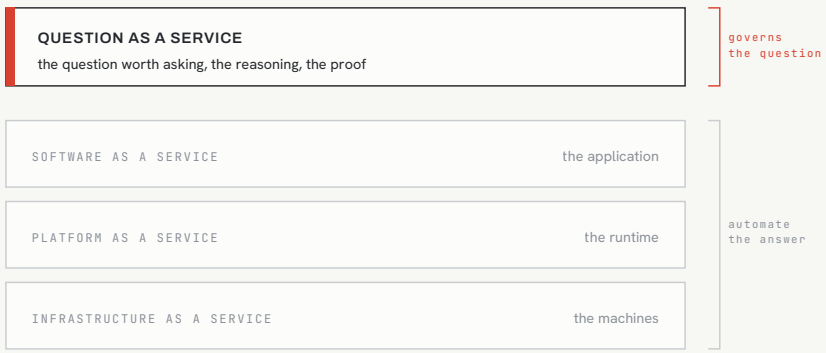

Exhibit 3. Infrastructure as a Service delivers the machines, platform the runtime, software the application. Each automates a layer of the answer. Question as a Service sits above them all and governs the layer none of them touches: which question is worth asking, how an expert reasons toward it, and how the result is proven.

08

Own, do not rent

Two forces make ownership the decisive question now, rather than a preference.

The first is defensibility. Regulators have stopped accepting “the model said so” as an account of a decision. Model risk management guidance written years ago for traditional models, SR 11-7, is now applied by examiners to artificial intelligence and machine learning, and it asks for exactly what a thesis built on provenance already produces: validation, documentation, governance, and monitoring.[12] The EU AI Act places creditworthiness and credit-scoring systems in its high-risk tier, with documentation and human-oversight obligations attached.[13] A United States rule that would have governed predictive analytics in advice was proposed and later withdrawn, but the pressure it expressed, that a firm must be able to explain and defend an automated recommendation, has not gone anywhere.[14] A decision you cannot defend is, increasingly, a decision you are not permitted to make.

The second is dependency. The incumbent platforms of institutional finance are formidable, and they are rented. BlackRock’s Aladdin runs risk and operations across roughly twenty-five trillion dollars of assets on, in its own framing, one platform and one data set; SimCorp and Bloomberg’s AIM occupy the same front-to-back operational ground.[15] They are very good at what they do, which is execution. But they are systems the institution operates inside, not systems it owns, and the value they create accrues in part to the platform. The criticism that trails them, lock-in, the black box, the quiet

convergence of everyone running the same model on the same data, is the criticism of rented judgment.

This paper's architecture is the other choice, and the choice is not close. Execution can be rented; it is a commodity and will stay one. Judgment should be owned. The encoded reasoning of your most experienced people, the policy that compiles into your systems, the provenance that defends your decisions to a board or a regulator: these are not assets to lease from a vendor whose incentives are not yours. That is the through-line of everything we build, and it has a short name. Own your AI. More precisely, and more durably: own your questions.

09

What an institution is actually buying

Everyone will have automation; it confers no advantage to have what everyone has. The durable assets are different. The first is a method for externalising the judgment that today lives tacitly in your most experienced people, judgment that walks out of the building when they do. The second is a record that proves every decision back to its source, on demand, to whoever is entitled to ask. The third is a discovery engine that turns your own institution's uncertainty into its next set of questions, so the system gets sharper precisely where it is currently weakest.

The moat, in other words, is not the model. It is the encoded reasoning and the provenance that compounds across decisions: a portable protocol for how a domain thinks, defended by a spine that makes every output auditable. That is a far harder thing to copy than a prompt, and a far more valuable thing to own than another layer of automation.

On the limits of this thesis

Stated plainly, because a thesis that hides its soft points is not worth defending.

The compounding is bounded, not magical. It rests on a repeatable method and on structure demonstrated across real systems, not on a claim that all domains converge. We reject the version that would promise more than the evidence supports.

Elicitation is the live frontier. Our bet is that recent capability shifts changed its economics, not that the problem is solved. The knowledge-acquisition bottleneck can reappear in new clothes, and we build against that.

Outcome latency is real. In slow-feedback domains the loop is calibrated elsewhere and carried in. Where we cannot yet observe ground truth quickly, we say so, and we do not dress confidence up as calibration.

The category is a claim, not yet a consensus. Naming a layer does not make a market. Question as a Service is our wager on where value is moving, argued from first principles and current evidence; the reader is entitled to hold us to it.

NOTES

1. A. Einstein and L. Infeld, *The Evolution of Physics* (Cambridge University Press, 1938). The widely circulated "if I had an hour, I would spend fifty-five minutes on the problem" line is not traceable to Einstein and is not used here.
2. J. W. Getzels and M. Csikszentmihalyi, *The Creative Vision: A Longitudinal Study of Problem Finding in Art* (Wiley, 1976); and Getzels, "Problem Finding: A Theoretical Note," *Cognitive Science* 3 (1979).
3. D. A. Schon, *The Reflective Practitioner: How Professionals Think in Action* (Basic Books, 1983), on "problem setting" as the work of naming and framing that precedes problem solving.
4. P. F. Drucker, on the danger of the right answer to the wrong question, a recurring theme across his management writing.
5. Harvard Business School research on generative AI and the value of expertise, including work by Bojinov and McFowland on where these systems help (conceptualisation and problem framing) versus where they do not (execution), and Fuller and Sigelman in *Harvard Business Review* (2025).
6. On the lineage of expert systems and knowledge engineering, see E. A. Feigenbaum's work on knowledge-based systems and the broader expert-systems literature of the 1980s.
7. The distinction between articulable and tacit knowledge follows M. Polanyi, *The Tacit Dimension* (1966); the "knowledge-acquisition bottleneck" is a term of art from the expert-systems field of the same era.
8. On policy as code and the Open Policy Agent, see the Cloud Native Computing Foundation's OPA project, and Styra, *The State of Policy as Code* (2023), in which a large majority of technical decision-makers describe policy as code as essential to security and compliance at scale.
9. H. Mohun and A. Roberts, *Cracking the Code: Rulemaking for Humans and Machines*, OECD Working Papers on Public Governance No. 42 (2020); and the New Zealand "Better Rules" and Rules as Code initiative.
10. For a financial-services example of policy treated as code for data governance and access control, see UBS's published work on policies as code for its enterprise data platform (2025).
11. "Query as a service" appears as a narrow technical feature in knowledge-graph and ERP tooling (for example, parameterised SPARQL endpoints and saved-query services). The acronym QaaS is separately used for Quantum-as-a-Service and Quality-as-a-Service. None refers to the institutional service defined here.
12. Board of Governors of the Federal Reserve System and OCC, Supervisory Guidance on Model Risk Management (SR 11-7, 2011); on its application to AI and machine-learning models, see the U.S. Government Accountability Office (2025).
13. Regulation (EU) 2024/1689 (the AI Act), Annex III, classifying creditworthiness and credit-scoring AI as high-risk, with attendant documentation and oversight duties.
14. U.S. Securities and Exchange Commission, proposed rules on conflicts of interest associated with predictive data analytics (July 2023), formally withdrawn in 2025; the underlying expectation of explainable, defensible automated advice persists.
15. Figure for assets on the Aladdin platform per BlackRock disclosures and management commentary (2024 to 2025). SimCorp One (Deutsche Borse) and Bloomberg AIM are cited as comparable front-to-back operational platforms. The market these platforms serve continues to expand: Cerulli projects the U.S. outsourced-CIO function to grow from roughly 3.3 trillion dollars (2024) toward 5.6 trillion dollars by 2029.

Dartmouth Advisory Partners builds governed AI for institutions that have to defend their decisions. The automation is table stakes. The questions, the policy, and the proof are what you own. If discovery is your bottleneck, that is the conversation we want to have.

partnerships@dap.solutions · dap.solutions · Toronto