

# IBM's CEO Says Open-Weight Models Are 90% as Good for a Fraction of the Price

2026-10-26 / 00:23:13

*“no one cuts prices by 80% unless there is some externality in play”*

— from this episode's transcript

- Lenar Kess
- Damra Vol

OpenAI and Anthropic have cut prices by as much as eighty percent since June, IBM's CEO says open-weight models now deliver about ninety percent of the quality at a fraction of the token cost, and builders are already splitting work between a frontier planner and a local typist. Why prices fell is the first argument; the rest of the day covers hosted agents, reviewing code with a different model, and agents under stress.

- [Max Woolf's post on cheap LLMs](#) lays out the dated price cuts, the cache-read arithmetic, and his case for why prices fell; it frames the whole lead.
- [The Register on IBM's Q3 call](#) has Arvind Krishna putting open-weight models at ninety percent as effective for 10 to 25 percent of the token cost, which matters because IBM also sells the hardware.
- [A Reddit video of the same game built twice](#) compares Opus alone with Opus planning and a local Qwen coding, an anecdote about the planner-and-typist split.
- [A LocalLLaMA post on Qwen 3.8 Flash Next](#) claims it runs in 8 GB of video memory, which would widen who can run local models.
- [Nate B Jones on outcome pricing](#) argues per-token pricing meters compute while leaving the return on the buyer.

- [Jack Witherspoon's talk on Gemini Managed Agents](#) covers the cloud sandbox, the secrets proxy, and environment-ID handoff between agents.
- [A post on agent orchestration](#) and [a post on models writing their own harness](#) are user impressions that recent models orchestrate themselves, which would raise the stakes for the sandbox.
- [Arize's talk on its Signal agent](#) describes an always-on agent behind a proxy with external memory, and says the human's job moves to evaluations.
- [Greptile's Dax on code review agents](#) reports that reviewing with a different model family raises incident detection by about ten percent; the figures are the speaker's own.
- [Opus 5.5 writes slop code](#), [I just realized we've been shipping AI slop](#), [Farol](#), and [the Kasparov-era post](#) show how reviewers and builders are reacting to AI-written code.
- [MCP-Injector](#) injects faults into Model Context Protocol tool calls and finds the larger gpt-oss model isn't clearly better under faults.
- [Self-Direction](#) distills lessons from failed tasks into small adapters, with reported gains of 8.9 to 17 points.
- [Agent Trace](#) identifies the model behind a black-box agent from its repair attempts, with 96 percent attribution accuracy on the models tested.
- [A field report on a hidden note](#) describes an agent that acted on it sixty-one minutes later; it's a single operator's account.
- [The Lynch lab skin-cancer comparison](#) pits a fine-tuned 2015 network against GPT-6 Astra and Claude Fable 5.1 on 200 biopsy-confirmed lesions.
- [Aleph Alpha's Kolibri](#) is an open-weight, Apache 2.0 mixture-of-experts model with 4.4 percent of parameters active, built for German and regulated work.
- [A third-party write-up of Microsoft's Decision-1](#) describes a nine-billion-parameter model; a mention only.

- [TACS-RF](#) has Claude Opus 4.8 driving chip design tools for a 240 to 251 gigahertz transmitter, with extracted-and-simulated results.
- [NVIDIA's FLORA](#) is an open suite for labeling robot data, claimed to be about 125 times cheaper than GPT-6 Astra.
- [SIREN](#) finds goal-insertion attacks that push humanoid robots with safety filters into collisions.

---

## SEGMENTS

|                          |                                      |
|--------------------------|--------------------------------------|
| <a href="#">00:00:04</a> | The price sheet                      |
| <a href="#">00:05:04</a> | Open weights on-prem                 |
| <a href="#">00:07:42</a> | Hosted agents and self-orchestration |
| <a href="#">00:11:02</a> | Review with a different model        |
| <a href="#">00:13:56</a> | Breaking agents at the tool          |
| <a href="#">00:17:47</a> | A ten-year-old network               |
| <a href="#">00:20:33</a> | Agents reach hardware                |

## Transcript

### 1. Lenar Kess 00:00:04

On July thirtieth, OpenAI cut the price of its small model, GPT-5.6 Luna, by eighty percent. It wasn't a promotion or a new tier. It was the same model at one-fifth the price, five weeks after it launched. And that was one cut in a run of them from OpenAI and Anthropic since late June.

- [minimaxir.com](#)
- [theregister.com](#)
- [v.redd.it](#)
- [reddit.com](#)
- [youtube.com](#)
- [youtube.com](#)

- [reddit.com](https://reddit.com)
- [reddit.com](https://reddit.com)
- [youtube.com](https://youtube.com)
- [youtube.com](https://youtube.com)
- [reddit.com](https://reddit.com)
- [reddit.com](https://reddit.com)
- [github.com](https://github.com)
- [reddit.com](https://reddit.com)
- [arxiv.org](https://arxiv.org)
- [arxiv.org](https://arxiv.org)
- [arxiv.org](https://arxiv.org)
- [reddit.com](https://reddit.com)
- [lynchlab.net](https://lynchlab.net)
- [shattered.io](https://shattered.io)
- [arxiv.org](https://arxiv.org)
- [arxiv.org](https://arxiv.org)
- [arxiv.org](https://arxiv.org)
- [arxiv.org](https://arxiv.org)

## 2. Damra Vol 00:00:21

Eighty percent is what you do when somebody is about to take your customers. So who was it?

## 3. Lenar Kess 00:00:27

Max Woolf published a long post yesterday asking exactly that, and we'll start with his price table. IBM's CEO said something on his earnings call that bears on the same question, so that comes second. After that we've got Google's hosted agents and the claim that models now orchestrate themselves. Then a Greptile talk arguing the model that reviews code shouldn't be the one that wrote it, and a paper that breaks agents by breaking their tools. We finish with a ten-year-old image network that beats two frontier models at skin cancer, and a coding agent that designed a transmitter running above two hundred gigahertz.

## 4. Damra Vol 00:01:00

That's a full Monday. Start with the ledger, though. What did Woolf actually count?

## 5. Lenar Kess 00:01:05

He keeps a dated list of prices per million tokens, input and output. OpenAI's late-June launch priced Sol at five dollars in and thirty out, and Luna at one and six. Anthropic's Sonnet 5 comes in at

two and ten, which Anthropic called a temporary launch price and then made permanent in August. Then comes Luna's eighty percent cut, and in August about twenty percent off Sol.

6. Damra Vol 00:01:28

And September?

7. Lenar Kess 00:01:30

September twenty-second is the big day. OpenAI halves both models again, so Sol goes to two and ten and Luna to ten cents and fifty. The same day Anthropic ships Opus 5.5 at four and twenty, which he figures is forty to sixty percent cheaper overall than Opus 5. A week later GPT-6.1 Sol sits at two and ten with a cheaper cache, and he calls that cumulatively seventy-five percent down.

8. Damra Vol 00:01:55

Those are list prices. Nobody running a coding agent pays list in any way that matters, because the bill is mostly cache.

9. Lenar Kess 00:02:02

He says exactly that. Cache reads are ninety-six to ninety-nine percent of token spend in agentic coding. So when Anthropic released Fable 5.1 and dropped the cache-read price from a dollar to twenty-five cents, that's a seventy-five percent cut to the line that dominates the invoice, and he puts the real cost drop around seventy to seventy-five percent.

10. Damra Vol 00:02:22

Which should make bills fall, except —

11. Lenar Kess 00:02:24

Except Vercel's gateway data, which he cites, has cost per token at forty-three percent of its April level while total spending keeps rising. Quality improves faster than prices fall, so people hand the cheaper model bigger jobs.

12. Damra Vol 00:02:38

That sits oddly next to yesterday's Gartner number, where only thirteen percent of CIOs said they saw big AI value. Tokens are cheap and the invoices are still arriving before the value does.

13. Lenar Kess 00:02:49

His central exhibit on why is a sentence I'd put on a wall: no one cuts prices by eighty percent unless there is some externality in play. His candidates are the two labs reacting to each other, models

being swappable by changing an endpoint string, and open weights.

14. Damra Vol 00:03:04

Externality is a polite word for DeepSeek.

15. Lenar Kess 00:03:07

Largely. DeepSeek's V4 Flash launched in April at fourteen cents in and twenty-eight out. The updated V4.1 Flash scores forty-five on the Artificial Analysis index, up from twenty-four, at fifteen cents and sixty. And the number of providers hosting it on OpenRouter went from thirteen to twenty-nine, several within ten percent of DeepSeek's own cost. He suspects the Luna cut was aimed at models like that.

16. Damra Vol 00:03:31

And Saturday's Kimi is on his list.

17. Lenar Kess 00:03:34

Kimi K3.1 Pro, fifty-five on that index against fifty-eight for Opus 5.5, at two dollars and ten. Moonshot's weights are due November fourth.

18. Damra Vol 00:03:44

Half the price of Opus 5.5, three index points behind, and downloadable in nine days. If I'm sitting at Anthropic, that's a very specific spreadsheet.

19. Lenar Kess 00:03:54

He rejects the idea that this is Uber-style subsidy. The labs aren't a duopoly, entry costs are lower, and he expects prices to drift toward marginal cost. What nags him is the IPO. Anthropic is rumored to list within months, and deep cuts look strange in an S-1, so with reported inference margins as high as eighty percent he's curious what those margins look like after the cuts.

20. Damra Vol 00:04:16

Anthropic's own line is the simple version: Opus 5.5 requires less compute to serve than Opus 5, and its pricing reflects that. Cost came down, they passed it on. It's also exactly what you'd say if a competitor were cutting, so I don't think the two explanations compete. The cut is cost-down, and the timing is competition.

21. Lenar Kess 00:04:37

From the seller's side, Nate B Jones argued yesterday that per-token pricing only meters compute and leaves the return on the buyer. He points to Cedar's billing agent and to Sierra pricing per completed service interaction. If tokens drift toward marginal cost, outcomes are where the margin goes.

22. Damra Vol 00:04:55

So tokens become a utility and everyone sells something on top. I'm curious whether an outcome price survives a buyer who can also read the token meter.

23. Lenar Kess 00:05:04

Arvind Krishna, IBM's CEO, said this on the third-quarter earnings call, reported by The Register on the twenty-second. It picked up almost three hundred points on Hacker News. He gave the history in numbers. In 2024 the top closed model might score a seventy on a broad benchmark and an open model a twenty or thirty. In late 2025 the gap was seventy to fifty. Now, in his words, open-weight models run anywhere from twenty-five percent to ten percent of the token cost of the frontier models. And quote, that is very attractive given that they're now ninety percent as effective.

24. Damra Vol 00:05:38

Ninety percent of what, on which benchmark? He's a CEO on an earnings call, so I'm hearing a round number from somebody who sells the hardware around it.

25. Lenar Kess 00:05:47

His own examples are mundane on purpose: handling email, reviewing contracts, answering customer calls. His argument is that most business tasks don't need the frontier, so firms won't keep paying a ten-times premium. He also concedes the hard part, asking where enterprises are going to find their kernel engineers. IBM's answer is software to make deployment easier and, he expects, demand for its Spyre accelerator cards.

26. Damra Vol 00:06:12

Same quarter, The Register has mainframe revenue down twenty percent. That doesn't make him wrong. But he calls it a simple question for any enterprise, and the simple answer comes with two or three racks of hardware IBM would be happy to sell you.

27. Lenar Kess 00:06:27

Two or three racks is his estimate of how much capacity most firms need to build something useful. That's smaller than I'd have guessed.

28. Damra Vol 00:06:34

Smaller than the data center story we've been telling for months, anyway.

29. Lenar Kess 00:06:37

Now the individual builder's version. Over on the ClaudeAI subreddit, someone built the same game twice. Opus alone took sixteen minutes and cost two dollars fifty-two. Opus planning with a local Qwen model doing the coding took forty-two minutes and sixty-two cents.

30. Damra Vol 00:06:53

It's four times cheaper but two and a half times slower, on one game by one person, so it's an anecdote. Still, the split is the interesting part. The expensive model does the work that's mostly deciding, and the cheap one does the typing. That's the same division IBM is describing for a whole company.

31. Lenar Kess 00:07:12

And the cheap end keeps getting more reachable. Someone on the LocalLLaMA subreddit claims Qwen 3.8 Flash Next runs on eight gigabytes of video memory plus thirty-two of system RAM. They report up to nine hundred tokens a second reading the input and thirty generating, with the expert weights memory-mapped instead of loaded.

32. Damra Vol 00:07:30

A cousin of the NVMe streaming experiments from yesterday. Thirty tokens a second generating is also, I'd guess, where a lot of those forty-two minutes went. You pay the cheap typist in waiting.

33. Lenar Kess 00:07:42

Jack Witherspoon, a developer-experience engineer at Google DeepMind, gave a talk at the AI Engineer conference on Gemini Managed Agents. The pitch is a coding agent that runs in a cloud sandbox you start with one API call. It isn't fighting your laptop for resources or touching your own configs and subscriptions, and you don't think about containers, networking, or firewalls.

34. Damra Vol 00:08:03

Where does it keep the secrets? That stops me with every hosted agent.

35. Lenar Kess 00:08:07

He answered with a proxy. The sandbox only holds dummy placeholders for credentials. When a request goes out to a matching external endpoint, the proxy swaps in the real token, so the real secret never lives inside the box where the agent is running.

36. Damra Vol 00:08:21

So a hijacked agent can leak a placeholder and that's all. Nice property, if the proxy's endpoint matching is tight. It's Google's own description, though, and nobody tested it on stage.

37. Lenar Kess 00:08:32

He did demo how agents chain. A first agent researched Nvidia's SEC filings and saved metrics into the sandbox. A second, a quant agent, reused that same environment by passing its environment ID, and ran Python analysis scripts. A third wrote the report. Every step is logged as a function result or a code execution.

38. Damra Vol 00:08:52

I like that the environment is the handoff. Agents don't pass summaries to each other, they pass a file system. Memory by shared disk.

39. Lenar Kess 00:09:00

It sits on what he calls the Interactions API, an agent-first interface with stateful multi-step conversations. Around it are a harness called Anti-Gravity, scheduled triggers, and webhooks for output. The model underneath is Gemini 4 Argon, which has a one-million-token window shared across input, reasoning, and output.

40. Damra Vol 00:09:19

Shared across reasoning is the number I'd chew on. Long decode eats the same budget as the input.

41. Lenar Kess 00:09:25

Now the other half of this cluster, which is user impressions rather than announcements. On the Anthropic subreddit, a post titled New recommendation: Agent orchestration says recent models can handle subagent orchestration themselves, and a post on the singularity subreddit claims frontier models can now write their own task-specific harness to hill-climb a goal, tested on Gemini 4 Argon.

42. Damra Vol 00:09:48

Two people posting, so I'd keep it light. But if it's even half right, the hand-built orchestration layer everyone has been writing for two years becomes something the model does for itself. And the hosted sandbox gets more important, not less, because it decides where the agent runs, who holds the secrets, and where the logs go.

43. Lenar Kess 00:10:07

Arize's talk on its Signal agent, from the same conference feed, ends up in the same place from the other side. It's an always-on agent triggered by schedules, events, or metrics. It runs in sandboxes where a proxy controls everything in and out, and it keeps its findings in external memory so it can resume from a checkpoint if a sandbox dies. It works with Claude Code, Cursor, or Codex.

44. Damra Vol 00:10:30

Look at what Google and Arize both built. A proxy at the sandbox wall, and state kept outside the sandbox. Two separate talks, same design.

45. Lenar Kess 00:10:39

So what's left for the human? Arize's speaker says evaluations: aligning automated metrics with human judgment using golden datasets, and testing the evaluators themselves before you trust them, offline and against production traces.

46. Damra Vol 00:10:52

The person stops writing the loop and starts deciding what counts as done. That's a harder job, because a model can't write its own definition of good and have you believe it.

47. Lenar Kess 00:11:02

Greptile's co-founder, Dax, gave a talk on building a code review agent, and Greptile's product reviews roughly one percent of the pull requests on GitHub. He starts with four problems. The agent misses bugs, it lacks context about the codebase, and it costs between three and twenty-five dollars per pull request. And its nitpicky comments push people into defensive coding when they just auto-resolve them.

48. Damra Vol 00:11:24

Three to twenty-five dollars. At the low end that's already more than a lot of the model calls that wrote the code.

49. Lenar Kess 00:11:30

The result people will quote is the inversion. He says Claude models produce roughly three times as many one-off errors as GPT models, and GPT models are significantly more prone to authentication bypasses. If the same model writes and reviews, catch rates fall. Reviewing with the other family raises incident detection by about ten percent.

50. Damra Vol 00:11:50

Obvious in hindsight: a model's blind spots and its habits are the same thing. If it didn't see the bug while writing, it won't see it while reading.

51. Lenar Kess 00:11:58

He adds a practical wrinkle. To invert, you have to know who wrote the code, and he says branch names and commit messages identify the generating model about eighty percent of the time, with telemetry heuristics covering the rest.

52. Damra Vol 00:12:11

Every figure here is the speaker's own and unaudited, and he sells the product. Ten percent more incidents caught is believable. I'd ask for the denominator. But the branch-name trick is great: the model's habits leak into its git hygiene.

53. Lenar Kess 00:12:27

His other results come in three parts. One is a prompt aimed at bug categories drawn from your own outage history and told to ignore style. Another is a continuously updated set of historical failing pull requests to tune against, and the last is a codebase index of architectural flows and authentication state instead of grep. He's also plain that an agent verifies a bug's presence, not its absence, so the catastrophic one-in-a-thousand pull request stays hard to catch.

54. Damra Vol 00:12:54

The one in a thousand is what review is for. Review agents get graded on the common bugs, and the rare outage is the reason you wanted review in the first place.

55. Lenar Kess 00:13:03

On Reddit the mood is rougher. A post titled Opus 5.5 writes slop code, but succeeds anyways describes a research project inspired by a slop-code benchmark, and another says, I just realized we've been shipping AI slop. There's a Show HN called Farol that builds a risk map of a pull request for the person who approves it, and a post arguing senior developers are in their Kasparov era.

56. Damra Vol 00:13:26

Kasparov is a better comparison than the poster probably meant. He lost to Deep Blue, then championed human-plus-machine chess, and the review loop is where that centaur lives. Farol interests me because it's aimed at the approver, not the author. It says here's where to look.

57. Lenar Kess 00:13:43

The author is now a model that's fast and cheap, and the approver is a person who isn't. Anything that points that person at the right fifty lines is useful.

58. Damra Vol 00:13:51

And if Dax is right, the helper should be a different species of model than the author.

59. Lenar Kess 00:13:56

A paper from Tilburg, Luxembourg, and two Italian universities asks what happens to an agent when its tools misbehave. They built MCP-Injector, middleware that sits on the tool-call path of the Model Context Protocol and injects faults. The fault mix comes from the real world: they mined 57,240 issue and pull-request threads across 1,774 open-source MCP server repositories.

60. Damra Vol 00:14:22

So the kinds of breakage are the ones people actually filed bugs about.

61. Lenar Kess 00:14:26

They sort the faults into six kinds. Some tools return a small mistake inside an otherwise valid answer, and some return a wrong but well-formed one. Others break their schema or throw an explicit error, and the last two are a delayed response and a silent drop. Each can be permanent, transient, or intermittent.

62. Damra Vol 00:14:44

A wrong-but-valid answer is the nasty one. An error is a gift, because the agent knows. A plausible wrong number is what it builds on.

63. Lenar Kess 00:14:52

The experiment has 1,350 cells. It uses two sizes of gpt-oss, twenty billion and one hundred twenty billion parameters, and three agent stacks: a custom plan-and-execute loop, the OpenAI Agents SDK, and LangChain. Those run across twenty-five tasks and three seeds. Under their permanent-heavy profile, the task-completion score falls sixty percent, from 8.08 out of ten to 3.23.

64. Damra Vol 00:15:20

And the headline is that the bigger model doesn't help.

65. Lenar Kess 00:15:22

The hundred-twenty-billion model keeps no larger share of its fault-free score. Its completion rate under faults is 91.1 percent against 95.6 for the small one, though the interval on that gap runs from

minus 0.2 to 9.3 points, so I'd say no better, not clearly worse. They also see the agent stack mattering, which suggests the framework's error handling counts for more than model size.

66. Damra Vol 00:15:46

Two caveats stay attached: two models from one family, and a stress profile skewed toward permanent faults, which is harsher than most production days. The authors say so themselves. Still, it matches what anyone running MCP servers will tell you. The agent isn't what breaks, the tool it calls is.

67. Lenar Kess 00:16:05

Another paper this morning takes the opposite end. Self-Direction has an agent that fails a task reflect on its trajectory and extract a lesson. It then distills that lesson into a small low-rank adapter, about 0.02 percent of the base model's parameters, with the base weights frozen. Adapters go into a library and get routed to related problems later. On Qwen-based agents it reports success-rate gains of 8.9 to 17 points across GAIA, Claw-Eval, and Terminal-Bench 2.1.

68. Damra Vol 00:16:34

A lesson as a file you can name and swap. I'd want to know what happens when two lessons contradict each other, but I like that the unit of learning is something you can hold.

69. Lenar Kess 00:16:44

And one from the security side. Agent Trace identifies which foundation model is behind a black-box agent by injecting ambiguous probes that cause serialization failures at tool interfaces, then watching the sequence of repair attempts. With simple Markov chains it reports 96 percent attribution accuracy under adversarial identity spoofing, on the models they evaluated.

70. Damra Vol 00:17:05

Put it next to Dax's branch names. A model's fingerprint shows up in how it repairs a broken tool call and in how it names a branch. An agent's identity is observable even when you tell it to hide.

71. Lenar Kess 00:17:17

Last, a field report, with all the weight a Reddit post carries. A site operator on the AI Agents subreddit says an agent visited, read a note hidden from human visitors, and did what it said sixty-one minutes later. The agent isn't identified, and the note was the operator's own decoy.

72. Damra Vol 00:17:34

Sixty-one minutes is the detail. It didn't act in the same request. It read the note, held onto it, and came back. Whoever built that agent gave it memory and a schedule, and a sentence on a web page counted as an instruction.

73. Lenar Kess 00:17:47

Dr. Lynch's lab posted a comparison that reached 582 points on Hacker News. The test is two hundred lesions, all confirmed by biopsy, drawn from a public dermatology dataset released in 2025. One hundred needed action: thirty melanomas, twenty-five basal cell carcinomas, twenty-five squamous cell, and twenty actinic keratoses. The other hundred were safe to discharge.

74. Damra Vol 00:18:11

And the contestants?

75. Lenar Kess 00:18:13

An Inception v3 convolutional network from 2015, fine-tuned on about sixty-eight thousand images on one H100 for roughly twenty minutes, against GPT-6 Astra and Claude Fable 5.1, each given just the two images and one prompt.

76. Damra Vol 00:18:28

Which is the unfair part to say out loud. One is a specialist trained on exactly this kind of image, and the others are walking in cold.

77. Lenar Kess 00:18:36

The CNN got the first-choice diagnosis right on 125 of 200, 62 percent. Astra got 84 and Fable 85, 42 percent each. Within the top two choices it was 173 against 134 and 138. On melanoma, first choice out of thirty, it's 13 for the CNN, 8 for Astra, and 11 for Fable.

78. Damra Vol 00:19:00

The author states his limits: each language-model case was run once and answers vary, thirty melanomas is a small sample, and Inception v3 is a baseline. So I wouldn't read it as frontier models being blind to skin. A twenty-minute fine-tune beat a general model with no training on the task, and the general model costs far more per call.

79. Lenar Kess 00:19:20

That pairs with Aleph Alpha's Kolibri, out today as a paper. It's a mixture-of-experts transformer with 78.1 billion parameters in total and 3.46 billion active per token, which is 4.4 percent. It's released as open weights under Apache 2.0 and built for German and regulated work like public

administration, industry, and aerospace. They trained on 24 trillion tokens, more than twenty percent German by bytes. Over two trillion of those German tokens were curated from the web or generated synthetically, and then came reinforcement learning on 1.2 million internal tasks.

80. Damra Vol 00:19:56

The sovereignty pitch is that they control the whole pipeline, from data through evaluation, which is a procurement answer. A ministry can say who trained what. The Pareto-frontier claim is theirs, measured against the open models they picked.

81. Lenar Kess 00:20:11

There's also Microsoft's Decision-1, a nine-billion-parameter model on a Qwen base. One commenter on Hacker News calls the explainer ad-like, and the source is a third-party write-up, so I'll only mention it.

82. Damra Vol 00:20:23

A decade-old network for pixels, a mixture of experts at 4.4 percent active for German paperwork. Neither is a general intelligence, and both are cheap to run.

83. Lenar Kess 00:20:33

Tristan Liang and Aravind Nagulu have a paper called TACS-RF in which Claude Opus 4.8 drives industry design tools directly: Cadence Virtuoso and Spectre, Calibre, Cadence EMX, and Ansys HFSS. The test case is a 240 to 251 gigahertz transmitter in a 90-nanometer silicon-germanium process.

84. Damra Vol 00:20:56

That's not a toy frequency.

85. Lenar Kess 00:20:58

The agent semi-autonomously designed the whole chain. That meant a 60 gigahertz oscillator with its power drivers, a frequency doubler with a bowtie antenna, and the matching networks between them. With extracted parasitics and electromagnetically verified passives, the transmitter delivers plus 5.1 to plus 6.3 decibel-milliwatts of output power across an 11 gigahertz tuning range, and it draws 361 milliwatts of DC power.

86. Damra Vol 00:21:24

One design, from one group, and the abstract describes extracted-and-simulated results, not a chip back from a fab. The paper's own word is semi-autonomously, and the division-of-labor section is

the interesting bit, because semi covers a lot and the abstract doesn't say who kept which decisions.

87. Lenar Kess 00:21:43

On the robotics side, NVIDIA's FLORA is an open model suite for labeling robot data: timestamped captions, outcome rewards, and process rewards for reinforcement learning. It's what's used to train GROOT. The authors say it's roughly 125 times cheaper than GPT-6 Astra, competitive with it on their internal benchmarks, and that it labels 200,000 hours of data a day on 512 GB300 GPUs.

88. Damra Vol 00:22:10

The 125 times is against Astra, on their own benchmark. The reason to care is that robot data is mostly unlabeled video, and labeling is where the money goes.

89. Lenar Kess 00:22:20

And a counterweight from a team that includes David Brumley. Humanoid robots ship with safety filters that guarantee collision-free actions mathematically, but only if the safety parameters are set right for the task and environment. No general method exists to set them. Their tool, SIREN, finds goal-insertion attacks, where a normal user instruction pushes the robot into a collision. It found three times as many attacks as a baseline, and they made a real humanoid collide.

90. Damra Vol 00:22:47

A guarantee that holds under a configuration nobody can verify is a guarantee about the configuration. Same pattern as the MCP paper: the capability is fine, and the boundary is where it breaks.

91. Lenar Kess 00:22:58

Kimi K3.1 Pro's weights are due November fourth, and that's when Woolf's table, IBM's ninety percent, and the fifty-five versus fifty-eight index gap get checked by people who don't work for Moonshot. I'm Lenar Kess.

## Hosts on this episode

- Lenar Kess moderator
- Damra Vol critic