ABOVE THE NOISE

THE UNBIASED AI BRIEF FOR ENTERPRISE LEADERS

The Enterprise Leader's AI Playbook

The lies you'll be told, the questions to ask, the architecture to understand, and the build-versus-buy decision — from someone who builds this for a living.

A COMPANION RESOURCE · 2026 EDITION

BEFORE WE BEGIN

Why This Guide Exists

Almost everything an enterprise leader hears about AI comes from someone with something to sell. This guide doesn't.

If you lead a function inside a large organization right now, you are being briefed on AI constantly — by vendors, by consultants, by analysts, by your own teams. The volume of information is not the problem. The problem is that very little of it is disinterested, and almost none of it comes from people who have actually shipped these systems into production and lived with the consequences.

This playbook is written from the practitioner's side of the table. Not the keynote stage, not the sales deck — the side where the system has to work on Monday morning, pass a security review, survive an audit, and justify its line in the budget. The recommendations here reflect what holds up in production at enterprise scale, not what demos well.

It is organized around four things every enterprise leader needs but rarely gets in one place:

- **A clear-eyed read on the industry's most common false claims** — what's being said, why it persists, and what's actually true.
- **The questions that separate real solutions from theater** — what to ask your engineering team (or a vendor) when a solution lands on your desk, and what a good answer sounds like.
- **A working mental model of how AI applications are actually built** — the standard layers, in plain language, so architecture conversations stop being a foreign country.
- **A disciplined build-versus-buy framework** — because this is the single most expensive decision you'll make, and most organizations make it on instinct.

You don't need a technical background to use any of it. You need about forty-five minutes and a willingness to ask uncomfortable questions in your next vendor meeting.

> HOW TO USE THIS PLAYBOOK
>
> Read Parts I and III once for grounding. Keep Part II open during solution reviews — the questions are designed to be asked verbatim. Bring Part IV to any meeting where a platform purchase or internal build is on the table. The reference sections at the back are built for the thirty seconds before a meeting starts.

CONTENTS

What's Inside

I Ten Lies You Will Be Told About Enterprise AI

WHAT'S BEING CLAIMED · WHY IT PERSISTS · WHAT'S ACTUALLY TRUE

None of these claims are usually told as deliberate deceptions. Most are repeated by people who believe them — which is precisely why they're dangerous. A lie that the teller believes survives every polygraph your procurement process can administer. Each entry below names the claim, explains why it keeps circulating, states the truth as observed in production environments, and ends with a concrete action.

LIE № 01

"If you don't move now, you'll be left behind."

WHY IT PERSISTS

Urgency closes deals and unlocks budgets. Fear of being last is the single most effective sales motion in enterprise AI, and it works on boards as well as buyers.

THE TRUTH

The organizations winning with AI are not the ones who moved first — they're the ones who moved *deliberately*. A rushed deployment built on weak data foundations costs more to unwind than a careful one costs to delay. The technology is getting better and cheaper every quarter; the penalty for starting six months later is small. The penalty for institutionalizing a bad architecture is measured in years.

WHAT TO DO

Replace "we need an AI strategy" with "we need to fix the three measurable problems where AI plausibly helps." Urgency should come from the business case, never from the calendar.

LIE № 02

"Hallucinations are basically a solved problem now."

WHY IT PERSISTS

Accuracy genuinely has improved, demos are curated to avoid failure cases, and admitting unsolved error rates complicates the sale.

THE TRUTH

Hallucination is mitigated, not solved — and it can't be fully solved, because generating plausible text is what these models *do*. Grounding techniques, retrieval, and guardrails reduce error rates dramatically, but the residual failures are confident, fluent, and hard for non-experts to spot. Any system presented to you should be designed on the assumption that it will sometimes be wrong.

WHAT TO DO

Ask every solution owner: "What happens when it's wrong, who catches it, and how do we measure how often that occurs?" If there's no answer, the system isn't production-ready — regardless of the demo.

LIE № 03

"It's essentially plug-and-play. You'll be live in a few weeks."

WHY IT PERSISTS

The model layer really is fast to stand up, and a proof-of-concept genuinely can be built in days. The timeline quoted is the timeline to a demo — not to production.

THE TRUTH

In an enterprise, the model is perhaps a fifth of the work. Identity and access integration, data pipelines, security review, compliance sign-off, evaluation frameworks, monitoring, and change management dominate the timeline. Six-week pilots routinely become nine-month deployments — not because the AI failed, but because the demo never had to log in through your SSO or pass your audit.

WHAT TO DO

Ask for the deployment timeline *including* security review, data integration, and user rollout — and ask for a reference customer at your scale and in your regulatory posture.

LIE № 04

"Our AI agents can run this process end-to-end, autonomously, today."

WHY IT PERSISTS

"Agent" is the most fundable word in software right now, and the gap between a scripted demo and open-ended autonomy is invisible from the audience.

THE TRUTH

Agents do real work today — in narrow, well-instrumented domains, with clear guardrails, checkpoints, and human escalation paths. What they do not reliably do is run long, open-ended, high-stakes processes unsupervised. Error compounds across steps: an agent that's 95% reliable per step is roughly 60% reliable across ten of them. Production agent systems succeed by constraining scope, not by trusting autonomy.

WHAT TO DO

Ask: "Show me the boundaries — what is the agent *not* allowed to do, and what triggers a human?" A vendor who can't answer crisply hasn't run this in production.

LIE № 05

"You'll need your own custom model to get real value."

WHY IT PERSISTS

Custom model work commands premium consulting fees and long engagements, and "our own model" flatters organizational ego.

THE TRUTH

The overwhelming majority of enterprise use cases are served best by frontier models accessed through an API, combined with retrieval over your own data, careful prompting, and solid orchestration. Fine-tuning is a legitimate but narrow tool — useful for style conformance, classification at volume, or latency/cost optimization. Training a model from scratch is justified for almost no enterprise. Your data is your advantage; baking it into custom weights is rarely the best way to use it.

WHAT TO DO

When custom model work is proposed, require a comparison baseline: "What does the best API model with retrieval score on the same evaluation?" Fund the delta, not the ambition.

LIE № 06

"Your data is ready. We'll just connect to it."

WHY IT PERSISTS

Data readiness is your problem, not the vendor's — so it's priced out of the pitch and discovered after the contract is signed.

THE TRUTH

In nearly every enterprise AI program, data is the longest pole in the tent: scattered across systems, inconsistently labeled, stale, duplicated, and governed by access rules nobody fully remembers. An AI system that retrieves from bad data doesn't fail loudly — it answers *confidently from the wrong page*. The unglamorous work of data quality, lineage, and access control is where most AI timelines actually go to die.

WHAT TO DO

Before approving any AI initiative, ask for a data readiness assessment of the specific sources it will touch — freshness, ownership, access model, and quality. Budget for remediation up front.

LIE № 07

"Our platform will cover all your AI use cases."

WHY IT PERSISTS

Platform consolidation is a rational instinct after a decade of SaaS sprawl, and every vendor's roadmap slide shows them at the center of your stack.

THE TRUTH

The AI landscape is moving too fast for any single platform to lead everywhere. The leader in document intelligence is not the leader in agents, code, or analytics — and next year's rankings will differ from this year's. Sensible enterprises run a small portfolio: one or two strategic platforms, plus an abstraction layer that keeps model and vendor choices swappable. Total consolidation today is lock-in to a snapshot of a market in motion.

WHAT TO DO

Make exit cost a formal evaluation criterion. Ask: "If we left in eighteen months, what comes with us — and in what format?"

LIE № 08

"The ROI takes care of itself — productivity gains show up immediately."

WHY IT PERSISTS

Early personal-productivity wins are real and visible, and extrapolating them to the enterprise P&L makes a beautiful slide.

THE TRUTH

Individual time savings do not automatically become enterprise value. Saved minutes diffuse into the workday unless the surrounding process is redesigned to capture them. The AI deployments that show durable, attributable ROI share three traits: a baseline measured *before* launch, a specific process metric the initiative moves, and an owner accountable for that metric. Deployments without those three produce enthusiastic anecdotes and unverifiable claims.

WHAT TO DO

No baseline, no launch. Require every AI initiative to name its metric, its current value, and its target before funding — and review actuals at ninety days.

LIE № 09

"AI is too risky for a regulated business like yours."

WHY IT PERSISTS

This one comes from inside the building. Caution is professionally safe, and one well-publicized AI failure elsewhere justifies years of internal "no."

THE TRUTH

Regulated industries — banking, insurance, healthcare — are deploying AI at scale right now, with governance to match: scoped use cases, human review on consequential decisions, audit trails, and model risk management. Risk is real, but it is *managed*, not avoided — and inaction carries its own risk: shadow AI spreading through unsanctioned tools, talent leaving for organizations that let them use modern tooling, and competitors compounding learning you're not doing. The riskiest AI posture in a regulated industry is an unmanaged one.

WHAT TO DO

Replace blanket prohibitions with a tiered use-case policy: low-risk uses approved broadly, high-stakes uses gated by review. Measure shadow AI honestly — it's already in your building.

LIE № 10

"Pick the best model and you've won."

WHY IT PERSISTS

Model benchmarks make headlines, leaderboards make procurement feel objective, and "we use the #1 model" is an easy story to tell upward.

THE TRUTH

Frontier models leapfrog each other every few months, and the gap between the top several is narrow and shrinking for most enterprise tasks. The model is becoming the most *replaceable* layer in the stack. Durable advantage lives in the layers you control: the quality of your data, the depth of workflow integration, the rigor of your evaluation discipline, and an architecture that lets you swap models as the market moves. Organizations that bet on a model rent their advantage; organizations that build those four things own it.

WHAT TO DO

Ask your teams: "If the best model changed next quarter, how long would it take us to switch?" If the answer is months, the architecture — not the model — is your problem.

II The Questions to Ask When a Solution Is Presented

FOR YOUR ENGINEERING TEAM, YOUR VENDORS, AND YOUR INTEGRATORS

You don't need to understand transformers to govern AI well. You need to ask questions whose answers reveal whether the people in front of you have done the work. The questions below are designed to be asked verbatim. Each comes with a note on what a credible answer sounds like — because the goal isn't to catch people out; it's to raise the standard of the conversation. A strong team will *enjoy* these questions.

1 · Problem & Fit

The most expensive failures aren't broken systems — they're working systems aimed at the wrong problem.

Q. What specific business problem does this solve, and how do we measure that problem today?

LISTEN FOR — a named metric with a current value. "Improves productivity" is not an answer; "reduces average claims-review time from 4.2 days, which we currently track in X" is.

Q. Why does this need AI at all? What would the non-AI solution look like?

LISTEN FOR — genuine engagement. Some problems are solved better by a form, a rule, or a process change. A team that has considered the boring alternative and can explain why AI wins is a team you can trust.

Q. Who is the user, and have they been in the room?

LISTEN FOR — evidence of real user contact. Systems designed about users rather than with them get quietly abandoned within a quarter.

2 · Data

Data is where most AI timelines, budgets, and accuracy claims actually break.

Q. What data does this system rely on, who owns it, and how current is it?

LISTEN FOR — named sources and named owners. Hesitation here predicts a stalled project three months from now.

Q. How does the system respect our existing access controls? Can it show a user something they couldn't otherwise see?

LISTEN FOR — a concrete permissions story. AI retrieval that bypasses entitlements is one of the most common and least discussed enterprise risks.

Q. Is our data used to train anyone else's models? Where does it live, and where is it processed?

LISTEN FOR — contractual specifics, not reassurance. Data residency, retention, and training-use terms should be in writing.

3 · Accuracy, Evaluation & Failure

Every AI system is wrong sometimes. The mature question is not "is it accurate?" but "how do we know, and what happens when it isn't?"

Q. How do we evaluate this system's quality — before launch and continuously after?

LISTEN FOR — a test set, defined scoring criteria, and a cadence. "We tried it and it looked good" is the wrong answer at enterprise stakes.

Q. What does failure look like for this system, and who catches it?

LISTEN FOR — specific failure modes (wrong answer, stale answer, leaked context, refused task) mapped to specific safeguards: human review, confidence thresholds, escalation paths.

Q. What accuracy is actually required for this use case — and what happens at the gap?

LISTEN FOR — recognition that a drafting assistant can tolerate 85% while an automated customer-facing decision cannot. The required bar should drive the design, including where humans sit.

4 · Architecture & Dependencies

Today's choices determine tomorrow's optionality. (Part III gives you the vocabulary for this section.)

Q. If we wanted to swap the underlying model in six months, what would that take?

LISTEN FOR — "days to weeks, it's abstracted" rather than "that would be a rewrite." Model-agnostic design is the single best hedge in this market.

Q. What are we locked into, and what's the exit path?

LISTEN FOR — an honest inventory: proprietary formats, embedded workflows, data egress terms. Every platform has lock-in; unacknowledged lock-in is the dangerous kind.

Q. How does this fit with what we already run — identity, data platforms, existing automation?

LISTEN FOR — integration named as the major cost it is, not waved through as "it has an API."

5 · Security, Privacy & Compliance

AI introduces genuinely new attack surfaces alongside the familiar ones.

Q. How is this system protected against prompt injection and data exfiltration through the model?

LISTEN FOR — awareness that malicious instructions can arrive *inside the content the system reads* (documents, emails, web pages), plus concrete mitigations: input sanitization, output filtering, least-privilege tool access.

Q. What gets logged, and could we reconstruct why the system gave a specific answer?

LISTEN FOR — audit-grade logging of inputs, retrieved context, and outputs. In a regulated business, "the model decided" is not an acceptable root cause.

Q. Has our security team reviewed this — and was the review designed for AI systems specifically?

LISTEN FOR — a review covering AI-specific risks, not just a standard vendor assessment with a new logo on it.

6 · Cost & Scale

AI costs behave differently from traditional software: they scale with usage, and the pilot price tells you almost nothing.

Q. What does this cost at pilot scale, and what does it cost at full deployment?

LISTEN FOR — a per-unit cost model (per query, per document, per seat) projected to real volumes. Token-based costs at 50,000 users are a different universe from a 50-person pilot.

Q. What's the total cost beyond licenses — integration, evaluation, monitoring, support, change management?

LISTEN FOR — a number meaningfully larger than the license fee. (See "The Real Cost of AI" in Part V; the license is typically a third or less of the true total.)

7 · Operations & Lifecycle

The launch is the cheapest day. Everything after is where systems quietly degrade or quietly compound.

Q. Who owns this system after launch — and is that in their actual job description?

LISTEN FOR — a named team with allocated capacity. Orphaned AI systems decay fast: prompts go stale, data drifts, quality erodes invisibly.

Q. How will we know if quality degrades over time?

LISTEN FOR — continuous monitoring against the evaluation set, user feedback channels, and drift alerts — not "users will tell us."

Q. What's the rollback plan if this goes badly?

LISTEN FOR — a real answer. If the old process is being dismantled at launch, you have no rollback plan — you have a one-way door.

THE PATTERN BEHIND ALL OF THESE

Every question above reduces to one of three things: **How do we know it works? What happens when it doesn't? What does it really cost?** If you remember nothing else from this section, ask those three — in every AI conversation, every time.

III The Anatomy of an AI Application

THE SIX STANDARD LAYERS — AND THE TWO DISCIPLINES THAT WRAP AROUND ALL OF THEM

Every serious AI application — whether your team builds it or a vendor sells it — is some arrangement of the same six layers. Once you can see the stack, vendor pitches become legible: you can ask *which layers they actually provide*, where your data sits, and which layers you'd be locked into. This is the most useful mental model an enterprise leader can carry into an AI conversation.

1 **Experience Layer**
What users touch — chat interfaces, copilots embedded in existing tools, APIs feeding other systems.

2 **Orchestration Layer**
The conductor — workflows, agents, tool-calling, routing. Decides what happens in what order, and when a human steps in.

3 **Context Layer**
What the model knows in the moment — prompts, retrieval (RAG), memory. Where your institutional knowledge meets the model.

4 **Model Layer**
The reasoning engine — foundation models accessed by API, occasionally fine-tuned variants. Increasingly swappable.

5 **Data & Knowledge Layer**
Your sources of truth — documents, databases, pipelines, and the indexes that make them searchable by meaning.

6 **Infrastructure Layer**
Where it all runs — cloud platforms, compute, networking. Largely a solved procurement problem for most enterprises.

\+ **Wrapped around everything: Security & Governance · Observability & Evaluation**
Not layers but disciplines — they must touch every layer above, or they're decoration.

Each Layer, in Plain Language

1 · The Experience Layer

This is where adoption is won or lost. The strongest model in the world fails if it lives in a separate tab nobody opens; a modest one succeeds if it appears inside the tools people already use, at the moment they need it. When evaluating a solution, weight the experience layer more heavily than the demo encourages you to — it's the layer that determines whether the system gets used at all.

> THE EXECUTIVE QUESTION
>
> *"Does this live where my people already work, or does it ask them to go somewhere new?"*

2 · The Orchestration Layer

Modern AI applications are rarely one model call. They're sequences: retrieve, reason, call a tool, check the result, escalate if needed. The orchestration layer is where "agents" live, where business rules constrain what the AI may do, and where human checkpoints are placed. This layer embodies your risk posture in code — it deserves executive attention for exactly that reason.

> THE EXECUTIVE QUESTION
>
> *"Where are the human checkpoints, and what is the system not allowed to do without one?"*

3 · The Context Layer

Models know nothing about your business until this layer tells them. Retrieval-augmented generation (RAG) — fetching relevant internal content and handing it to the model alongside the question — is how most enterprise AI grounds its answers in your reality. The quality of this layer, far more than the choice of model, determines whether answers are right. When an AI system gives wrong answers about your business, the fault is almost always here or in the data layer below.

> THE EXECUTIVE QUESTION
>
> *"When it answers, can it show me where the answer came from?"*

4 · The Model Layer

The famous layer — and increasingly the most commoditized. Frontier models from the major labs leapfrog each other every few months, and for most enterprise tasks the practical gap between the leaders is small. The strategic implication is architectural: this layer should be *swappable*. A system hard-wired to one model is a system that ages in dog years.

> THE EXECUTIVE QUESTION
>
> *"How long would it take us to switch models — days or months?"*

5 · The Data & Knowledge Layer

Your documents, databases, and records — plus the pipelines that clean them and the indexes that make them searchable by meaning rather than keyword. This is the layer where enterprises hold a genuine, durable advantage: nobody else has your data. It is also, reliably, the layer in the worst shape on day one. Most "AI projects" are, in honest accounting, data projects wearing an AI badge.

> THE EXECUTIVE QUESTION
>
> *"Is the data this will draw from accurate, current, and governed — and who owns making it so?"*

6 · The Infrastructure Layer

Compute, networking, hosting. For most enterprises consuming models through APIs, this is the least differentiated layer — a procurement and cost-management exercise on your existing cloud relationships rather than a strategic frontier. It matters intensely to your platform teams; it should rarely dominate an executive conversation.

> THE EXECUTIVE QUESTION
>
> *"Is anything here a genuine constraint, or is this layer business as usual?"*

+ The Two Wrap-Around Disciplines

Security & governance must run through every layer: who can ask what (experience), what actions are permitted (orchestration), what content is retrievable by whom (context and data), and what's logged throughout. **Observability & evaluation** is the discipline of knowing whether the system works — test sets, quality scoring, drift monitoring. A vendor who presents six polished layers and hand-waves these two has shown you a sports car without brakes.

> THE EXECUTIVE QUESTION
>
> *"Show me the evaluation results and the audit log — not the demo."*

IV Build vs. Buy: A Decision Framework

TWO AXES · A WEIGHTED SCORECARD · AND THE THIRD OPTION MOST ENTERPRISES ACTUALLY NEED

Build-versus-buy is the most consequential recurring decision in enterprise AI, and most organizations make it on reflex: engineering-led cultures default to build, procurement-led cultures default to buy, and both defaults are wrong about half the time. Two questions cut through the reflex: **Is this capability strategically differentiating for us?** and **Do we honestly have the capability to build and run it?**

HIGH DIFFERENTIATION · LOW INTERNAL CAPABILITY

Partner & Grow

The capability matters strategically but you can't yet build it well. Co-build with a partner under terms where the IP, the data, and — critically — the learning accrue to you. Treat the engagement as paid apprenticeship, with a planned handover date.

HIGH DIFFERENTIATION · HIGH INTERNAL CAPABILITY

Build

This is where building is justified: the capability touches your competitive core, leverages data only you hold, and you have the talent to build *and operate* it. Build here is investment; build anywhere else is hobby.

LOW DIFFERENTIATION · LOW INTERNAL CAPABILITY

Buy

Commodity capability, no special advantage in owning it. Buy the leading product, configure minimally, and spend the saved energy on adoption. Custom-building here is the most common and most expensive unforced error in enterprise AI.

LOW DIFFERENTIATION · HIGH INTERNAL CAPABILITY

Buy — and Resist the Urge

Your team *could* build it, which is exactly the trap. Every engineer-month spent rebuilding a commodity is a month not spent on the differentiating quadrant. Buy it, integrate it well, and aim your builders where they compound.

← INTERNAL CAPABILITY: LOW | HIGH → · VERTICAL AXIS: STRATEGIC DIFFERENTIATION (TOP = HIGH)

The Weighted Scorecard

When the 2×2 isn't decisive, score the decision. Rate each dimension 1–5 for your specific use case, multiply by the weight, and compare totals honestly.

DIMENSION	WEIGHT	FAVORS BUILD WHEN…	FAVORS BUY WHEN…
Strategic differentiation	×3	The capability is part of how you win — customers would notice it missing.	It's table stakes; every competitor will have the same thing.
Proprietary data leverage	×3	Value comes from data only you hold, deeply integrated.	It runs on generic or public data any vendor can access.
Customization required	×2	Your workflow is genuinely unusual and products force bad compromises.	Your process is standard, even if it doesn't feel that way internally.
Time to value	×2	You can afford quarters; the payoff justifies the wait.	Value is needed this quarter; the market window is now.
Internal talent & capacity	×2	You have proven AI engineering capacity — uncommitted, not theoretical.	The build team would be hired, borrowed, or wished into existence.
Three-year total cost	×2	Vendor pricing scales painfully with your volumes.	Build estimates ignore the ongoing operate-and-maintain cost (they usually do).
Vendor market maturity	×1	The category is young; products are thin wrappers you'd outgrow.	Mature products exist with reference customers at your scale.
Lock-in & exit risk	×1	Vendor lock-in is severe: proprietary formats, painful egress.	Standards-based product; your data leaves cleanly if you do.

THE THIRD OPTION: ASSEMBLE

The honest answer for most enterprises is neither pure build nor pure buy — it's **assemble**: buy the commodity layers (models via API, platform infrastructure, standard tooling) and build only the thin layer where your advantage lives — your data integration, your workflow logic, your evaluation standards. This captures most of build's differentiation at a fraction of its cost and risk. When a team proposes "build," ask whether they mean building everything or assembling intelligently; the gap between those two answers is usually several million dollars.

Three rules that survive contact with reality. First, never build what you aren't prepared to *operate* — the build cost is the down payment, and the operating cost is the mortgage. Second, beware "build" proposals from teams whose real motivation is wanting to do interesting work; the motivation is human and even healthy, but it isn't a business case. Third, beware "buy" decisions made solely because the budget cycle favors OpEx — a strategic capability rented from a vendor is a strategy with a landlord.

V The Reference Section

RED FLAGS · THE REAL COST OF AI · A 90-DAY EVALUATION PLAYBOOK · AN EXECUTIVE GLOSSARY

Eight Red Flags in Any AI Pitch

Any one of these is a yellow card. Two or more, and you should slow the conversation down considerably.

01 **The demo only ever uses their data.**
A solution that hasn't touched your messy, real-world data hasn't been tested on anything that matters. Insist on a pilot against your own content before any commitment.

02 **Accuracy claims with no methodology.**
"98% accurate" means nothing without knowing: measured on what data, scored by whom, against what definition of correct? Real vendors share their evaluation methodology readily.

03 **No reference customers at your scale, in your industry.**
"We work with leading enterprises" should convert, on request, into a phone call with one. If it can't, you are the reference customer — and should be priced accordingly.

04 **Vague answers on where your data goes.**
Residency, retention, training use, and subprocessors should be answerable in one breath and confirmable in the contract. Hesitation here is disqualifying for regulated businesses.

05 **"Humans in the loop" with no specifics.**
It's the phrase every pitch includes and few can operationalize. Ask exactly where humans sit, what they see, and what authority they hold. Watch for the answer dissolving.

06 **Pricing that only makes sense at pilot volume.**
If per-unit economics at full deployment make everyone uncomfortable, the discomfort is the data point. Model year-two costs at realistic usage before signing year one.

07 **The roadmap does the heavy lifting.**
If the capabilities you actually need are "coming next quarter," you're buying a promise. Evaluate what ships today; treat the roadmap as marketing until it isn't.

08 **Urgency as a closing technique.**
End-of-quarter discounts are normal commerce. "This market is moving too fast for due diligence" is not. The vendors most confident in their product are the most patient with your process.

The Real Cost of AI

The license fee is the part of the iceberg you can see. In practice it is rarely more than a third of what an AI capability actually costs to run well.

COST CATEGORY	TYPICAL SHARE	WHAT IT ACTUALLY CONTAINS — AND WHERE IT HIDES
Licenses & usage fees	25–35%	The visible number: seats, tokens, platform fees. The only line that appears in the pitch deck.
Integration & data work	20–30%	Connecting identity, permissions, and source systems; cleaning and structuring the data the AI depends on. Reliably the most underestimated line.
Evaluation & quality assurance	10–15%	Building test sets, scoring quality, regression-testing every model and prompt change. Skipping this doesn't remove the cost — it converts it into incidents.
Monitoring & operations	10–15%	The ongoing run cost: drift detection, log review, prompt maintenance, model migrations. Permanent, not project-phase.
Change management & training	10–20%	The line that determines whether anyone uses the system. Chronically underfunded because it isn't technical — and decisive because adoption is the whole point.
Governance & risk	5–10%	Review processes, audit support, policy work, vendor reassessment. Scales with how regulated you are.

THE PLANNING RULE

When a proposal lands with a license cost of X, plan for **2.5–3X all-in** over the first full year. If the proposal's own budget is already in that range, you're dealing with a team that has done this before. If it's 1.1X, the missing 1.5X hasn't disappeared — it's waiting for you in next year's budget cycle, unannounced.

PART V · REFERENCE

The 90-Day Evaluation Playbook

A disciplined cadence for taking any AI initiative — bought or built — from idea to a defensible go/no-go decision, with explicit gates.

DAYS 0 – 30

Frame the Problem and Baseline It

- Name the business problem and the **single metric** the initiative must move. Measure that metric's current value — this baseline is non-negotiable.
- Run the data readiness check: which sources, who owns them, how current, what access rules. Surface remediation needs now, not at week ten.
- Define the evaluation standard: what "good" looks like, who judges it, and the test cases that will be used unchanged for the duration.
- Identify the real users and recruit a pilot group from them — not from the project team.

GATE № 1 — No baseline metric, no Phase 2. A pilot that can't be measured can only be marketed.

DAYS 31 – 60

Pilot Under Real Conditions

- Deploy to the pilot group inside their actual workflow — real data, real permissions, real stakes (with appropriate human review on anything consequential).
- Score outputs weekly against the evaluation standard. Track the failure modes, not just the win rate.
- Hold structured user feedback sessions at weeks two and four. Watch what users *do*, not just what they say — quiet abandonment is the loudest signal.
- Project full-scale costs from observed pilot usage. This is when per-unit economics become real numbers.

GATE № 2 — Quality at or above the bar, users returning voluntarily, unit economics that survive scaling math. Two of three is a conversation; one of three is a stop.

DAYS 61 – 90

Decide Like It's Permanent

- Compare the metric against its baseline. Attribute honestly — separate the AI's effect from the attention effect of any pilot.
- Complete security and compliance review under production assumptions, not pilot exemptions.
- Name the permanent owner, the operating budget, and the quality monitoring cadence — before the scale decision, not after.
- Make a clean call: *scale, iterate with a specific hypothesis, or stop.* Record the reasoning either way — it's the cheapest institutional learning you'll ever buy.

GATE № 3 — "Promising, let's keep piloting" is the most expensive sentence in enterprise AI. Pilots that can't graduate must end.

An Executive Glossary

Twenty terms, defined the way you'd want them explained two minutes before a board meeting — plain, honest, and free of mystique.

Large Language Model (LLM) — Software trained on enormous amounts of text to predict language, which turns out to make it useful for writing, summarizing, analysis, and reasoning. The engine inside most current "AI."

Foundation model — A large, general-purpose model (from labs like Anthropic, OpenAI, Google) that applications are built on top of — rented via API rather than built yourself.

Token — The unit models read and write — roughly three-quarters of a word. Matters to you because usage-based AI pricing is usually priced per token.

Context window — How much material a model can consider at once — its working memory. Bigger windows let it read longer documents, at higher cost.

Prompt — The instructions and material given to a model. "Prompt engineering" is the craft of writing instructions that reliably produce good output; it's a real skill, not a buzzword.

RAG (Retrieval-Augmented Generation) — The standard technique for making AI answer from *your* content: fetch the relevant internal documents, hand them to the model with the question. How enterprise AI gets grounded in your reality.

Hallucination — When a model states something false with complete confidence. Reduced by grounding techniques; never eliminated. Design assumption, not edge case.

Fine-tuning — Additional training that adjusts an existing model's behavior using your examples. A specialized tool for specific jobs — not the default path to "AI with our data" (that's usually RAG).

Embedding / vector database — The technique (and storage) that lets systems search by meaning rather than keywords — how "What's our parental leave policy?" finds the right document even if it never says those words.

Agent — An AI system that takes multi-step actions toward a goal — searching, calling tools, writing, checking results — rather than answering a single question. Powerful in narrow, well-guarded domains; oversold as universal autonomy.

Orchestration — The coordination logic around models: what happens in what order, which tools may be called, where humans approve. Where your business rules and risk posture actually live.

Inference — Running a model to get output (as opposed to training one). "Inference costs" are your ongoing usage bills.

Guardrails — Controls that constrain AI behavior — blocking topics, filtering outputs, limiting tool access, enforcing escalation. The difference between deployed and deployed responsibly.

Evals (evaluations) — Structured tests measuring AI quality against defined criteria — the AI equivalent of QA testing. If a team can't show you their evals, they're measuring quality by vibe.

Prompt injection — An attack where malicious instructions hide inside content the AI reads (an email, a document, a web page) and hijack its behavior. The signature new security risk of this era; ask your security team about it by name.

Model drift / data drift — Quality degradation over time as the world (or the model behind an API) changes while your system stands still. Why monitoring is permanent, not a launch-phase activity.

Multimodal — Models that handle more than text — images, audio, documents, video. Increasingly standard; relevant wherever your workflows involve PDFs, scans, photos, or calls.

MCP (Model Context Protocol) — An emerging open standard for connecting AI systems to tools and data sources — reducing custom integration work and vendor lock-in at the connection layer. Worth asking vendors whether they support it.

Human-in-the-loop — Design pattern where a person reviews or approves AI output at defined points. The phrase is cheap; the operational specifics — who, where, with what authority — are what matter.

Shadow AI — Unsanctioned AI use inside your organization — personal accounts, unapproved tools, pasted confidential data. Already happening at scale in most enterprises; governable, but only if acknowledged.

The Signal, Not the Noise

This playbook is a companion to **Above the Noise**, a biweekly brief for enterprise leaders navigating AI — built on production experience rather than press releases. Every episode: what's real, what's hype, and what to do about it.

No sponsors. No affiliate deals. Nothing for sale. If this was useful, the most valuable thing you can do is pass it to one other leader who's drowning in the noise.

ABOVE THE NOISE · THE UNBIASED AI BRIEF FOR ENTERPRISE LEADERS