How Fast Does AI Find a New Page?

By William Tygart · November 2026

On one of my sites, an AI system found a new post in under twenty minutes. On the same site, in the same fifty-five days, another AI system took a median of 92.85 hours to show up.

Same content. Same server. Same everything except the visitor.

That gap is why I don't trust any single answer to the question people keep asking me: how long until AI finds my new page? There isn't one clock. There are fleets, and each fleet is running a different job. Discovery speed is a property of the job, not of your page.

How I measured it

For every post published across five of my sites over fifty-five days — September 18 through November 12, 2026 — I took the publish timestamp and matched it against the first AI hit in the origin server logs. Not an analytics tool. Not a dashboard. The log line the server wrote when the request actually arrived.

Origin logs are the only record that sees all of this. Anything that depends on the visitor politely executing your JavaScript misses the fleets that show up, take what they came for, and leave in half a second. Those are most of the visits that matter here.

Here is what the clocks actually looked like, site by site:

Site	Median publish → first AI hit	Notes
tygartmedia.com	~4.2 hours	—
BC ESG	7.2 hours	Across 35 articles
Healthcare Facilities Hub	7–8 hours	bingbot was the fastest discoverer — and the biggest crawler, at 11,016 hits
Continuity Hub	10.8 hours	28 of 28 posts AI-discovered; PerplexityBot was first on 27 of the 28
Risk Coverage Hub	14.07 hours	PerplexityBot median — but some posts were found in under 20 minutes

Read that table carelessly and you'll conclude some of my sites are "faster" than others. Read it again and a different pattern shows up: the medians are mostly a record of *which bot happened to be first*, and the bots are not doing the same work.

Two jobs, two clocks

Risk Coverage Hub makes it impossible to miss. PerplexityBot's median time to find a new post there was 14.07 hours, with some posts picked up in under twenty minutes. ClaudeBot, on the same site, over the same period, showed a median lag of 92.85 hours.

That's roughly 6.6 times slower. Same site. Same content. If discovery speed were a property of the page — its structure, its tags, its internal links — that gap shouldn't exist. It exists because the two systems wanted different things.

PerplexityBot behaves like retrieval. Somebody asks a question, the answer needs sources, and the system goes looking. That job runs constantly, it is driven by live demand, and when your new page matches something people are asking about, it can arrive in minutes. The under-twenty-minutes cases weren't flukes in the data; they were what retrieval looks like when the timing

lines up.

ClaudeBot, in these logs, behaves like a batch indexer. It arrives in sweeps. It works through what it knows about, picks up what's new, and leaves. A median of 92.85 hours isn't slowness — it's a cycle. The page didn't fail to be found for four days. It was waiting for the next pass, and the next pass came on schedule.

Continuity Hub sharpened the same point from the other side: twenty-eight posts published, twenty-eight discovered, and PerplexityBot was first on twenty-seven of them. When one retrieval fleet decides you're in its territory, your discovery time stops being about your site at all. It's about their sweep pattern against your publish times.

And Healthcare Facilities Hub is a useful reminder not to turn any of this into religion. There, the fastest discoverer wasn't an AI-native bot at all. It was bingbot — 7 to 8 hours, and the biggest crawler on the site by volume. The old search infrastructure is still feeding the new answer layer, and on some sites it gets there first.

So the honest answer to "how fast will AI find my page" is a question back: which AI, doing which job? A retrieval-driven visitor can be there before you've finished sharing the link. An indexing fleet might take four days, and it won't be because you did anything wrong.

The part you'll argue with

Tags didn't speed any of this up.

In a separate set of experiments in the lab, I changed the tagging on hundreds of already-published articles — properly, with substantive tags a reader would recognize as topics, not keyword stuffing. The effect was real and measurable: AI re-crawls of those known articles went up roughly 23 to 28 percent.

First discovery did not move.

I've sat with that for a while, because it cuts against a lot of advice, including some I would have given a year ago. The resolution is that discovery and re-crawl are separate systems. A tag, apparently, is a signal the fleets use when deciding whether something they've already catalogued is worth another visit. It is not a flare that brings them to a page they've never seen. By the time a tag can help you, discovery has already happened — or hasn't — for reasons that have nothing to do with it.

Which means two popular instincts are both wasted effort. Tagging a new post more heavily will not get it found faster. And panicking at the four-day silence from an indexing fleet is misreading a batch cycle as a penalty. If you want to know which situation you're in, you can't guess it from a dashboard. You have to watch the log line land.

What I'd actually check

If I publish something and want to know where I stand, I look for three things. Who arrived first, and is it the same fleet that usually arrives first on that site? How long until the *second* fleet shows up — because being found by retrieval and being indexed are different assets, and I want both. And then, over the following weeks, whether the page earns re-crawls, which is the part tagging and structure actually influence.

Discovery is weather. Re-crawl is climate. I can measure both. I can only work on one of them.

We do AI citation restoration. When the AI can't find you, can't read you, or keeps citing somebody else, that's the work — send us the worst one you have. Your RFPs. Your disasters. The problem everyone else already took a pass at. We come from the restoration industry; we know what's on the floor, and cleaning it up is the job. And if you're a developer who just needs help on the hard part, get a hold of us. We'll jump in on that piece with you.