tray.ai

The AI agent strategy playbook

A PRACTICAL FRAMEWORK FOR OPERATING AI AGENTS AT ENTERPRISE SCALE

Opening move:

From early wins to enterprise orchestration

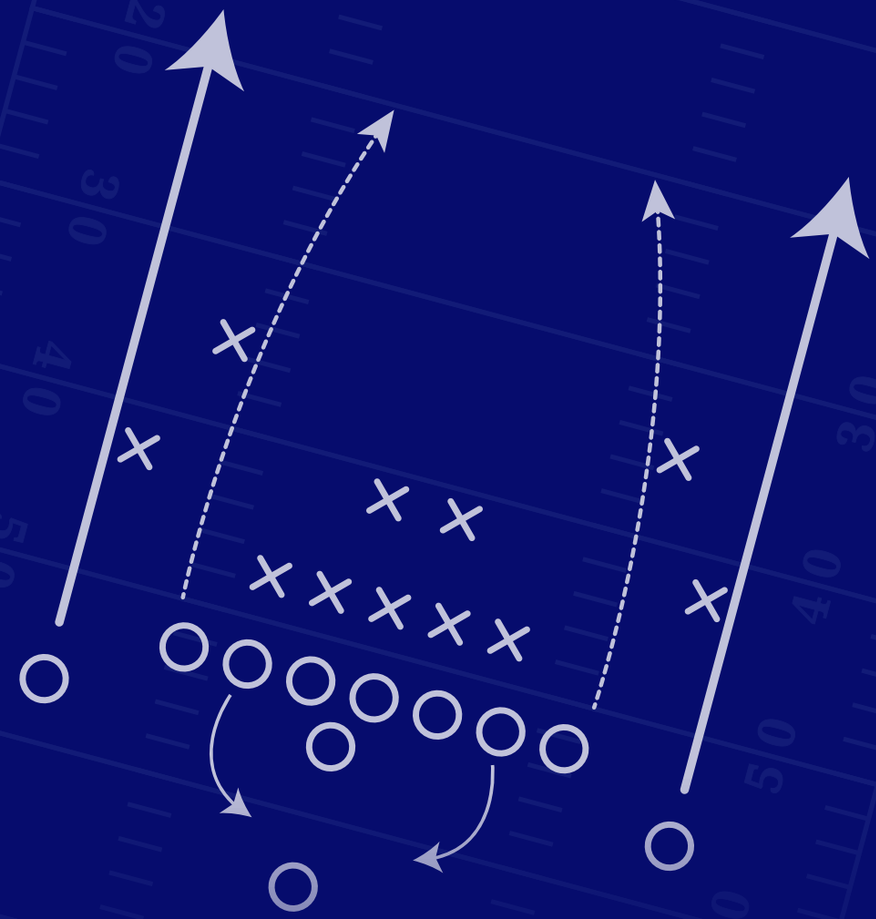
Enterprises have moved past asking whether to use AI agents. The challenge now is to turn early successes, which are often driven by chat tools, copilots, or single-purpose agents, into systems that can scale reliably across the business without creating fragmentation, risk, or uncontrolled costs.

As agent usage expands and use cases grow more complex, gaps in integration, governance, and coordination surface quickly. Agents built for isolated tasks struggle when they are expected to operate across teams, data, and systems.

This playbook breaks down the most common AI agent operating models, the trade-offs of each, and how to determine which model fits your architecture, security, and ability to scale.

We'll walk through:

- **What each operating model looks like in practice**
- **Where each model fits (and doesn't)**
- **How to evaluate agent strategies based on control, speed, scalability, and governance**
- **Why AI orchestration platforms are designed for production-grade, cross-functional agent systems**



Enterprises want AI agents—but their tech stacks aren't ready for game day

ENTERPRISE AI STATS

GARTNER FORECAST

80%

of automation-mature enterprises will consolidate onto a single orchestration platform for agentic automation by 2029.[1]

TRAY ENTERPRISE SURVEY

86%

need stack upgrades to properly deploy and scale AI agents.[2]

FROM ISOLATED PILOTS TO CROSS-FUNCTIONAL PLAYS, FOUNDATIONAL GAPS ARE PUSHING ENTERPRISES INTO THE RED ZONE

THE DEFENSIVE LINE: WHAT'S BLOCKING YOUR AGENTS?

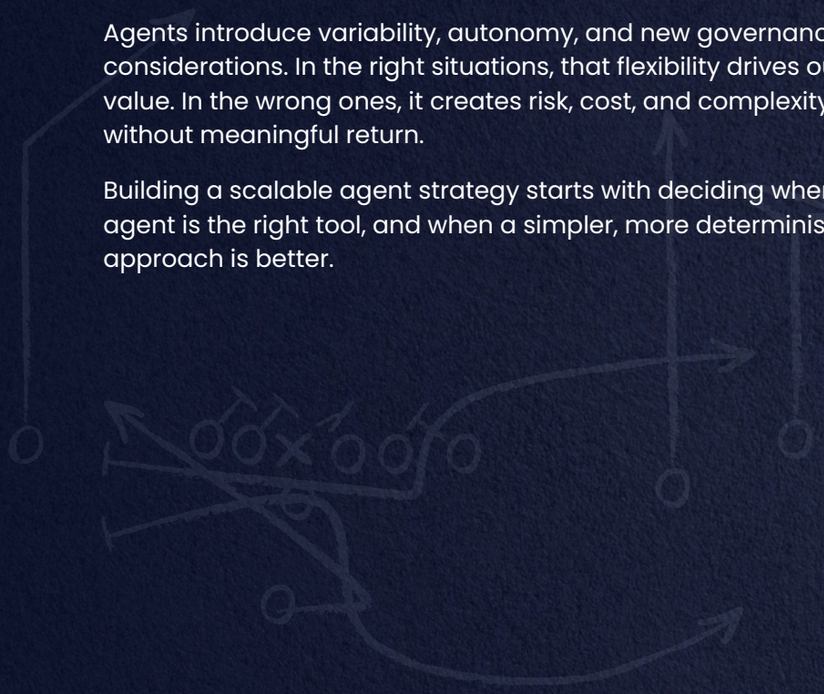
Governance and security	Without clear ownership and guardrails, AI agents introduce risk through unauthorized actions, policy violations, and data leakage.
Scalability and performance	As agent usage grows and complexity increases, coordinating tasks, state, and execution across systems breaks down under sustained load.
Integration complexity	Fragmented system connectivity limits agents' ability to act across workflows and data, reducing reliability and preventing agents from delivering consistent outcomes.
Talent and ownership gap	While building agents is easier than before, unclear ownership and accountability slow progress and create friction across teams responsible for operating them.
Cost and ROI pressures	When agents proliferate without a clear operating model, duplication and inefficiency drive up costs and make ROI difficult to measure or sustain.
Deployment timelines	Misaligned execution paths and rework slow delivery, causing agent initiatives to lose momentum before they reach meaningful scale.

Only run agents when they win

As AI agents become easier to build, it's tempting to use them everywhere. But ease of creation doesn't guarantee operational success.

Agents introduce variability, autonomy, and new governance considerations. In the right situations, that flexibility drives outsized value. In the wrong ones, it creates risk, cost, and complexity without meaningful return.

Building a scalable agent strategy starts with deciding when an agent is the right tool, and when a simpler, more deterministic approach is better.



WHEN AN AI AGENT MAKES SENSE

- Inputs are unstructured or ambiguous
- Decisions require interpretation or judgment
- Context changes over time
- Multiple systems must be consulted before acting
- Human review or escalation may be required
- Outcomes improve through iteration

WHEN AN AI AGENT DOESN'T MAKES SENSE

- Work follows fixed rules
- Inputs and outputs are well defined
- Deterministic execution is required
- Latency or cost sensitivity is high
- Compliance demands predictable outcomes
- Failure tolerance is low

THE BOTTOM LINE

USING AGENTS WHERE AUTONOMY ADDS VALUE (AND AVOIDING THEM WHERE IT DOESN'T) IS THE FASTEST WAY TO PREVENT SCALE, COST, AND GOVERNANCE ISSUES LATER.

Strategy map:

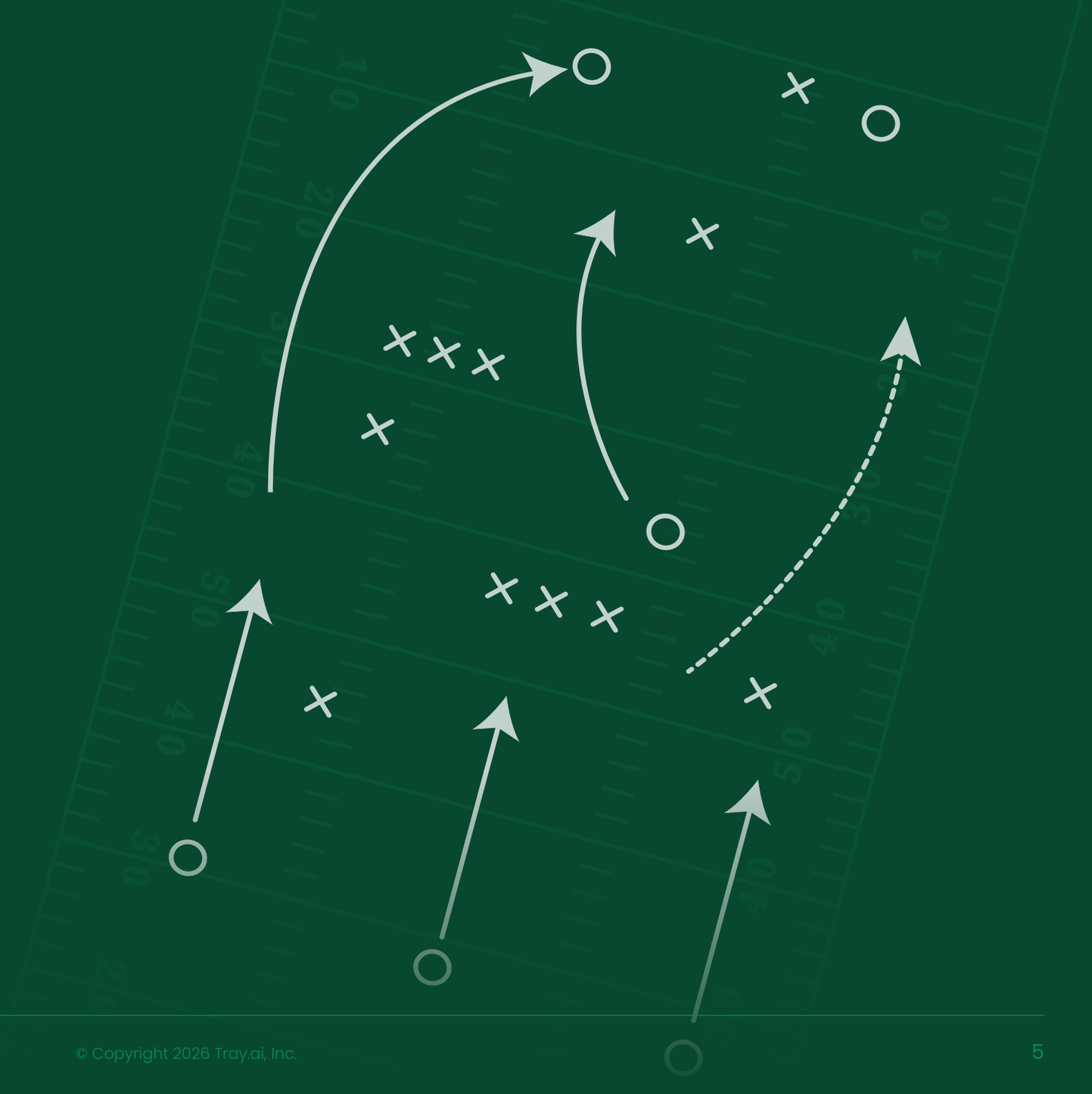
Four AI agent operating models

There's no single way to operate AI agents at scale. Different organizations make different trade-offs depending on how much control, coordination, and governance they require.

The models that follow represent the most common ways enterprises are structuring agent ownership and execution today. Each approach carries distinct implications for scalability, reuse, and long-term operability.

We'll explore these four strategic plays

- **Play 1: Internal custom agents**
- **Play 2: Off-the-shelf agents**
- **Play 3: SaaS app agents**
- **Play 4: AI orchestration platforms**



What they are

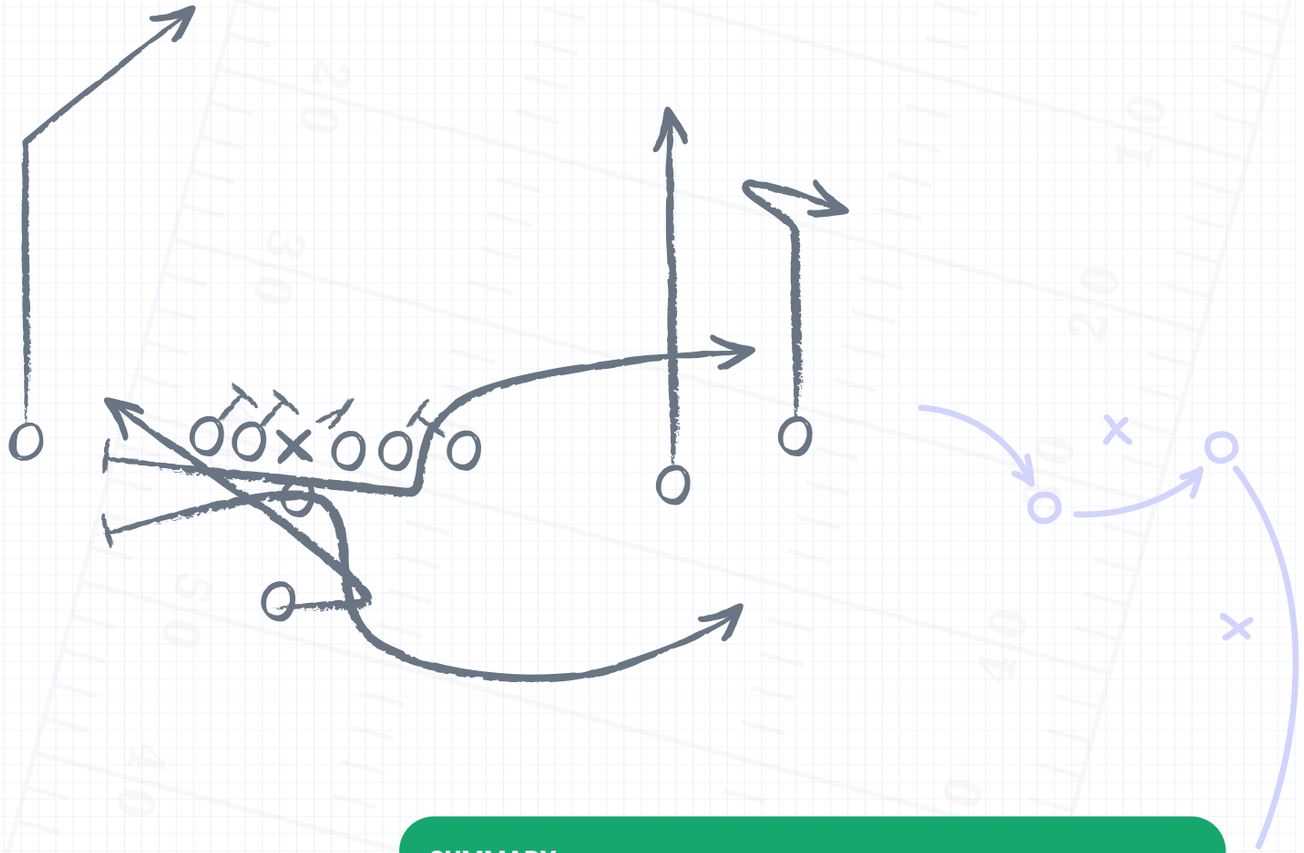
Custom-built AI agents developed and operated in-house by engineering teams.

How it's done

Internal teams design and maintain agents end to end, including data pipelines, orchestration logic, integrations, and execution paths. Each component is built specifically for internal systems, policies, and workflows.

Where it fits

Best for highly specialized use cases where full control over architecture, data, and execution is required, and where teams are prepared to own long-term maintenance, governance, and scale.



Trade offs

- Requires sustained engineering ownership, not just initial build effort
- Long build cycles delay time to business impact
- High upfront and ongoing costs compound as use cases expand
- Custom integrations increase operational and maintenance burden
- Difficult to reuse logic across teams and workflows
- Governance, security, and compliance must be designed and enforced internally

Run this when:

You need full control over architecture, data, and execution, and are willing to own the complexity that comes with it

Watch out for:

High cost, long timelines, and growing maintenance overhead as agents scale across teams

SUMMARY	
Speed to deploy	Slow
Governance	Designed and enforced internally
Customization	Full control
Scalability	Hard to scale across teams
Integration depth	Custom for every system

What they are

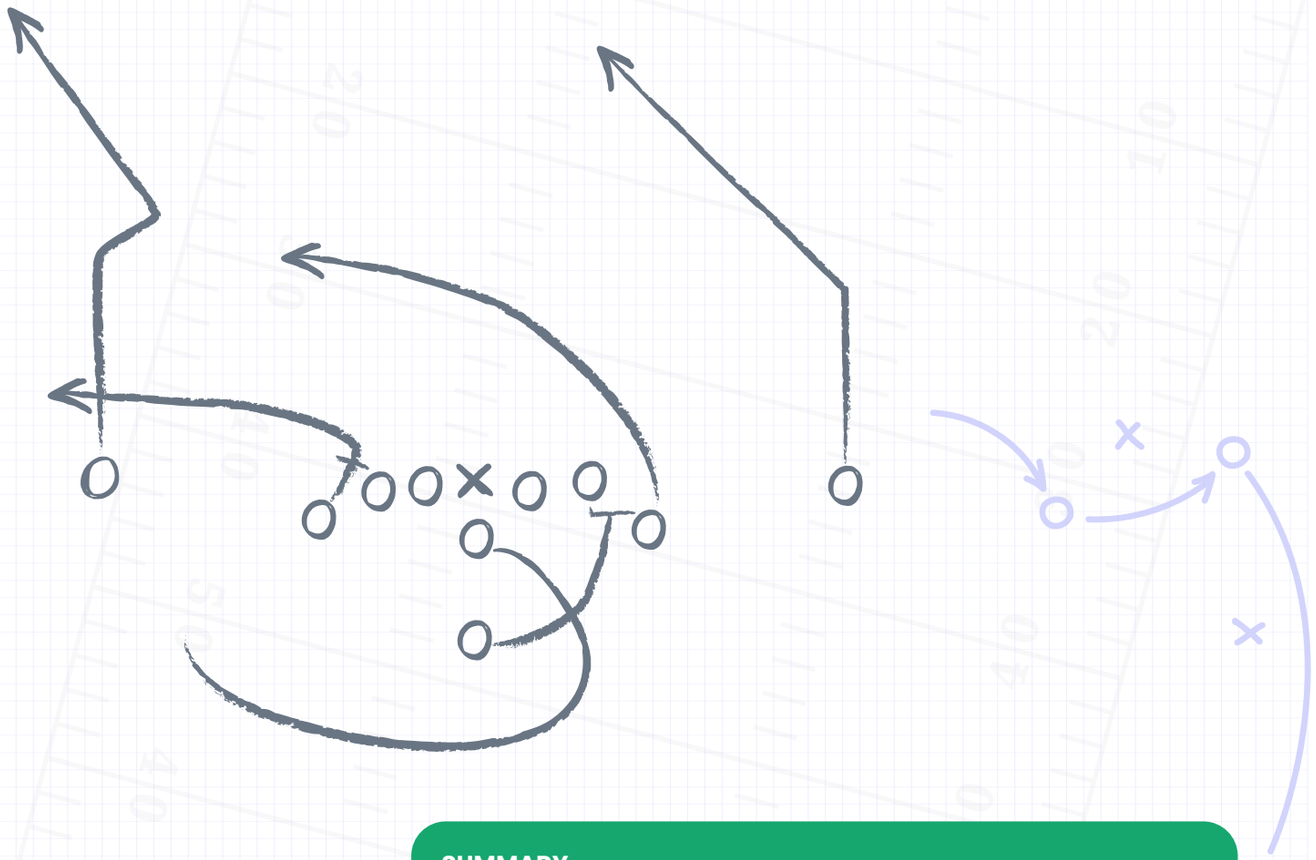
Pre-built AI agents designed to perform a narrow, predefined function such as chat, scheduling, or basic Q&A.

How it's done

Enterprises license agents from a vendor and configure them within fixed parameters. Behavior, integrations, and execution paths are largely defined by the vendor's product boundaries.

Where it fits

Best for isolated, low-risk tasks where speed matters more than integration depth, reuse, or long-term scalability.



Trade offs

- Limited integration constrains agent effectiveness
- Rigid designs restrict adaptation to new workflows or requirements
- Governance and policy enforcement vary by vendor
- Difficult to coordinate across teams or departments
- Vendor-specific logic creates long-term dependency
- Oversight fragments as agents proliferate across tools

Run this when:

You need to address a narrow use case quickly and are comfortable operating within vendor-defined constraints

Watch out for:

Integration limits, policy gaps, and growing dependency as agent usage expands

SUMMARY	
Speed to deploy	Fast
Governance	Vendor-defined
Customization	Minimal
Scalability	Not scalable
Integration depth	Low

What they are

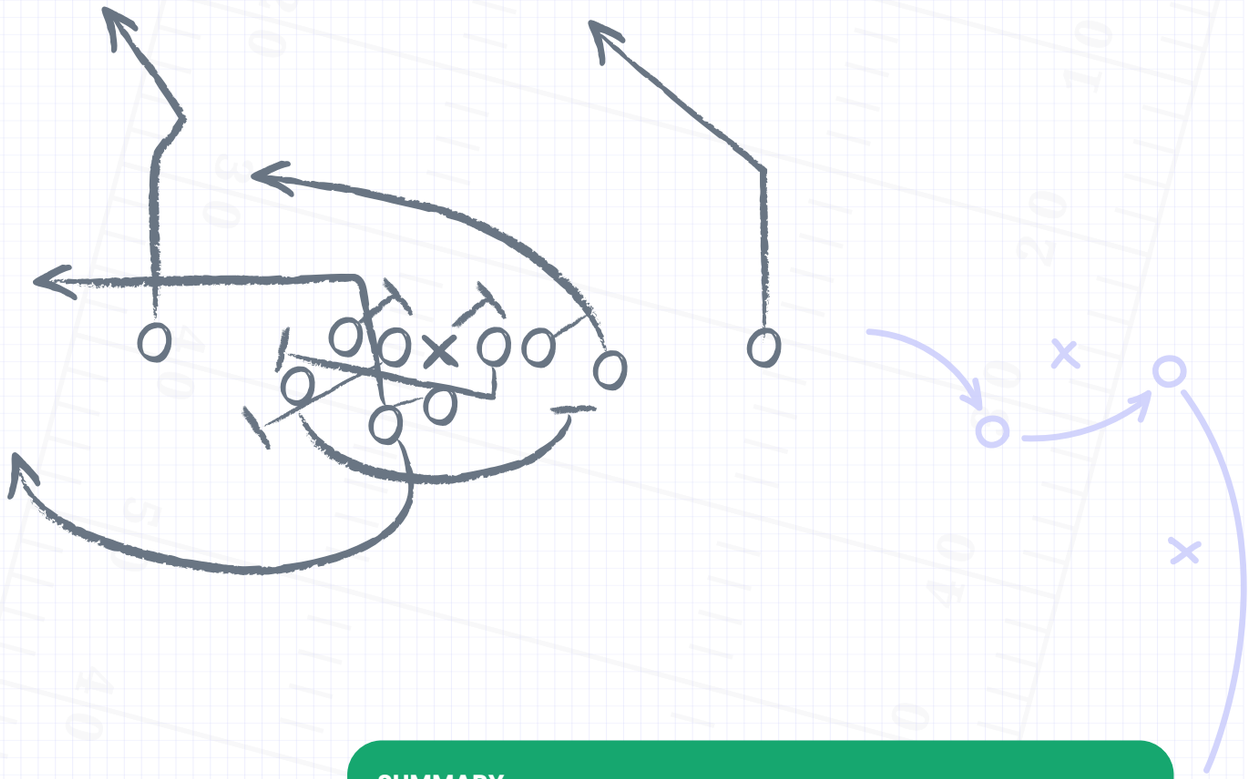
AI agents embedded within a specific SaaS platform that automate tasks tied to that application's data and workflows.

How it's done

Delivered as part of a SaaS product, these agents are configured and governed by the vendor to support platform-specific actions such as data entry, routing, and recommendations.

Where it fits

Best for improving efficiency within a single application, especially when teams already operate primarily inside that platform and don't require cross-system coordination.



Trade offs

- Confined to a single vendor ecosystem
- Limited flexibility beyond predefined workflows
- Interoperability with external systems is constrained
- Governance and policy enforcement are vendor-dependent
- Switching and licensing costs accumulate over time
- No shared orchestration across teams or functions

Run this when:

You want to improve execution inside a specific SaaS platform without extending automation beyond it

Watch out for:

Fragmented agent behavior, limited reach, and growing complexity as multiple SaaS agents accumulate

SUMMARY	
Speed to deploy	Fast (within platform)
Governance	Vendor-defined
Customization	Limited
Scalability	Confined to the platform
Integration depth	Platform-only

What they are

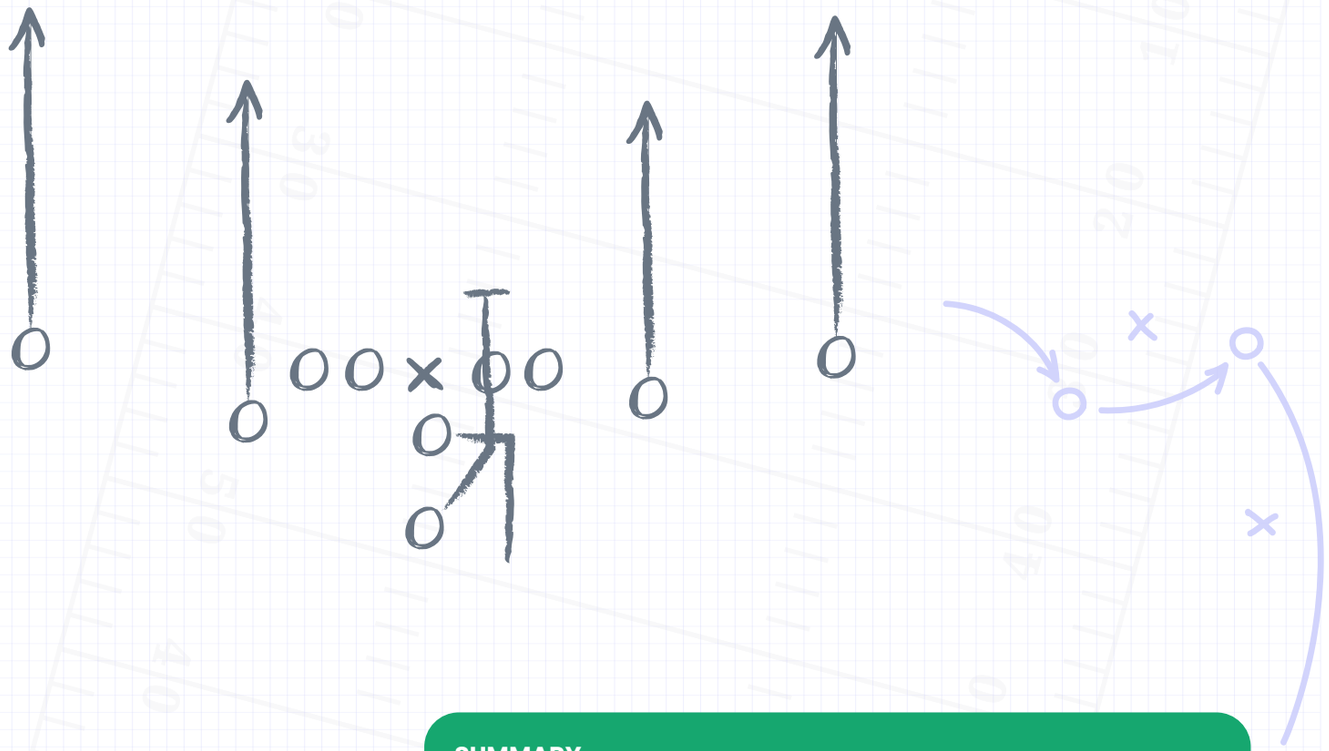
Platforms designed to coordinate, govern, and operate AI agents across systems, teams, and workflows, providing a shared foundation for agent execution at enterprise scale.

How it's done

Provides shared services for agent execution, including integration, workflow coordination, observability, and policy enforcement. Agents connect to systems through governed interfaces, so teams can build, run, and monitor agents across environments while maintaining centralized control.

Where it fits

Best for organizations operating multiple agents across teams and systems that need consistent governance, shared infrastructure, and coordinated execution without rebuilding foundations for every use case.



Trade offs

- Requires upfront platform onboarding and alignment
- Demands clear ownership and operating standards
- Introduces change management across teams and tools

Run this when:

You're moving from isolated agent deployments to coordinated, enterprise-wide operation

Watch out for:

Lack of ownership or governance clarity, which can undermine the benefits of orchestration

SUMMARY	
Speed to deploy	Moderate
Governance	Centralized
Customization	High
Scalability	Enterprise-wide
Integration depth	Cross-system

The lineup:

How AI agent operating models work together

Most enterprises don't rely on a single way to use AI agents. Different operating models emerge based on speed, specialization, and where work happens inside the business.

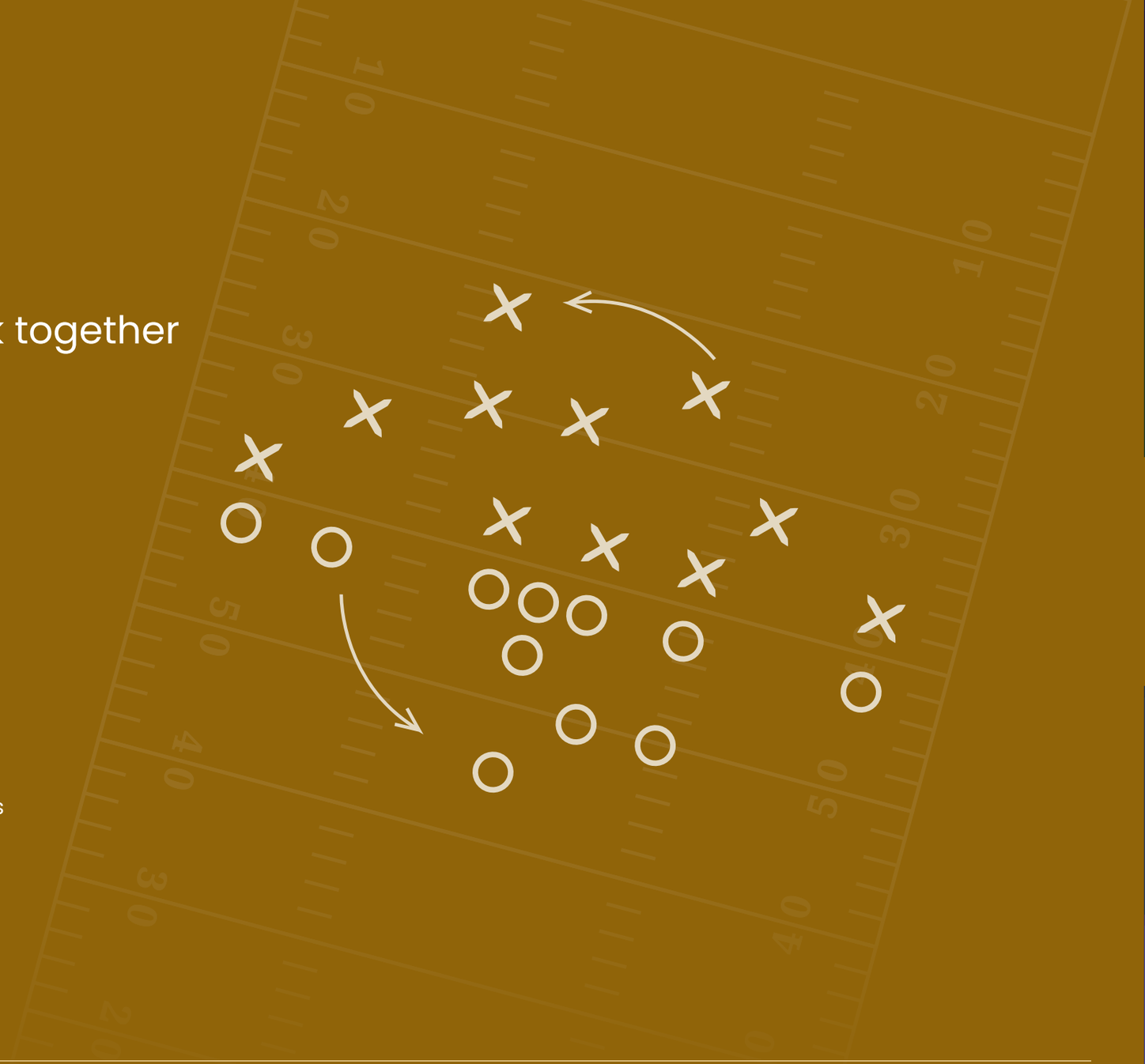
In practice, these models play for the same team:

- ➜ **Off-the-shelf agents handle narrow, predefined tasks**
- ➜ **SaaS app agents operate inside individual platforms**
- ➜ **Internal custom agents support specialized logic and workflows**

Each plays a role, but they're limited by where they live and what they can access.

As organizations move beyond isolated use cases, a different class of agent becomes necessary: agents that can operate across systems, act on trusted data, and run within consistent enterprise controls.

That's where AI orchestration platforms fit.



Winning move:

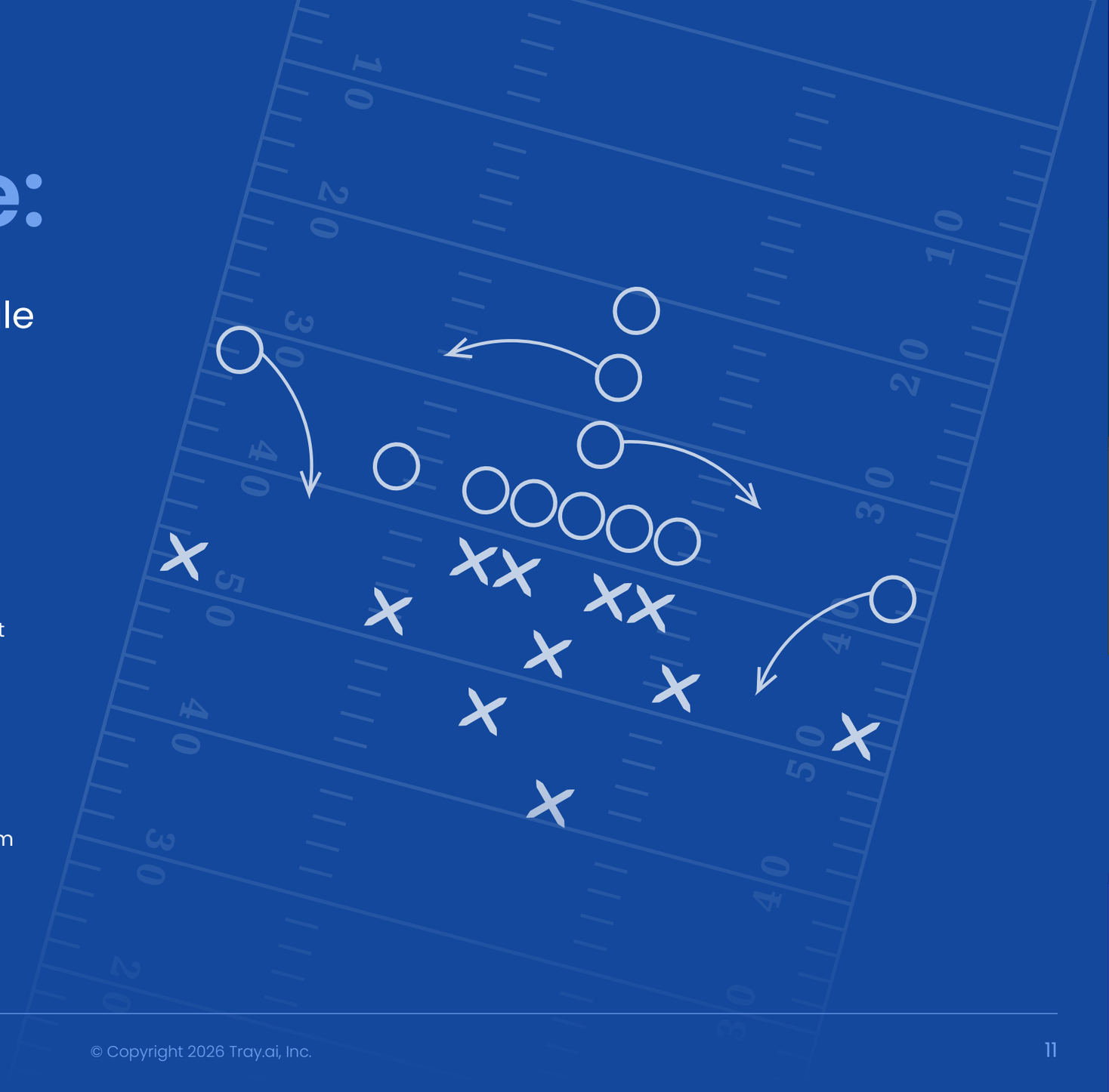
Why AI orchestration matters at scale

Enterprise AI adoption is accelerating, but success depends on more than getting agents live. The harder problem is sustaining them across teams, systems, and use cases without creating fragmentation, risk, or operational drag.

Off-the-shelf agents and SaaS-native agents help teams move quickly inside narrow boundaries. Custom-built agents offer control, but are costly to maintain and difficult to scale. As these models accumulate, enterprises face a coordination problem: disconnected agents, duplicated logic, and inconsistent governance.

AI orchestration platforms address this gap. They provide a standardized way to build, deploy, and operate agents that must work across systems, while enforcing shared policies, access controls, and observability.

That's the difference between launching agents and running them as enterprise infrastructure.



Why Tray is built for enterprise AI agents

AI agents need more than a prompt and a model. They must connect to enterprise systems, follow governance policies, and operate reliably at scale. Tray is an AI orchestration platform designed to support the full lifecycle of enterprise AI agents—from build to execution—across teams, systems, and data.

Build and govern agents	**Merlin Agent Builder**	• Design and configure agents using guided workflows and reusable components • Provides shared tools, data sources, and logic blocks to standardize agent development • Create multi-agent orchestration while maintaining consistent guardrails as agents scale across use cases
Build and govern MCP	**Agent Gateway**	• Build and deploy MCP Tools based on sophisticated workflows for total control over what they do • Deploy MCP Servers that securely connect across your stack • Centralized instrumentation, detailed logs, and security to monitor, audit, and manage MCP.
Work with data	**Vector Tables & IDP**	• Enable agents to retrieve and reason over structured and unstructured enterprise data • Automate document ingestion, classification, and extraction • Support multiple LLMs for generation, summarization, and decision support
Connect	**700+ integrations**	• Pre-built connectors to SaaS applications, databases, and internal systems • Native support for REST, GraphQL, SOAP, webhooks, and event-driven triggers • Allow agents to take action where work actually happens
Scale	**Serverless, cross-system orchestration**	• Coordinate agent execution across departments and workflows • Elastic, serverless infrastructure handles high-volume execution without manual provisioning • Centralized monitoring and observability across agents and automations

Learn from the teams already winning with AI agents

Each team faced different constraints, but won by designing AI agents to be orchestrated, governed, and scalable from the kickoff.

PLAY 1:

Apollo — WHEN AGENTS ARE ALLOWED TO TAKE ACTION

THE SITUATION

Apollo identified 80+ agent use cases but needed a safe way to deploy agents across systems without rigid, prebuilt tools.

THE PLAY

Starting in IT, they deployed a Slack-based ITSM agent that auto-resolves issues and escalates when needed, then reused the model across teams.

WHY IT WORKED

40%	TICKET DEFLECTION
4.9	IT CSAT

"IT'S NOT JUST ANSWERING QUESTIONS, IT'S TAKING ACTION. THAT'S WHY PEOPLE TRUST IT."

RAMIRO MEYER, HEAD OF IT

PLAY 2:

Life360 — WHEN ONE AGENT COORDINATES MANY

THE SITUATION

Single-purpose bots confused employees, while all-in-one agents didn't scale, driving constant pings to IT.

THE PLAY

Life360 built one orchestrator agent on Tray to route requests to specialized sub-agents, governed and scaled through shared workflows.

WHY IT WORKED

1	FRONT DOOR FOR EMPLOYEE REQUESTS IN SLACK
2500+	SLACK CHANNELS AUDITED AUTOMATICALLY

"WE NEEDED ONE PLACE TO ASK AND LET THE SYSTEM DECIDE WHICH AGENT TO USE."

MATT CURRIE, SENIOR AUTOMATION ENGINEER

PLAY 3:

J.W. Pepper — WHEN GOVERNANCE COMES BEFORE SCALE

THE SITUATION

As teams adopted MCP and AI agents, J.W. Pepper needed to prevent over-permissioned access and unmanaged integrations without slowing adoption.

THE PLAY

They introduced a governed MCP layer with Tray Agent Gateway between agents and enterprise systems.

WHY IT WORKED

FASTER	TICKET HANDOFFS WITH FULL CONTEXT PRE-WORK
SAFER	AI ENABLEMENT

"YOU DON'T REALLY WANT PEOPLE INTEGRATING THEIR OWN MCPS. YOU WANT SOME GOVERNANCE AROUND IT."

MARCUS DUBREUIL, DIRECTOR OF SYSTEMS ARCHITECTURE

From idea to impact:

It’s not about launching one agent. It’s about building a foundation that scales, adapts, and stays in your control.

Tray gives you the platform to operationalize your AI agent strategy—across teams, systems, and use cases.

Talk to a strategist →

Get direct, technical input on how this model will work in your stack.

Request a 1:1 workshop →

Work with our team to discuss your goals, systems, and current constraints, and how AI orchestration fits into your agent and automation strategy.

Plan your ITSM agent impact →

ITSM is one of the fastest, most valuable places to start with agents. Use our ITSM AI agent capacity model to estimate hours saved, tickets resolved, and service desk capacity gained.

About Tray.ai

Tray.ai is the leader in enterprise orchestration for data and AI that enterprises use to build smart, secure AI agents at scale. It eliminates the need for disparate tools and technologies to integrate and automate sophisticated internal and external business processes and speeds the creation and deployment of high-value, production-ready AI agents. Enterprises can now avoid the traps of high costs and long lead times typical in custom agent development as well as the constraints and silos created by implementing and managing single-purpose agent offers from each SaaS application in the enterprise tech stack. With Tray.ai, the development of integrations, the delivery of intelligent apps and the integration of trusted data anywhere is fast, flexible and safe. Learn more at Tray.ai.

Citations

1. Gartner (2023, October). Strategic roadmap for automation. Gartner, Inc.
2. Tray.ai (2024). Enterprise AI agent survey report. https://tray.ai/resources/reports-ebooks/enterprise-ai-agent-survey