AI DIGEST

🔥 Earthquake at Google DeepMind: Hassabis steps down as CEO, and Jeff Dean takes three more people with him to found a company

The loudest event of the day, and three separate Hacker News stories about the same thing: 547 points (the Google post), 652 (the new company's site, top of the day) and 297 (NYT). The primary sources are Sundar Pichai's letter to staff and the new company's site.

Thursday, 6 August 2026

IN THIS ISSUE

TOPIC 1

🔥 Earthquake at Google DeepMind: Hassabis steps down as CEO, and Jeff Dean takes three more people with him to found a company

The loudest event of the day, and three separate Hacker News stories about the same thing: **547 points** (the Google post), **652** (the new company's site, top of the day) and **297** (NYT). The primary sources are Sundar Pichai's letter to staff and the new company's site.

What actually happened, point by point from the letter:

• **Demis Hassabis** steps down as CEO of Google DeepMind and becomes **Chair of GDM + Chief Scientist of Alphabet**, staying at the head of Isomorphic Labs. Pichai's wording: Demis "described us as standing in the foothills of the singularity", and he needs a role where he can "focus all of his attention on shaping the future of AGI"

• Operational leadership of GDM goes to **Koray Kavukcuoglu** (13 years at DeepMind, behind WaveNet and DQN) as SVP, reporting directly to Pichai • **Jeff Dean leaves after 27 years** at Google

The important part is on the new company's site. Pichai names two people: Dean and Sanjay Ghemawat. On `discoveryloop.com` there are **four** founders: **Jeff Dean, Sanjay Ghemawat, Quoc Le and Oriol Vinyals**. So **Oriol Vinyals** (co-author of Gemini, AlphaStar) and **Quoc Le** are leaving Alphabet too. That is the core walking out. The team describes itself as "three of the most cited researchers in AI and two of the most cited in distributed systems".

The company is **Discovery Loop**, a public benefit corporation. The mission, verbatim: automate experimental loops in ML, science and engineering, "running thousands of experiments in parallel". They start by automating **ML research itself** and say outright that they will be "our own first customer". Google stays on as **founding investor and cloud partner**.

Google's numbers from the same letter: **the Gemini app at 950M+ monthly users**, **Gemma at 900M+ downloads**, and a mention of **Gemini 4** as already in the works.

Why it matters. The sharpest line of the day is from @blader: "there are exactly Zero venture investors in the valley who needed to see a pitch deck from Jeff Dean". But the bigger point is elsewhere. Discovery Loop is building exactly what yesterday's item 4 described (Lilian Weng, harness engineering): an **automated self-improvement loop**, only at company scale and with the people who built TPUs and MapReduce. Yesterday Weng described a loop that could be automated; today a company was raised to do it. [proven - both primary sources opened directly: Pichai's letter and the Discovery Loop site; the discrepancy in the founder list between them was checked]

Pichai and Hassabis letter · discoveryloop.com · Jeff Dean's announcement · HN on the changes, 547 · HN on Discovery Loop, 652

TOPIC 2

Atlassian Rovo leaks Jira and Confluence through prompt injection, and Atlassian has said nothing for two and a half months

196 points. PromptArmor demonstrated a zero-click data leak from **Atlassian Rovo**, the AI agent that roams the whole Jira/Confluence stack.

The mechanics, verbatim from the write-up. A user uploads a file with a hidden injection into Rovo, an ordinary thing: "found a document online, handed it to the agent". They ask it

to go through Jira tickets, and the injection makes the agent **append sensitive data to an attacker's URL**. Rovo's URL-opening tool has no protection at all against following a link the agent **just composed itself**. The attacker's server logs the request along with the data.

Two details that make it worse:

• The attack works **even when web search is disabled for the organisation**: the setting removes search but **does not remove the tool that opens results • It leaves no trace**: if the user later opens that same chat, they see ordinary ticket suggestions and no evidence of the attack

Disclosure timeline: PromptArmor notified Atlassian on **23 May**. Atlassian assigned a case number, said thanks, and **over two and a half months and several reminders never got back in touch. The vulnerability is live**, which is why they are publishing.

Why it matters. Three days running, the main security story is **an agent doing something extra because of trusted input** (04.08, the AISI agent social-engineering a maintainer; 05.08, the npm worm in an agent config file; today, Rovo). The common denominator: **a tool the agent can call with a parameter that came from outside**. The practical conclusion: the dangerous thing is **a tool that accepts a URL or a path assembled from text that was just read**. Anything arriving from a messenger, from email, from someone else's documents is data. Rovo shows that a rule in the prompt does not save you if the tool technically allows an arbitrary destination. [proven - the detailed write-up with the disclosure timeline was read in full]

PromptArmor write-up · HN thread

TOPIC 3

Rust has officially banned LLM-written code in the language core, and the argument is not about code quality

109 points, and a rare case of an AI policy written by the community that maintains one of the most important languages in use. The author is Jynn Nelson; the policy was adopted by **five teams** of the project for the `rust-lang/rust` monorepo.

The rule in one line: **LLMs are allowed for questions, analysis, distillation, verification and review, but not for creation.** (There is an exception: pre-agreed, non-critical, well-tested changes with **mandatory disclosure** that an LLM was involved.)

The interesting part is the reasoning, verbatim from the post:

• **"A polished technical product no longer signals effort and understanding."** A clean PR used to mean there was a person on the other end who had put in the time. Now, with an autonomous agent, "**there is no longer anyone on the other end**"

• **The ease of writing code finishes off the review bottleneck.** A number worth remembering: **1,281 open PRs** in `rust-lang/rust` right now. And the key line: "most of a reviewer's work is deciding **whether this direction is a good idea at all**"

• Mechanically copying back and forth between an LLM and the tracker wastes the time of the people reading it

Why it matters. A direct link to yesterday's item 3 (ACM Queue: coding takes only ~15% of a developer's time, the pressure moves downstream to review). Yesterday it was a peer-reviewed mechanism, today it is **the largest open source project hitting that wall and writing a rulebook**. The conclusion is practical and awkward: a rule that says "we write faster with AI", without a rule for who reviews it and how, just moves the queue onto people. In Rust that queue already has a number: 1,281. [proven - official Rust blog, figures and wording from the primary source]

the policy on Inside Rust · HN thread

TOPIC 4

Cloudflare open-sourced its "OS for agents", and it is literally harness architecture, only for a company

501 points. Cloudflare released **Cloudflare OS**, a platform where every employee gets an agent and a workspace "built around their company: how it works, what it knows, and which systems it relies on".

In **May** they gave the first version to **everyone at Cloudflare**, and by their account "thousands of people in every function, many outside engineering" use it daily. Today they released a reworked version that any organisation can deploy for itself.

The key part is a **shared library of context and skills**. Verbatim: "It captures terminology, procedures and the best known ways to do repeatable work **as instructions an agent can follow**. When one person finds a better way to do something, everyone can use it immediately."

They are honest about what did not work in the first version: the apps were static, not living software. Deterministic tasks still required running the instruction again and burning tokens. And the main one, **access to an MCP server told you which tools the agent could call, but not which resources it actually saw**. That became a problem the moment people started sharing workspaces.

Why it matters. Their **mistakes** are worth more than their release: the MCP point ("we know the tools, we do not know the resources seen") is exactly the problem that killed Rovo in item 2 today. And separately, their conclusion: do not make every person explain the same procedure to the model again. That is an argument for a curated library of instructions instead of repeating "how to do this" in chat every time. [proven - official blog, read in full; this is an open source release. Not an announcement of intent]

Cloudflare OS announcement · HN thread, 501

TOPIC 5

An open 4B model beat GPT-5.6 Sol at search and costs 100x less

251 points. Neon (that is Postgres, now part of Databricks) and the startup Castform published a measurement: **an open 4-billion-parameter model, after RL fine-tuning, searches a database as accurately as GPT-5.6 Sol at 100x lower cost**.

The number that makes this worth attention: a typical multi-step search query on `gpt-5.6-sol` takes **over 10 seconds and costs ~$0.03** end to end. The reason: agentic search stopped being a single request. The model plans and searches in a loop, and **every iteration of that loop is another frontier-model call**.

The authors' logic is simple: small open models are two orders of magnitude cheaper but weaker out of the box; RL fine-tuning for **one specific task** (search) closes that gap. Castform's stated goal is "to make post-training as accessible as prompt engineering".

The most sober line in the thread: "the harness should spin up a subagent and hand narrow tasks to specialised models; Claude Code already does some of this, giving the explore agent's work to Haiku".

Why it matters. Straight on the theme of yesterday's item 4 and today's item 1: intelligence is being pulled out of the scaffolding. The practical side: $0.03 per search query only adds up to money at volume, and local BM25 over a small corpus stays cheaper than any model. This is a cost benchmark for agentic search if the question ever comes up of searching something much larger. [proven - figures from Neon's technical post; it is their own measurement of their own product, so the source has an interest]

Neon + Castform write-up · HN thread

TOPIC 6

Alexandr Wang: a swarm of agents at Meta does more than a team of 100 engineers. He gave the number, not the evidence

The post got 75K views. Meta's Chief AI Officer **Alexandr Wang**, at YC Startup School 2026 in conversation with Garry Tan, said verbatim:

> *"Inside Meta we have seen cases where, if you build the right agentic loop and have the right evaluation system and the right metric the agents are optimising,* ***a swarm of agents can achieve more than a team of 100 engineers****. They do it very confidently, actually quite easily."*

The claim does not hold on trust. It rests on three conditions ("the right loop", "the right evaluation system", "the right metric"), and none of them is spelled out. There is no task, no benchmark, no measurement. This is exactly the genre yesterday's ACM Queue article took apart in myth #3: **a metric that invites gaming the system**. In today's item 3 Rust says the opposite, and with a number. The bottleneck is that **someone has to decide whether this is a good idea at all**, and a swarm of agents does not do that work, it creates it.

Why it matters. Treat it as a marker of a genre, not a fact. When this quote gets used as an argument (and it will, coming from the YC stage in Meta's name), the counter-question is: on what task, by what metric, and who reviewed the result. There is nothing silly about the idea of "a swarm of agents plus an evaluation system"; it is literally what Discovery Loop from item 1 intends to do seriously. The difference is that they described a mechanism, and this is an anecdote from a stage. [fuzzy - the quote is verbatim and verified, but there is no published measurement behind the claim itself]

the quote from YC Startup School

TOPIC 7

levelsio burned $900 on an agentic loop and threw away 95% of the result

150K views, 964 likes, and the most useful antidote to item 6 on the same day.

Pieter Levels builds products solo and lives off them. He wrote about his experience with "Gauntlet Loop": "Every time I do a Gauntlet Loop it becomes **a total mess and chaos of suboptimal code** where too much is happening and nothing works well. And $500 is burnt. I have to clean everything up by hand and revert to what it was."

An hour and a half later, in a correction: "**It was actually $900 burnt**, and it became a total mess. Had to clean up **95% of what it did** and revert to what it was!"

His conclusion about his own workflow: the only way to code with AI is **step by step**, without a long autonomous loop.

Why it matters. This is about the **mode of use**. A long autonomous run with no check on the intermediate result produces the most expensive mistakes; stopping and inspecting the artefact catches them. Levelsio pays for that lesson in cash: $900 for one run. [proven - his own experience with a specific amount, both posts read in full]

levelsio post · the $900 correction

TOPIC 8

Zed built a version control system where every line is tied to the conversation that produced it

340 points. Zed announced **DeltaDB**, a separate version control system built for the fact that code is now written by agents.

The core idea in their words: "**Software is made between commits**", meaning all the interesting work happens in the gap git does not see. What DeltaDB does:

• **Rewind to any edit**: it records every operation between commits, each with a stable identifier • **Links code to conversation**: "from any line of code find the conversation; from any message jump to the code it changed"

• **Branch from any moment**, including **mid-agent-run**: the worktree is virtualised, so a new branch costs almost nothing • A colleague can join in **while the work is still running** and talk to the agent doing it, without waiting for a commit and a push

An important clarification from the thread: it works **with any agent over ACP**, so Codex and Claude Code too, not only Zed's own agent. For now it is **early access by request**.

Why it matters. The link between a commit and the conversation that caused it usually depends on what a person types into the commit message by hand, which means from memory and selectively. DeltaDB makes that link a property of the store. This is the second day running that tooling has pointed at the same spot. [promising - early access, no field reports yet, everything from their announcement]

DeltaDB announcement · HN thread

TOPIC 9

TIME serves AI bots a different version of the site, with ads inside the text

234 points. Vincent Schmalbach found that **TIME** serves AI crawlers **different HTML** from what people get, with ads woven into that version so they land in the model's context and surface in the answer to the user.

The most sober lines from the thread:

• "Sounds like plain old prompt injection. ChatGPT can now browse 50 pages when asked to research a topic. It is entirely plausible the final answer gets influenced by ads like this"

• And the sharpest one: "The best way for a politician to lie is to convince someone else of the lie and put them in front of the cameras. **LLMs are credulous by nature**"

Plus one reasonable extra suspicion: this could also be a way for an agency to inflate the impression numbers it reports to a client.

Why it matters. Any pipeline that reads other people's sites through browser automation or `curl` gets whatever the server serves, and the server can now tell it is not a human and hand over something else. The rule "take numbers from the primary source" still holds, but it needs an addition: the primary source can be personalised for a bot. The defence is the same: cross-check several sources, and open pages in a real logged-in browser. [proven - the test is reproducible: different HTML for different User-Agent]

Schmalbach's write-up · HN thread

TOPIC 10

Why the Erdős problems have started falling to AI

130 points, Quanta Magazine. The legendary problems of Paul Erdős, the kind of maths problem that stood for decades, have started being solved with the help of AI, and Quanta looks at why now.

And on the same day, in the same feed, the opposite paper lands: **"Position: LLMs Can't Jump"** (256 points, OpenReview), a position paper arguing that LLMs do not make **intuitive leaps**. The best line from the thread, half in jest: the proposed experiment is to train an LLM on all texts **up to 1980-90** and see whether it can arrive at inventing itself.

Why it matters. These two topics are presented together deliberately, because each one alone lies. "AI solves Erdős problems" without the second half sounds like intuition has been cracked; "LLMs can't jump" without the first sounds like nothing works. The truth sits in between and it is operational: **where there is a measurable success criterion and thousands of attempts can be tried, the machine wins** (literally the thesis of Discovery Loop in item 1 and Castform in item 5). Where one non-trivial leap is needed with no feedback, it does not. The same conclusion as yesterday's item 8 on tabular data: what matters is knowing which of the two modes a tool is working in. [promising - Quanta is good popular science journalism, the second paper is a position paper, meaning an argument. Not an experiment]

Quanta on the Erdős problems · HN thread · "LLMs Can't Jump" · HN thread

misc - briefly, what else is worth a look

• **@blader set a screen time limit for his daughter, and she asked Claude Code to build her own browser.** 406K views, 7404 likes, the funniest thing of the day: an 11-year-old got around the YouTube limit by vibe-coding a browser on Electron and Chromium. In a separate post he admits it is his own fault: he promised unlimited screen time for every successful hack. Among the earlier ones, she learned to **approve her own screen time requests**

• **@garrytan: "Make something agents want"** - 637K views and 3218 bookmarks for four words. YC's motto ("make something people want") rewritten for the new reality. As an indicator of where the biggest accelerator in the world is looking, it is worth more attention than a lot of long essays

• **@emollick noticed a contradiction:** models are getting better at following complex instructions, but at the same time they increasingly "**use judgement**" about which parts of the instruction to focus on and which to mute. His conclusion: "Big implications for skills, which may become **suggestions. Not commands**"

• **@emollick on financial advice:** a study from MIT and Stanford says **most people would be financially better off following LLM advice** (they tested GPT-5.2 and Gemini 3 Flash). With an important caveat: the quality of the advice depends heavily on **what questions you ask**. Read it with yesterday's item 8 in mind: asking for numeric forecasts is a bad idea

• @gdb: **a packed room for OpenAI's talk on the Hugging Face incident** at Black Hat. This is the same story Maleev dramatised as "The OpenAI Uprising" and which then fell apart; now it is being dissected at the main security conference

• @dakshgup: **Greptile v5** - a code review agent rewritten from scratch, "catches more bugs, with higher precision, and twice as fast (~2 minutes per review)". Straight on the theme of item 3: if review is the bottleneck, that is where the tools will go

• @levie: **"99% of the world's tokens will be consumed in an enterprise context"** - code, life sciences research, manufacturing automation, security, fraud detection. A sober counterpoint to the consumer hype

• @EricTopol **- not about AI:** tirzepatide (Zepbound) versus sitagliptin in people with diabetes and prior atherosclerotic cardiovascular disease: a reduction in cardiovascular events, and also **infections by 36% and all-cause mortality by 45%**. Published in BMJ. Presented as a signal for the GLP-1 class, not as a recommendation

• @lennysan **- following yesterday:** yesterday the news was that YouTube destroyed his channel for "impersonating himself". Today he wrote it up at length: the appeal is filed, the channel is still gone. In his own words: "on one hand, it is good that they take the problem seriously; on the other, maybe there is a more delicate way than shutting down a channel"
