

■ The State of Knowledge Report 2026

Table of Contents

01

Introduction

03

02

Part 1: Agents: The Readers You Didn't Write For

04

Who They Are04How They're Different05

03

Part 2: The Price of Persistence

06

All Roads Lead to Readership06Who Pays?07Markdown vs. Maps06

04

Part 3: Inside the New Docs Ecosystem

09

Skepticism Doesn't Scale09Reader to Writer10

05

Part 4: From Documentation to Knowledge Infrastructure

12

Spread Across Surfaces12Knowledge Infrastructure in Practice14Why It Often Stops at Docs12

06

Conclusion

15

METHODOLOGY

This report combines server-side traffic data from Mintlify-hosted documentation sites, Mintlify product telemetry, a controlled open-source benchmark, external sources, and survey findings. Traffic analysis covers February 1 through August 31, 2026.

Human traffic is captured through client-side analytics—page loads and events recorded in the browser. Agent traffic is measured through server-side requests for machine-readable versions of pages, agent-friendly assets like llms.txt, and HTML page fetches by recognized AI user-agents. Because these two collection methods differ, direct comparisons are estimates.

ABOUT MINTLIFY

Mintlify is the AI-native knowledge infrastructure company that makes knowledge accessible to humans and agents.

As AI agents become the primary consumers of documentation, Mintlify is building the layer that makes every product understandable, usable, and discoverable by AI.

Mintlify powers documentation for over 20,000 companies, including Anthropic, Microsoft, Coinbase, Spotify, and HubSpot. Knowledge on Mintlify reaches over 50M people a month.

Visit mintlify.com to learn more.

INTRODUCTION

Your number-one knowledge consumer is no longer human.

2026 marks the first year on record that AI agent readership of Mintlify-powered sites crossed the majority line. August alone saw 257 million agent web requests and 131 million human page loads. This means agents now account for over half of total traffic. And the numbers keep compounding.

There are still people behind every one of these requests. They've just stopped doing the reading themselves. Agents skip your images, your design, and your catchy headline. They want structure, yet even the most forward-thinking companies have made only one knowledge surface ready: docs. And it's not as ready as many think.

AGENT VS HUMAN TRAFFIC IN AUGUST 2026

257,000,000

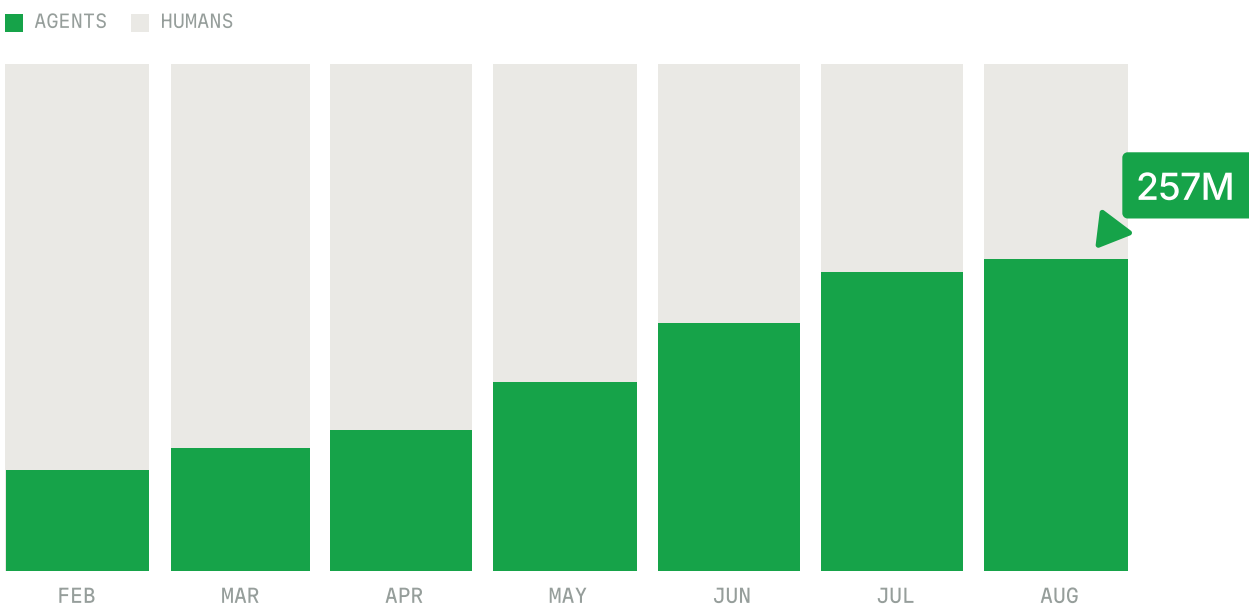
Agent requests

131,000,000

Human page loads

66%

Agent share



Docs vs. Knowledge



Documentation

is what you publish for people building with your product: guides, tutorials, API reference, etc.



Knowledge

is all of that plus everything else that explains your product, your company, and how it all works: the help center, the community forum, the changelog, support tickets, sales enablement, premium customer portals behind auth, and more.

As agents begin moving from reading documentation to writing it, knowledge stops being a static destination one team publishes and becomes living infrastructure the entire business runs on.

The data insights from this year's report show how quickly this transition is happening, what it costs companies that aren't ready, and what it takes to go beyond docs and turn company-wide knowledge into a system that keeps up.

PART 1

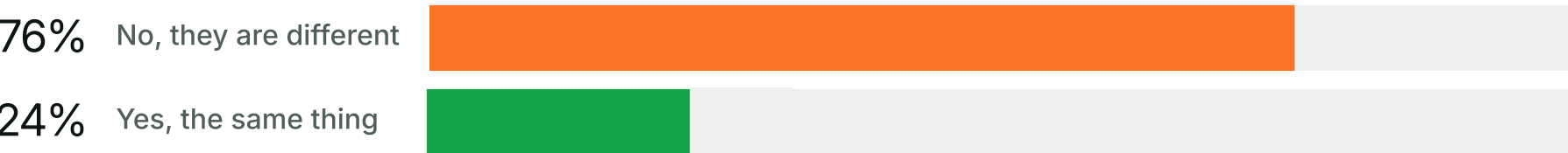
Agents: The Readers You Didn't Write For

Who They Are

The instinct to write agent traffic off as “bot traffic” is strong, but the difference is measurable. “Agents are a new kind of user,” explains Chris Riley, Sr. Manager, Developer Relations at HubSpot. “What’s good for the humans is good for the agents, but you can’t treat them the same way.

Treating non-human traffic as something to filter out means you’re now filtering out your customers.” Our survey results show this opinion is now the consensus among most companies, yet a surprising 24% of respondents still don’t see the distinction.

Do you consider "bot traffic" to be the same as "agent traffic"?



BASE: ALL 329 RESPONDENTS

FIGURE 1

While bots crawl indiscriminately, agents work with purpose. Most agents eschew HTML rendered pages. In August, 83% of agent traffic arrived through machine-friendly routes: the .md version of a page, llms.txt files, or agent skills. Agents only request these resources when they know they exist.

And Model Context Protocol (MCP) tool calls exceeded 2 million the same month, roughly 3x growth since February. MCP requires a client that speaks the protocol and calls tools by name — a crawler can’t produce a single one.

A Third Channel

Web traffic isn't the full picture. Docs assistants operate as a separate channel that’s tracked outside human and agent requests entirely. This channel returned nearly 1.3 million responses in August alone.

83%

Of agent traffic arrived through machine-friendly routes in August 2026

2M+

MCP tool calls in August 2026

3×

MCP growth since February 2026

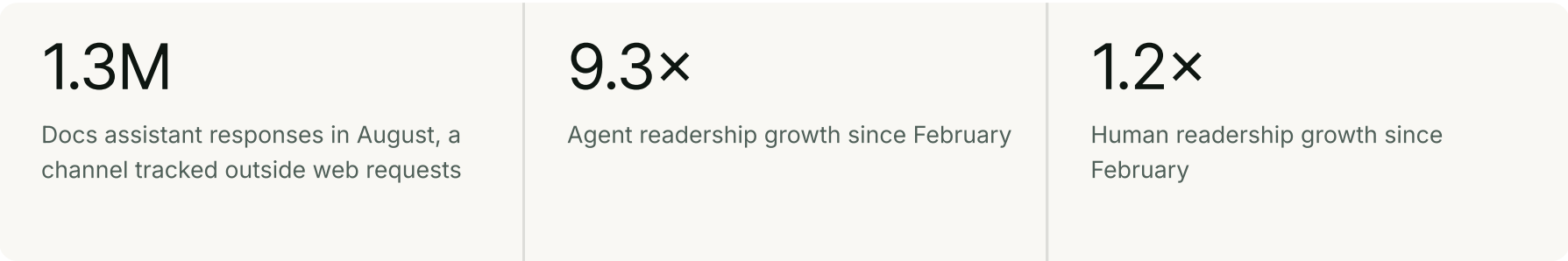
This is real readership, and it’s grown 9.3x since February this year, compared to just 1.2x for human readership. While part of this can be attributed to more agents, the rest reflects the differences between how agents and humans read.

Agent and human readership of Mintlify-powered sites



AUGUST 2026

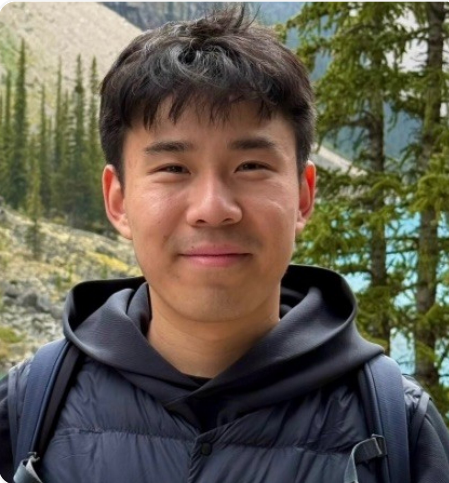
FIGURE 2



How They’re Different

A developer with a question used to open the documentation and read, generating a single visit. That same developer now asks an agent, and the agent pulls the map, checks three pages, follows a reference, and verifies a parameter before answering. The question and person are identical, but the traffic is an order of magnitude higher.

Agents don’t browse—what they come for is structure. Case in point: /llms.txt was fetched roughly 10.8 million times in August: a file humans rarely type into browsers. MCP calls split 53% search / 47% filesystem retrieval, depending on whether the agent already knows the page path. Either way, it orients before it acts, which makes your information architecture the interface, not your prose.



“Agents are the new audience for your knowledge, and that comes with real product requirements: for how it's shaped, what it contains, and how retrievable it is, so the agent can perform meaningful tasks end-to-end for users.”

David Hou
Product Education Lead, Decagon



PART 2

The Price of Persistence

All Roads Lead to Readership

The question is no longer if agents will try to read your docs, it’s what happens when they do. And the data shows they keep trying.

We noticed agents repeatedly requesting pages that didn’t exist: 404ing, backing up, guessing a new path, trying again. So we ran 2,400 controlled tests across 20 documentation sites, asking each the same set of questions, served four ways, with every request an agent made observed and counted.

The agents rarely failed to answer. Accuracy remained between 94-99%, regardless of format. Reading was never the problem—navigation was.

Markdown vs. Maps

The cause was structural, and counterintuitive. Serving pages as stripped-down markdown, the standard move for making content machine-readable, removes the navigation along with the HTML, leaving agents to infer URLs that aren’t real.

Documentation served as HTML produced 2.23 failed requests per task. Clean markdown brought that down to 1.42, and adding a single link to a map dropped it to 0.11. That’s a 20x improvement over HTML, with markdown accounting for roughly a third of the gain and the map delivering the rest. The pattern held across every model tested, on 19 or 20 of the 20 sites in each case.

In our testing across five different models, we saw a 26% to 60% savings in tokens consumed when the agents fetched pages as Markdown + llms.txt compared to HTML alone. This showcases that optimizing your documentation for agents helps reduce cost for your customers in addition to speeding up the path to better and more accurate results.

2,400

Controlled tests run

20

Documentation sites tested, four ways each

94–99%

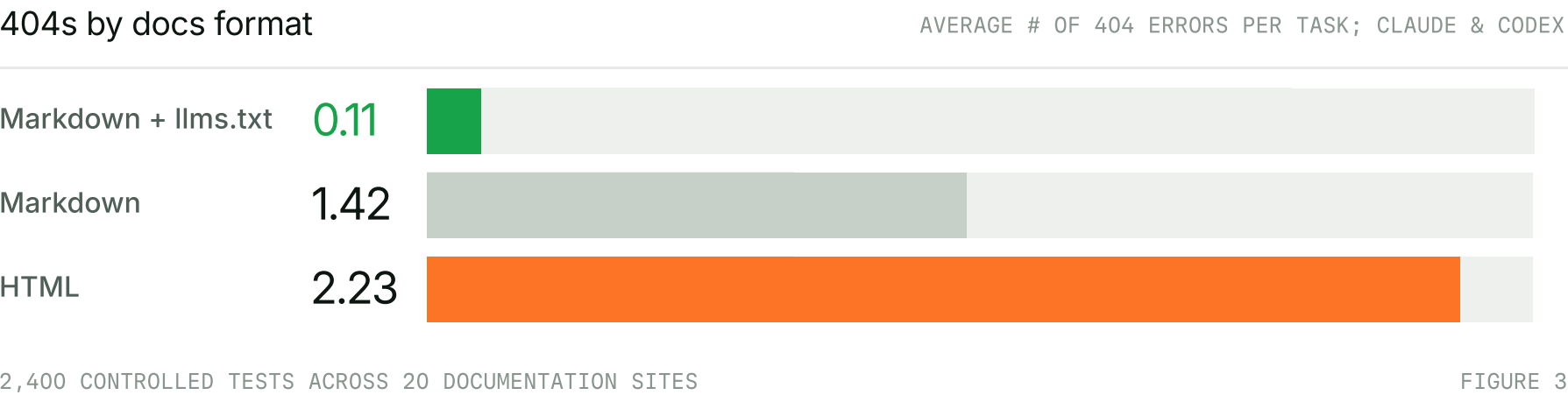
Answer accuracy, regardless of format

20x

Fewer failed requests with Markdown + a map, compared to HTML

26–60%

Token savings across five models fetching Markdown + llms.txt



 Test your own docs

We made our benchmark tool open source. Point it at your own docs to find out how many failed requests agents are hitting trying to navigate your content.

See your results →

Who Pays?

The cost of failed navigation lands on your customers first. Their agents go in circles inside your docs, burning their tokens and time. But it doesn't stay there. Every wasted call and information gap shapes your user experience and the impression your product makes.

"For coding agents, docs are an essential part of choosing which software to implement," says Ethan Finkel, Founding Product Manager at Gauge, a platform that runs coding agents in real repositories to measure which tools they choose. "They may start with a list of tools that they would commonly use, but selection and implementation success are highly coupled to the quality of your documentation."



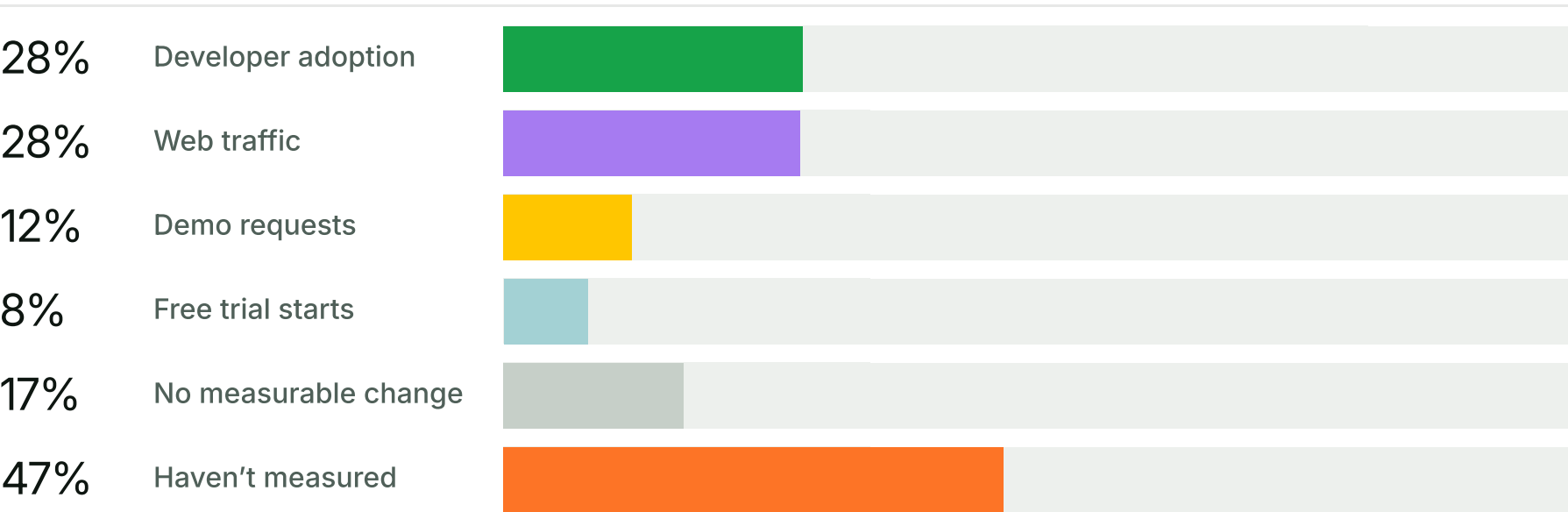
"In prompts focused on choosing a tool with a hard limit on building, Gauge found that agents were more likely to read documentation than they were to read marketing materials."

Ethan Finkel
Founding Product Manager, Gauge



We asked Mintlify customers whether improving documentation discoverability and structure has positively influenced their growth metrics. While many note increases in signups, website traffic, demo requests, and free trial starts, almost half have yet to measure the impact. This indicates a major opportunity to improve understanding of the relationship between content discoverability and business outcomes.

Since improving your documentation’s discoverability and structure with Mintlify, have you seen measurable changes in any of the following?



216 MINTLIFY CUSTOMERS

FIGURE 4

Unfortunately, nothing on your dashboard shows an agent wasting twenty fetches to answer a question that should have taken one. And what happens if what they eventually find isn't accurate?

PART 3

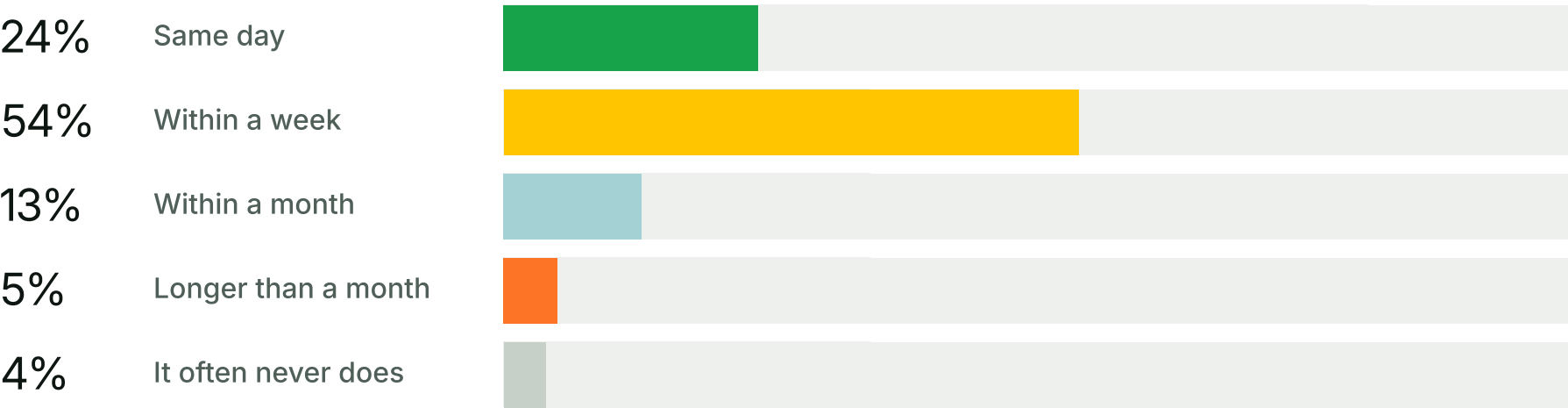
Inside the New Docs Ecosystem

Skepticism Doesn't Scale

Ask any technical writer or developer: keeping documentation current has always been the hardest part of the job. “GitHub recently published [an article](#) that says the number of pull requests merged has increased by 3.6x since 2023,” shares Sarah Deaton, Anthropic’s Technical Content Engineer for Claude Code. “Nobody is manually documenting 3.6x more. Keeping up with this volume requires a new approach.”

Just 24% of survey respondents say product changes are reflected in docs the same day. About 54% cite a one-week lag, while 22% have to wait a month or longer, clearly illustrating the challenge most teams face keeping up with the constant rate of new features.

How long does a product change typically take to reach your documentation?



ALL 329 RESPONDENTS

FIGURE 5

Human skepticism has quietly absorbed decades of documentation debt. A developer who hits a stale page and notices the version doesn't match will try something else, ask a colleague, or check the changelog.

But an agent that reaches your documentation doesn't then evaluate whether it's right. It acts on it, writing the wrong parameter into a codebase without hesitating.

Documentation accuracy has stopped being an editorial problem and become a systemic issue, requiring a solution that scales just as fast.



“The same page that misled one developer now misleads an unknowable number of agents, which then propagate that misunderstanding to downstream users,”

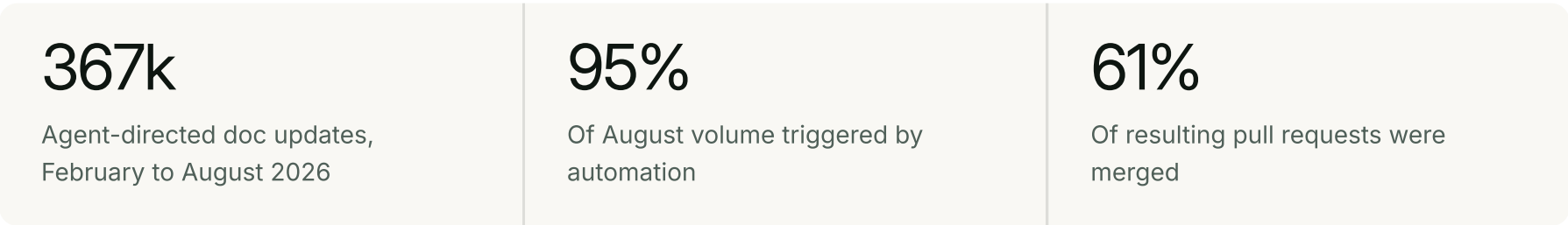
Sarah Deaton
Technical Content Engineer,
Claude Code, Anthropic

ANTHROPIC

Reader to Writer

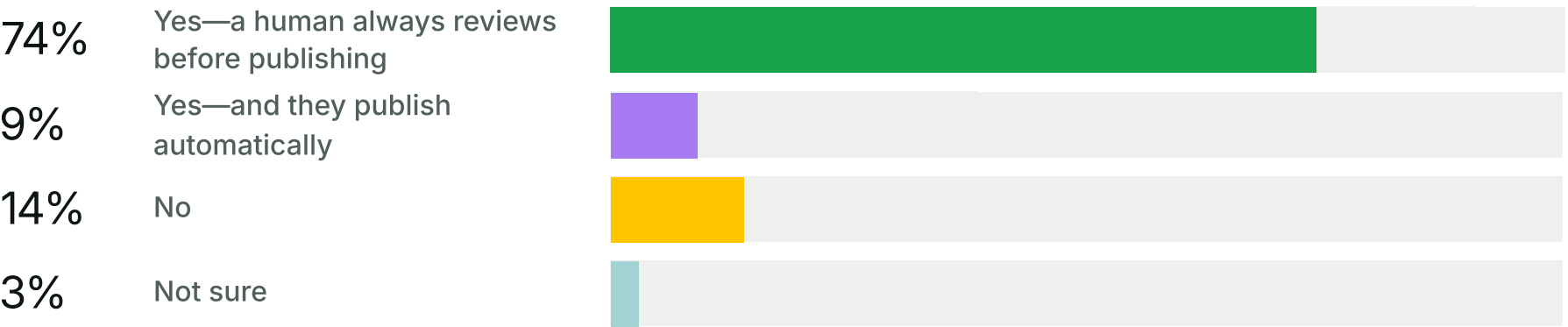
Leading companies aren't just structuring their documentation for agents to read. They're now leveraging them to update it with every code push as well. From February to August 2026, users directed the Mintlify agent to update their documentation nearly 367,000 times.

In August alone, 95% of that volume was triggered by automation, either a webhook or a scheduled cron job. More than 61% of resulting pull requests were merged. And now teams have started pointing the same agents at resolved support tickets, turning the questions documentation failed to answer into patches.



Our survey findings confirm this trend. A whopping 83% of respondents say AI agents now draft documentation updates for their team. However, only 9% let those agents publish without a human in the loop.

Do AI agents draft documentation updates for your team?



ALL 329 RESPONDENTS

FIGURE 6

A How Anthropic closes the loop

Sarah Deaton is the only person running documentation for Claude Code at Anthropic—a setup made possible by agents that surface knowledge gaps and draft the fixes for her review.

Scheduled jobs pull new feedback every few hours through the Mintlify analytics API: broken code-sample reports, written comments, and more. An agent triages that stream alongside Slack mentions and GitHub issues, sorting each signal into user error, support issue, or a real documentation gap.

From a dashboard she built with Claude Code, an autofix button turns the ones that matter into pull requests. And when feedback points to friction in the product, she can submit a PR to fix that instead, so the issue gets solved rather than documented around.

Read the full story →

Documentation is transitioning from a static surface teams publish into a system that reads itself and drafts its own improvements—with people deciding what's true and what ships. And it isn't going to stop there.

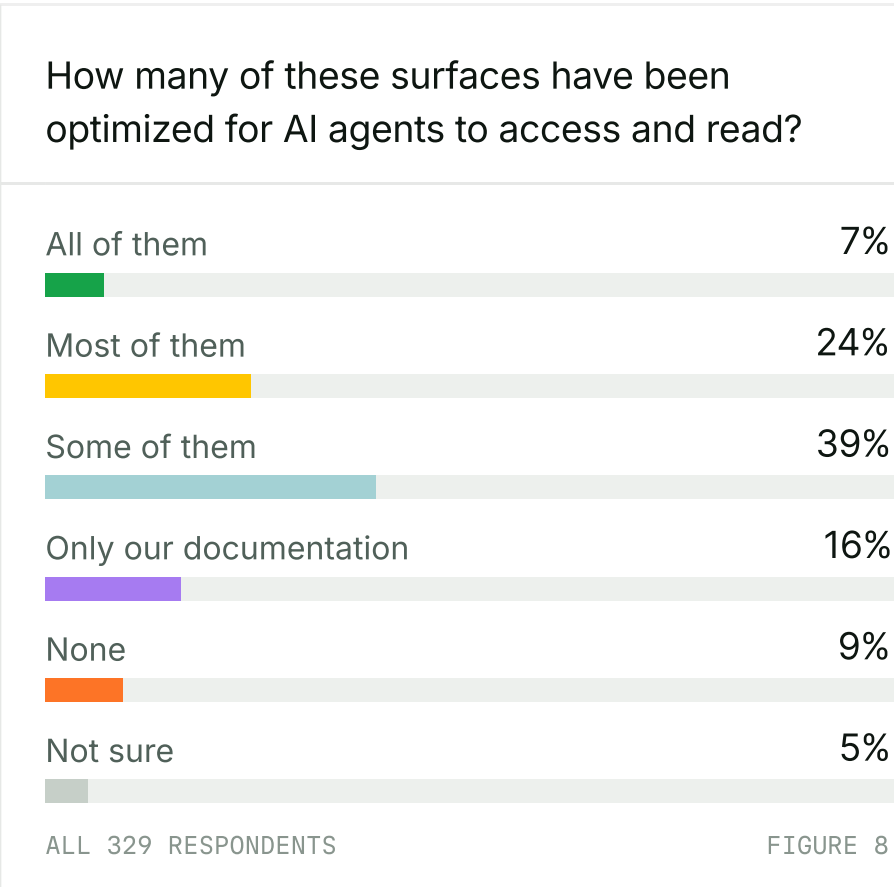
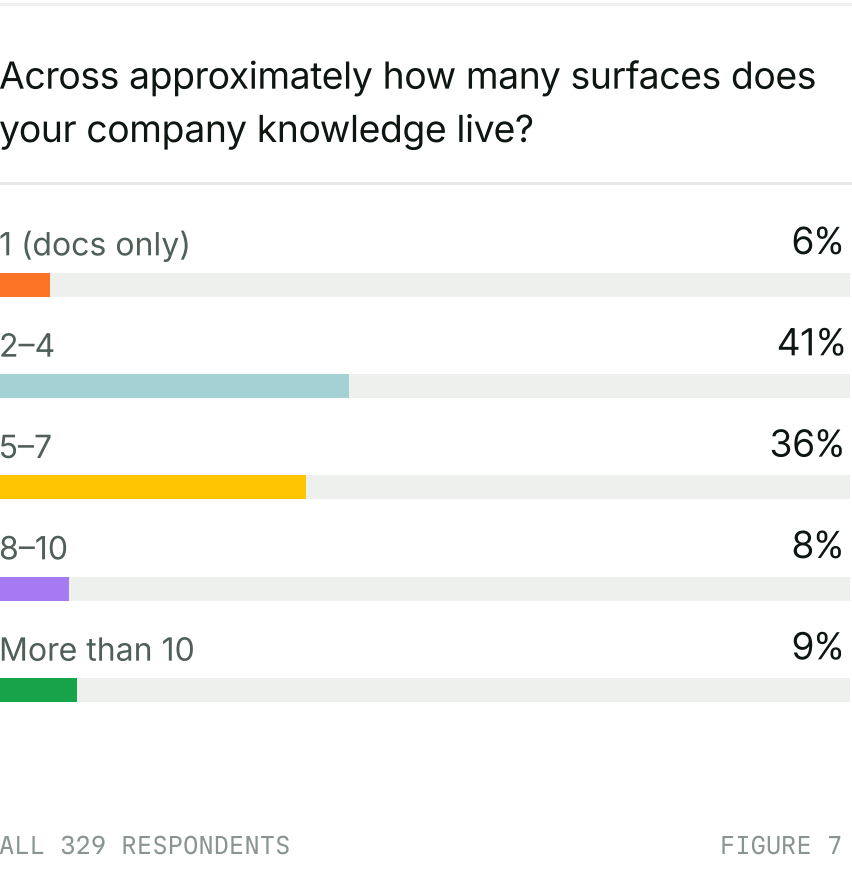
PART 4

From Documentation to Knowledge Infrastructure

Spread Across Surfaces

Agents answer from whatever they can reach, and a lot of what a company knows isn't in the docs. It lives in a help center article, support ticket, or community post. More than half of survey respondents say their company knowledge lives across 5+ knowledge surfaces.

Respondents also reveal that few of these surfaces have been optimized for AI discovery and consumption, let alone self-improvement. Less than a third classify "most" or "all" of their knowledge as AI-ready.



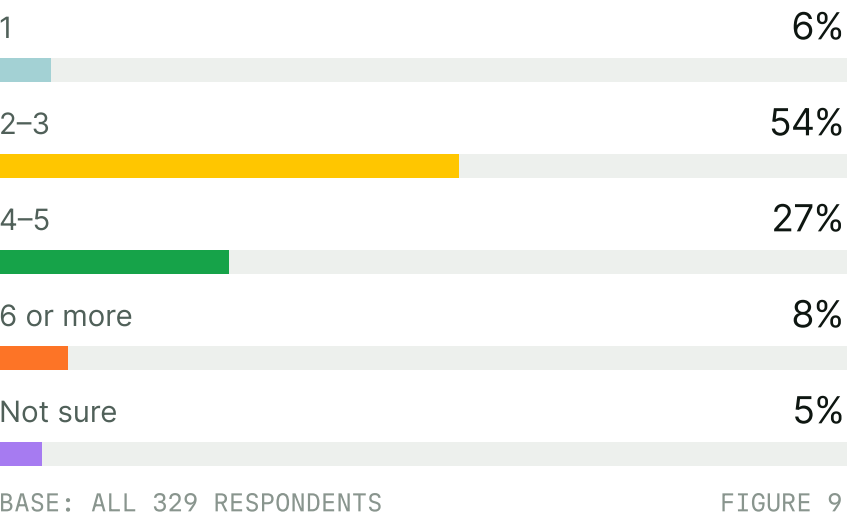
Why It Often Stops at Docs

As Mintlify Co-founder and CEO, Han Wang, explains, "Making a surface agent-ready is a massive engineering lift, from clean markdown and a queryable interface to a way to report what's missing and a path back to the source. Once it's 'done,' the target has already moved."

llms.txt was proposed in September 2024. MCP followed that November, and by late 2025 had been donated to a foundation with more than 10,000 public servers running on it. A company that made its knowledge base agent-ready in early 2025 built against a landscape that has already changed several times since.

Then there’s ownership. Docs sit with DevRel or engineering, the help center with support, community with marketing, and the changelog with product. That means four teams, four tools, and four versions of “truth.”

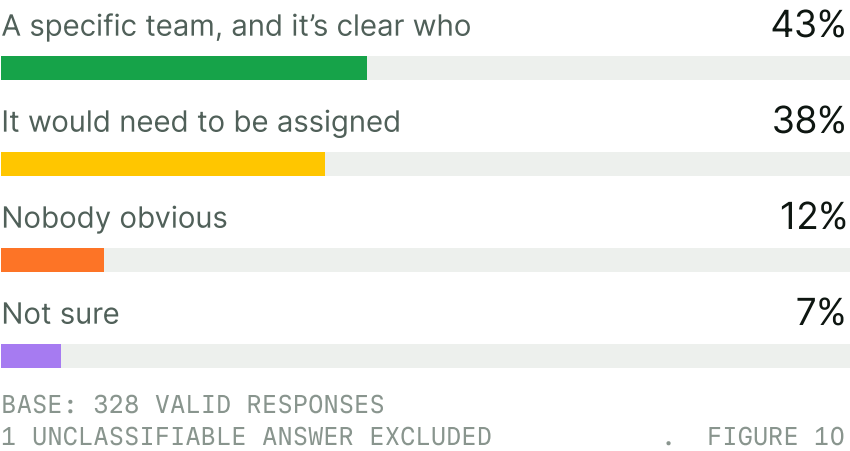
How many separate tools or platforms does your company use to publish and manage its knowledge?



Some 35% of survey respondents say their companies use four or more tools to publish knowledge. This group is also 5x more likely to rate their companies’ AI-readiness as “behind” compared to those with just one knowledge publishing platform.

Less than half of respondents have a specific team that’s responsible for ensuring non-docs surfaces are AI-ready. The majority, 57%, has no clear owner.

If your company wanted to make knowledge other than docs readable by AI agents, such as its help center or community site, who would be responsible for that work?



“Knowledge and the ability to make it agent-accessible often don’t sit in the same place,” says Han. “The people typically closest to what customers actually ask, like sales and support, are the least equipped to make those answers reachable, and the people who could make them machine-readable aren’t the ones holding that knowledge.”

 Enter the Knowledge Engineer

Today, teams struggle with a lack of ownership for making knowledge AI-ready. To solve this, more companies are hiring designated “knowledge engineers” to build and maintain systems that keep knowledge synchronized across the company.

Rather than manually choosing and writing every word, this role focuses on designing the prompts, context, automations, and agents that produce and maintain content. A successful knowledge engineer sets the standards and processes for their entire team to capture and use information.

Learn more about Knowledge Engineering →

Top companies are recognizing these issues and that knowledge doesn't stop at docs. They've stopped treating it as a collection of static, siloed repositories and started running it as operational infrastructure the whole business depends on. Documentation, changelog, community, and everything else live on a single system that's agent-accessible by default and designed to maintain itself at scale.

"This layer is what actually unlocks a future where an agent can take action and run long-horizon tasks for your company and your customers," says Hou. "These outputs are what drive actual business outcomes, so getting the foundation right matters more than most teams realize."

Knowledge Infrastructure in Practice

One company already doing this is Microsoft Foundry. As Microsoft's unified platform for building, grounding, and governing AI apps and agents, Foundry needed more than a docs site. But its existing solution constrained design, left roughly 40% of API docs with missing or inaccurate return values, and offered no llms.txt or Markdown hosting.

The team decided to create an entirely new developer experience. Not a docs site, but a knowledge center embedded directly in the Foundry portal.

Documentation and API reference, roadmap and labs pages, a model showcase, an auto-generated changelog, and a community hub—all in one place, inside the product.

Every page can be copied or viewed as Markdown. An auto-generated llms.txt indexes the content for LLMs. Semantic HTML maps to typed MDX components, so the pages are structured for machines as much as rendered for people.



"Docs alone just don't cut it anymore. As agentic AI keeps advancing and doing more work for people, we'll see more companies join the ranks of leaders like Microsoft Foundry, Decagon, Coinbase, and Anaconda by putting knowledge infrastructure in place that makes content accessible to both humans and agents."

Han Wang
CEO, Mintlify



Conclusion

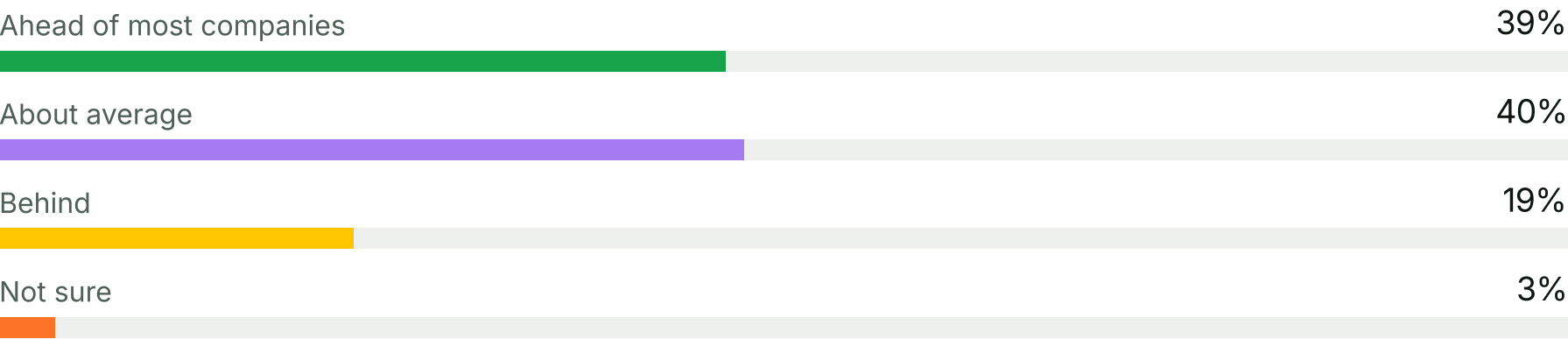
On the one surface companies have made ready, documentation, agents went from reading to writing in a single year. It’s quickly becoming a system that maintains itself, with people deciding what ships.

But everywhere else a company’s knowledge lives—the community, sales enablement, the release notes—hasn’t followed. So the depth races ahead on one part of company knowledge while everything else sits stagnant, written for an audience that’s now in the minority.

That gap is the work of the next few years, and it isn’t only technical.

Ironically, most companies don’t yet realize this applies to them. Some 79% rate their own knowledge as “about average” or “ahead of most other companies.” Yet just 7% have made every surface they own readable by an agent, illustrating that AI confidence is running well ahead of readiness.

How would you rate the AI-readiness of your company’s knowledge?



BASE: ALL 329 RESPONDENTS; PERCENTAGES DO NOT SUM TO 100 DUE TO ROUNDING

FIGURE 11

“Building the knowledge layer takes more than technical infrastructure. It takes people and operating infrastructure too—the whole organization has to buy in, or a real source of truth never happens. That’s the operational problem: getting a company to actually treat knowledge as one shared asset, and building a system where this knowledge can grow and self-heal over time.”

David Hou
Product Education Lead, Decagon.



In the meantime, agents aren’t waiting for any of this to get sorted. They’re already reading everything they can reach, and answering from whatever they can find.

This raises an uncomfortable question: How much are you investing in knowledge that’s becoming increasingly invisible?

Documentation is
where the work starts.
It can't be where it ends.

To learn more, visit www.mintlify.com