The AI Deployment Policy

AI governance as a runway to launch.

Original Publication Date	10/26/2026
Revised	
Version	1.1
Author	Jeremy Taylor

© 2026 Jeremy Taylor · CC BY 4.0

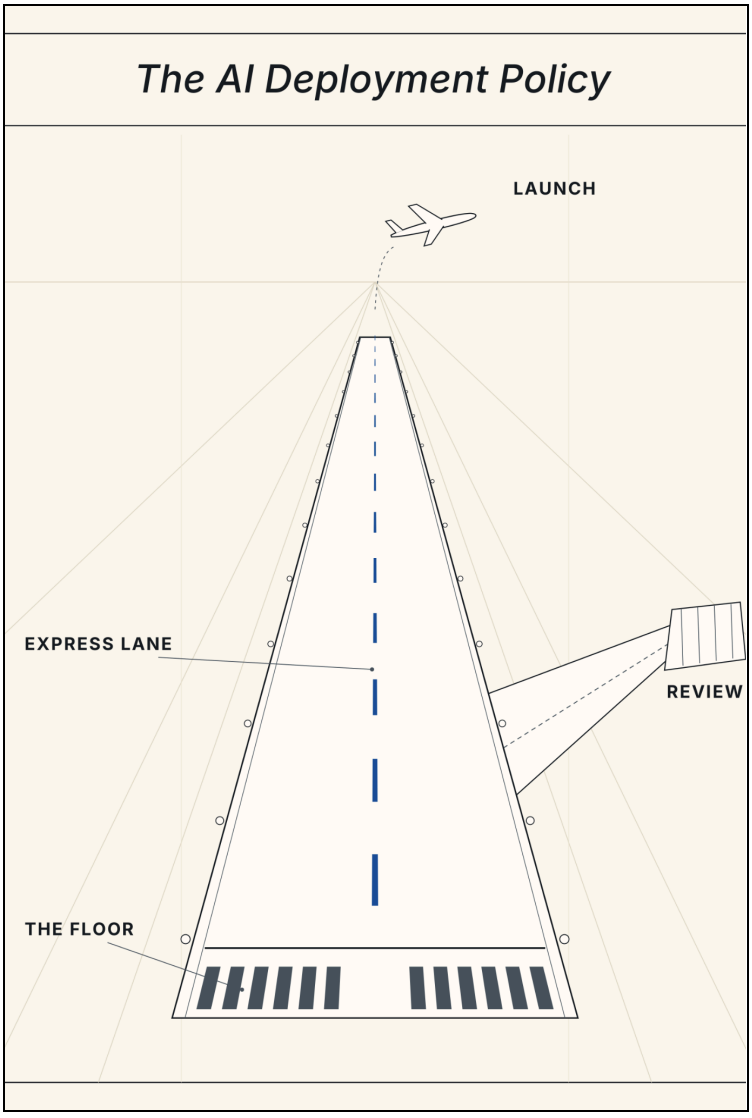


Table of Contents

Introduction 3
Part 1: The Model 5
Part 2: Risk 10
Part 3: Controls 15
Part 4: Decisions and Lifecycle 17
What Changes 21
Continuing the Conversation 21
About the Author 21
Appendix A: Adapting This to Your Organization 22
Appendix B: Starter Prohibited Uses 22
Appendix C: Starter Standing Positions 23
Appendix D: Registry Fields 24
Appendix E: Glossary 24
Sources and Influences 25
License 27

Introduction

As AI moves into higher-value work, risk rises with it. Audiences get larger, decisions multiply, and the stakes get bigger. Without a framework, one of two things happens: risk goes unmanaged or deployments stall.

Both cost the organization. Risk that isn't mitigated comes back as a problem the organization has to tackle, with costs. Or teams assume they can't build an innovation, because no one gave them the requirements to make it safe to deploy.

The intent of a deployment policy is to answer the questions people ask about AI governance and to provide a runway to launch, so teams can plan, prepare, and build. AI Governance is about enabling deployment while balancing risk with mitigation.

The goal: 80% or more of AI applications launch without waiting for approval. They clear the Express lane or Moderate self-service, so review time goes to the few that carry real risk. If far fewer clear, the policy is set wrong, not the teams (§4.3).

This is a companion to Pillar 6 of The Enterprise AI Operating Model, and it builds on The Federated Data Governance Program.

How to Use This

Start with a few use cases your organization is considering, and map them to this model. Rate the ten Risk Factors in §2.1. Identify which cases clear the Express lane, need Settings dialed up, or go to a reviewing body.

The document has four parts: The Model, Risk, Controls, and Decisions and Lifecycle. Figure 1 shows how they connect. Two tables carry the core method: The Ten Risk Factors (§2.1) and the Twelve Controls (§3.1). The rest of the document explains them.

Everything specific is a placeholder and freely adaptable to your organization. Appendix A says what to swap and what to keep.

Companion Documents

This policy builds on two of my earlier guides at ossnetworks.com. You don't need to read them first. Where this document borrows a term, here is what it means.

The Enterprise AI Operating Model organizes AI work into six pillars, and "Pillar 2" here means the second of them:

- Pillar 1, Strategy & Value Realization: value hypotheses and the investment funnel
- Pillar 2, AI FinOps: cost tagging and spend
- Pillar 3, Data Management & Intelligence: the data catalog
- Pillar 4, Technology & Operations: building and releasing, including regression testing
- Pillar 5, People & Enablement: adoption
- Pillar 6, Risk, Ethics, & Policy: the Usage Policy, the data classifications (Public, Internal, Confidential, Restricted), and this Deployment Policy

The Federated Data Governance Program sets how data is owned and trusted. Four of its terms appear here:

- **Data Owner:** the named person accountable for a data asset. Each AI Application also has an Owner (§4.1).
- **Tier:** how important a data asset is, set by its Owner. Tier 1 is business-critical, Tier 2 is shared beyond its team, and Tier 3 is local. Each concept has one Tier 1 asset, its authoritative source.
- **Certification:** how far you can rely on a data asset: Bronze, Silver, or Gold.
- **The data Council:** the program's cross-domain governing body.

Where to Start

Start with one application and one page.

Pick an AI application that is already wanted, used by a small team, and read-only. Fill in its Registration Record (§1.3) and rate the ten Risk Factors (§2.1). If every rating sits in the Low column, it clears the Express lane and runs at the Floor. That first registration is the worked example every later team copies.

Then work outward: find what already runs, give it one quarter to register, and issue model credentials only to registered applications (§4.6).

Write the one-page Risk Appetite statement (§4.4) before the first High or Critical application arrives. The reviewing body will need it.

Part 1: The Model

1.1 What the Policy Covers

The Deployment Policy governs **AI Applications**. An AI Application is one deployed thing with one Owner: an agent, a workflow, an LLM feature inside a product, or a shared no-code assistant. Each one links to a value hypothesis.

Bought AI counts too. An AI feature inside a vendor's product registers like one built in-house. The internal team that configures and runs it is the Builder, and any Control the vendor cannot meet is recorded as a gap.

A use case is different. It is an idea in the Pillar 1 investment funnel, and one use case can produce several AI Applications with different risk. The policy registers the applications.

Personal use of an enterprise AI tool stays under the **Usage Policy** from Pillar 6: company data only in tools under a vendor agreement and never with Restricted data. Something leaves the Usage Policy and enters the Deployment Policy the moment either of these is true:

- someone other than the builder uses it, or
- it acts on a system or runs without a person starting it.

An analyst using an enterprise chat tool for her own work is under the Usage Policy. The same analyst sharing a custom assistant over a sales dataset with twelve colleagues registers it.

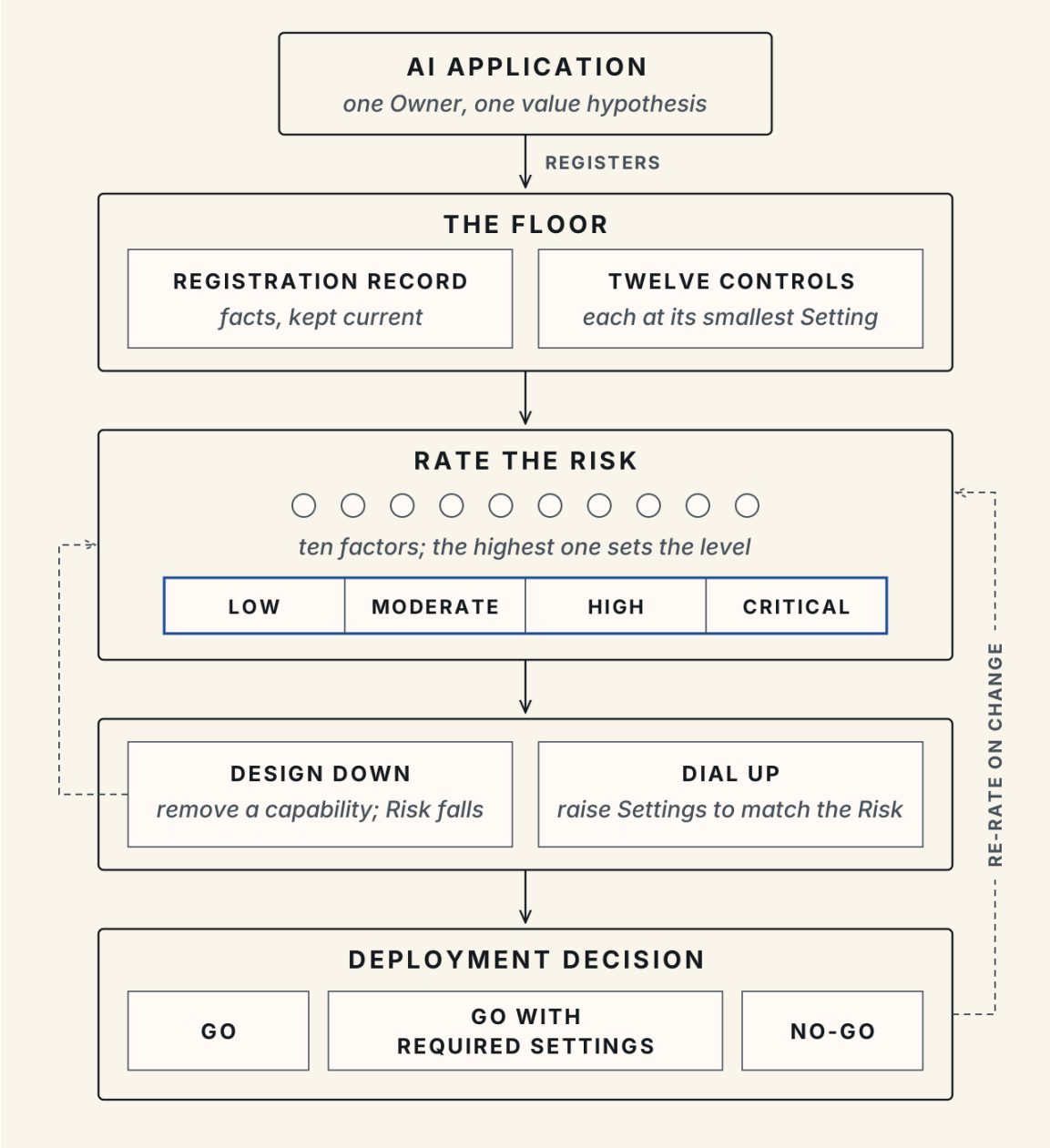


Figure 1. One path for every application.

1.2 The Registry

The **Registry** is the system of record for AI Applications. It is its own application, with a schema, an API for engineering workflows, and a UI for owners and reviewers. It links to the data catalog's assets, and through those links it reads each input's Tier and Certification. It

does not copy them.

The catalog is not the Registry. An AI Application needs a richer record than a general catalog holds: its value hypothesis, its autonomy, its tools, its privileges, its controls, and its review history. Catalogs may grow an object type for this. Until they do, the Registry stands beside the catalog.

1.3 The Floor

Every AI Application meets the **Floor** before it runs. The Floor has two halves.

The Registration Record. Facts recorded once and kept current:

- the Owner and the Builder (§4.1)
- the linked value hypothesis
- the AI model in use
- the ratings for the ten Risk Factors
- the classifications of the data it reads
- FinOps labels
- lifecycle status

All of them must be complete to register. This half is where Pillars 1 and 2 meet the policy: no value hypothesis and no cost tags means no registration, so every dollar answers to a value hypothesis at the door.

The Controls. Twelve mechanisms every application carries, each present at its minimum **Setting** (§3.1). The Floor version of each is small and real: one regression scenario, a scheduled judge run on a sample, every run logged. The mechanism exists in the deployment model from day one. Dialing a Setting up means adding to it, not building it.

1.4 Risk and Control Settings

Risk is how much harm an AI Application could do if it did the wrong thing and nothing stopped it. It is rated on ten **Risk Factors** (§2.1) before any Controls are counted, and it lands on one of four levels: Low, Moderate, High, Critical.

Risk sets the **Required Settings**: how far each Control must be dialed up before the application may run (§3.1). Low needs the Floor. Each level above it dials some Controls up.

Settings also rise on their own. A team adds regression scenarios as it learns. A reviewer asks for more human review. Both are normal. Required Settings are the minimum for the Risk level, never the ceiling of good practice.

1.5 Design Down or Dial Up

There are two honest ways to make an application acceptable.

Design down. Remove a capability, and Risk falls. Drop write access. Remove outbound email. Cap a refund at $50 in code. Each change lowers a factor's rating and with it the Risk.

Dial up. Raise a Setting, and the Risk is matched with mitigation. Add human approval. Expand evals. Tighten privileges.

Take a support agent that can issue refunds up to $500. Requiring approval on every refund dials up Human review, and the Risk stays the same. Capping refunds at $50 in code lowers the Risk. Both work. The Registry records which one the team chose.

Designing down is usually cheaper. A smaller capability needs fewer controls, which makes it the minimally invasive path. Figure 2 shows both paths.

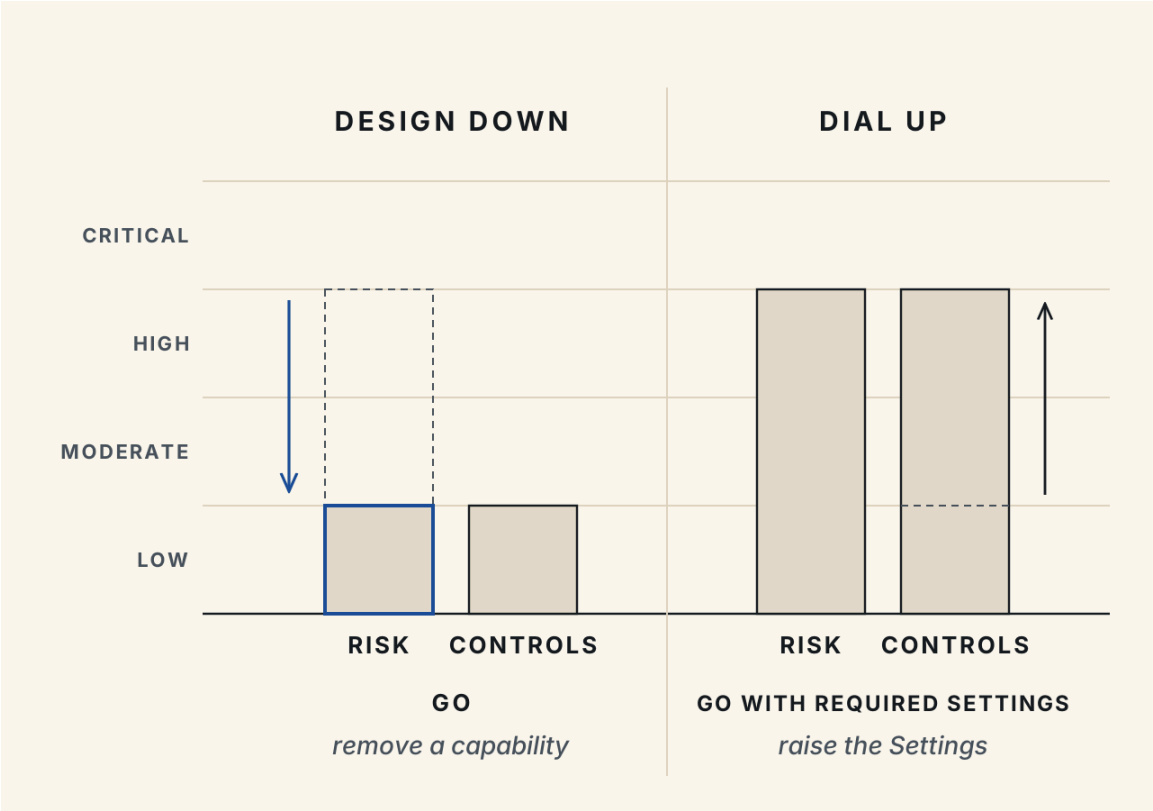


Figure 2. Design changes lower Risk. Settings answer it.

1.6 Go or No-Go

Applying the policy to an AI Application produces one of three outcomes.

- **Go.** Risk is Low and no Review Trigger fires. The Floor is enough.
- **Go with Required Settings.** The Controls are dialed up to what the Risk requires, and the application runs.
- **No-go.** The Risk exceeds the organization's Risk Appetite (§4.4), and no Settings can contain it.

1.7 Principles

Principle	What it means
Every application registers	If it runs and someone besides its builder relies on it, it is in the Registry. Credentials follow registration.
The Floor is real	Every Control exists from day one at a small Setting. Dialing up is adding, never starting.
The highest factor wins	Risk is the worst single rating. Nothing averages it away.
Design down before dialing up	Removing a capability is usually cheaper than controlling it.
The business owns the risk	The Owner is the business leader where the application acts, and holds value and risk together.
Most applications should flow	If fewer than 80% of applications clear the Express lane or Moderate self-service, the lanes are set wrong.

Part 2: Risk

2.1 The Ten Risk Factors

Each factor is a question about the application, and each one is rated Low to Critical on its own. A cell marked Review also sends the application to a named reviewer, whatever its Risk level (§2.5).

The factors are not exhaustive. They cover the situations that come up most often, not every one. When an application doesn't fit the table, rate its Risk in the spirit of it: ask how much harm the application could do if nothing stopped it, and choose the level that matches. Judgment can raise the Risk above the table's rating, never lower it (§2.3).

#	Factor	Low	Moderate	High	Critical
1	Audience reach	the builder's team	one department	enterprise-wide, or named external partners and customers under contract *Review: reviewing body (external only)*	the open public, or anyone without a contract *Review: reviewing body*
2	Audience type	staff using it for their own work	staff acting on it for others	business customers, partners *Review: reviewing body*	consumers or vulnerable people *Review: reviewing body*
3	Decision consequence	informational	internal operations	money, customers, regulators	a person's rights, access, or opportunity *Review: legal*
4	Reversibility	undone instantly	fixable with effort	hard to undo	cannot be undone
5	Action authority	answers	recommends	acts with approval	acts on its own
6	Trigger	a person asks	scheduled	event-driven	continuous
7	Lethal trifecta legs (§2.4)	none	one	two	all three *Review: security*
8	Data sensitivity	Public, Internal	Confidential	Confidential with personal data	Restricted
9	Identity	the requesting user	a narrowly scoped service identity	a broad service identity	a privileged or admin identity
10	Tools and memory	retrieval only, no memory	a few internal tools, session memory	many tools or any write tool, persistent memory	tools or agents outside your control, or memory that outside content can write to

Regulated domains (financial controls, health privacy, fair lending, EU AI Act high-risk uses) go to legal whatever the ratings above. §2.5 lists every Review Trigger.

Four ratings are easy to misread:

- **Decision consequence, High:** a wrong answer reaches customers, money, or regulators. Below that, mistakes stay inside the business.
- **Trigger** matters as much as capability. A simple assistant that runs on a schedule acts with nobody watching. A sophisticated agent that only answers when asked has a person there every time.
- **Lethal trifecta legs** counts how many of three capabilities the application holds: access to sensitive data, exposure to untrusted input, and a way to send data out (§2.4).
- **Data sensitivity, Restricted** rates Critical, but it is not banned. The Usage Policy keeps Restricted data out of everyday AI tools. A purpose-built application with a limited audience can use it after Critical review.

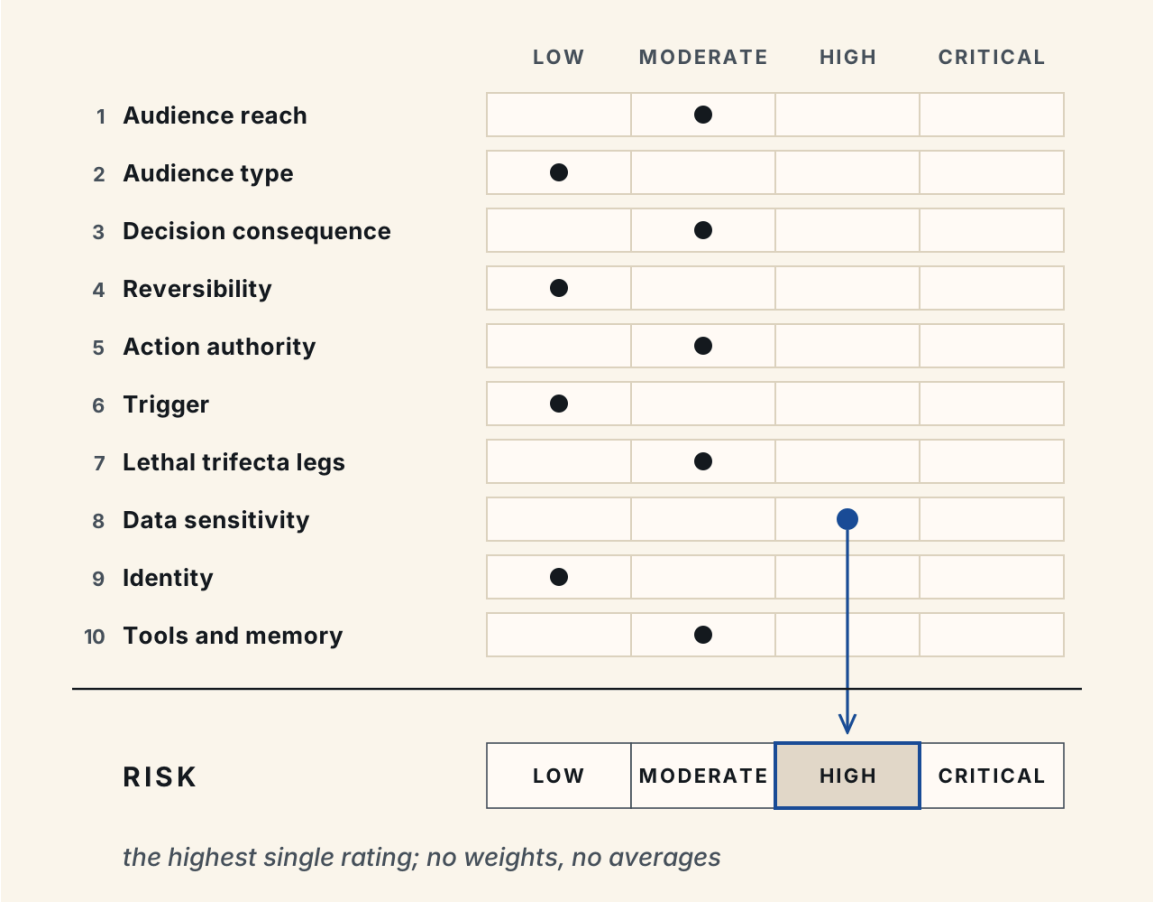


Figure 3. Risk is the highest single rating.

2.2 The Four Risk Levels

Risk	Plain test
Low	A wrong answer costs someone a minute.
Moderate	A wrong answer costs the business something it can fix.
High	A wrong answer reaches customers, money, or regulators.
Critical	A wrong answer harms a person or cannot be undone, at scale.

2.3 The Highest Factor Wins

Risk is the highest level any single factor reaches. There are no weights and no averages. Ten Low ratings do not add up to Moderate, and one Critical rating is not diluted by nine Low ones. Figure 3 shows an example.

Judgment only ratchets up. The Owner, the reviewing body, or legal may raise Risk above the rating for reasons the factors miss. A reasonable test: would the organization be comfortable seeing this application on the front page? Nobody may lower Risk below the rating. The only way down is a design change that lowers a factor's rating.

2.4 The Lethal Trifecta

Three capabilities are manageable in pairs and dangerous together:

- access to sensitive data
- exposure to untrusted input (web pages, email, documents, other agents)
- the ability to send data out or change state outside the application

An application with all three can be instructed, through content it reads, to send what it can see to someone who should not have it. A single language model cannot reliably tell the instructions it was given from the text it was handed to read. Filtering lowers the odds; it does not fix this.

Separation by design. The separation can be built around the model instead. The part that can act or send never reads untrusted text. The part that reads untrusted text cannot act, and hands back only typed values (a choice from a fixed list, a number, a field) that deterministic

code checks before anything happens. Models that answer only from predefined options fit the reading side well: injected text can at most push the answer to a different allowed option, never to a new command. When the separation holds, untrusted input no longer reaches the part that acts, and that leg is broken. A security reviewer confirms it. A filter or a second model watching the first does not count.

Memory counts as input. If anything untrusted can write to an application's memory, a poisoned entry can steer it days later, long after the session that planted it. Memory that outside content can write to is an untrusted input leg.

The response is a design change. Remove one leg, by design or by separation, and the trifecta rating drops from Critical to High. Filtering and isolating the application as a whole harden it but do not reclassify it. It stays Critical with security review.

2.5 Review Triggers

A **Review Trigger** sends an application to a named reviewer on its own, whatever its overall Risk.

Trigger	Goes to
Decisions about a person's rights, access, or opportunity	legal
Regulated-domain rules apply (financial controls, health privacy, fair lending, EU AI Act high-risk uses)	legal
All three legs of the lethal trifecta (§2.4)	security
Any external audience, or synthetic content shown to one	the reviewing body

Review Triggers apply at any Risk level. A Moderate application that drafts performance review summaries still goes to legal, because those summaries feed decisions about a person's job. Every legal review produces out-of-scope topics for the application (§3.3).

Part 3: Controls

3.1 The Twelve Controls

Each row is one Control. The Floor column is the smallest real version of the mechanism. Each column to the right is what that Risk level requires. Figure 4 shows the table as sliders for an application rated High.

Control	Floor (Low)	Moderate	High	Critical
Regression testing	one scenario, gated on change	a scenario per task	edge and injection cases	a separate team tries to break it
Evals	scheduled judge on a sample	more criteria, trended	continuous on live traffic	evals run by an outside team
Decision traces	every run logged	plus tool calls	retained, reviewable	reproducible with citations
Human review	asker sees sources	sampled after the fact	approval above thresholds	100% approval before action
Privileges	as the user, read-only	limited write scopes	hard limits in code	isolation
Input Certification	Bronze	Silver	Silver, Tier 1 sources	Gold
Compliance guards	content and injection filters	personal-data redaction	out-of-scope topics set by legal	second-model check on every output
Notice and recourse	AI use disclosed	reasons shown	human appeal path	appeal outcomes reviewed
Kill switch	Owner can stop it	tested quarterly	tested monthly, with rollback	automatic stops
Rollout	direct	pilot group	shadow run, output not used	widened in stages, each approved by the reviewing body
Review cadence	yearly	twice a year	quarterly	quarterly and after incidents
Spend cap	hard cap	same	same	same

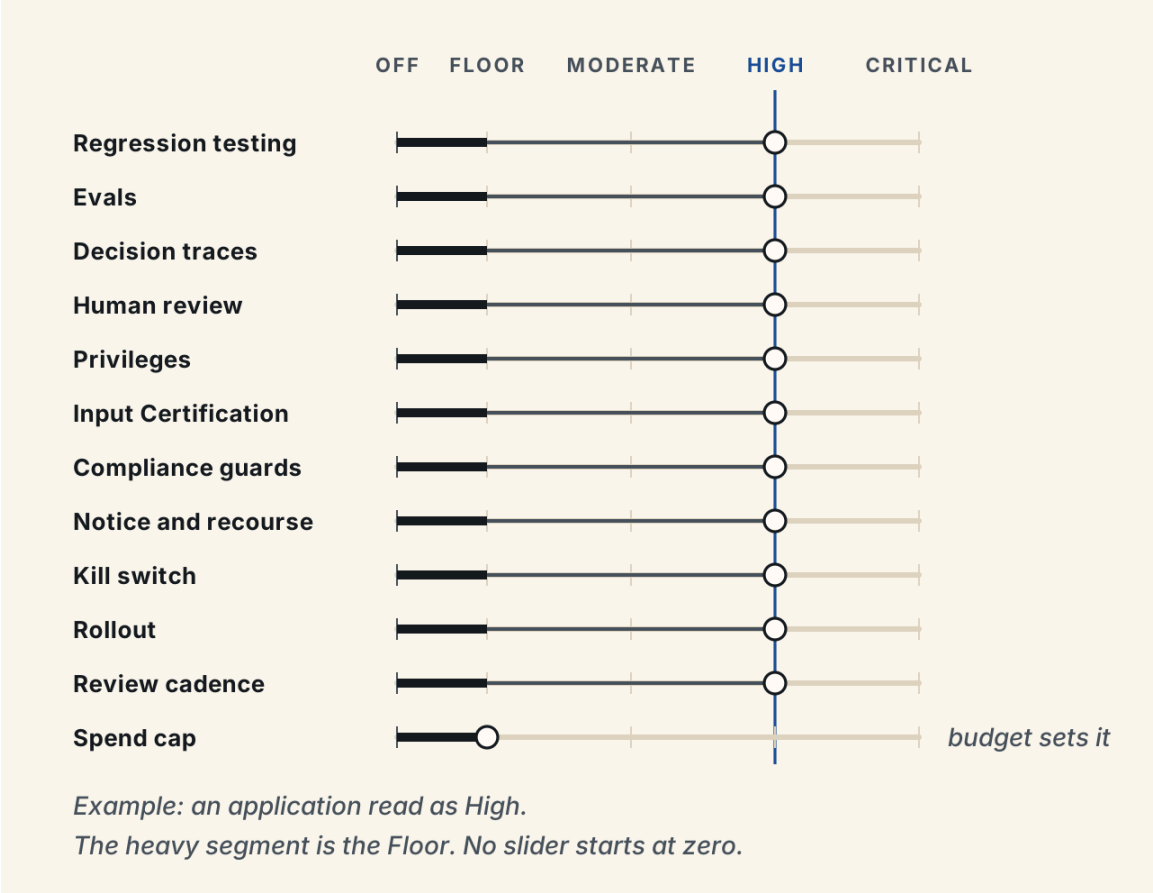


Figure 4. Risk sets how far each Control must move. The Floor is never zero.

3.2 Ground Rules

- The regression gate runs on any change to the prompt, the model, or the context.
- Evals add fairness criteria whenever an application's decisions touch people.
- Human review counts only if reviewers have the capacity for the volume.
- A legal Review Trigger adds out-of-scope topics, even below High.
- Privileges at the Floor include a tool allowlist.
- The spend cap alerts at 75% and stops at 100%.

3.3 Controls in Practice

Regression testing and Evals answer different questions. Regression testing asks whether a change broke what used to work, and it gates every change (Pillar 4). Evals ask whether the

application is any good in production, and they run on a schedule. Decision traces record what happened. The three are dialed up separately.

Human review is a range. At the low end, someone checks a sample after the fact. At the top, a person approves every action before it happens. Someone asked to approve four hundred cases a day approves all four hundred without looking, which is why capacity is a ground rule (§3.2).

One approach for legal to identify out-of-scope topics comes from legal using the application. Legal runs test sessions against the application, asking the questions that carry known or perceived litigation risk in the industry. The session logs are summarized into risk areas, and the risk areas are finalized as out-of-scope topics. Those become Compliance guards, and each one becomes a regression scenario the application must refuse, so the boundary is tested on every change.

Notice and recourse protects the people the answers reach. Telling people they are dealing with AI costs almost nothing, so it sits at the Floor. Reasons, an appeal to a person, and review of appeal outcomes follow as Risk rises.

Input Certification is the bridge to the data program. Its High and Critical Settings lean on the data program: at High, inputs come from each concept's Tier 1 asset, certified Silver or better; at Critical, they are certified Gold. Every application rated High or Critical needs better-certified inputs, and that demand is what moves data up the certification ladder.

The spend cap does not move with Risk. Budget and volume size it: a Critical application can be cheap, and a Low one can loop expensively. Every application carries a hard cap from day one, which answers Pillar 6's financial risk for all of them (Pillar 2).

Part 4: Decisions and Lifecycle

4.1 Owner and Builder

The Owner of an AI Application is the business leader accountable for the process where it acts. The Owner is also the sponsor on its value hypothesis. One person holds value and risk

together on purpose, because only that person can trade them: is this application worth dialing Human review up to 100%?

The Builder is the team that builds and runs the application, or configures and runs it when it is bought. The Builder answers to the Owner for the Controls working: tests passing, traces recording, privileges holding, the kill switch stopping.

The Owner may delegate work to the Builder, never the accountability. Data ownership does not change. The Builder's team still owns any datasets the application creates, under the data program's federated ownership rule.

When the data team builds an agent that drafts collection notices for Finance, the Controller is the Owner and the data team is the Builder. A wrong notice is the Controller's to answer for. A kill switch that failed is the data team's to answer for, to the Controller.

4.2 Funding vs. Deployment

Two different questions get asked about every AI application.

- **The Funding Decision:** should we pay for this? It is made against the value hypothesis and kill criterion, and it ends in funded or killed (Pillar 1).
- **The Deployment Decision:** may this run? It is made against Risk and Required Settings, and it ends in Go, Go with Required Settings, or No-go.

In a small organization, the same AI Center of Excellence leads hold both. Larger organizations may give the Deployment Decision to the data Council or a separate body. Either way the decisions stay distinct, because "is it worth it" and "is it safe" are argued differently. Both outcomes sit on the same Registry record.

4.3 Who Decides

Risk	Who decides	Who accepts the risk
Low	the Express lane: Go at Registration	the Owner
Moderate	the Owner, using the Required Settings table	the Owner
High	the reviewing body	the reviewing body
Critical	the reviewing body, with legal	a named executive, in writing

The Express lane rule. If every rating sits in the Low column and no Review Trigger fires, the Deployment Decision is Go at the Floor. No form is routed and no meeting is held.

Moderate is self-service too. The Owner reads the Moderate column, dials the Settings up, and records it. The reviewing body sees High and Critical, plus anything a Review Trigger sends it. The reviewing body is the group your organization names for this: an AI Center of Excellence, the data Council, or a risk committee.

Lane calibration. Hold the lanes to the policy's goal: 80% or more of applications clear Low or Moderate. If far fewer clear, widen the Low column instead of adding reviewers. Controls that make teams wait drive them to tools nobody governs. Track the median time from registration to Go. It is the plainest evidence that the policy is a runway, not a gate.

4.4 Risk Appetite

No-go depends on Risk Appetite, so the organization writes it down. The **Risk Appetite statement** is one page, set by the executive layer and reviewed yearly. It has three parts.

1. **Prohibited uses.** Uses that are No-go regardless of Settings. Starters are in Appendix B.
2. **Who accepts what.** The right-hand column of §4.3. The named executive signature at Critical is what keeps "the business owns the risk" from being a slogan.
3. **Standing positions.** Short answers to recurring hard cases, so the reviewing body does not re-argue them. Starters are in Appendix C.

4.5 When Things Change

Applications change after they ship. One rule covers it.

Re-rate the Risk when a rating could change. The Owner updates the Registration Record and the Risk is rated again when the application gains:

- a new or wider audience
- a new tool, a new write scope, or the ability to send outside
- a new data source, or data of a higher classification
- a different trigger
- more action authority

An incident also forces a re-rating. If the new rating crosses into a higher level, the change goes through that level's review path before it ships.

Re-test when behavior could change. Prompt edits, model version changes, and context changes do not change the ratings. They run the regression suite as the gate. Pin model versions where the vendor allows it. Where it does not, the scheduled regression run is how a silent vendor update gets noticed.

A Moderate internal sales assistant that gains a "send follow-up email" button adds a trifecta leg and more action authority. The Risk moves to High, and the button waits for the reviewing body. A prompt change the following week only runs the regression suite.

4.6 Existing Applications

On the first day of the policy, AI applications are already in production, and some of them nobody has mentioned.

- **Find first.** Inventory model keys, AI spend by project, and shared assistants.
- **Register on a clock.** Existing applications get one quarter to register and keep running in the meanwhile. Applications rated High or Critical are reviewed in the same quarter. Nothing is switched off for being found.
- **New applications cannot run unregistered.**

- **Credentials follow registration.** Model API keys and gateway access are issued only to registered applications. An unregistered application does not get a reminder; it does not get a key. When the quarter ends, the stragglers are in the same position.

A scheduled script in Finance that has called a model with a personal key for six months keeps running when it is found. Finance names an Owner within the quarter. It probably rates Moderate (scheduled, Confidential data) and moves to a gateway-issued key. The personal key is revoked through the normal security process.

What Changes

When this is working, 80% or more of applications register and flow without review. The largest, most valuable cases get the focus time.

Teams know the requirements before they build, so they can plan, prepare, and build toward a launch instead of waiting on one.

One number tells you it's working: the share of AI spend running through registered applications with a live value hypothesis.

Continuing the Conversation

If you're working through any of this or think part of the model is wrong, I'd like to hear about it: jeremy.taylor@ossnetworks.com or linkedin.com/in/jeremytaylor00. Additional articles are at ossnetworks.com.

About the Author

I lead global data and AI organizations with twenty years in technology. I've spent the past eleven years building and transforming cloud data platforms. Most recently, I've focused on adding an AI platform layer and agents in production. I completed the Carnegie Mellon

University Chief Data and AI Officer executive certificate in 2026.

Appendix A: Adapting This to Your Organization

The policy names no products. Where the policy needs a capability (a registry, a gateway, an eval harness), it describes the capability and leaves the tool to you.

Substitute Throughout

Placeholder	Replace with
The reviewing body	Your AI Center of Excellence, data Council, or risk committee
Factor ratings in §2.1	Your own boundaries, if your business draws them differently (for example, what counts as a vulnerable audience)
Settings in §3.1	Your own figures: sample sizes, cadences, thresholds
The 80% calibration target	Your own, once you have a quarter of data
Appendices B and C	Your own Prohibited uses and Standing positions

Keep, Unless You Have a Deliberate Reason

- **The six Principles (§1.7).** Each one holds up part of the policy.
- **Judgment only ratchets up.** Averaging or self-assessing Risk downward quietly removes the policy.
- **Applications are never certified.** Keep maturity words for data.

Adjust Freely

- The wording of factors and Settings, if your teams frame them differently.
- Review cadences, to fit rhythms your teams already keep.
- Whether the Funding and Deployment Decisions sit with the same people.

Appendix B: Starter Prohibited Uses

Starter ideas for each organization to consider. None applies until your executive layer adopts

it.

- AI that decides to hire, discipline, or dismiss a person without a human making the decision.
- Manipulative techniques that exploit a person's age, disability, or circumstances.
- Scoring people on social behavior or personal traits for unrelated decisions.
- Inferring an employee's emotions from face, voice, or other biometric signals, for example scoring engagement on video calls. Sentiment analysis of text, such as the tone of customer reviews, is a different and ordinary use.
- Categorizing people by sensitive attributes (race, religion, health, sexual orientation, political views) from other data.
- Impersonating a real person without disclosure.

Several of these mirror the EU AI Act's prohibited practices, which make a useful outside reference even for organizations it does not reach.

Appendix C: Starter Standing Positions

Starter answers to cases that recur. Adopt, change, or drop each one.

- Agents do not send outbound messages to customers or the public (emails, posts, notices) unless a person approves each message or the template it uses. Replies in a conversation the customer started follow the application's Human review Setting.
- Restricted data may be used in a purpose-built application with a limited audience, after Critical review.
- An application that keeps all three legs of the lethal trifecta needs a named security owner.
- AI-generated code is reviewed by a person before it reaches production, and review capacity keeps pace with how much code is generated.
- Vendor model updates are treated as changes: pinned where possible, regression-tested where not.

Appendix D: Registry Fields

The minimum record for each AI Application. A spreadsheet can hold it on day one; the Registry makes it an API.

Group	Fields
Identity	name, description, Owner, Builder, lifecycle status
Value	linked use case and value hypothesis, funding status
Cost	FinOps labels, spend cap, current spend
Model and tools	AI model and version, tools and write scopes, memory (session or persistent)
Data	inputs (linked to catalog assets, with their Tier, Certification, and classification)
Risk	the ten factor ratings, the Risk level, any ratchet with its reason, Review Triggers fired
Controls	current Setting of each of the twelve Controls, Required Settings, gaps
Decision	Deployment Decision, who decided, who accepted the risk, date, conditions
History	re-ratings, incidents, kill switch tests, reviews

Appendix E: Glossary

Term	Definition
AI Application	One deployed AI thing with one Owner, built or bought: an agent, a workflow, an LLM feature, or a shared no-code assistant.
Builder	The team that builds or configures an AI Application, runs it, and answers to the Owner for its Controls.
Control	One of twelve mitigation mechanisms every AI Application carries.
Deployment Decision	Go, Go with Required Settings, or No-go.
Deployment Policy	The Pillar 6 rules for AI Applications.
Express lane	The Low review path: Go at the Floor, with no meeting.
Floor	A complete Registration Record and every Control at its minimum Setting.

Term	Definition
Funding Decision	Funded or killed, against the value hypothesis (Pillar 1).
Lethal trifecta	Sensitive data, untrusted input, and the ability to send out or change state, held by one application.
Notice and recourse	The Control that protects the people an application's answers reach.
Out-of-scope topics	Subjects an application must not engage with, set by legal from test sessions.
Owner	The business leader accountable for the process where an AI Application acts.
Prohibited use	A use that is No-go regardless of Settings.
Registration Record	The facts recorded for an AI Application before it runs.
Registry	The system of record for AI Applications.
Required Settings	The Settings a Risk level demands.
Review Trigger	A condition that sends an application to a named reviewer, whatever its Risk.
Risk	How much harm an AI Application could do if it did the wrong thing and nothing stopped it: Low, Moderate, High, or Critical.
Risk Appetite	How much Risk the organization will contain, written as a one-page statement.
Risk Factor	One of ten questions about an application, each rated Low to Critical.
Setting	How far a Control is dialed up.
Standing position	A pre-decided answer to a recurring hard case.
Usage Policy	The Pillar 6 rules for a person using an AI tool for their own work.

Sources and Influences

This document is a synthesis. Where an idea came from someone else, they are named here. Everything else is my own.

From the Carnegie Mellon University Chief Data and AI Officer program (2026). These ideas came from people who taught in the program. The idea is theirs; how it is applied here is mine.

- **Omar Khawaja** (Field CISO, Databricks): an express lane that clears most AI use cases

without manual review, with 80% as the target, and the warning that heavy controls push teams to tools nobody governs.

- **Anand Rao** (Carnegie Mellon University): autonomy is a property of the trigger, not the capability. A simple assistant on a schedule is more autonomous than a capable agent that waits to be asked.
- **Beibei Li** (Carnegie Mellon University): the same model can be low or high risk depending on the decision it touches, and human oversight only counts if the reviewer can really review.
- **Craig Brabec** (instructor, Carnegie Mellon University CDAIO program): when AI acts, accountability belongs to the business owner of the process, not the technologist who built it.
- **Chris Hughes** (VP of Security Strategy, Zenity): govern what agents do, not only what they say, and use hard boundaries instead of soft guardrails for the worst-case action.

Published sources.

- The lethal trifecta is Simon Willison's term, and separation by design follows his Dual LLM pattern and Google DeepMind's CaMeL. The two-of-three rule is Meta's "Agents Rule of Two."
- NIST AI Risk Management Framework (AI RMF 1.0) and its Generative AI Profile (NIST AI 600-1); the EU AI Act, for prohibited practices and high-risk uses; COSO Enterprise Risk Management, for risk appetite; FIPS 199, for the high-water mark; NIST SP 800-53, for graduated control baselines; ISO 31000, for risk treatment.

Companions. Both are my earlier guides at ossnetworks.com. Data Tier, Certification, Owner, and federated ownership come from The Federated Data Governance Program. Pillar references are to The Enterprise AI Operating Model.

No endorsement. Naming someone here credits an idea. It does not imply that they reviewed, endorse, or are affiliated with this document.

What Is Original Here

The individual mechanisms have earlier sources. Rating Risk by the highest single factor is the high-water mark from FIPS 199. Recording a design change apart from a control follows

standard risk treatment. Graduated control levels echo the NIST SP 800-53 baselines, and AI inventories and gateway-issued credentials are common practice. What this document adds is how the pieces fit together for AI deployment: every Control present from day one at a small working Setting, so Risk dials up mechanisms that already exist; an 80% self-service target that makes the policy accountable for speed; Input Certification, which ties how far an application can be trusted to the certification of the data it reads; and Funding and Deployment Decisions kept separate on one Registry record, with one business Owner holding both.

License

This document is licensed under Creative Commons Attribution 4.0 International (CC BY 4.0). You may copy, adapt, and redistribute it for any purpose, including commercially, provided you give appropriate credit and indicate whether changes were made. Adapting it for your own organization is the intended use; Appendix A tells you where to start.

Attribution: “The AI Deployment Policy: AI governance as a runway to launch” by Jeremy Taylor, licensed under CC BY 4.0. Adapted from the original where changed.