

INDUSTRY REPORT - AUGUST 2026

The Hidden Disconnects in Enterprise AI

Tactics for aligning coding agents, business agents
and the APIs they depend on

Research conducted by:



© Jentic 2026

Contents

Executive Thesis

The Six Disconnects

Disconnect 1: Coding agents outside the platform model

Disconnect 2: Rigidity over flexibility

Disconnect 3: AI-first without API-first

Disconnect 4: Projects without purpose

Disconnect 5: Symbolic security

Disconnect 6: Limited sandbox leverage

Conclusion

Platform maturity checklist

Methodology

Respondent profile

Acknowledgements

Sources

03

04

05

07

09

12

14

16

18

19

20

21

23

24

About this report

Enterprise AI is increasingly being adopted through two tracks: coding agents and business agents. Both depend on the same platform foundations: APIs, data, identity, access control, observability, governance and sandbox environments.

This report identifies six disconnects that appear when those tracks are implemented separately. It draws on desktop research, a practitioner survey and enterprise interviews to show where fragmentation is emerging, and what platform, API, architecture and AI leaders can do to close the gaps.

Written by Erik Wilde, Sean Blanchfield and Mark Boyd.
Design by Elena Anagnostatou.

The enterprise push for agents is creating a fragmented platform model

Enterprise success with agents depends on platform maturity.

Coding agents and business agents are both operating on the same enterprise foundations: APIs, data, identity, access control, observability, testing environments and governance. But in many organisations, they are being introduced through separate processes.

That separation is most visible in how teams access business systems, how APIs are prepared for machine consumption, how security controls are implemented, and how AI projects move from experimentation to production.

The result is not one enterprise AI platform, but **multiple local AI stacks**. Controls are duplicated. API access is handled inconsistently. Platform teams lose visibility. And agentic systems become harder to scale safely.

The answer is not to rebuild the stack from scratch. It is to **extend platform engineering disciplines** across the full agent lifecycle, so coding agents and business agents can be governed, integrated and improved through shared platform capabilities.

Research at a glance

**183**

reports &
articles
reviewed

**114**

AI & API
practitioners
surveyed

**16**

enterprise
leaders
interviewed

DEFINITIONS

Coding agents = AI tools that interpret natural-language instructions to produce software code, specifications, tests, documentation or other software-development outputs.

Business agents = AI agents designed to execute business workflows within bounded digital environments.

Six disconnects are fragmenting enterprise AI adoption

01

Coding agents outside the platform model

Coding agents are now used across teams with no consistent platform visibility.

Platform response:
Treat coding agents as platform components, not just productivity tools.

02

Rigidity over flexibility

Governance is built around approved-tool lists that cannot keep pace with AI change.

Platform response:
Govern usage patterns, access, validation and monitoring.

03

AI-first without API-first

Agents are being connected to APIs before those APIs are ready for machine consumption.

Platform response:
Assess and modernise priority APIs for agent use.

04

Projects without purpose

Experimentation is moving faster than strategic prioritisation and ROI discipline.

Platform response:
Keep experimentation open, but gate production with business-value criteria.

05

Symbolic security

Guardrails exist, but identity, monitoring, downscoping and incident response remain weak.

Platform response:
Build operational security controls for agents.

06

Limited sandbox leverage

Simulation sandboxes are treated as test environments rather than core infrastructure for agent/workflow development and governance.

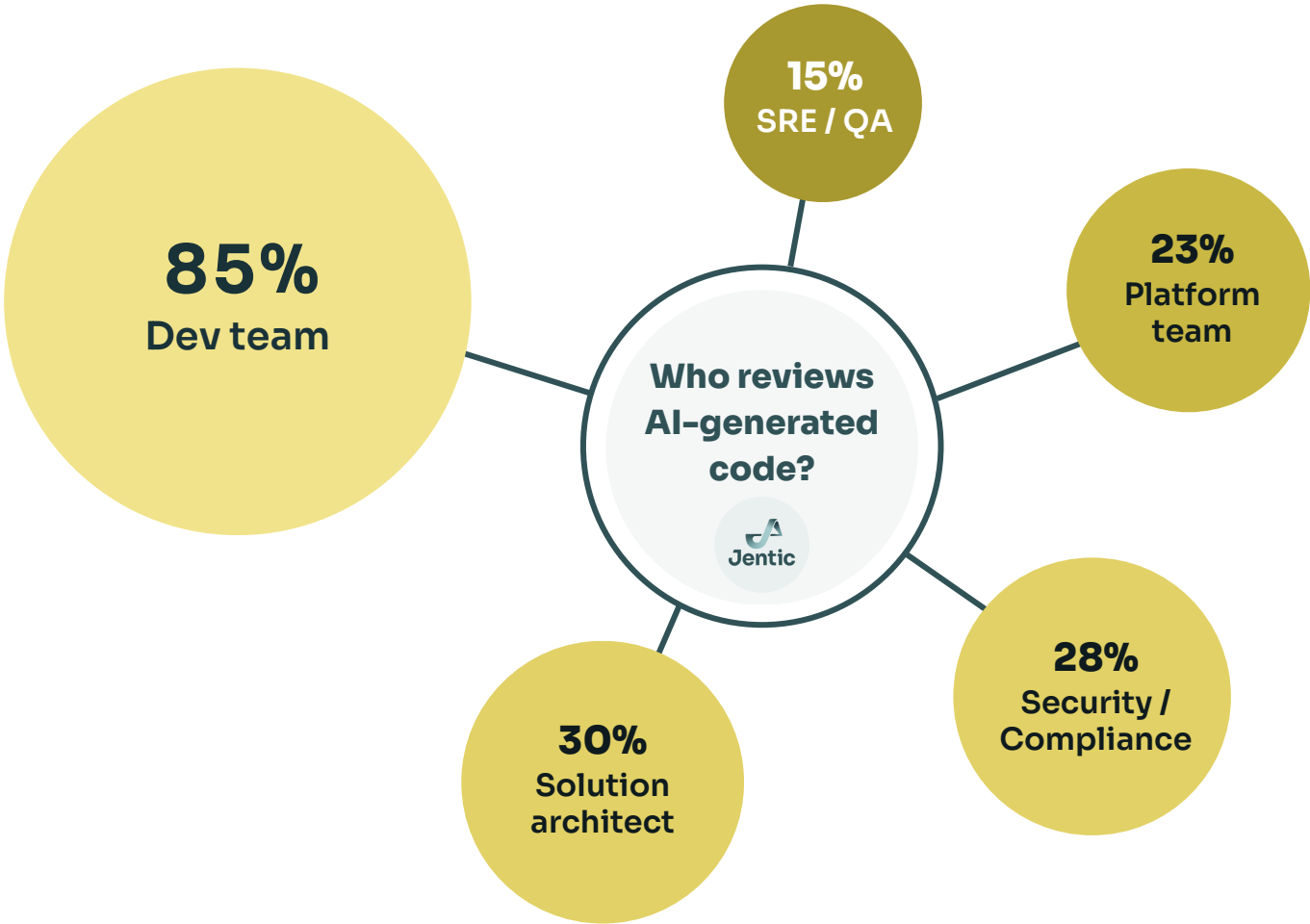
Platform response:
Use sandboxes across the full agent lifecycle.

01: Coding agents outside the platform model

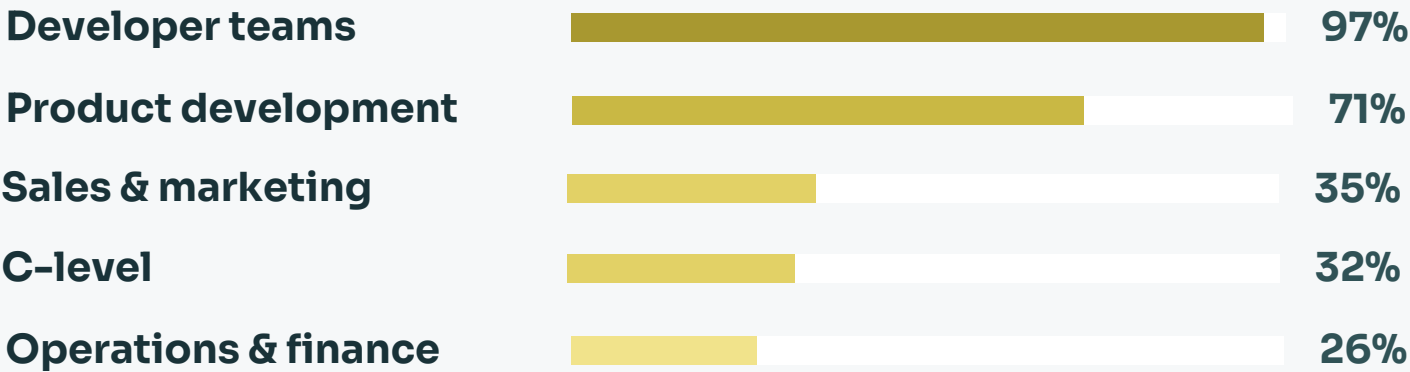
Coding agents are no longer confined to engineering teams. In our survey, **89%** of respondents reported coding-agent use across one or more enterprise teams, and **51%** said their organisation uses both coding and business agents.

But **only 16%** reported well-defined processes for AI use. That gap matters because coding agents increasingly touch the same assets as business agents: internal APIs, datasets, workflows, specifications and production-adjacent systems.

When coding agents are used as **productivity tools**, each department develops its own approach to access, validation, monitoring and risk.



Teams using coding agents:



You will need to speak with each individual department to understand how they are using coding agents.

— Enterprise interview comment
(received more than once)



Platform response:

Treat coding agents as platform components

1. Create one governance path for agentic work

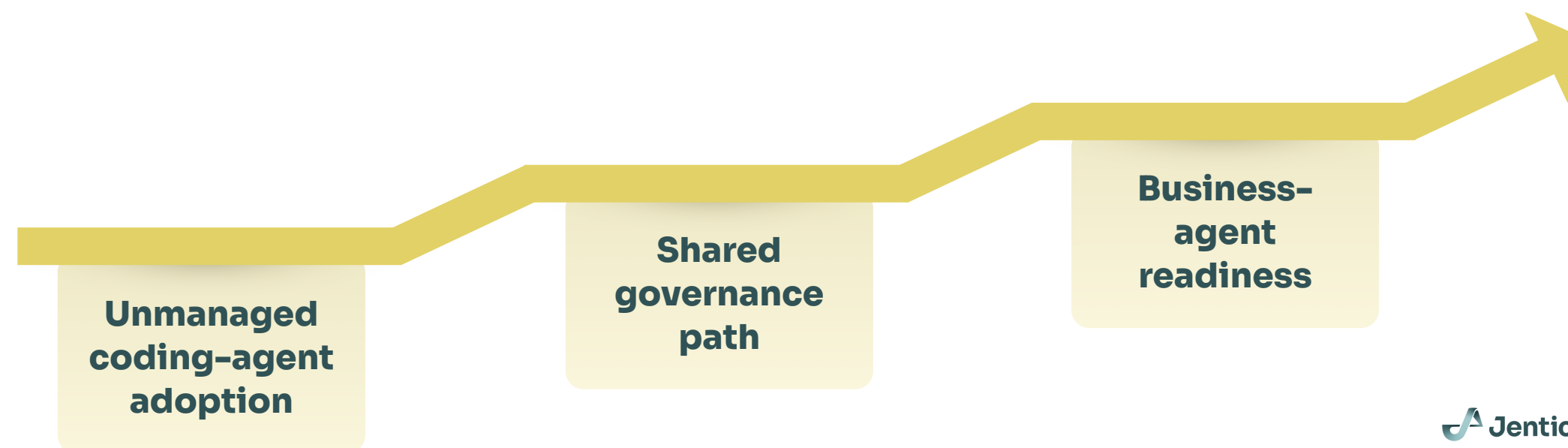
Apply consistent governance wherever agents touch enterprise systems: identity, access, data exposure, output validation, logging and escalation.

2. Use coding-agent adoption as business-agent preparation

Coding agents are often the first place teams learn how agents interact with enterprise assets. Treat that adoption as preparation for business agents, not as a separate tool rollout.

3. Take platform enablement beyond engineering

Platform teams need to support sales, marketing, finance, operations and product teams directly, not just developer teams.



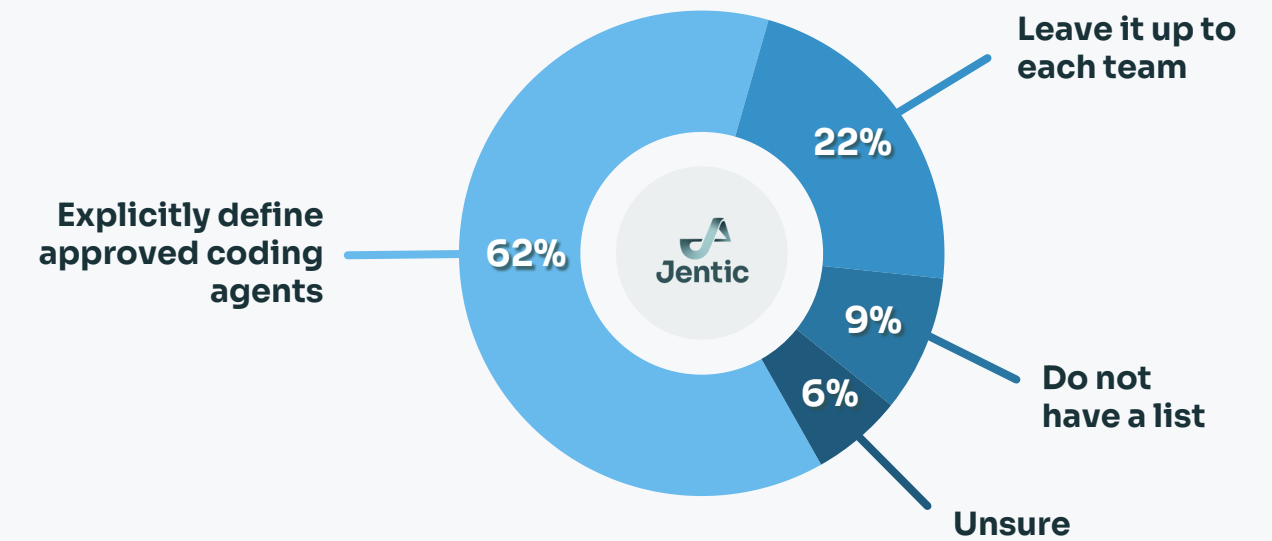
02: Tool lists are not enough governance

Many enterprises govern AI by defining which tools teams are allowed to use. Approved-tool lists are useful, but they are **not a complete governance model**.

AI tools, models and agent frameworks are changing too quickly for governance to depend on static lists alone. The risk depends less on the name of the tool than on **how it is used**: what data it can access, what systems it can call, what outputs it creates, and whether those outputs are validated.

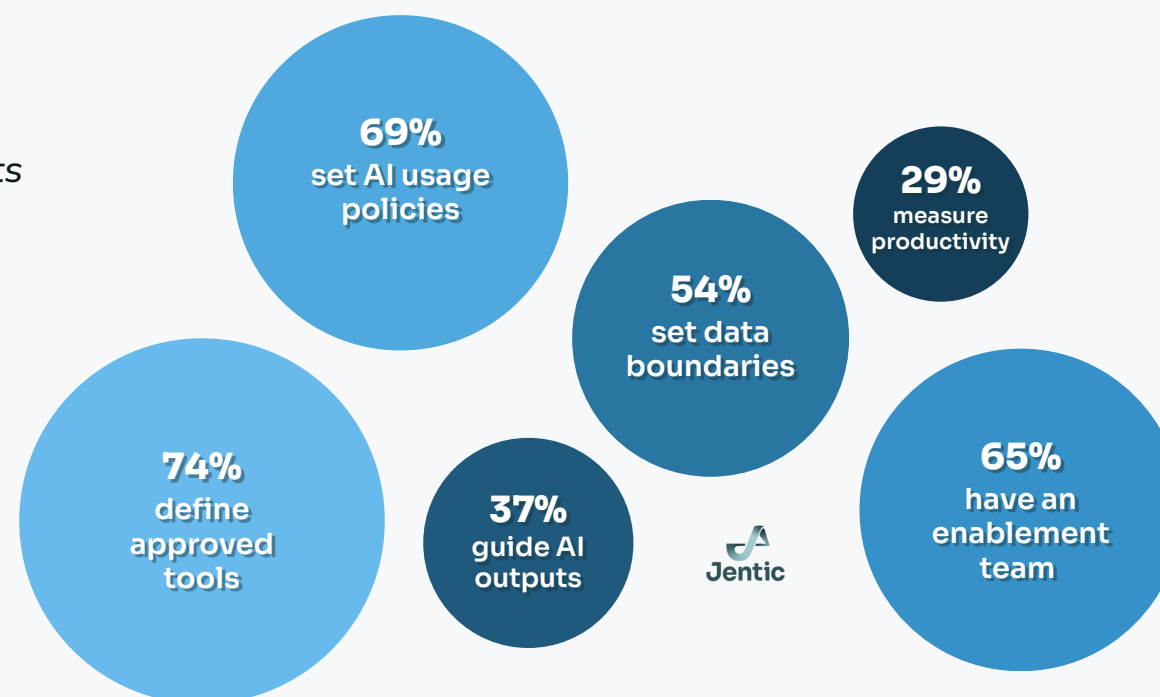
When approved tools frustrate users, teams often move to their own preferred stack. That creates **shadow AI**: usage that is invisible to platform, security and governance teams.

How are approved coding agents approved?



The “Check Gap”

Companies govern the inputs (which tools) but not the outputs (quality of results).



DEFINITIONS

AI governance = The policies, processes and controls that enable responsible, secure and effective AI use in line with organisational goals and constraints.

Shadow AI = AI used without authority or oversight in an organisation.

Harness engineering = Establishing the infrastructure, context, access policies, and so on to ensure agents operate within clear parameters.

Platform response: Govern usage, not only tools

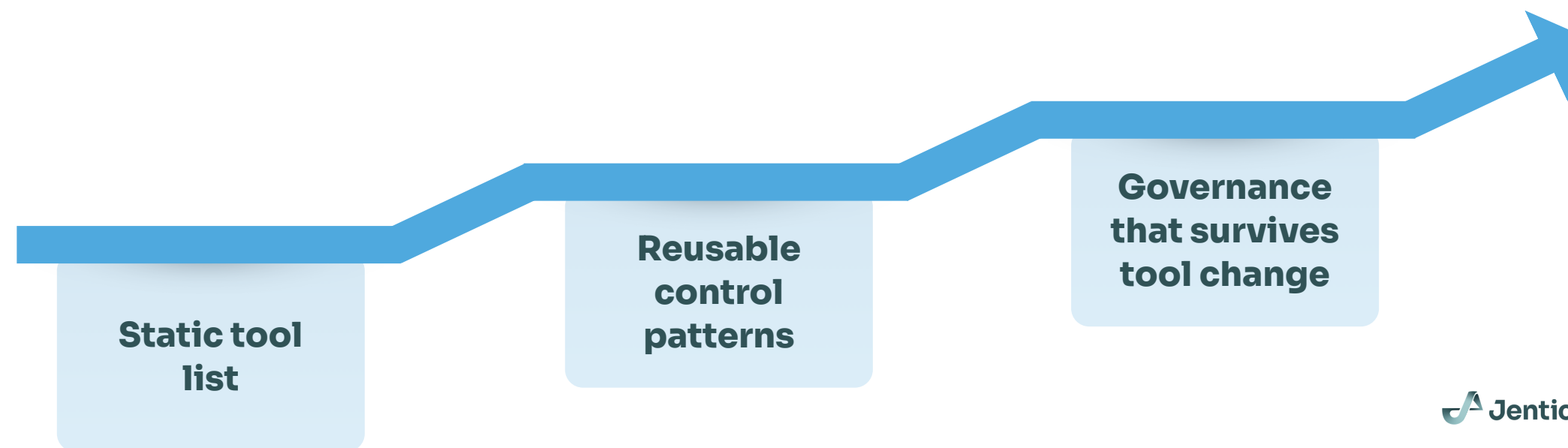
1. Govern usage patterns

Define controls for identity, access rights, data boundaries, output validation, logging, human review and escalation. **The safe path should be the easiest path**, even as tools change.

2. Invest in ongoing learning and evaluation

Teams need time and resources to understand how agents behave, how outputs should be reviewed, and where exposure risks emerge. An approved tool does not automatically produce an approved outcome.

Emerging practices such as [harness engineering](#) can make validation part of the agent workflow, not a separate afterthought.



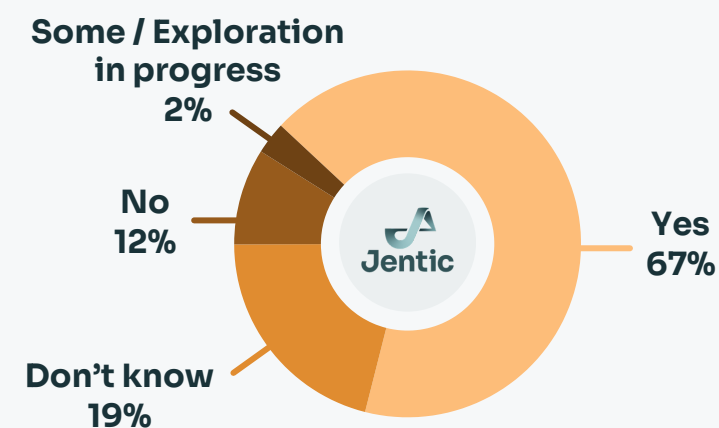
03: AI-first fails without API-first

Enterprises are connecting agents to APIs directly and through MCP servers. But many are doing this **before assessing whether those APIs are ready** for AI agent consumption.

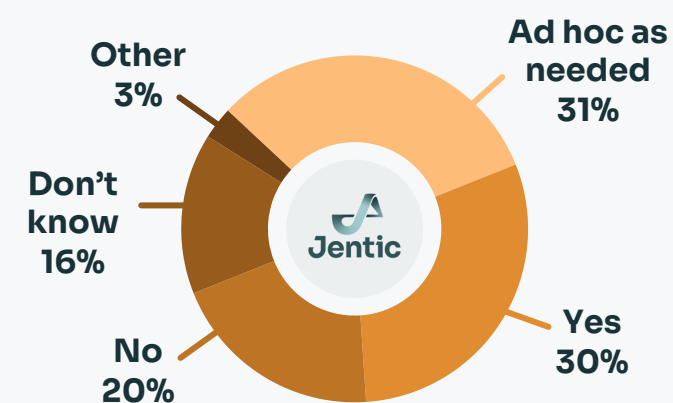
Agent reliability depends on API quality. Clear, well-documented and consistent APIs become more valuable because agents can understand and use them. Ambiguous, inconsistent or poorly governed APIs become failure points.

Our survey suggests the gap is already visible: **67%** of respondents say their APIs are used for business agents, but only **30%** say their APIs have been assessed for AI readiness. Another **31%** say assessment is done ad hoc as needed.

Are your APIs used for business agents?

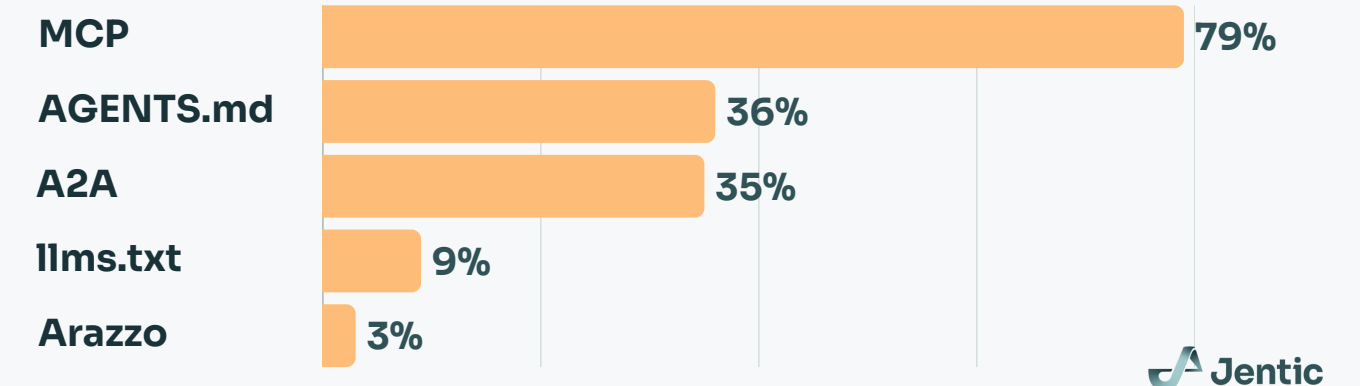


Have your APIs been assessed for AI readiness?



Context support approaches

Which emerging standards and approaches do you use with your agents? (multiple selections allowed)



DEFINITIONS

API-first = An architectural approach that defines interfaces, boundaries and API contracts before implementation, so digital capabilities are easier to reuse, govern and compose.

AI-first = A platform orientation in which APIs, data and workflows are prepared primarily for AI agent consumption.

MCP servers = A standardised plugin-like interface to connect AI agents to tools, resources and prompts.

API readiness is agent readiness

Agents need clear context, predictable behaviour, consistent definitions, usable authentication and useful error handling.

When APIs are not ready for agent consumption, three risks appear:

1. Agents become unreliable

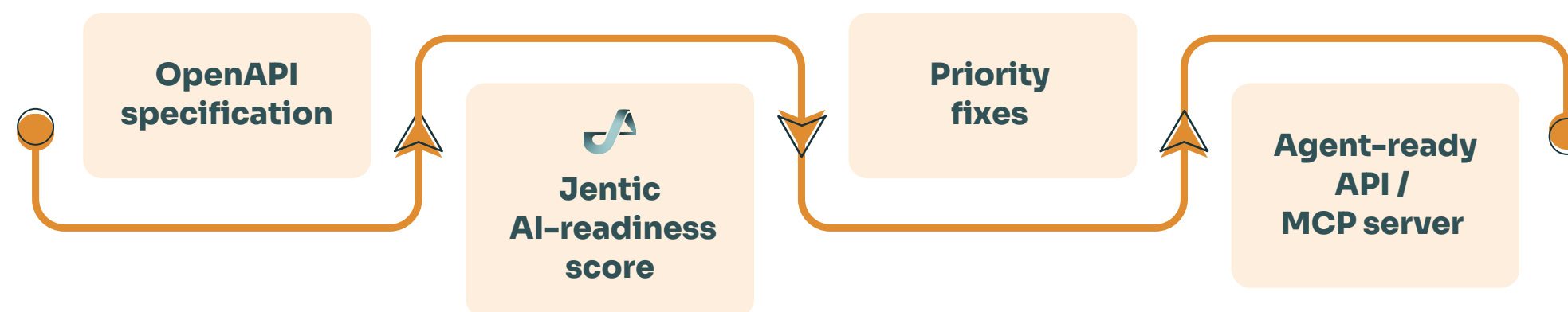
They fail to successfully plan tool calls, return incomplete outputs, or continue workflows with misunderstood context.

2. Teams bypass governed interfaces

Internal teams may connect agents directly to data, systems of record or informal workflows rather than using governed APIs.

3. External agent ecosystems move elsewhere

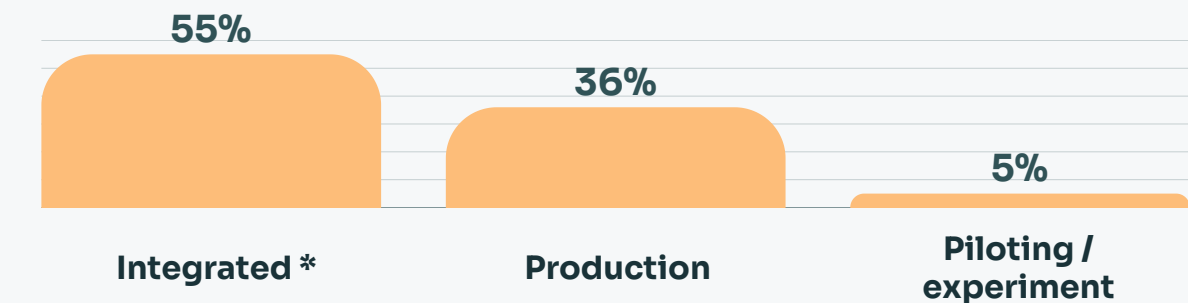
If external agents cannot consume your APIs reliably, customers and partners may look for easier-to-use alternatives.



AI integration stage correlates with API readiness



% of companies that have formally assessed API readiness, by AI integration stage:



Organisations in production are **7x more** likely to have formally assessed API readiness than those in pilots.

* Integrated = AI embedded in business systems with well-defined processes

Jentic opportunity:

Use AI-readiness scoring to identify which APIs are ready for agent consumption, which need documentation or contract improvements, and where MCP servers may help expose capabilities safely.

Check your AI-Readiness Score with Jentic's free scoring tool: <https://jentic.com/scorecard>

Platform response: Fix the foundations

1. Score and prioritise APIs for AI readiness

Identify the APIs most important to internal agents and external agents. Assess them with a repeatable AI-readiness methodology, then identify common fixes across the portfolio.

Free AI-Readiness scoring tool:
<https://jentic.com/scorecard>

2. Modernise API design and governance

Agents need the same foundations developers need, but with **less tolerance for ambiguity**: clear contracts, consistent patterns, accurate documentation, well-defined authentication, useful errors and governed change management.

3. Avoid MCP-washing

MCP servers can expose useful capabilities to agents, but they should not become a way to hide poor API design. Wallpapering over weak APIs with MCP increases technical debt instead of reducing it. API design problems should be **fixed at the API layer** before wrapping them with MCP.

Weak API + weak MCP server = duplicated technical debt

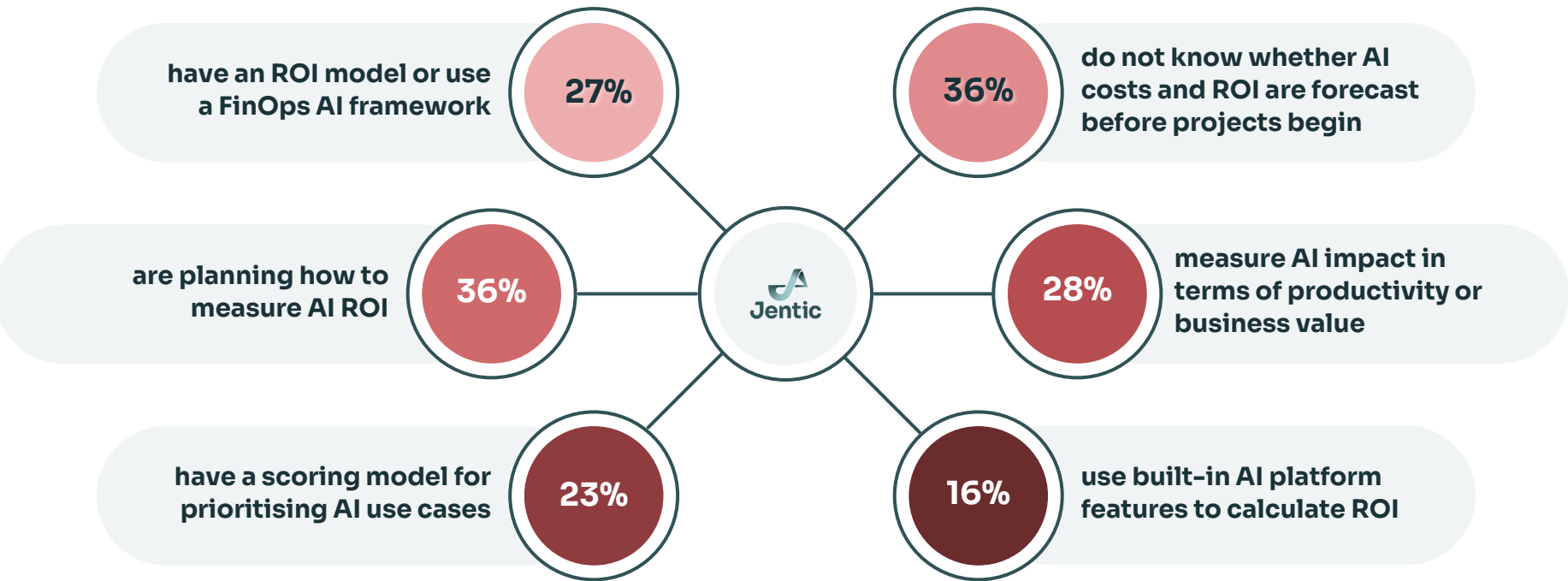
Agent-ready API + governed MCP server = reusable agent capability

04: Agent projects are moving faster than business prioritisation

Experimentation is necessary. Teams need space to learn what agents can do and how they interact with enterprise systems.

But the path to production AI should be linked to expected ROI. Without that link, the portfolio fills with point solutions that incur mounting **complexity and technical debt**.

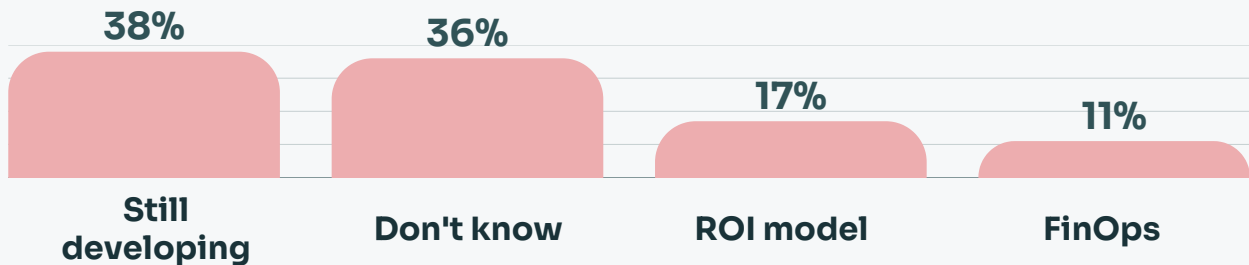
The current signs are mixed: only **28%** of respondents measure AI impact in terms of productivity or business value, **23%** have a scoring model for prioritising AI use cases, and **35%** do not know whether AI costs and ROI are forecast before projects begin.



The ROI black hole



74%* of organisations have **no working ROI model** for AI



* Combined framing: 38% "still developing" and 36% "don't know"

"We just execute for the task at hand, no strong prioritization beyond 'is it solving a client problem / is it moving the business forward'."

— Survey respondent

"Use cases are prioritized based on impact, feasibility, and suitability for automation, especially where multi-agent workflows can replace manual or fragmented processes."

— Survey respondent

Platform response:

Keep experimentation open; gate production on strategy

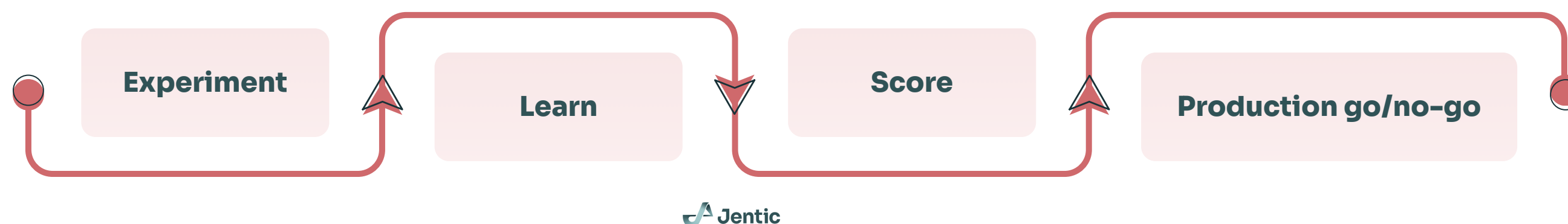
1. Protect low-risk experimentation

AI literacy improves when teams can explore use cases, test agent behaviour and learn how AI connects to systems. Not every experiment needs a direct business case, but every experiment should teach something about capability, risk or readiness.

2. Add a go/no-go assessment before production

Before an AI use case moves into production, require a simple assessment of strategic fit, expected value, feasibility, API and data readiness, operating cost, risk level, monitoring plan and success metrics.

**Experimentation
should be **broad**.
Production should
be **selective**.**

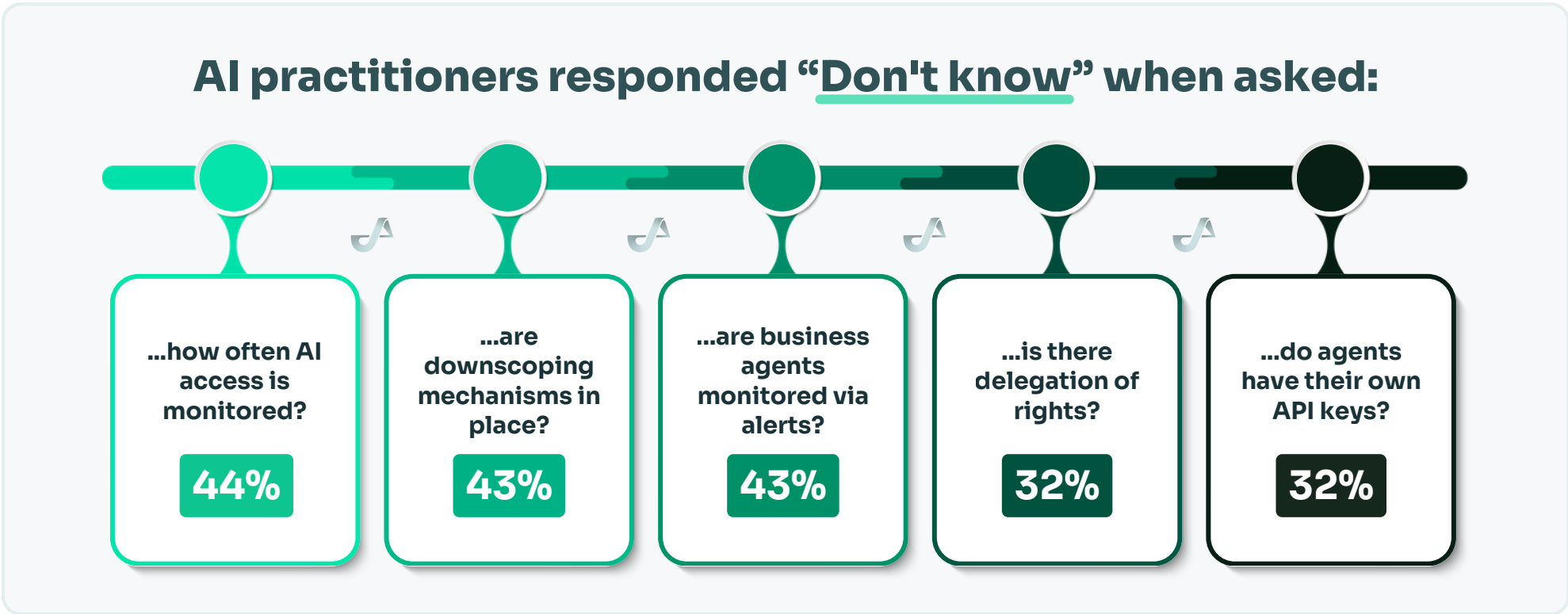


05: Security cannot stop at guardrails

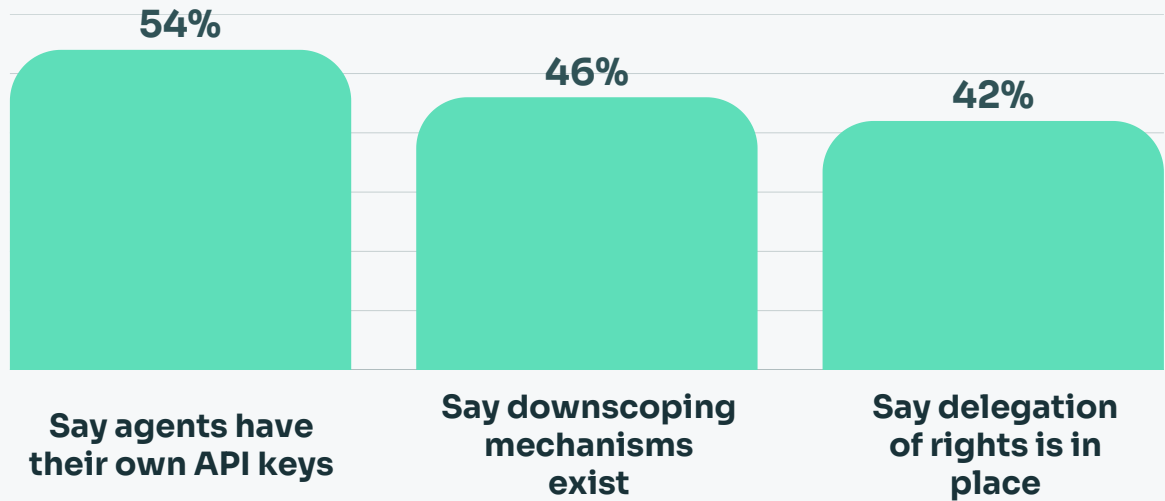
AI security has two sides. Enterprises need to know whether they can **trust agent outputs**, and whether agents can **use enterprise systems without creating unacceptable risk**.

Guardrails are useful, but they are not enough for production agents. Agents need identities, delegated rights, downscoped access, logs, alerts and incident investigation. Without those controls, security becomes symbolic: **visible in policy, but weak in operation**.

The current picture is uneven. Just over half of respondents say business agents have their own API keys, but **44%** are unsure how often AI access to systems is monitored, and **43%** are unsure whether alerting exists for unusual business-agent activity.



Controls reportedly in place



But confidence is thin; for every control reported, a near-equal share has no visibility into whether it works.

"Monitoring is currently manual and review-based during execution, with potential future plans to introduce automated alerting and anomaly detection."

— Survey respondent



Platform response:

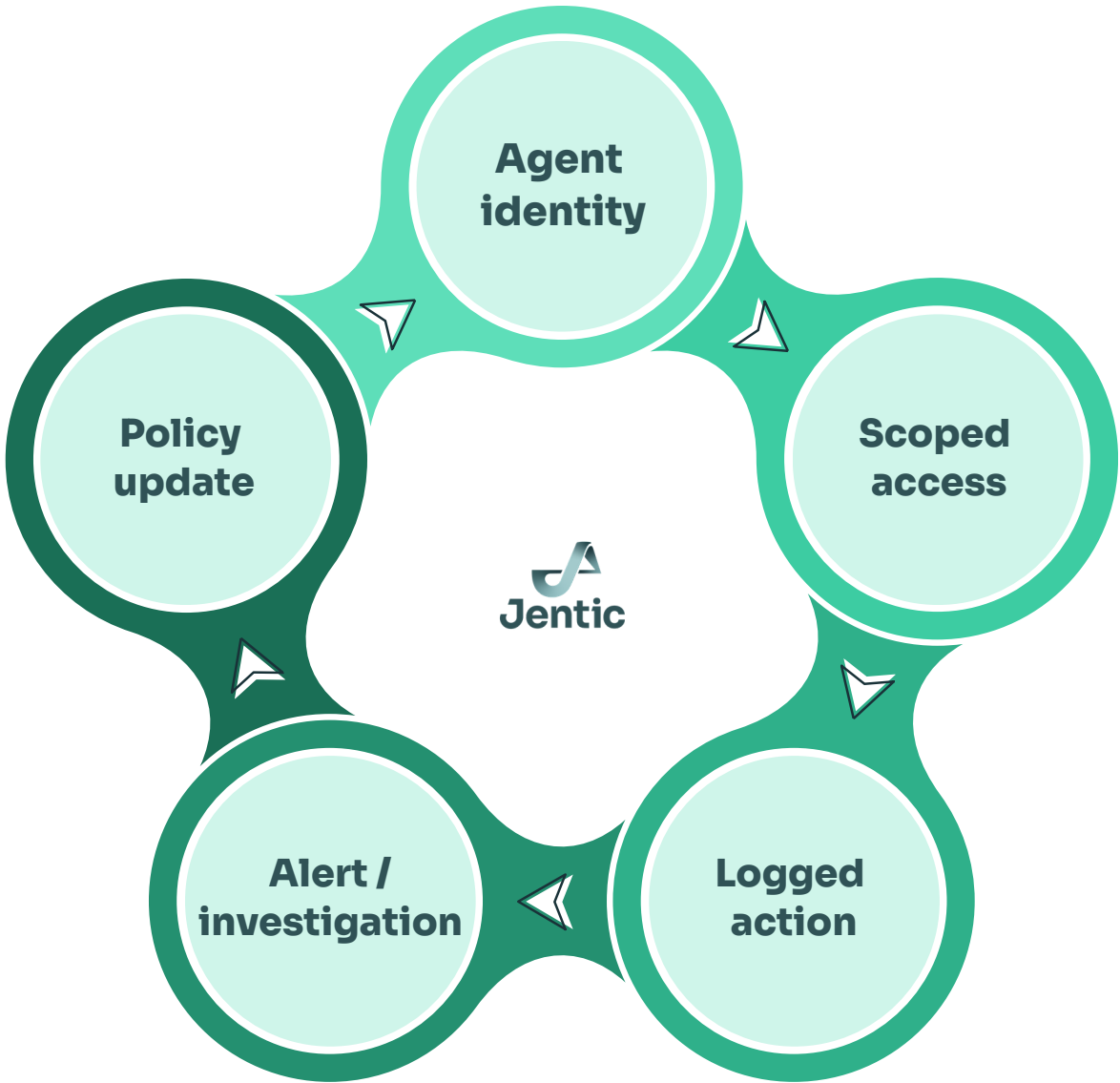
Build identity, monitoring and response infrastructure

1. Provide a security intermediary layer

Use an intermediary layer to **issue agent identities, API keys and scopes**. This layer should enforce policy, constrain access, log actions and support investigation when agent behaviour is unexpected.

2. Use agents to improve alerting and investigation

Business agents can help define thresholds, detect anomalies and support root-cause analysis using operational data. High-risk actions still need clear escalation paths and human oversight.



06: Simulation sandboxes are underused platform assets

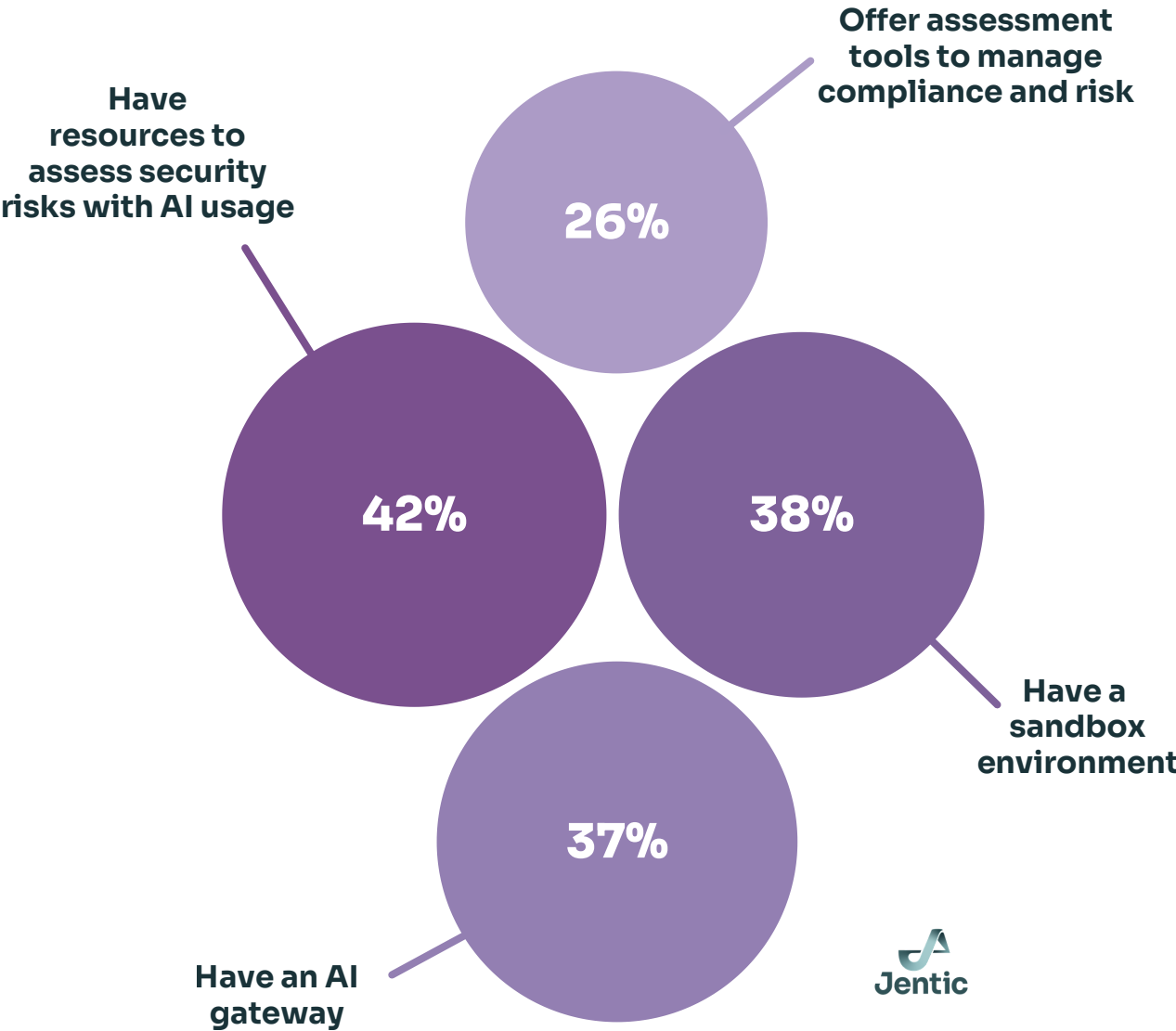
Many teams treat simulation sandboxes as final pre-production test environments. In an AI-first enterprise, the sandbox has a **larger role**.

A sandbox can emulate representative enterprise APIs and data, allowing agents to execute workflows, fail, recover and improve without interacting with live systems. This accelerates development in an isolated, non-production environment.

In our survey, **38%** of respondents said they have a sandbox environment, but definitions vary widely. Some treat sandboxes as runtime containers, some as regulatory testing environments, and some as API or data-workflow testing environments.

DEFINITIONS

Simulation sandbox infrastructure = An isolated environment that emulates representative enterprise APIs and data so agents can execute and validate workflows before reaching production.



Platform response:

Use sandboxes across the full agent lifecycle

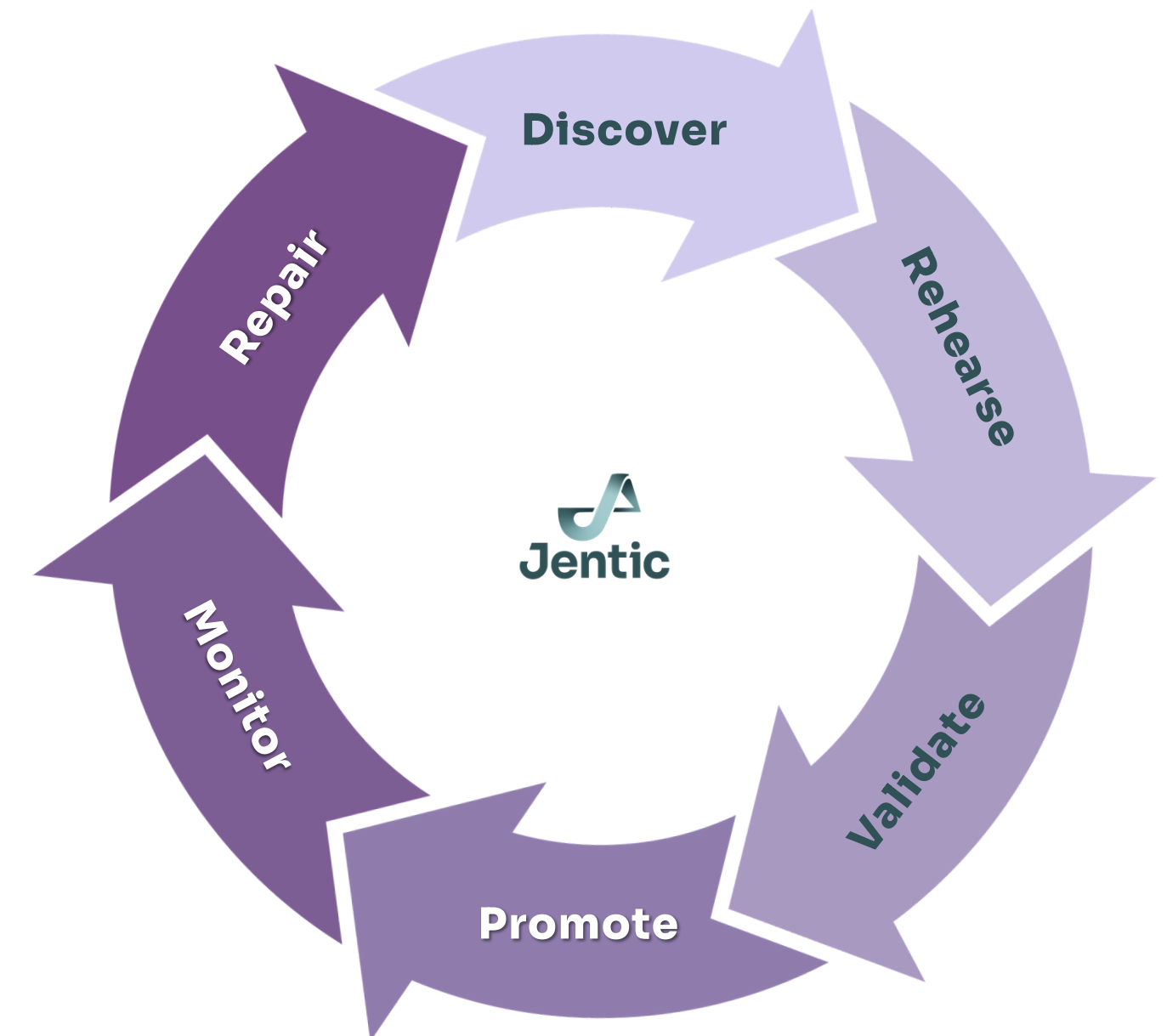
1. Use internal sandboxes to move faster safely

Implement sandboxes where internal teams and agents can rehearse behaviour, run acceptance tests, perform compliance and security checks, and rebuild workflows when APIs, business requirements or regulations change.

2. Offer external sandboxes to customers and ecosystem partners

Customers are also learning how to connect their agents to your APIs and MCP servers. A sandbox gives them a safe place to experiment while protecting production systems and giving you visibility into how external agents consume your platform.

The sandbox is not the last stop before production.
It is where agent readiness is created.



Platform maturity is the path to agent scale

Agent adoption is moving quickly, but it is moving on two tracks. Coding agents are spreading across teams and departments. Business agents are being connected to APIs, data and workflows. When those tracks are implemented separately, the enterprise accumulates **agentic technical debt**.

The six disconnects in this report are symptoms of the same platform maturity gap: coding agents outside the platform model; rigid tool-list governance; AI initiatives built on weak APIs; projects not tied to strategy; security controls without operational monitoring; and sandboxes used too late.

You do not need to rebuild your API and AI stack from scratch. You need to align it.

The practical path is to make platform capabilities reusable across all agents: governance, API readiness, identity and access, observability, strategic portfolio management and sandbox-based validation. Enterprises that do this will move from fragmented agent experimentation toward AI-first operating models with less risk and greater trust.

Platform maturity checklist for enterprise agents



1. AI-ready APIs

Can you identify which APIs are ready for agent consumption, which need improvement, and which should not yet be exposed to agents?



2. Agent workflow discovery

Can agents discover the APIs, tools, skills, workflows and other agents without relying on brittle documentation, guesswork or one-off integrations?



3. Governed MCP exposure

Are MCP servers exposing well-designed capabilities, or are they wrapping deficient APIs and creating a second layer of technical debt?



4. Scoped agent access

Can agents have their own identities, credentials and permissions appropriate to the systems they need to access?



5. Observable execution

Do you have a single system where you can see which agents accessed which systems, what calls they made, what data they used and where failures occurred?



6. Sandbox validation

Can agents rehearse workflows safely before production, with successful traces converted into validated, reusable workflows?



7. Ecosystem readiness

Can customers, partners and external agents test against your APIs and MCP servers without putting production systems at risk?

- **Start with the APIs and workflows agents already need.**
- **Score the APIs using a scoring tool at <https://jentic.com/scorecard>.**
- **Expose the right capabilities.**
- **Validate workflows in a sandbox.**
- **Promote only what is ready.**

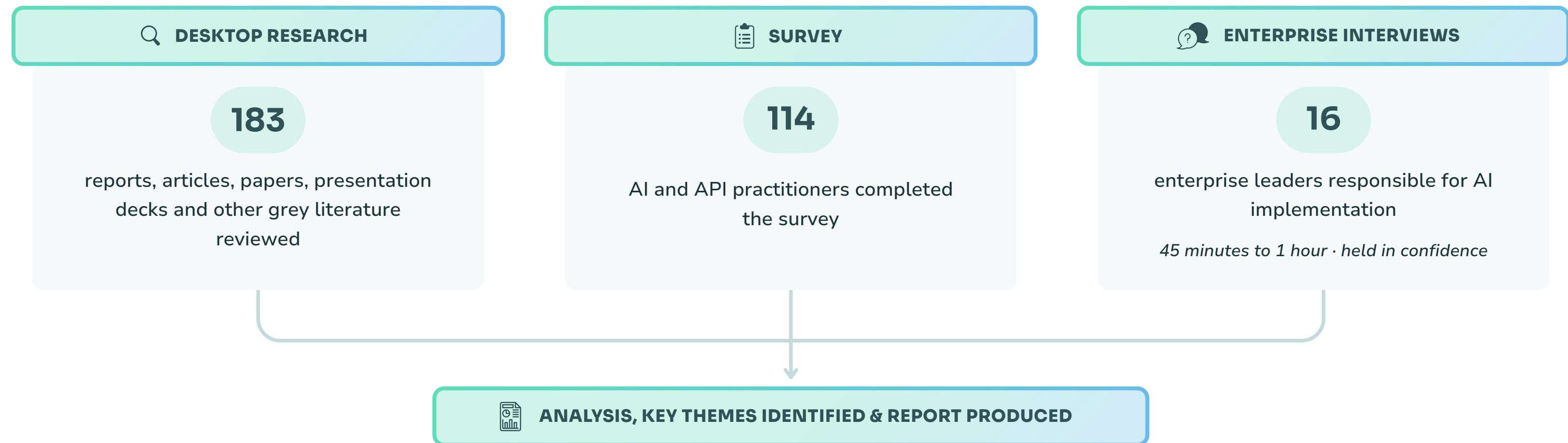
Wherever you are on this checklist, we can help you take the next step.
Talk to our team at jentic.com/contact.



Research Methodology

MARCH - MAY 2026

The research was designed to identify common disconnects in enterprise adoption of coding agents and business agents, and to identify practical tactics being used by platform, API, architecture and AI leaders.



Findings should be read as directional practitioner insight rather than a statistically representative enterprise benchmark. The survey and interview samples were skewed toward technology, banking/finance and senior architecture/platform roles.

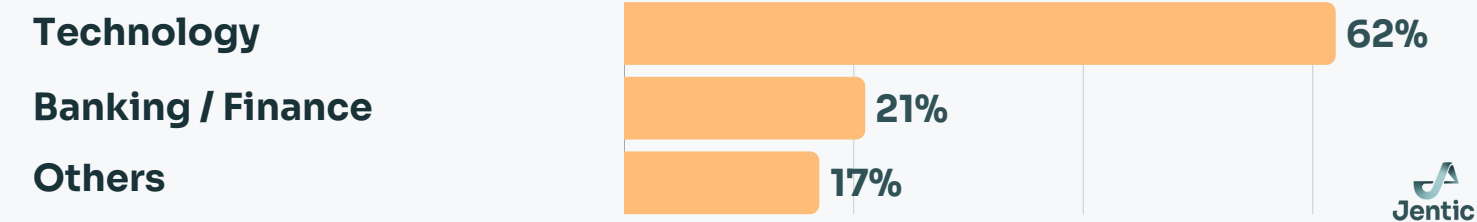
252 practitioners began the survey and 114 completed it; as not every question was answered by every respondent, percentages throughout this report are based on the number of respondents who answered each question.

Respondent profile

From our Survey:

114 survey responses (March-May 2026)

INDUSTRY



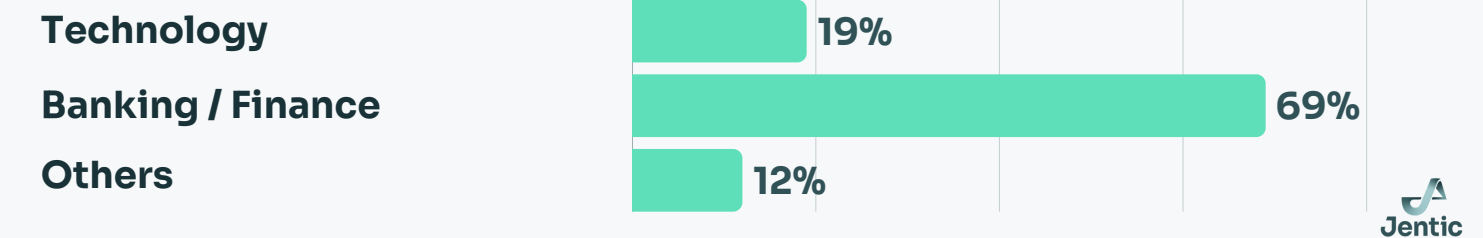
ROLE



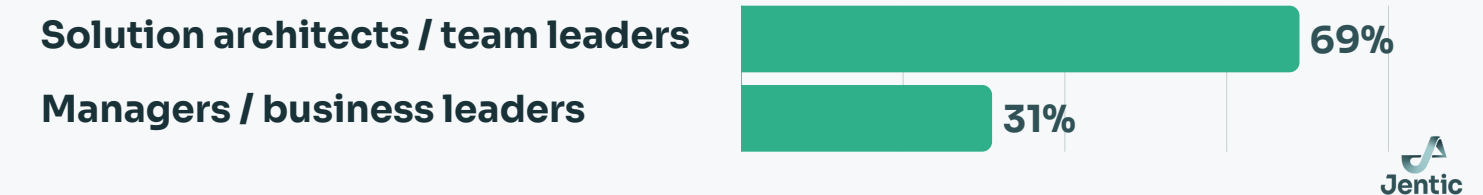
From our Interviews:

16 interviews (March-May 2026)

INDUSTRY



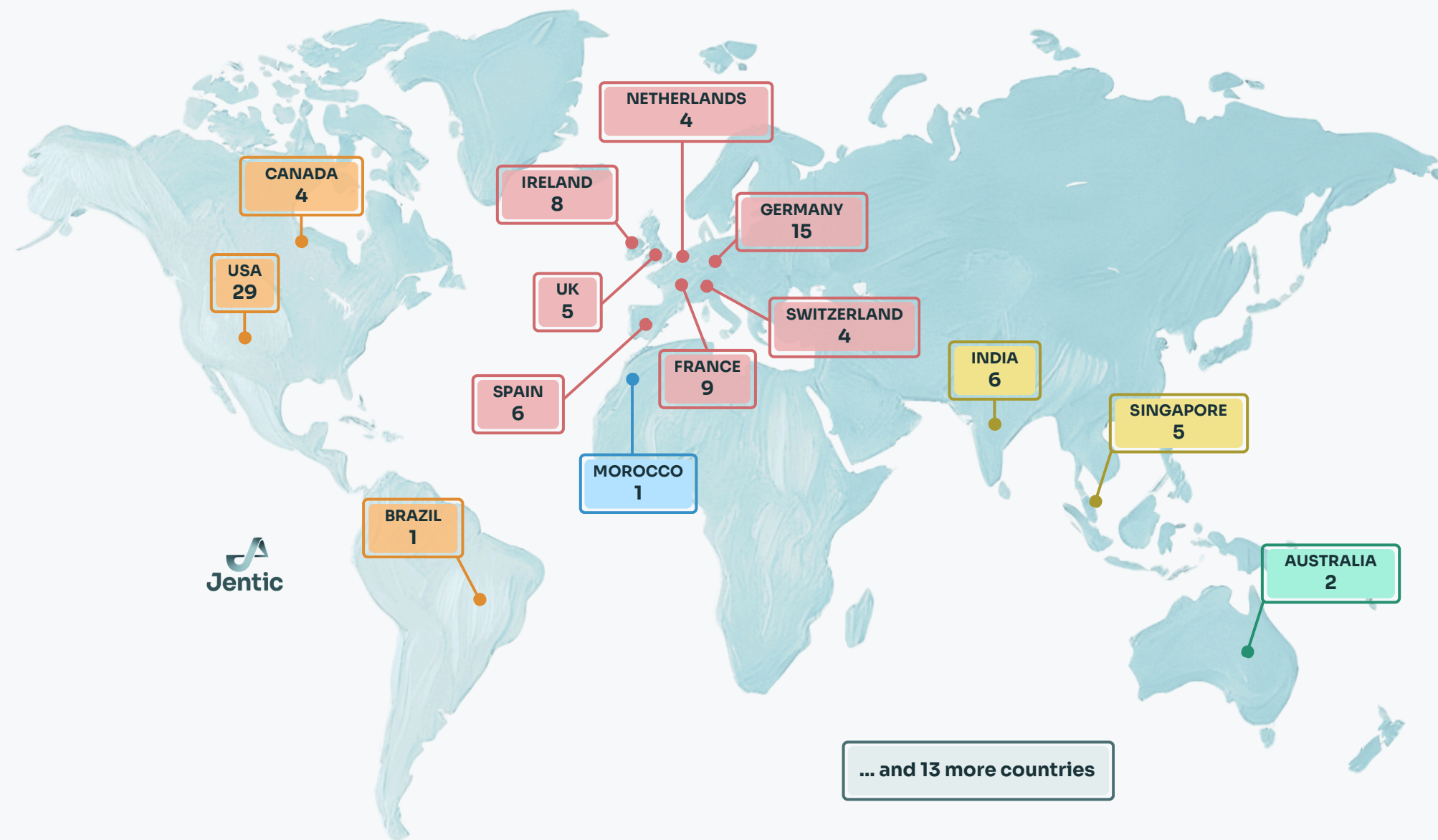
ROLE



Respondent profile

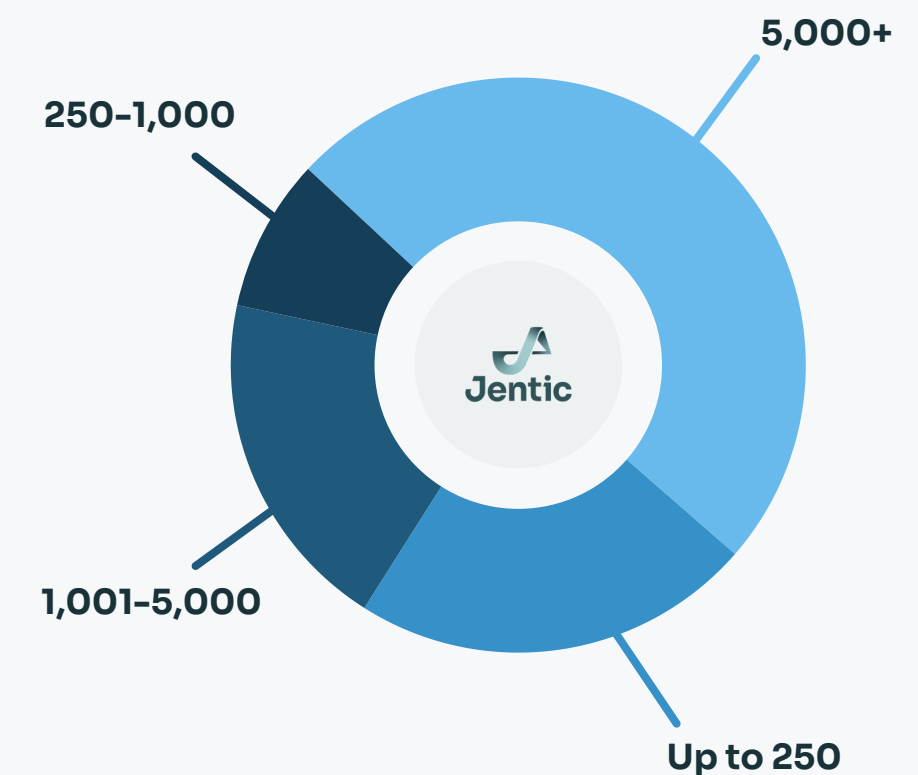
Geographic distribution

27 countries represented across 5 continents:



Company size

The sample skews toward organisations where platform decisions are already active. Self-reported by 113 survey respondents (March–May 2026).



Acknowledgements

We would like to thank the following people for their insights, contributions and time in developing this report:

Koen Adolfs, Aleksey Akimov, Mike Amundsen, Tony Blank, Abhijit Dey, Bill Doerrfeld, Zabeer Farook, Alexei Galban, Quincy Hseh, Andrzej Jarzyna, Joe Joyce, Frank Kilcommins, Kin Lane, John Musser, Ani Radouncheva, Jennifer Riggins, Sophie Rutard, Bartosz Sokolinski, Joyce Stack, Ado Trakic and Kristen Womack.

Written by: Erik Wilde, Sean Blanchfield and Mark Boyd

Design by: Elena Anagnostatou

Sources

We reviewed 183 reports, articles, papers, presentation decks and other grey literature. The following sources were among the most influential for this report.

- ADP, Inc. "ADP Marketplace Launches AI Agents to Help Make Work Easier, Smarter." ADP Newsroom, March 2, 2026.
- Agrawal, Shreya. "How to design and run an agent in rehearsal – before building it." DataRobot Blog, April 2, 2026.
- Bartoletti, Ivana, and Keri Dawson. "How to design for trust in the age of AI agents." World Economic Forum, February 18, 2026.
- Beveridge-Raffo, Anita. "Stop Building Super Agents; Build Effective AI Teams Instead." The New Stack, September 11, 2025.
- Böckeler, Birgitta. "Harness engineering for coding agent users." Martin Fowler, April 2, 2026.
- Bridgpal, Anil, and Seán McTernan. "The GTM Agent We Wished Existed, So We Built It on Netlify." Netlify Blog, April 3, 2026.
- Buhr, Martin. "Arazzo and AI agents – your essential starter guide." Tyk API Gateway, May 30, 2025.
- Buntz, Brian. "Salesforce brings agents to enterprise scale in life sciences." R&D World, August 12, 2025.
- Claassen, Ailis. "My API Story: TM Forum's Open APIs empower Innovation Incubator to fast-track end-to-end telecom product development." TM Forum, 2026.
- Corrigan, Louise. "APIs and Agents: What Developers Need to Know." O'Reilly, July 11, 2025.
- Defranchi, Lydia. "OpenClaw, Moltbook, and the new AI Wild West: what enterprises must learn about AI governance." Axway Blog, March 2, 2026.
- Doerrfeld, Bill. "How To Prepare Your API for AI Agents." The New Stack, June 25, 2025.
- Doerrfeld, Bill. "Why CIOs see APIs as vital for agentic AI success." CIO, July 9, 2025.
- DZone Contributor. "Designing a Real-Time Data Activation Platform Using Segment CDP, Databricks, and Iterable." DZone, 2026.
- Einy, Zohar. "The hidden technical debt of agentic engineering." The New Stack, April 2, 2026.
- Eliyahu, Roey. "An AI Agent Didn't Hack McKinsey. Its Exposed APIs Did." Security Boulevard, March 13, 2026.

Sources

- Fankhauser, Karl. "How Agentic AI and APIs Are Powering the Next Wave of Enterprise Innovation." digitalML, March 10, 2025.
- Freyfogle, Ed, and OpenCageData. "opencage-skills: Agent skills for using OpenCage services." GitHub, March 2026.
- Furze, Leon. "A Taxonomy of Agentic AI." Leon Furze Blog, March 30, 2026.
- Galileo Technologies, Inc. The State Of Eval Engineering - Q1 2026. Research Report. Galileo Technologies, Inc., 2026.
- Gerred. "The Agentic Systems Series." The Agentic Systems Series, 2026.
- GitHub. "Usage-based billing for organizations and enterprises." GitHub Docs, 2026.
- Heritage, Laura. "Why your API infrastructure isn't ready for AI (and what to do about it)." Tyk API Gateway, April 2, 2026.
- Hetamsara, Prachi. "Your agentic AI pilot worked. Here's why production will be harder." DataRobot Blog, April 3, 2026.
- Higginbotham, James. "What is ADDR? - ADDR Process Methodology." LaunchAny, 2026.
- Humble, Charles. "Before you let AI agents loose, you'd better know what they're capable of." The New Stack, March 12, 2026.
- Janakiram MSV. "Anthropic, OpenAI, Google, and Microsoft agree that the harness is the product. They disagree on the price." The New Stack, April 18, 2026.
- Joyce, Joe, and Mark Boyd. "APIs in the Agentic Era - Designing, Testing, and Governing AI-Ready APIs." Presentation, Apidays, 2025.
- Kilcommins, Frank. "From Endpoints to Intent: Rethinking Agent API Workflows with Arazzo." SmartBear Blog, August 14, 2025.
- Lim, Julian. "Cursor's Claude agent wipes production database and backups in 9 seconds." Startup Fortune, April 27, 2026.
- Margaryan, Anoush. "AI is not an agent – AI is a tool." Oxford Internet Institute, August 6, 2025.
- Miles, Ruston. "What Agentic Commerce Actually Means 2026." YouTube video. Wrap Up, 2026.
- Mota, Adriano. "Gateway Federation: How to Manage Multiple API Gateways at Scale." Nordic APIs, 2026.
- Nashawaty, Paul, and Scott Hebner. "The Agentic AI Revolution in AppDev and DevOps." theCUBE Research, January 24, 2025.
- NTT DATA Group. "NTT DATA Partners with Google Cloud to Accelerate Agentic AI Adoption and Cloud Modernization for Enterprises Globally." NTT DATA News, August 13, 2025.

Sources

- Pandey, Rajat. "How to Govern Agentic AI Before It Governs You." HackerNoon, November 14, 2025.
- Plumb, Taryn. "Adobe bets on agentic AI to rewrite SaaS for customer experience." CIO, April 20, 2026.
- PYMNTS. "BNPL Providers Aim to Stay in the Checkout as AI Agents Take Over Shopping." PYMNTS, March 4, 2026.
- Rasmus, Daniel W. "AI Agent Taxonomy: Struggling with AI Agent Classification? Explore The Serious Insight's Focused Take on Agents." Serious Insights, April 7, 2025.
- Riggins, Jennifer. "HPE's AI agents cut root cause analysis time in half." The New Stack, March 25, 2026.
- Rodrigues, Patricia. "Data management: the unsung hero of the agentic AI revolution." Sync NI, November 21, 2025.
- Sapkota, Ranjan, Konstantinos I. Roumeliotis, and Manoj Karkee. "AI Agents vs. Agentic AI: A Conceptual taxonomy, applications and challenges." Information Fusion 126 (2026): 103599. <https://doi.org/10.1016/j.inffus.2025.103599>.
- Subramanian, Krish. "Platform Engineering Just Got Its Brain." Rishidot Research, September 2024.
- Sukharevsky, Alexander, Dave Kerr, Klemens Hjartar, Lari Hämäläinen, Stéphane Bout, Vito Di Leo, and Guillaume Dagorret. "Seizing the agentic AI advantage." McKinsey & Company, June 13, 2025.
- SYUTHD Team. "Mastering Agentic API Design: How to Build Endpoints for Autonomous AI Tool-Calling." SYUTHD, March 4, 2026.
- Taskade Team. "AI Agents vs Copilots vs Chatbots: The Complete 2026 Taxonomy." Taskade Blog, April 6, 2026.
- The Paypers. "Google Cloud, Wells Fargo collaborate on AI agents." The Paypers, 2026.
- Trindel, Kelly. "Workday's Bold Future With Responsible Agentic AI." Workday, March 5, 2026.
- Vaitsman, Mark, and Eric Saraga. "A Look Inside Claude's Leaked AI Coding Agent." Varonis Threat Labs, April 3, 2026.
- Verdent AI. "Verdent AI." Verdent AI, 2026.
- Whiteley, Rob. "Vibe Coding in a Post-IDE World: Why Agentic AI Is the Real Disruption." The New Stack, June 17, 2025.
- Yee, Lareina, and Rodrigo Liang. "Preparing for tomorrow's agentic workforce." McKinsey & Company, June 12, 2025.



*The enterprises that govern AI well today
will define what enterprise AI looks like tomorrow.*

Explore more at jentic.com