PARAKLETOS

LONDON

# The Governance Gap in Enterprise AI

*Why models, tools and agents need a defensible control plane, not just gateways, provider dashboards and policy documents.*

A WHITE PAPER

Parakletos AI Limited · London

◆ CONTENTS

# How this paper is organised


## I THE PROBLEM

**At a glance** 3
*AI access is being solved. Authority is not.*

**Executive summary** 4
*For many enterprises, AI adoption is entering a second phase*

**The governance gap** 5
*Useful access on one side. Defensible control on the other.*

**A scenario** 6
*A useful session, three months later*

**Three composite cases** 7
*What broke when organisations tried to reconstruct AI activity*

**Reconstructed versus recorded** 8
*The same question, answered two ways*

**Where the gap hides** 9
*An accumulation of reasonable shortcuts*

**From chat to authority** 10
*The trust boundary is expanding*

## II CONTEXT AND JUDGEMENT

**Gateway and control plane** 11
*Routing traffic is not governing authority*

**Proportionality** 12
*When a full control plane is the wrong answer*

**Five objections** 13
*Five objections, taken seriously*

**Legal and regulatory pressure** 14
*Demonstrable control is already expected*

**From capability to obligation** 15
*What the evidence record can help demonstrate*

## III THE ARCHITECTURE

**The control framework** 16
*Nine domains separate AI access from an enterprise-controlled AI estate*

**A practical definition** 17
*A practical definition*

**Policy evaluation in real time** 18
*How a single tool operation is decided*

**Preventing bypass · Evolving the control plane** 19
*Making the governed path authoritative*

**Long-running agents** 20
*Coherent evidence for multi-step execution*

**The evidence record** 21
*One AI action. One defensible record.*

## IV MEASURING AND ADOPTING

**The maturity model** 22
*From unmanaged, to managed, to governed*

**Measuring progress** 23
*Leading indicators for each level*

**Prioritising the nine domains** 24
*Where to start depends on what the path can do*

**The governance dividend** 25
*Control that can make adoption faster*

**Failure modes and cost** 26
*What goes wrong, and what it costs*

## V GETTING STARTED

**Where to start** 27
*Five day-one AI paths*

**Readiness check** 28
*Eleven questions, answered against one real path*

**Closing argument** 29
*From access to authority*

**References** 30
*Sources referred to in this paper*

◆ AT A GLANCE

# AI access is being solved. Authority is not.

THE SHIFT

## Enterprise AI is moving beyond chat.

Coding assistants, model APIs, MCP tools, RAG systems and agents are becoming part of normal work.

THE GAP

## Giving people access to capable AI is increasingly easy.

Demonstrating who was allowed to do what, with which data, through which model or tool, under which policy, is not.

THE FIX

## One organisation-controlled layer between users and AI execution.

**Identity** · **Policy** · **Routing** · **Tools** · **Evidence**

THIS PAPER IS FOR

| | |
|---|---|
| PRIMARY AUDIENCE | AI Platform · Developer Platform · Security Engineering · Enterprise Architecture |
| ALSO RELEVANT TO | CIO / CTO · CISO · Risk · Compliance · Data Protection |

WHAT THIS PAPER IS NOT

It is not an opinion against Claude Code, Copilot, OpenAI, Anthropic, MCP, enterprise AI gateways or agents.

It is an argument that enterprise AI needs a ***control boundary*** around those capabilities.

## ◆ Who should read this, and why

The governance gap looks different from each chair at the table. The same AI path raises a distinct question for each function.

| READER | THE QUESTION THEY NEED TO ANSWER |
|---|---|
| **CIO / CTO** | Can we scale AI access without losing control? |
| **CISO** | Can we demonstrate who accessed which models, data and tools? |
| **AI Platform** | Can we provide a governed route developers actually want to use? |
| **Risk / Compliance** | Can we show which policy applied to a specific AI action? |
| **Enterprise Architecture** | Can providers and gateways change without changing the trust model? |

◆ EXECUTIVE SUMMARY

# For many enterprises, AI adoption is entering a second phase

The first phase was about access. Employees gained chat assistants. Engineers gained coding copilots. Teams connected frontier models through APIs. Now MCP and agentic systems are connecting models directly to repositories, documentation, databases and internal services. The tooling is increasingly capable, cheap and easy to adopt. That is precisely why the governance problem is becoming harder.

**66%** of respondents to Stack Overflow's 2026 Developer Survey use AI coding assistants or agents at work.[12]

**23%** of organisations in McKinsey's 2025 global survey were scaling an agentic AI system; a further 39% were experimenting.[13]

**WHO IS MOST EXPOSED**

**Organisations with:**

- large engineering teams
- regulated data
- widespread coding-assistant adoption
- internal AI APIs
- MCP experimentation
- early agents
- multiple AI providers

A successful model response does not prove that the request was authorised.

A functioning MCP call does not prove that the agent was entitled to invoke the tool.

A provider dashboard does not necessarily establish which enterprise policy applied when the request was made.

For a serious enterprise, the requirement is therefore no longer simply access to AI. It is the ability to demonstrate:

| 01 | 02 | 03 | 04 | 05 |
|---|---|---|---|---|
| who initiated an AI action | what they attempted | which policy governed it | what model or tool actually executed it | what happened as a result |

The practical answer is not an enterprise-wide AI transformation programme. It is to take one real AI path and put it under control, end to end.

◆

*Capability is becoming a given. Governed execution is not.*

**HOW TO USE THIS PAPER**

Read it with one real AI path in mind. It also covers when a lighter approach is enough (page 12) and where control planes fail (page 26). The readiness check on page 28 can be completed against that one path.

◆ THE GOVERNANCE GAP

# Useful access on one side. Defensible control on the other.

There is a gap between useful AI access and enterprise-controlled AI. Access is what a tool gives a user. Control is what the organisation can stand behind afterwards. An AI action can be entirely successful while the process that permitted it remains unaccountable.

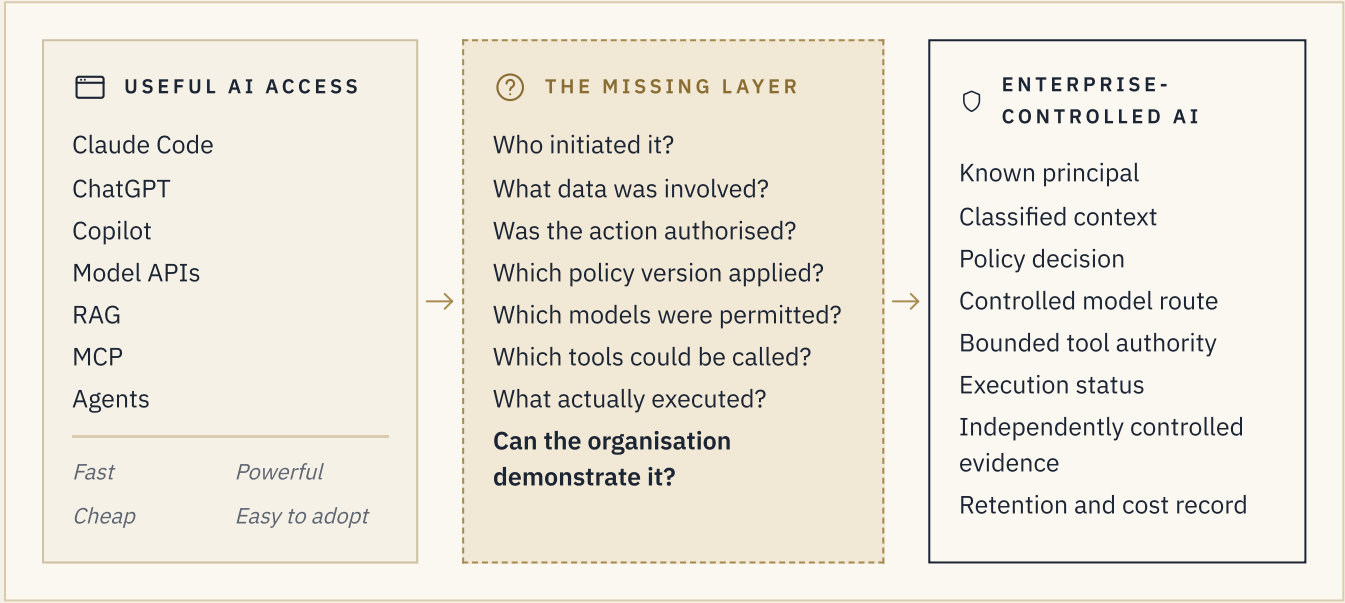



**Figure 1.** *The governance gap. The questions in the middle are the ones asked after the event.*

◆

*A successful AI action can still be produced through a process nobody can later defend.*

**WHAT PROCUREMENT OR ENGINEERING TESTS**

Capability · Latency · Model quality · Developer experience · Price

*Tested at adoption.*

**WHAT SECURITY, RISK AND AUDIT TEST LATER**

Identity · Policy · Authority · Execution · Provider route · Evidence

*Tested after the fact.*

Closing that gap requires a control boundary, not another model feature.

◆ A SCENARIO THAT EXPOSES THE PROBLEM

# A useful session, three months later

An engineer at a financial-services firm uses Claude Code against an internal repository. The session uses an approved frontier model and a GitHub MCP server. During it, the assistant reads repository code, retrieves architecture documentation, reads open issues and proposes a change. The work is useful.

Three months later a security review, an incident or an internal audit asks a set of reasonable questions.

- Which employee initiated the session?
- Which repositories did the system read?
- What information left the organisation?
- Which model processed it?
- Which MCP operations were attempted?
- Was write access technically possible?
- Which policy authorised those operations?
- What did the enterprise itself record?

**WHAT HAPPENS NEXT**

The organisation reconstructs the event from provider dashboards, gateway logs, GitHub logs, SIEM events, employee memory and support tickets. Each log describes its own system; none records the decision that connected them.

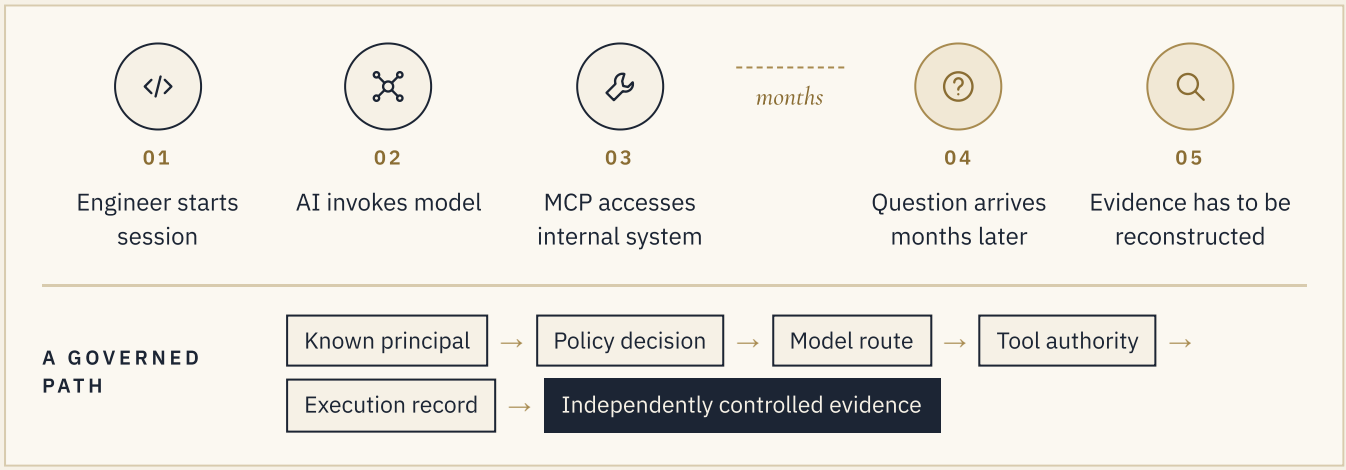



**Figure 2.** *The failure scenario. Each step works and each system keeps its own log, but no single organisation-controlled decision record explains who initiated the action, what authority applied, which policy decision permitted it and which downstream system executed it.*

◆

*The AI worked. The control model didn't.*

◆ THREE COMPOSITE CASES

# What broke when organisations tried to reconstruct AI activity

Each case below is a composite. It combines recurring patterns from enterprise AI reviews, with sector, scale and detail altered so that no organisation is identifiable. None involved negligence. Each involved a reasonable architecture meeting a question it was never designed to answer.

CASE A

## Coding assistant

UK retail bank · c.1,400 engineers · two model providers

**THE QUESTION**

Following a supplier notice: did confidential source code from the payments estate reach a provider region outside the UK or EU during a three-week window?

**WHAT BROKE**

Provider usage was attributed to API keys shared across teams. The gateway logged model and token counts, but not repository. MCP servers ran locally on laptops, so no central record of which repositories were read existed.

**OUTCOME**

The answer was assembled from sampling, gateway logs and interviews with around thirty engineers. It was given with an explicit low-confidence qualification.

CASE B

## Internal assistant with retrieval

Insurer · c.9,000 employees · one provider on an enterprise tier

**THE QUESTION**

In response to a data subject request: was this customer's personal data processed by an AI system, by whom, and for what purpose?

**WHAT BROKE**

Prompts were retained by the provider for a limited period, and internally only as free-text chat history. No classification was applied at request time, so the only available search was for the customer's name across unstructured logs.

**OUTCOME**

The response could not rule processing out. Retention settings were changed afterwards; past activity could not be recovered.

CASE C

## Early agent with write access

B2B software company · ticket-triage agent

**THE QUESTION**

After roughly 300 customer tickets were closed in error overnight: which actions did the agent take, under whose authority, and which were safe to reverse?

**WHAT BROKE**

The agent used a service account shared with an older integration, so the ticketing system attributed both to one identity. Retries after a timeout duplicated some updates. Nobody could say which steps of the final run had completed.

**OUTCOME**

Reversal took two days of manual comparison against backups. The agent was disabled for six weeks while its credentials were separated.

THE COMMON PATTERN

- Identity shared or inferred
- No record of the decision
- Tool activity attributed to the wrong principal
- Partial execution unknowable

◆ RECONSTRUCTED VERSUS RECORDED

# The same question, answered two ways

The difference a control plane makes is easiest to see in a single investigation. The comparison below takes the question from Case A and contrasts how each element of the answer is obtained: reconstructed after the event from the logs each system happens to keep, or read from a record written at the point of decision.

| ELEMENT OF THE ANSWER | RECONSTRUCTED FROM PROVIDER AND SYSTEM LOGS | READ FROM A CONTROL-PLANE RECORD |
|---|---|---|
| **Who initiated it** | Inferred: API key → team → interviews to narrow to individuals | Recorded: principal and workload on each request |
| **Which data was involved** | Inferred from repository access logs if MCP ran centrally; otherwise unknown | Recorded: resource and data classification per operation |
| **Which policy applied** | Inferred from the policy document in force on the date, assuming it was followed | Recorded: policy ID, version and decision |
| **Where it was processed** | Provider console, per key; region often not exposed | Recorded: resolved provider, model and region |
| **What executed** | Downstream system logs, attributed to whichever credential called them | Recorded: child decisions carrying the parent execution ID |
| **Partial or failed steps** | Usually unknowable | Recorded: outcome state, including indeterminate |
| **Who must be involved** | Platform, Security, each provider, affected teams, individual engineers | One team querying the evidence store |
| **Indicative effort** | Days to weeks | Minutes to hours |
| **Confidence of the answer** | Qualified | Stated, with known gaps declared |

Figure 3. *Reconstructed versus recorded. Effort ranges are indicative, drawn from the composite cases, and are not measurements.*

### ON THE EVIDENCE BASE

The adoption figures on page 4 come from two large surveys, and both have limits. The Stack Overflow Developer Survey is a self-selected sample of developers who engage with Stack Overflow. McKinsey's survey relies on executives' self-reporting, and what counts as 'scaling an agent' is left largely to respondents.

Neither measures governance, and we are not aware of public longitudinal data on the cost of investigating AI activity. The composites are offered as illustration, not proof. An organisation that times its own investigations before and after governing one path will have better evidence than this paper can provide.

◆ WHERE THE GAP USUALLY HIDES

# An accumulation of reasonable shortcuts

The gap is rarely one catastrophic architecture decision. It is usually an accumulation of individually reasonable shortcuts, each taken by a team trying to get useful work done.

### PERSONAL CREDENTIALS

Provider keys or personal AI accounts become de facto enterprise identity.

### DIRECT PROVIDER ACCESS

Applications bypass the approved gateway because the provider SDK is easier.

### OPAQUE MCP ACCESS

An agent can call tools, but nobody centrally records its effective authority.

### PROVIDER-DEFINED MODELS

Applications specify vendor SKUs directly, bypassing enterprise routing policy.

### LOGGING AFTER THE FACT

The SIEM records activity, but did not participate in the decision that allowed it.

### UNVERSIONED POLICY

A rule exists in documentation, but nobody can show which version governed a specific request.

### STANDING AGENT PRIVILEGE

Tools and credentials remain accessible beyond the task that required them.

### SPLIT OWNERSHIP

Security thinks Platform owns AI governance. Platform thinks application teams own it.

> None of these is negligent on its own. Together, they leave no single point at which an AI action was decided, constrained and recorded by the organisation.

◆ FROM CHAT TO AUTHORITY

# The trust boundary is expanding

> *The governance question is moving from 'what did the model say?' to 'what was the system allowed to do?'*

Early enterprise AI governance concentrated on content: what a user typed into a chat window and what came back. That framing is becoming too narrow. Each new integration pattern hands the AI system more of the organisation's authority: first its data, then its knowledge, then its systems, and finally the ability to act on them.

| | STAGE | AUTHORITY DELEGATED TO AI | EXTENT |
|---|---|---|---|
| 01 | CHAT | Generate text | |
| 02 | MODEL API | Process enterprise data | |
| 03 | RAG / CONTEXT | Retrieve internal knowledge | |
| 04 | MCP / TOOL ACCESS | Read internal systems | |
| 05 | WRITE ACTIONS | Change enterprise state | |
| 06 | AUTONOMOUS AGENTS | Act continuously with delegated authority | |

Shaded stages reach into, or change, enterprise systems.

Figure 4. *The expansion of AI authority. Each stage delegates more of the organisation's authority to the system.*

Every step increases the amount of enterprise authority delegated to AI.

Governance therefore has to move closer to execution.

◆ WHY A GATEWAY ALONE IS NOT THE CONTROL PLANE

# Routing traffic is not governing authority

Model gateways are useful infrastructure. They can centralise provider access, routing, credentials, retries, quotas and telemetry, and most enterprises that are serious about AI should have one. But routing traffic is not the same as governing enterprise authority. Questions a gateway alone may not answer:

- Who is the real human or workload principal?
- What data classification applies?
- Which enterprise policy was evaluated?
- Why was this model permitted?
- Which MCP operations were available?
- Was approval required?
- What happened when execution became ambiguous?
- Does the organisation control the evidence of the decision?

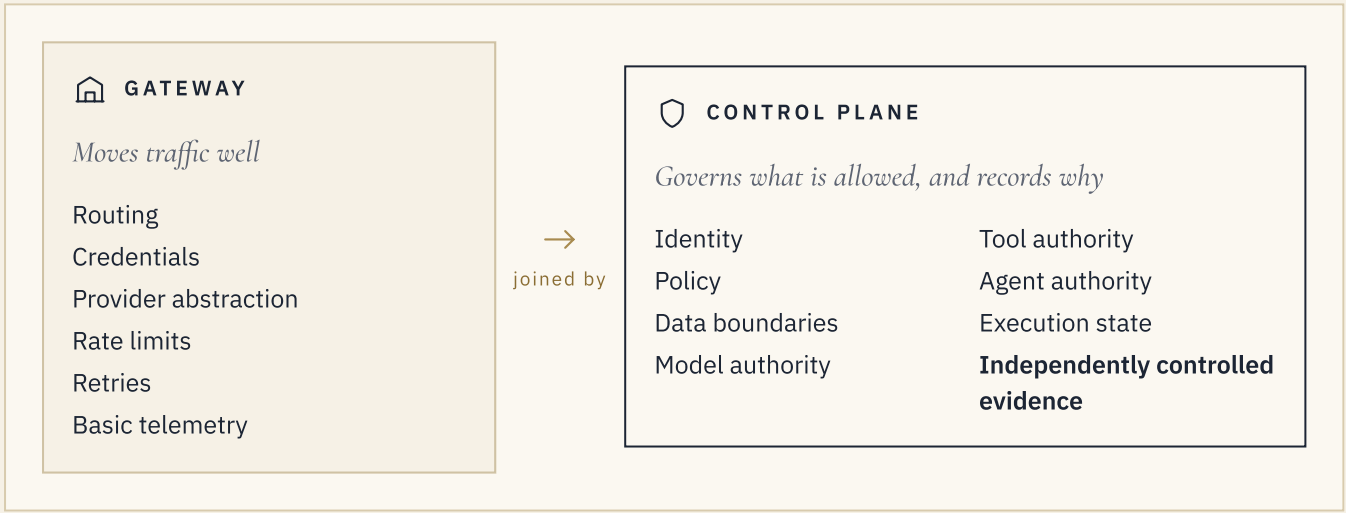



Figure 5. *Gateway and control plane. The gateway is a component; the control plane is the boundary around it.*

**IF YOU BUILD IT YOURSELF**

An enterprise can assemble these capabilities itself. The components are available: identity providers such as Microsoft Entra ID or Okta, gateways such as LiteLLM or Kong, policy engines such as OPA or Cedar, observability built on OpenTelemetry, workloads on Kubernetes, and MCP infrastructure. The harder engineering problem is maintaining consistent governance semantics across identity, policy, models, tools, execution and evidence as those components and providers change.

> *"The components already exist. The governance boundary joining them usually does not."*

> *"Logging what happened is not the same as proving why it was allowed to happen."*

◆ PROPORTIONALITY

# When a full control plane is the wrong answer

Not every organisation needs everything in this paper, and not every capability needs to be bought. Governance should be proportionate to the authority an AI path holds. The question is not whether to buy a new category, but whether the semantics joining existing components exist, and who owns them.

### LIGHTER CONTROLS ARE LIKELY ENOUGH

WHEN

- Chat and drafting only, with no tool access
- One provider on an enterprise agreement with SSO, admin controls and retention settings
- No regulated or confidential data in scope
- One team owns all AI use

*Enterprise tier, SSO, an acceptable-use policy and a periodic access review.*

### EXTEND WHAT YOU ALREADY RUN

WHEN

- A mature identity provider with workload identity
- A policy engine already used for application authorisation
- An AI or API gateway in place
- A SIEM with suitable retention

*Add AI-specific policy inputs, a shared evidence schema and per-operation tool decisions to components you already own.*

### A DEDICATED BOUNDARY IS JUSTIFIED

WHEN

- Several providers and models
- MCP or tools with write access
- Agents acting with delegated credentials
- Regulated data, or obligations to produce records
- Many teams building independently

*An organisation-controlled layer of the kind described from page 16 onwards.*

Figure 6. *Matching the control to the authority. Most estates contain paths in more than one column.*

WHAT EXISTING PLATFORMS OFTEN ALREADY COVER · NAMED PRODUCTS ARE EXAMPLES, NOT ENDORSEMENTS

| CAPABILITY | OFTEN ALREADY COVERED BY | WHAT IS USUALLY MISSING |
|---|---|---|
| **Identity** | Identity providers such as Microsoft Entra ID, Okta or Ping Identity; workload identity via SPIFFE / SPIRE or cloud managed identities | The link from a model or tool call back to the human or agent behind it |
| **Routing, keys, quotas** | AI or API gateways such as LiteLLM, Kong AI Gateway, Azure API Management, Cloudflare AI Gateway or Portkey | The policy reason a route was permitted |
| **Authorisation** | Policy engines such as Open Policy Agent, Cedar (and Amazon Verified Permissions) or Cerbos | AI-specific inputs: data class, model alias, tool operation |
| **Logging** | OpenTelemetry pipelines into SIEM and observability tools such as Microsoft Sentinel, Splunk, Datadog or Grafana | Decision context, and parent–child linkage across a run |
| **Provider controls** | Enterprise tiers such as ChatGPT Enterprise, Claude Enterprise or GitHub Copilot Business: SSO, audit logs, retention settings | A cross-provider, cross-tool view held by the organisation |

◆ COUNTER-ARGUMENTS

# Five objections, taken seriously

Each objection below is reasonable, and some are partly right. The strongest version of each is stated first, followed by where it holds and where it stops holding.

| OBJECTION | THE STRONGEST VERSION | WHERE IT HOLDS, AND WHERE IT BREAKS |
|---|---|---|
| *"We already have a gateway, API management and a SIEM."* | Those components already route, authenticate and log. Another layer adds latency, cost and one more thing to operate. | Often right for model traffic owned by one team. It breaks at tool operations and agents: a model gateway sees calls to models, not MCP operations on internal systems, and the SIEM records events without the decision that permitted them. The answer may be to extend those components rather than replace them (page 12). |
| *"This will slow developers down."* | Every check adds latency and friction, and friction drives shadow AI: the opposite of the intended outcome. | This is the most common way control planes fail (page 26). A locally evaluated policy decision adds little latency next to a model call; the real costs are onboarding, approvals and unexplained denials. If the governed path is harder to adopt than a personal API key, it will lose. |
| *"Policy-as-code is too rigid for AI."* | AI behaviour is probabilistic and context-dependent. Static rules cannot judge whether a prompt or an output is appropriate. | Agreed, for content. Policy-as-code is not used to judge outputs. It decides deterministic questions: may this principal use this model for this data class; may this agent call this operation. Content judgements stay with classifiers and people, and their results become policy inputs. |
| *"We can just use provider enterprise features."* | Enterprise tiers now offer SSO, admin consoles, audit logs, retention controls and data-handling commitments. | For a single-provider estate without tool access, this may well be sufficient. It breaks across several providers and tools, and when the record of authorisation must be held by the organisation rather than by the supplier being assessed. |
| *"This is just Zero Trust and IAM applied to AI."* | Verify explicitly, grant least privilege, assume breach. Nothing here is new. | Largely correct, and that is a strength. What is new is the subject (a model acting on a principal's behalf), the unit of authority (one tool operation inside a session) and the evidence (a decision chain across model, tool and agent). Treat this as Zero Trust's next scope, not a rival to it. |

**A NOTE ON BIAS**

The author of this paper builds control planes, and so has an interest in the answer. Readers should weigh the responses above accordingly, and test them against their own estate using the readiness check on page 28.

◆ LEGAL, REGULATORY AND OPERATIONAL PRESSURE

# Demonstrable control is already expected

None of these obligations requires an 'AI control plane' as such. Many of them, however, require an organisation to demonstrate control over how personal data, suppliers and critical systems are handled, and AI activity now touches all three. *This is not legal advice.* Whether and how each applies depends on role, sector, data and circumstances.

## UNITED KINGDOM

**UK GDPR and Data Protection Act 2018**
Accountability, data minimisation, security of processing, and controller–processor obligations that extend to suppliers.

**Data (Use and Access) Act 2025**
Amends the UK GDPR, DPA 2018 and PECR. Most data protection changes, including revised automated decision-making rules, have applied since 5 February 2026.

**Operational resilience (FCA / PRA)**
For in-scope financial firms: important business services and impact tolerances, alongside outsourcing and third-party risk management expectations.

## EUROPEAN UNION

**EU GDPR**
Accountability, records of processing, processor obligations and international transfers.

**EU AI Act**
Applies in phases. Transparency duties for certain systems from 2 August 2026. Logging, human oversight and deployer duties for Annex III high-risk systems from 2 December 2027, as amended in 2026.

**DORA**
Since 17 January 2025, for in-scope financial entities: ICT risk management, ICT third-party risk and major incident reporting.

**NIS2**
Only for essential and important entities, through national law: risk management, supply-chain security and incident reporting. Transposition is incomplete in some Member States.

## AND THE PRESSURE THAT IS NOT REGULATORY

*The same questions, asked more often*

**Internal audit**
Asks for evidence that a control operated, not only that it was documented.

**Cyber incident response**
Needs to reconstruct what an AI system accessed, did and was permitted to do.

**Third-party risk**
Treats each model provider and MCP server as a supplier to be assessed.

**Architecture review**
Asks where the trust boundary sits before a new AI path goes live.

**Vendor due diligence**
Enterprise customers may ask how a supplier's own AI use is controlled.

**AI governance committees**
Need a mechanism, not only a policy, to make their decisions enforceable.

## THE EVENTS THAT EXPOSE THE GAP

Security incident
Regulatory enquiry
Audit
AI supplier review
Data subject request
Agent misuse
Unexpected tool action
Leak investigation

◆ FROM CAPABILITY TO OBLIGATION

# What the evidence record can help demonstrate

The table maps control-plane capabilities to obligations they can help evidence. A capability that supports an obligation is not compliance with it: most obligations also require organisational measures no system provides. *This is not legal advice.*

| CAPABILITY | WHAT IT PRODUCES | OBLIGATIONS IT CAN HELP EVIDENCE |
|---|---|---|
| **Principal attribution** | Human, workload and agent identity on every material action | UK / EU GDPR Art. 5(2) accountability and Art. 32 security of processing · DORA Art. 9 access control · NIST AI RMF GOVERN 2 (accountability) |
| **Versioned policy decisions** | Policy ID, version, decision and attached obligations | GDPR Art. 5(2) and Art. 25 data protection by design · EU AI Act Art. 26(1), use in accordance with instructions · ISO/IEC 42001 AI policy and Annex A controls · NIST GOVERN 1 |
| **Data-aware routing and residency** | Route resolved against data class and permitted region | GDPR Art. 5(1)(c) minimisation, Art. 28 processors, Chapter V transfers · DORA Arts. 28–30 ICT third-party risk |
| **AI asset inventory** | Register of models, providers, gateways, MCP servers and agents | DORA Art. 28(3) register of information · GDPR Art. 30 records of processing · NIST MAP function · ISO/IEC 42001 resource documentation |
| **Human approval events** | Who approved, when, and under which policy version | EU AI Act Art. 14 human oversight and Art. 26(2) (high-risk systems) · UK GDPR Arts. 22A–22D automated decision-making safeguards, as inserted by the DUAA 2025 |
| **Execution evidence and retention** | Exportable record per request or run, retained under organisational control | EU AI Act Art. 12 record-keeping and Art. 26(6) deployer log retention of at least six months · DORA Arts. 17–19 incident management and reporting · NIST MANAGE function |
| **Central constraint and revocation** | Ability to narrow or withdraw AI authority immediately | DORA Art. 11 response and recovery · NIS2 Art. 21 risk-management measures · GDPR Art. 32(1)(b) ongoing resilience |

Figure 7. *Control-plane capabilities mapped to obligations they can support. Article references are pointers for counsel and compliance teams, not a compliance matrix.*

### ON THE EU AI ACT

Articles 12, 14 and 26 bite on high-risk systems, and depend on whether the organisation is a provider or a deployer. Most enterprise coding and chat uses will not be high-risk. The evidence record matters there for GDPR accountability and operational resilience, not AI Act conformity.

### WHY ONE RECORD HELPS

Different regimes ask overlapping questions in different words. One record answering who, what, under which policy, where and with what result can serve several of them, so long as its fields and retention are agreed with the teams that will rely on it.

◆ ENTERPRISE AI CONTROL FRAMEWORK

# Nine domains separate AI access from an enterprise-controlled AI estate

The framework combines two kinds of capability. Seven domains are controls applied on the path of an AI action. Two of them, AI asset inventory and lifecycle and accountability, are management capabilities that govern the wider estate rather than steps executed on every request.

| | | |
|---|---|---|
| 1<br>**IDENTITY AND ATTRIBUTION**<br>Every request belongs to a known person, service or agent. | 2<br>**AI ASSET INVENTORY**<br>Know which models, gateways, agents and MCP servers exist.<br>MANAGEMENT PLANE | 3<br>**POLICY-AS-CODE**<br>Material actions are evaluated against identifiable, versioned policy. |
| 4<br>**DATA BOUNDARIES**<br>Classification, location and handling constraints travel with the request. | 5<br>**MODEL AUTHORITY**<br>Applications request approved capabilities rather than hard-coded providers. | 6<br>**TOOL AUTHORITY**<br>MCP and tool operations are authorised individually, including read versus write. |
| 7<br>**AGENT EXECUTION**<br>Budgets, credentials, time limits and execution boundaries are explicit. | 8<br>**EVIDENCE AND OBSERVABILITY**<br>Every decision and material execution event is attributable and traceable. | 9<br>**LIFECYCLE AND ACCOUNTABILITY**<br>Retention, incident investigation, approval and evidence export are defined.<br>MANAGEMENT PLANE |

Figure 8. *The nine-domain control framework. Shaded domains are management-plane capabilities; the other seven apply on the path of an AI action.*




Figure 9. *The request-path domains in sequence. Evidence is written along the path, not assembled afterwards.*

◆ WHAT AN ENTERPRISE AI CONTROL PLANE SHOULD MEAN

# A practical definition

THE DEFINITION

An enterprise AI control plane is an organisation-controlled layer that mediates access to models, tools and agent execution so that each material action is *attributable*, *policy-evaluated*, *constrained* and *supported by independently controlled evidence*.

In practical terms, for any governed AI action, the organisation should be able to answer each question on the right from the stage on the left.

| REQUEST PATH | WHAT THE ORGANISATION CAN ANSWER |
|---|---|
| PRINCIPAL | Who initiated it |
| REQUEST | What they requested, and which data or resource was involved |
| POLICY | Which policy was evaluated, and at which version |
| OBLIGATIONS | Which constraints the decision attached: redaction, region, read-only, approval |
| MODEL / TOOL | Which model or tool was permitted |
| EXECUTION | What actually executed, and what happened |
| EVIDENCE | What evidence remains, written by the organisation-controlled path, not borrowed solely from suppliers |

Figure 10. *The control-plane request path, and the question each stage must be able to answer.*

A NOTE ON PLACEMENT

The control plane is not necessarily the model gateway. It can sit above, beside or integrate with existing gateway infrastructure. What defines it is the boundary it enforces, not the product category it is bought from.

◆ POLICY EVALUATION IN REAL TIME

# How a single tool operation is decided

The hard part of governing MCP is not the protocol. It is that authority is exercised one operation at a time, inside a session that was authorised once. A control plane evaluates each operation at the point it is dispatched, not only the session that preceded it.

1 **INTERCEPT**
The client's tool call reaches an organisation-controlled MCP proxy, not the server directly.

2 **ASSEMBLE INPUT**
Principal, agent run, server identity, tool, normalised arguments, resource and data class.

3 **EVALUATE**
A policy decision point, such as OPA or Cedar, evaluates the input against the current, versioned policy bundle, in-process or as a sidecar.

4 **DECIDE**
Allow, deny, or allow with obligations: redact, read-only, require approval.

5 **DISPATCH OR REFUSE**
Only allowed calls are forwarded, with a credential injected for that operation alone.

6 **RECORD**
Decision and outcome are written with the parent execution ID.

**POLICY INPUT**

| | |
|---|---|
| principal | `a.mehta · human` |
| via | `agent:code-assist` |
| server | `mcp:github@org · pinned` |
| tool | `create_pull_request` |
| resource | `repo:payments/ledger` |
| data_class | `confidential.source` |
| session | `ai.code.session@v14` |

**DECISION**

| | |
|---|---|
| decision | `DENY` |
| policy | `ai.code.repo-write@v6` |
| reason | `write tools not permitted in coding-assistant sessions` |
| dispatch | `not forwarded` |
| event | `evt_02d81e.14` |

Figure 11. *Evaluating one MCP operation. The deny decision shown is the one recorded in the evidence record on page 21. Illustrative values.*

## THE HARD PARTS

**Tool descriptions are not authority**

Servers declare their own tools, and descriptions can change or be written adversarially. Pin server identity and tool schemas; treat a changed schema as a new tool that needs approval.

**Arguments matter**

The same tool can be harmless or dangerous depending on path, repository or query. Policy needs normalised arguments, not just tool names, which means per-server adapters.

**Fail closed, deliberately**

If the decision point is unavailable, material operations are denied and the denial recorded. Any fail-open behaviour should be an explicit policy for a named class of operation, not a default.

◆ PREVENTING BYPASS

# Making the governed path authoritative

A control plane that can be routed around is a logging system with opinions. Bypass is prevented by layered controls, none of which is sufficient alone. The aim is that the governed path becomes easier than the alternative, and that departures from it become visible.

| LAYER | CONTROL | WHAT IT STOPS | LIMITS |
|---|---|---|---|
| NETWORK | Egress to model-provider and public MCP endpoints permitted only from the control plane, e.g. via Kubernetes NetworkPolicy or a mesh egress gateway (Istio, Cilium) | Direct SDK calls from applications and managed devices | Personal devices and off-network use |
| CREDENTIALS | Provider keys held only by the control plane; applications receive short-lived scoped tokens; tool credentials injected per operation from a secrets manager such as HashiCorp Vault, AWS Secrets Manager or Azure Key Vault | Shared keys, keys in code, standing tool credentials | Legacy keys must be found and rotated out |
| PROVIDER TENANCY | Enterprise tenant accepts traffic only from control-plane identity or addresses; personal sign-up blocked for corporate domains | Personal accounts using corporate identity | Support varies by provider |
| PLATFORM | Governed client configuration distributed through device management (Intune, Jamf) and development images; MCP servers registered centrally | Locally configured servers and endpoints | Users with local administrator rights |
| DETECTION | Egress monitoring, OpenTelemetry traces, secret scanning and spend reconciliation against evidence records | Residual bypass, made visible | Detective, not preventive |

Figure 12. *Layered bypass controls. Each layer closes a route the others leave open.*

## EVOLVING THE CONTROL PLANE ITSELF

*Models and protocols will change faster than policy*

**Policy bundles are versioned and signed.** Every decision records the bundle version, so a later change never rewrites what was decided earlier.

**Model aliases decouple applications from SKUs.** Applications ask for coding.standard; alias resolution is a governed change, recorded per request.

**The evidence schema is versioned.** Changes are additive, exports declare their schema version, and readers understand earlier versions.

**Protocol adapters are isolated.** A new MCP revision or provider API is an adapter change behind the proxy, not a policy change for every team.

◆ LONG-RUNNING AGENTS

# Coherent evidence for multi-step execution

A single request has one decision and one outcome. An agent run may last hours, call dozens of tools, start sub-agents and stop halfway. The evidence trail has to stay coherent across all of it, including when nobody is sure what happened.

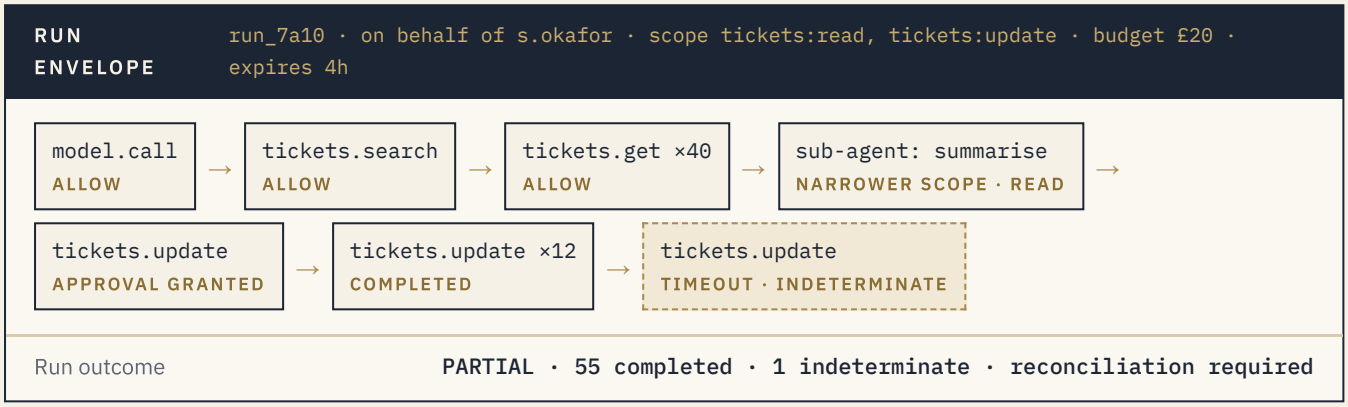



Figure 13. *An agent run as one envelope. Every step is a child event that inherits, and can only narrow, the run's authority. Illustrative values.*

**Delegation is explicit.** A run is created with a principal, scope, budget and expiry. Sub-agents receive a subset, never a superset.

**Every step is a child.** Events carry run and parent IDs, so the run can be rebuilt as a tree, sub-agents included.

**Authority is re-checked at the point of action.** Not only at run start, so revocation takes effect mid-run.

**Outcomes have four states.** Completed, failed, denied and indeterminate. The last is the one most systems omit.

## WHEN EXECUTION IS AMBIGUOUS

| SITUATION | WHAT THE RECORD SHOULD SAY | WHAT SHOULD HAPPEN NEXT |
|---|---|---|
| **Tool call timed out after dispatch** | Indeterminate: dispatched, no confirmation received | Reconcile against the target system before any retry; writes carry an idempotency key |
| **Run terminated mid-plan** | Partial: the completed child steps, listed | Surface to the run's owner. Compensating actions are new, separately authorised operations |
| **Retry produced a duplicate effect** | Two child events sharing one idempotency key | Downstream de-duplication, itself recorded |
| **Policy changed during the run** | Each step records the version it was evaluated against | Long runs re-evaluate; write authority is not grandfathered |

◆ THE EVIDENCE RECORD

# One AI action. One defensible record.

The most concrete expression of governance is not a policy document or a dashboard. It is an exportable record for one request or one agent run, written at the point of decision and execution, that answers the questions in this paper without depending on the supplier's own logs.

**EVIDENCE RECORD** exe_02d81e · illustrative

**WHO**

| | |
|---|---|
| Principal | a.mehta (human, SSO) |
| Team / workload | payments · coding-assistant |

**CORRELATION**

| | |
|---|---|
| Request ID | req_7f3c9a21 |
| Trace ID | 4bf92f3577b34da6 |
| Parent execution | exe_02d81e |

**WHAT**

| | |
|---|---|
| Action | code.assist.session |
| Resource | repo:payments/ledger |
| Data classification | confidential.source |

**ROUTE**

| | |
|---|---|
| Requested alias | coding.standard |
| Resolved provider | provider-a · eu-west |
| Resolved model | model-x-2026-05 |

**PARENT DECISION · SESSION**

| | |
|---|---|
| Policy ID | ai.code.session |
| Policy version | v14 · 2026-06-12 |
| Decision | **ALLOW** |
| Obligations | read-only tools; redact secrets |

**EXECUTION**

| | |
|---|---|
| Approval events | none required |
| Execution outcome | **COMPLETED · 1 child denied** |

**ACCOUNTING**

| | |
|---|---|
| Token usage | 184,220 in · 9,410 out |
| Cost | £2.71 |
| Timestamp | 2026-07-03T09:14:22Z |
| Retention state | retained · to 2027-07-03 |

**CHILD TOOL DECISIONS · EVALUATED PER OPERATION** parent: exe_02d81e

| EVENT ID | OPERATION | POLICY · VERSION | DECISION |
|---|---|---|---|
| evt_02d81e.01–12 | get_file ×12 | ai.code.repo-read · v14 | **ALLOW** |
| evt_02d81e.13 | list_issues | ai.code.repo-read · v14 | **ALLOW** |
| evt_02d81e.14 | create_pr | ai.code.repo-write · v6 | **DENY · not forwarded** |

**Figure 14.** *An evidence record for the scenario on page 6. The session decision and each tool decision are recorded separately; child events carry the parent execution ID. Outcomes may also be failed or indeterminate. Illustrative values; field names are indicative.*

◆

*Evidence written by the control plane, not borrowed from the provider.*

**ON INTEGRITY**

This paper describes independently controlled evidence: records written and retained by an organisation-controlled system, under the organisation's own access controls and retention policy. Provider logs remain useful corroboration; they are not the primary record. Stronger guarantees, such as cryptographic tamper evidence, are a separate design decision and should be specified explicitly where they are required.

◆ ENTERPRISE AI MATURITY MODEL

# From unmanaged, to managed, to governed

Many enterprises will recognise elements of the first two levels, often at different levels across different teams. The third level is where an AI action can be explained in terms the organisation itself controls.

| | LEVEL 1<br>Unmanaged AI | LEVEL 2<br>Managed AI | LEVEL 3<br>Governed AI |
|---|---|---|---|
| IDENTITY | Personal account / API key | SSO | **Human + workload + agent attribution** |
| MODELS | User choice | Approved list | **Policy-bound aliases** |
| DATA | Unclassified | Guidelines | **Classification drives routing** |
| POLICY | Documents | Gateway rules | **Versioned policy-as-code** |
| TOOLS | Ad hoc | Approved MCP servers | **Per-operation authority** |
| AGENTS | Unbounded | Basic controls | **Scoped execution** |
| EVIDENCE | Provider logs | Central logs | **Independently controlled evidence** |
| COST | Unknown | Aggregate | **Attributed per team and action** |

Figure 15. *The enterprise AI maturity model. Level 2 manages access; Level 3 governs execution.*

◆

*Managing AI access usually arrives before governing AI execution.*

◆ MEASURING PROGRESS

# Leading indicators for each level

The maturity model on the previous page describes states. The indicators below show movement between them. Each can be computed from data a control plane, a gateway or a finance system already holds.

| INDICATOR | DEFINITION | SIGNALS |
|---|---|---|
| **AI inventory coverage** | Known providers, models, MCP servers and agents, as a share of those found by egress and spend review | Level 1 → 2 |
| **Enterprise-account share** | Share of AI spend on enterprise-managed accounts rather than personal or expensed ones | Level 1 → 2 |
| **Governed coverage** | Share of in-scope AI calls, by volume and by spend, passing through the governed path | Level 2 → 3 |
| **Evidence completeness** | Share of AI actions whose evidence record has every mandatory field populated | Level 2 → 3 |
| **Time to evidence** | Median time from a question being asked to an exported record for one request or run | Level 2 → 3 |
| **Bypass rate** | AI calls or spend observed outside the governed path, as a share of the total | Level 2 → 3 |
| **Tool decision coverage** | Share of tool operations with an explicit, recorded policy decision | Level 3 |
| **Indeterminate rate** | Share of agent runs ending with at least one indeterminate step, and time to reconcile them | Level 3 |
| **Exception age** | Median age of open policy exceptions. A rising figure signals stale policy | Level 3 |
| **Time to first governed call** | Time for a new team to make its first call through the governed path | All levels |

Figure 16. *Leading indicators by maturity transition. The last is the best single test of whether the governed path is also the easiest one.*

**TRENDS, NOT THRESHOLDS**

The direction is what matters: governed coverage and evidence completeness rising; bypass rate, time to evidence and exception age falling. Fixed targets are better set by each organisation against its own baseline than borrowed from a paper.

◆

*If time to evidence is not falling, the control plane is producing logs, not answers.*

◆ PRIORITISING THE NINE DOMAINS

# Where to start depends on what the path can do

The nine domains do not carry equal weight for every path. A coding assistant's risk sits in identity, model route and repository access; a regulated-data assistant's in classification and retention; an agent's in tool authority and execution bounds.

| | DOMAIN | CODING ASSISTANT ON INTERNAL CODE | REGULATED DATA IN CHAT OR RAG | EARLY AGENT WITH WRITE TOOLS |
|---|---|---|---|---|
| 1 | Identity and attribution | ● | ● | ● |
| 2 | AI asset inventory | ◐ | ◐ | ● |
| 3 | Policy-as-code | ◐ | ● | ● |
| 4 | Data boundaries | ◐ | ● | ◐ |
| 5 | Model authority | ● | ● | ◐ |
| 6 | Tool authority | ● | ○ | ● |
| 7 | Agent execution | ○ | ○ | ● |
| 8 | Evidence and observability | ● | ● | ● |
| 9 | Lifecycle and accountability | ○ | ● | ◐ |

● First ◐ Next ○ Later

Figure 17. *Indicative priority of each domain by risk profile. Identity and evidence come first in every case; everything else depends on the authority the path holds.*

## WHAT 'GOOD ENOUGH' LOOKS LIKE

**SMALLER · ONE PROVIDER · LOW REGULATORY EXPOSURE**

Level 2 across the estate. Evidence records only for paths that touch confidential or personal data. A dedicated control plane is probably not justified.

**MID-SIZED · SEVERAL PROVIDERS OR TOOL ACCESS**

Level 3 for the one or two paths with the most authority, typically coding assistants with MCP and any agent. Level 2 elsewhere, reviewed annually.

**LARGE OR REGULATED · AGENTS IN PRODUCTION**

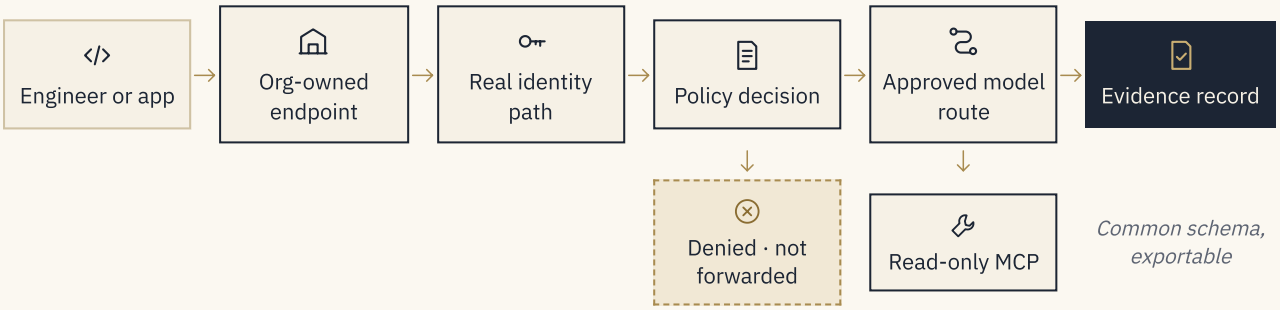
Level 3 as the default for new AI paths, with documented, expiring exceptions. Evidence export rehearsed as part of incident exercises.

◆ THE GOVERNANCE DIVIDEND

# Control that can make adoption faster

| | | |
|---|---|---|
| **FASTER SANCTIONED ADOPTION**<br>Teams get a paved path instead of waiting for one-off approvals. | **LESS SHADOW AI**<br>The governed route becomes easier than unmanaged alternatives. | **EASIER PROVIDER SUBSTITUTION**<br>Applications depend on organisational capabilities, not vendor SKUs. |
| **LESS DUPLICATED ENGINEERING**<br>Identity, policy and evidence are not reinvented per application. | **FASTER INVESTIGATION**<br>The evidence already exists when questions arrive. | **BETTER COST ATTRIBUTION**<br>Model and tool usage is attributable to teams and workloads. |

**MINIMUM VIABLE CONTROL PLANE**




**On bypass.** A deny record evidences that the governed route did not dispatch the operation. Preventing bypass requires the governed route itself to be made authoritative through network, credential or platform controls.

Figure 18. *The minimum viable control plane: one of each, end to end, including a deny path.*

**ON DAY ONE**

One organisation-owned endpoint
One real identity path
One approved model route
One policy decision service
One read-only MCP integration
One deny path that evidences the governed route did not dispatch the operation
One common evidence schema
One exportable evidence record

**NOT NEEDED ON DAY ONE**

Every model provider
Every MCP server
Autonomous agents
Enterprise-wide rollout
A giant governance programme

**WHAT ACTUALLY MATTERS**

Demonstrable control over one real AI path, end to end.

◆ IMPLEMENTATION FRICTION

# What goes wrong, and what it costs

The governance dividend on the previous page is real only if the control plane is adopted and kept accurate. Most failures are not technical. They are drift, workaround and ownership, and each has an early sign that can be watched for.

| FAILURE | WHAT IT LOOKS LIKE | EARLY SIGN | MITIGATION |
|---|---|---|---|
| **Stale policy** | Rules written for last year's models and tools; exceptions granted 'temporarily' become permanent | Exception count rising; old exceptions unreviewed | A named owner per policy domain; exceptions expire; review on every model or tool onboarding |
| **Evidence schema drift** | Teams add their own fields; exports from different paths stop joining | Investigations need per-team special cases | One versioned schema; additive changes only; conformance tests in CI |
| **Developer workaround** | The governed path is slower, narrower or harder to join than a personal key | Provider spend outside the control plane; secret-scanning hits | Make the governed path the tooling default; measure time to first governed call |
| **Ownership disputes** | Platform builds it, Security writes policy, application teams carry the pager; nobody owns exceptions | Routine decisions escalating to committees | Agree ownership before the pilot: Platform owns the path, Security the policy, Risk the evidence requirements |
| **Divergent semantics** | Identity, data class or model alias mean different things in different components | The same action allowed on one route, denied on another | A shared vocabulary defined in code; contract tests across components |
| **Single point of failure** | An outage stops all AI use, creating pressure to fail open permanently | Emergency bypass flags left switched on | Availability objectives; explicit fail-open policy per operation class; bypass use recorded |

Figure 19. *Common failure modes. None is exotic; most appear within the first year of operation.*

### THE ENGINEERING COST

Expect a running cost, not a one-off build: a small platform team to operate the path, policy-authoring time from Security, and integration work each time a provider, model or MCP server is added. Keeping identity, policy, routing, tool authority and evidence semantics consistent is the hardest part, and it never finishes. For an estate with a handful of low-authority paths, this can exceed the benefit (page 12).

### MAKING THE GOVERNED PATH THE EASIEST PATH

- Self-service onboarding measured in hours, not weeks
- SDKs, CLIs and IDE configuration point at the governed endpoint by default
- Broader model and tool access inside the path than outside it
- Denials explain themselves and say how to request access
- Cost and usage reports that teams actually want

◆ WHERE TO START

# Five day-one AI paths

The first governed path should be chosen for what it demonstrates, not for its ambition. Each of the paths below is likely to exist in some form already, and each has an identifiable owner who can approve a pilot. Use the table to choose one, and to scope what must be governed first.

| | PATH | WHY IT IS A GOOD START | WHAT TO GOVERN FIRST |
|---|---|---|---|
| 1 | **CODING ASSISTANTS**<br>Claude Code or Copilot against internal repositories. | High value and already adopted. Touches sensitive intellectual property, with a clear engineering owner. | Identity · Model route · Repository access · Read / write distinction · Evidence |
| 2 | **INTERNAL AI CHAT**<br>Approved model access for employees handling company data. | Broad employee usage across functions, and a common route by which company data reaches a model. | Identity · Data classification · Model / provider route · Retention |
| 3 | **READ-ONLY MCP**<br>GitHub, Jira, documentation or internal knowledge sources. | A clear, contained demonstration of tool authority before any write operation is allowed. | Server identity · Per-operation policy · Read vs write · Evidence |
| 4 | **SENSITIVE DOCUMENT ANALYSIS**<br>Legal, compliance, risk or research workflows. | Sensitive enterprise content with explicit handling requirements and owners who already ask these questions. | Classification · Provider route · Residency · Retention |
| 5 | **EARLY AGENTIC WORKFLOW**<br>One constrained agent with explicitly bounded tools and credentials. | Delegated authority is visible. Establishes the execution controls later agents will need. | Tool scope · Credentials · Budget · Time limit · Approval · Termination |

Figure 20. *Choosing a first governed path. Each column on the right is a starting scope, not a complete control set.*

> Pick one workflow where AI is already useful and where the consequences of uncontrolled authority are easy to understand.

◆ ENTERPRISE AI CONTROL READINESS CHECK

# Eleven questions, answered against one real path

AI PATH ASSESSED _____ COMPLETED BY _____ DATE _____

| | CONTROL QUESTION | YES | PARTIAL | NO | OWNER | ACTION REQUIRED |
|---|---|---|---|---|---|---|
| 1 | Can we identify the human, service or agent behind every AI action? | ☐ | ☐ | ☐ | | |
| 2 | Can we show which models are permitted for which data? | ☐ | ☐ | ☐ | | |
| 3 | Can we show which policy version allowed or denied a request? | ☐ | ☐ | ☐ | | |
| 4 | Can we distinguish the requested model from the provider and model actually used? | ☐ | ☐ | ☐ | | |
| 5 | Can we control individual MCP or tool operations? | ☐ | ☐ | ☐ | | |
| 6 | Can we evidence that the governed path did not forward a denied operation? | ☐ | ☐ | ☐ | | |
| 7 | Can we prevent users, applications or agents from bypassing the governed path? | ☐ | ☐ | ☐ | | |
| 8 | Can we identify ambiguous or partially completed execution? | ☐ | ☐ | ☐ | | |
| 9 | Can we produce an independently controlled evidence record for one request or agent run? | ☐ | ☐ | ☐ | | |
| 10 | Can we revoke or constrain AI access centrally? | ☐ | ☐ | ☐ | | |
| 11 | Do Platform, Security and Risk agree where the enterprise AI trust boundary sits? | ☐ | ☐ | ☐ | | |

**Figure 21.** *A readiness check to complete internally against one real AI path, before any external conversation.*

◆

*If several answers are 'No', this is not primarily an AI capability problem.*
*It is a control-plane problem wearing the costume of an AI problem.*

◆ CLOSING ARGUMENT

# From access to authority

The first phase of enterprise AI adoption was about ***access***. Employees gained chat assistants. Engineers gained coding tools. Applications gained model APIs. The cost of powerful AI capability continues to fall.

The next phase is about ***authority***.

> *Which model may see which data?*
> *Which tools may an agent invoke?*
> *Which actions require approval?*
> *Which provider actually processed the request?*
> *Can the organisation demonstrate the answer later, without relying entirely on the supplier involved?*

The issue is no longer whether enterprise AI works. It is whether enterprise AI operates inside a boundary the organisation itself controls.

◆

*The danger is not unusable AI.*
*The danger is useful AI activity that nobody can later demonstrate was authorised.*

## WHAT TO DO NEXT

1

### COMPLETE THE READINESS CHECK

Against one real path, internally, with Platform, Security and Risk in the same room (page 28).

2

### DECIDE WHAT IS PROPORTIONATE

If most answers are Yes, or the path holds little authority, extend what you already run (page 12). Not every path needs a control plane.

3

### GOVERN ONE PATH END TO END

Where several answers are No on a high-authority path, build the minimum viable control plane for that path alone (page 25), and measure it (page 23).

**ABOUT THIS PAPER**

Parakletos AI Limited designs and builds enterprise AI control planes, so this paper argues for an approach in which we have a commercial interest. We have tried to make it useful whether or not you work with us, including on when not to build one. Readers who would like an external review of a single AI path can reach us at parakletos.co.uk or info@parakletos.co.uk.

◆ REFERENCES AND RELEVANT OBLIGATIONS

# Sources referred to in this paper

*A general summary of the obligations, standards and sources referred to in this paper, not legal advice. Whether and how each applies depends on role, sector, data and circumstances.*

## LAW AND REGULATION


1. **UK GDPR and Data Protection Act 2018**: accountability, Article 5(2); data minimisation, Article 5(1)(c); processors, Article 28; records of processing, Article 30; security of processing, Article 32; data protection impact assessment, Article 35.
2. **Data (Use and Access) Act 2025**: amends the UK GDPR, DPA 2018 and PECR. Royal Assent 19 June 2025; commenced in stages, with most Part 5 data protection changes in force from 5 February 2026 under the Commencement No. 6 Regulations 2026.
3. **FCA PS21/3 and PRA SS1/21**: operational resilience: important business services and impact tolerances; transitional period ended 31 March 2025. **PRA SS2/21**: outsourcing and third-party risk management; **FCA SYSC 8**: outsourcing.
4. **EU GDPR, Regulation (EU) 2016/679**: accountability, Article 5(2); processors, Article 28; records of processing, Article 30; security, Article 32; transfers to third countries, Chapter V.
5. **EU AI Act, Regulation (EU) 2024/1689**: record-keeping, Article 12; human oversight, Article 14; deployer obligations including log retention, Article 26; transparency, Article 50. General-purpose AI model obligations apply from 2 August 2025 and Article 50 from 2 August 2026. As amended by Regulation (EU) 2026/1744 (Digital Omnibus on AI), Annex III high-risk obligations apply from 2 December 2027 and Annex I from 2 August 2028.
6. **Digital Operational Resilience Act (DORA), Regulation (EU) 2022/2554**: ICT risk management, ICT third-party risk and the register of information, and major ICT-related incident reporting; applies from 17 January 2025.
7. **NIS2 Directive, (EU) 2022/2555**: cybersecurity risk-management measures including supply-chain security, Article 21; incident reporting, Article 23. Applies to in-scope essential and important entities through national transposing law; transposition remains incomplete in some Member States, and the Commission referred four to the CJEU in July 2026.


## STANDARDS AND TECHNICAL GUIDANCE


8. **NIST AI Risk Management Framework (AI RMF 1.0)**, January 2023: the Govern, Map, Measure and Manage functions.
9. **ISO/IEC 42001:2023**: artificial intelligence management systems.
10. **OWASP Top 10 for LLM Applications, 2025**: including excessive agency (LLM06) and sensitive information disclosure (LLM02).
11. **Model Context Protocol specification, revision 2026-07-28**: authorisation and security best practices for MCP clients and servers. modelcontextprotocol.io/specification


## MARKET AND ADOPTION CONTEXT


12. **Stack Overflow, 2026 Developer Survey**: AI section: 66% of respondents use AI coding assistants or coding agents at work; 26% use AI agents or automated workflows. survey.stackoverflow.co/2026/ai
13. **McKinsey & Company, The state of AI in 2025: Agents, innovation, and transformation**, November 2025: 88% report regular AI use in at least one business function; 23% are scaling an agentic AI system and 39% experimenting.
14. **Stack Overflow, 2025 Developer Survey**: 84% of respondents using or planning to use AI tools in development, up from 76% in 2024; 46% distrust the accuracy of AI output.


◆

*This paper is general guidance on enterprise AI governance and does not constitute legal advice.*

Parakletos AI Limited · London