Build the runway first:

How to scale AI agents without losing control

BRIGHTING
EMPOWER INNOVATION

A strategic guide for CTOs, CIOs, and AI Leads navigating the journey from AI pilots to enterprise-scale deployment.

INTRODUCTION

The board wants more AI. The business is already building it. And you're accountable for making it work...without knowing exactly what's running, who built what, or what it's connected to.

This is where almost every enterprise with an AI agenda is finding itself. Because the skills and tools that get you to working pilots aren't the skills and tools that get you to a governed, observable, cost-controlled agent ecosystem.

> **The missing piece? The platform underneath**
>
> The governance layer, identity controls, observability tooling, cost attribution logic... the runway that allows the aircraft to take off in an efficient, controlled way.

This whitepaper explains why most enterprise AI scaling efforts stall, what a properly built agentic AI platform looks like – drawn from real deployments – and how to assess readiness before committing to a build.

We've worked through these challenges with organizations including HEMA and ASICS. The patterns repeat, but the fix is more straightforward than most assume – if you build the right foundation.

01 Why AI scaling efforts stall – and why agents are almost never the cause

Nearly 78% of organizations have deployed generative AI in some form, yet roughly the same proportion report no material impact on earnings, according to McKinsey. The instinctive response to this value gap is to run more pilots, find better use cases, or bring in more AI talent. But that doesn't address the root cause: the lack of infrastructure that allows agents to deploy and operate at scale.

78%

deployed generative AI, yet see no material earnings impact

MCKINSEY

60%

of enterprise AI projects will fail to scale without proper governance

FORRESTER

75%

building agentic architectures on their own will fail

FORRESTER

As Forrester says, 60% of enterprise AI projects will fail to scale without proper governance frameworks, and 75% of companies that attempt to build agentic architectures on their own will fail.

What we've seen across enterprise AI deployments

Agentic AI is following the same trajectory cloud did a decade ago: teams moving fast, building governance per project rather than centrally, and ending up with sprawl, inconsistent controls, and no single view of what's running. The fix in cloud was the landing zone – a shared foundation every workload inherited. The fix is the same here.

The repeating pattern

Here's how it typically unfolds. Budget is allocated and someone is appointed to lead AI adoption. That person faces an immediate backlog, but they're under-resourced, still developing their own knowledge, and lack the tooling to comply with CISO requirements. So they build agents based on demand, with ad-hoc governance and access logic. Then – somewhere between the thirteenth and thirtieth agent – it becomes ungovernable.

02 The 5 gaps that break agentic AI at scale

Understanding why scaling fails is the first step to avoiding it - and the same five gaps appear when we assess an organization's readiness to scale AI agents.

1 Identity stops at the user, not the agent

Most enterprise IAM is designed around humans: a user authenticates, receives a token, and has their access governed accordingly. Agents break this model. An agent making an API call on a user's behalf may have more system access than the user – and in most organizations, no one has explicitly thought through what that means.

THE SOLUTION

The correct approach:

- Assigns each agent its own identity
- Scopes its permissions to exactly what it needs for each specific task
- Federates that identity with the enterprise IdP
- Passes user permissions to the agent, so least privilege prevails (so the same agent gives a different result to different users based on applicable permissions)

Then, every call is auditable, every permission is bounded, and revoking access to a compromised or decommissioned agent is a single action rather than a multi-team investigation.

2 Governance lives in a document, not the infrastructure

Shadow AI seems like a behavior problem, so most organizations respond by blocking tools, denying licenses, and refusing requests. This drives the activity underground, with agents built on infrastructure IT can't see, making both risk and spend become invisible.

THE SOLUTION

Make the governed path the easiest. When a shared platform gives every team the governance, identity, and observability they need automatically, shadow AI becomes governed AI.

This requires governance to be enforced at the infrastructure level, not declared in a policy document. Content safety guardrails, PII detection, topic restrictions, and spend limits all need to be active at runtime, applied to every agent and interaction, regardless of who built the agent or what framework they used.

3 Cost compounds invisibly

A single LLM call is cheap; a chained, multi-step agent workflow isn't. And when you have dozens of agents, total spend can spike fast. You don't know what's happening until the bill arrives, and even then, you can't always allocate the spend.

THE SOLUTION

A platform approach that provides real-time cost attribution per agent, team, and model – with budget alerts and spend limits enforced at the platform level, not self-managed by individual teams. Cost visibility is a property of the infrastructure, not a reporting task.

4 Drift no one notices until something breaks

LLM providers release new model versions on cycles measured in weeks. Each new version can change an agent's behavior – slightly different tone, edge-case handling, or interpretation of an ambiguous instruction. Without a baseline to measure against, this prompt drift is invisible until it manifests as a downstream problem.

THE SOLUTION

A robust agentic platform includes automated quality evaluation running continuously against production agent output, with regression detection before model updates are adopted. For any agent running a consequential business process, this is the difference between AI investments that compound in value and ones that degrade over time.

5 Incidents with no clear owner

Clear incident ownership requires 3 things to be defined before deployment:

- What constitutes an incident for each agent
- Who's notified when one occurs
- What the escalation path looks like when first-line response can't resolve it

THE SOLUTION

Building this per-agent is expensive and inconsistent. Building it once at platform level – with configurable alerting and escalation routing per agent – means incident response becomes an infrastructure capability, not a gap.

03 What a properly built agentic AI platform looks like

The core principle of the platform approach is simple, like with the cloud landing zone: deploy the foundation once, and every agent you build – today, next quarter, 2 years from now – inherits everything it needs automatically. Governance, identity, observability, cost controls, and guardrails are properties of the runway, not of each individual aircraft.

In practice, this comes down to 4 elements:

01
What the platform supports

02
What agents inherit automatically

03
How agents are discovered and governed

04
Who owns the infrastructure the platform runs on

1 platform, 3 agent types

A common misconception is that an agentic AI platform is only relevant for complex, custom-engineered agents. In practice, most enterprises need a governed, cohesive way to support 3 agent categories with different requirements.

AGENT TYPE	BUILD APPROACH	GOVERNANCE REQUIREMENT
Low-code agents	By business teams vibe coding or using tools like Glean, without dedicated engineering support.	Need governance and security to be inherited automatically because the teams building them can't be expected to implement it themselves.
Custom-built agents	By engineering teams on frameworks like Strands, LangGraph, LangChain, CrewAI, or bespoke orchestration.	Higher complexity, require more maintenance, and carry more risk if governance is inconsistent.
External SaaS agents	By third-party providers – e.g., Salesforce agents, ServiceNow agents, purpose-built vertical SaaS agents.	Need to be brought inside the organization's governance perimeter so at the very least they're observable and cost-monitored centrally.

You need all agent interactions to route through the platform so no matter who built it (internal or external) or which framework it was built on, the same governance is enforced at runtime.

What every agent inherits at runtime

When an agent is registered on the platform and receives a request, several things need to happen automatically – before the agent executes a single action:

01 **Identity and authorization:** Its identity is verified against the enterprise IdP, with scoped permissions for the tools and data it's authorized to access.

02 **Guardrails:** The request passes through configured guardrails including content safety checks, PII detection, topic restrictions, and any custom policies the CISO or governance team has defined. These are enforced at the gateway, not implemented per agent.

03 **Tool access:** Mediated through a central MCP tool gateway – a single catalog of approved APIs and internal services that any agent can invoke with appropriate access levels, with access controlled and audited centrally.

04 **Model routing:** The model call is routed to the appropriate LLM tier based on task complexity, with automatic failover across providers and regions, and semantic caching to reduce latency and token spend.

05 **Observability:** Every step of the interaction is traced in an OTEL-compatible format, feeding a central observability layer that tracks performance, cost, and output quality per agent and per team.

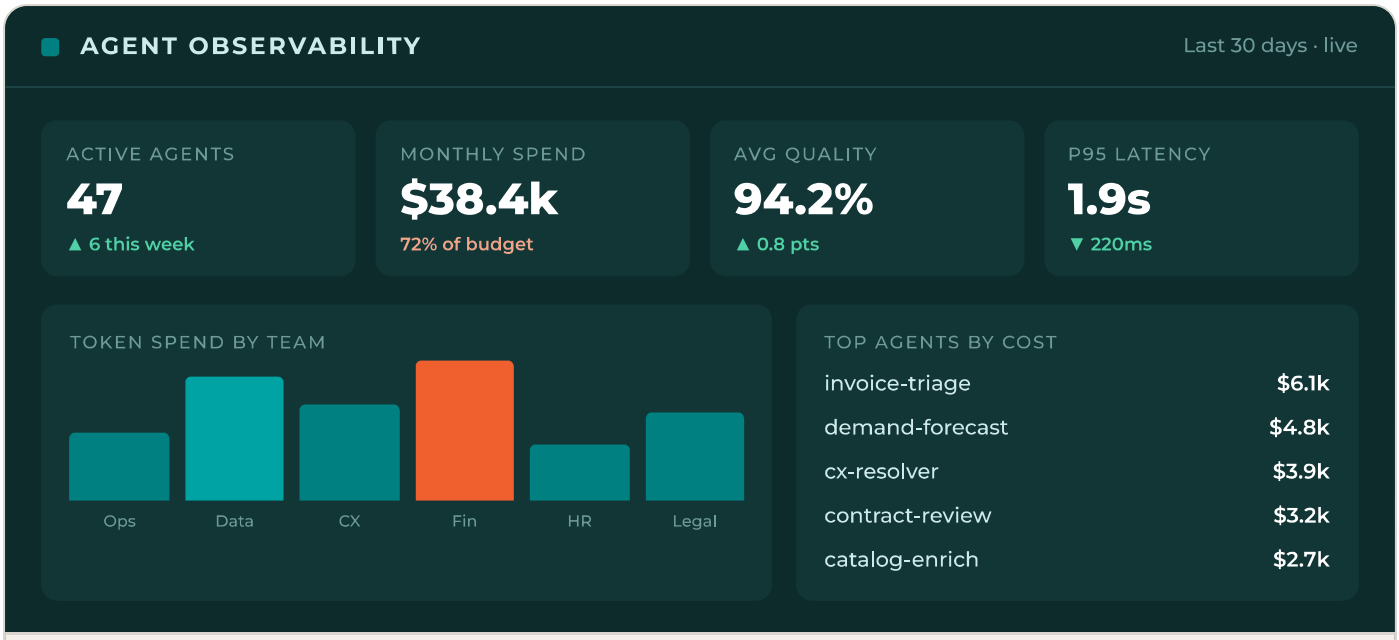

Figure 1. Observability dashboard: single view of agent performance, cost attribution, and quality metrics.

Illustrative interface — sample data.

Agent portal: governance as a feature, not a process

One of the most operationally significant components of the platform is the agent portal. You need a role-scoped interface through which teams discover, interact with, and manage agents.

The portal functions both as a consumer interface (teams find and use agents relevant to their work) and a governance tool (operators approve, monitor, and decommission agents from the same interface).

No agent goes live without review. For trusted teams with an established track record, this can be automated. For new teams or higher-risk agent types, it can remain a manual gate. You can configure this based on enterprise requirements.

The portal should also support agent-to-agent discovery. That way, an orchestrator agent can query the registry by capability and invoke specialist agents dynamically, without those integrations needing to be hard-coded. This is how multi-agent workflows become composable rather than brittle.

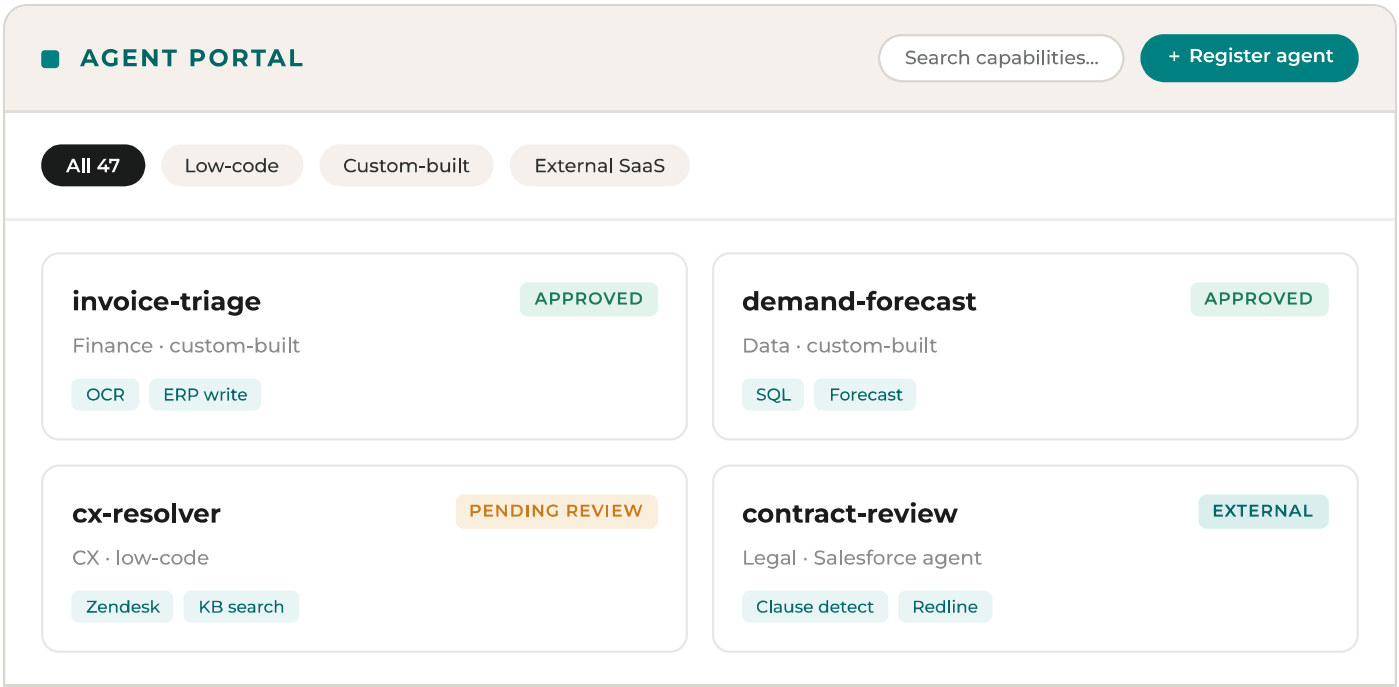

Figure 2. Agent portal: discovery catalog with role-based access and approval status.

Illustrative interface — sample data.

Complete control, avoiding vendor lock-in

A properly built agentic AI platform is built on open standards – MCP for tool integration, A2A for agent-to-agent communication, OTEL for observability, and OpenFGA for policy enforcement. This means agents, tools, and governance logic remain portable.

MCP · tool integration | **A2A · agent-to-agent** | **OTEL · observability** | **OpenFGA · policy**

Brighting takes this further. Our agentic AI platform is delivered as infrastructure-as-code into your own cloud platform account, with full source code under a perpetual license. Your data never leaves your environment, there's no SaaS subscription to renew, and no dependency on any single vendor to keep your platform operational.

04 Is your organization ready to scale agentic AI?

That's what's possible – and best practice. How ready are you to implement it?

Before committing to a platform approach – or to any significant agentic AI investment – it's worth being rigorous about where you currently stand. The questions below are drawn from the capability framework we use in our readiness assessments. This isn't the full list, but they're the questions that most reliably surface gaps that affect the organization's ability to scale agentic AI.

01 IDENTITY AND AUTHORIZATION

- [ ] Does every agent have its own identity with its own access controls and audit trail?
- [ ] Are least-privilege scopes defined and enforced per agent and per user?
- [ ] Does your identity setup integrate natively with your enterprise IdP for federated authentication and SSO?

02 AGENT DISCOVERY AND REGISTRY

- [ ] Is there a centralized catalog where teams can publish agents with their capabilities, ownership, and dependencies?
- [ ] Can agents and orchestrators query the registry by capability for dynamic composition?
- [ ] Do you have visibility into which downstream services, tools, and models each agent depends on?

03 COMMUNICATION & INTEGRATION

- [ ] Are agents able to communicate asynchronously, including for long-running tasks and partial updates?
- [ ] Are tools and APIs exposed through a unified MCP gateway that agents can discover and invoke at runtime?
- [ ] Does your setup support real-time, bidirectional streaming for voice or chat experiences?

04 RUNTIME AND EXECUTION

- [ ] Is agent execution framework-agnostic or locked to one vendor's stack?
- [ ] Does the runtime scale dynamically with demand, with session isolation between agents?
- [ ] Can agents handle complex async workloads running over extended periods without timing out?

05 MEMORY AND CONTEXT

- [ ] Do agents maintain conversational context within a session?
- [ ] Is there persistent memory across sessions for user preferences, prior decisions, and accumulated knowledge?
- [ ] Is there a centralized memory layer for cross-agent state sharing and session checkpointing?

06 GUARDRAILS AND POLICY

- [ ] Are there built-in and custom guardrails to filter unsafe or non-compliant outputs?
- [ ] Are agent boundaries defined by policy and enforced at the gateway, not just documented?
- [ ] Are there configurable escalation points where agents pause for human approval before high-risk actions?

07 OBSERVABILITY & EVALUATION

- [ ] Do you have end-to-end, OTEL-compatible traces across all agent execution?
- [ ] Are token usage, latency, error rates, and cost tracked per agent in real time?
- [ ] Is agent behavior continuously evaluated against correctness, safety, and goal success?

08 DEVELOPER EXPERIENCE

- [ ] Can teams register and deploy agents via SDK, CLI, or IaC without a central approval bottleneck?
- [ ] Is there a local development and testing environment before cloud deployment?
- [ ] Is there CI/CD integration for automated agent testing and deployment?

09 USER INTERACTION

- [ ] Are rate limits and access tiers managed at the platform level, with role-based controls per team?
- [ ] Is response caching in place for identical queries to reduce latency and cost?

10 COST AND GOVERNANCE

- [ ] Do you have a unified view of costs per agent, team, and model for chargeback and optimization?
- [ ] Is model routing matched to task complexity to optimize cost and performance?
- [ ] Can different business units operate with custom claims, tenant-scoped policies, and isolated data planes?

Most organizations find significant gaps in at least 3 of these domains. The question is whether those gaps are addressed before scaling or discovered during it.

Brighting's full readiness assessment covers **42 capabilities across these 10 domains**, scored against a maturity scale, with each gap mapped to a concrete business risk or scalability outcome. The output is a prioritized capability map – a sequenced view of which gaps carry the most risk,

which are fastest to close, and what a realistic path to production-ready looks like from your current position.

CONCLUSION

The organizations that scale AI fastest aren't the ones that build the most agents, they're the ones that build the right foundation.

Because when you have the right platform in place, every new use case becomes a configuration exercise rather than a rebuild.

The window to do this well is narrowing. Once shadow AI is deeply embedded and teams have built their own incompatible governance approaches, rationalizing it becomes a major organizational challenge as well as a technical one. The runway is significantly easier to build before the traffic arrives.

So if you're feeling the pressure – growing backlog, impatient board, CISO asking questions you can't fully answer – it's time to build the runway. That way, you'll have a clear path for making every future agent better, faster, and safer as the organization continues its AI journey.

READY TO PLAN YOUR RUNWAY?

Once people see how governance, access control, infrastructure, and observability work in the Brighting platform, their anxiety around scaling AI disappears. Are you ready for a similar light-bulb moment?

Contact us to discuss how we can address your scaling challenges – and help you stay a step ahead of AI demands.

BOOK A LIVE DEMO
brighting.nl/agentic-ai-platform

CHAT WITH AN EXPERT
info@brighting.nl

CALL US
+31 20 210 13 90

ABOUT BRIGHTING

Most firms talking about agentic AI are describing architectures they haven't yet built. Brighting has been building and running them in production. We're a cloud-native engineering firm – AWS Advanced Tier Partner, Lambda Service Delivery designated, and one of a small number of partners globally to hold the AWS Retail Competency.

Brighting's agentic AI platform gives you every capability every agent will ever need across governance, access control, infrastructure, and observability in a shared layer, inherited automatically. You get control without lock-in – it deploys in your own environment so you own the infrastructure, and you get the source code with a perpetual license.

It's not a consulting deliverable or proof of concept, it's infrastructure we've deployed, stress-tested, and iterated across enterprise environments at HEMA, ASICS and many more. If you're serious about scaling agentic AI and want a partner who's already done it, we're the partner to speak to.

Learn more at brighting.nl/agentic-ai-platform