Episode Book

EPISODE #6190

When Your AI API Quietly Changes Under You

PUBLISHED	21 October 2026 at 20:28 UTC
DURATION	29:13
LISTEN	myweirdprompts.com/episode/ai-api-silent-model-updates-42297/
HOSTS	Corn, Herman, Hilbert
SCRIPT MODEL	openai/gpt-6.1-sol
VOICES	chatterbox-regular (chatterbox-local)
PIPELINE	V5.18
SOURCES	12 cited by the research (listed at the end)
BOOK GENERATED	22 October 2026 at 07:46 UTC

SYNOPSIS

You ran the evals, you pinned the version, and your invoice parser worked perfectly in October. Then December arrives and the same API name returns something subtly different. This episode digs into what actually changes behind an unchanged endpoint name — weights, system instructions, routing, classifiers, quantization, even the GPUs — and why an alias like "latest" is effectively a subscription to retesting. Drawing on model equality testing across 31 commercial Llama endpoints (11 of which produced distributions different from Meta's reference implementation) and the disclosure practices of major providers, we separate model identity from service behavior, explain why pinning a snapshot removes one major source of change without freezing everything, and offer a practical stance: stop asking whether the model got worse overall, and start asking whether the specific part you're paying for moved.

TRANSCRIPT



Corn

Your October eval can be perfectly sound and still make a poor December purchasing decision. Daniel is asking what happens after the testing stops: you keep calling the same API name, but are you still getting the thing that won? We have been talking about repeated runs, judges, significance, and affordable evaluations. All of that work has a shelf life if somebody changes what sits behind the name.



Herman

The awkward part is that the name can stay completely still while several other things move. The weights, the instructions around them, the hardware executing them, even the decision about which request goes where. Your application sees a successful response and carries on.



Corn

A green status light on a moving target.



Herman

Daniel wants to know what changes, how much warning developers get, and how to notice without turning monitoring into another full-time occupation. Before attributing anything to a secret model replacement, though, we need to separate the model from the service delivering it.



Corn

Especially when the service is the thing we actually evaluated. I didn't test a tensor floating in space. I tested whether my invoice parser put the right amount in the right field.



Herman

That distinction matters immediately. An API name is a pointer. What answers you is a bundle: model weights, serving software, system instructions, request routing, safety classifiers, and sampling logic. A promise about one component doesn't automatically cover the others.



Corn

The restaurant has guaranteed the flour.



Herman

In some cases, yes. You might have the same weights but different numerical execution. You might have identical numerical execution but an added instruction. You might reach a different model entirely. Those are different events, even if your first symptom is identical: yesterday's reliable prompt starts producing rubbish.



Corn

I would separate identity from usefulness, too. If it changes and gets better, I still need to know. An extraction tool that helpfully expands an abbreviation can break a downstream lookup just as thoroughly as a wrong answer.



Herman

Would you reject an improvement that broke your existing format?



Corn

Until I could test and accommodate it, yes. Improvement is a claim about someone's objective. My objective includes not spending Thursday repairing the connection to the accounting system.



Herman

That is why the vocabulary is useful. Behavioral drift describes a change in outputs over time. Model equality testing asks whether two endpoints appear to generate from the same underlying distribution. A silent update is a change without public disclosure, a version increment, or re-evaluation. None of those terms automatically tells you what caused it.



Corn

Particularly the word silent. A provider can announce an update somewhere obscure, and I can still be operationally surprised. Disclosure and notification aren't interchangeable.



Herman

The striking audit here is Model Equality Testing. In summer two thousand twenty-four, it examined thirty-one commercial Llama endpoints. Eleven produced distributions different from Meta's reference implementations. That's enough to make identity testing a practical concern rather than a philosophical pastime.



Corn

Eleven endpoints selling the label with optional contents.



Herman

Some deviations were greater than the difference produced by substituting an entirely different language model. That doesn't establish that every flagged provider deliberately cheated. Serving choices can alter distributions. It does establish that the advertised family name doesn't settle what behavior you're buying.



Corn

Nor does a system card settle it. The evaluation might be accurate about an artifact I never actually reach.



Herman

No provider in the disclosure scorecard sample exposed a verifiable chain from the evaluated artifact to the API response. Today you can collect evidence that behavior changed. You generally cannot prove, from the returned text alone, that you received the exact artifact described in a report.



Corn

Then I can already see your next purchase, Herman. An exceptionally elegant detector with no answer to what it detected.



Herman

I would at least require a dashboard.



Corn

Naturally. Otherwise the uncertainty stays invisible.



Herman

The first practical distinction is between an alias and a pinned identifier. OpenAI's undated names, including GPT four o, GPT four o mini, and GPT four turbo, point to their respective latest versions. Historically that same pattern covers GPT four and GPT three point five turbo. The name in your request isn't enough. Inspect the returned model information.



Corn

Returned information being a provider's statement, not independent proof.



Herman

Correct. Useful operational evidence, not cryptographic identity. ChatGPT four o latest is a particularly explicit moving target. It follows the version used in ChatGPT and is updated frequently. Choosing that name means choosing movement.



Corn

Latest is a subscription to retesting.



Herman

There is a concrete repointing example. On October second, two thousand twenty-four, GPT four o moved to the August sixth snapshot. The unchanged alias now offered cheaper tokens and Structured Outputs.



Corn

That sounds like an update people would welcome. Which makes it a better example than a catastrophe. Nobody has to behave badly for the thing I tested to stop being the thing I'm using.



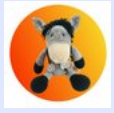
Herman

Input pricing dropped fifty percent, output pricing thirty-three percent. A substantial commercial change, alongside a capability change, under the familiar request name. An alias can be useful when you actively want improvements without editing configuration. It's a poor choice when your approval process assumes a frozen target.



Corn

You install things the week they appear. Would you actually choose a pinned version?



Herman

For an exploratory assistant, probably not. For a workflow making consequential decisions, yes. My enthusiasm doesn't get to sign the acceptance test.



Corn

It has certainly tried to.



Herman

You're the one who wants a machine that won't change during a nap. That would rule out quite a substantial maintenance window.



Corn

I accept scheduled maintenance. I object to waking up as the integration engineer.



Herman

Anthropic draws a stronger boundary around model IDs. Its documentation says each ID identifies a pinned version, with the underlying model constant for the lifetime of that ID. The important follow-up is that the weights stay fixed while the surrounding infrastructure can change.



Corn

Request routing, classifiers, sampling. The pieces you named earlier can still alter what reaches me.



Herman

Yes. Anthropic explicitly says infrastructure updates can produce minor observable differences with unchanged IDs and unchanged weights. It even says that unexpected changes on a previously stable ID are most likely due to infrastructure updates. That is a useful diagnostic admission.



Corn

Useful because it saves developers blaming themselves for every wobble. Less useful if their application cannot tolerate a minor difference.



Herman

There is also a naming wrinkle. The older Sonnet four point five convenience alias resolves to the latest dated snapshot for that minor version. Since the four point six generation, the dateless ID is itself the snapshot. You cannot determine alias behavior merely by checking whether the name contains a date.



Corn

The date vanished. The need to read the documentation survived.



Herman

OpenAI's compatibility language says prompting behavior between snapshots is subject to change. We should be precise about that quotation. Between snapshots means between versions. That sentence alone does not prove that a pinned snapshot silently changes its weights.



Corn

Thank you. Otherwise we'd be using an ordinary warning about upgrading as evidence of a secret upgrade.



Herman

Pinning still doesn't guarantee identical outputs or freeze the entire service around the weights. It removes a major source of change, namely an alias moving to a new snapshot. That is worthwhile without pretending it settles every layer.



Corn

Suppose my extraction prompt deteriorates under an unchanged ID. My first suspicion wouldn't be replaced weights. It would be that somebody added an instruction to be helpful, and now the model insists on supplying an introductory paragraph.



Herman

That is entirely plausible. System prompt revisions can alter verbosity, refusal behavior, formatting, or tool selection. Classifier updates can block material previously allowed through. Retrieval changes can give the same model different supporting documents. Routing changes can send requests down different paths.



Corn

Fine-tuning changes what the model learned. Retrieval changes what it gets told today. Those should leave very different traces if I save the right information.



Herman

Often, yes. Although identifying the cause from behavior remains difficult. A refusal might originate in the model, an instruction, or a separate classifier. Your logs show the refusal, not necessarily the component responsible.



Herman

Corn, would you count a retrieval update as the provider changing the model?



Corn

In conversation, probably. In an incident report, no. I'd call it a service change and ask which component moved. Otherwise we spend a day arguing about the definition while the document search keeps returning the wrong policy.



Herman

The computing layer adds another route. CUDA versions, inference kernels, and the GPUs serving requests can affect numerical results. Quantization stores or computes values at lower precision. Pruning removes parts of the computation. Those can change outputs without a new public product name.



Corn

Lower precision doesn't mean automatically worse at my task, though.



Herman

No. Good quantization can preserve much of the useful behavior. The objection is undisclosed substitution or untested change, not lower precision as a category. An INT8 or FP8 deployment may be economical and perfectly adequate. It can also differ in the particular corner your application depends on.



Corn

A provider's average is not my obscure surname.



Herman

Hosting expensive models creates pressure to reduce cost. That can mean efficient kernels and better batching. It can also create an incentive to quietly use a smaller model, a more aggressively compressed version, or a lighter route during peak traffic. These are possible substitution mechanisms, not a license to accuse every slow response of fraud.



Corn

Particularly because overload itself can explain strange latency. You need evidence that distinguishes explanations.



Herman

A router can also select variants based on query characteristics. Then the service you evaluate on short benchmark questions may behave differently on long production requests. No single permanent swap is required.



Corn

Your test traffic gets the velvet rope. The invoices use the side entrance.



Herman

Potentially. Even benign routing makes representativeness important. Your evaluation has to resemble your workload closely enough to encounter the paths your workload uses.



Corn

That makes me less interested in the grand question of whether the model got worse overall. I want to know whether the part I'm paying for changed.



Herman

Frequency is where we have to resist false precision. The endpoint audit gives us a cross-section of mismatches. It doesn't give us a universal timetable for silent updates. We cannot turn eleven flagged endpoints into a claim that providers change models every fortnight.



Corn

A photograph doesn't tell you the speed.



Herman

The Silent Updates Scorecard looked at nine first-party providers and seven third-party hosts. Providers commonly published substantial safety documentation, including quantitative evaluations. Yet none supplied enough information for an outsider to verify that the served artifact matched the documented one.



Corn

That leaves the answer to how often partly hidden by the thing we're asking about. If disclosure is missing, a changelog can't be our complete dataset.



Herman

Historical examples show real consequences. GPT three point five turbo had been updated twice by November two thousand twenty-three without those updates being visible to downstream developers. Separately, a toxicity-detection comparison found a major performance downgrade when moving from text davinci zero zero three to GPT three point five turbo instruct.



Corn

Separately matters there. Changing between those named models is a visible migration, even if the consequences surprise you.



Herman

Yes. The useful lesson from that comparison is that prompt rankings changed too. A prompt that worked best on the earlier model could lose its advantage on the replacement. Model and prompt form a tested combination. Keeping only the prompt text doesn't preserve its performance.



Corn

We should hear a listener's version of that uncertainty. This one's a voicemail from Beryl. Beryl: Hello Daniel, hello Corn, hello Herman, this is Beryl from Leeds, I'm on the number forty two bus going past the Corn Exchange, so I've got about four minutes before my stop. I wanted to ask about today's topic, which I think is about how you never know if the model behind the name has quietly changed, is that right? Because our local history group has been using one of these models to transcribe old parish records, and back in March it was brilliant with the handwriting, and then in June it started writing out dates in a completely different way, and I thought I was going mad. So my question is, when a model changes quietly like that, is there a way of telling whether it's the model or just me getting worse at the prompts? Thank you, love the show.



Herman

Keep a few unchanged record images, the exact prompt, and the earlier outputs, then rerun that fixed set. That separates a changed test from changed behavior. For dates, score the transcription separately from the formatting. A different date convention needn't mean worse handwriting recognition.



Corn

That also preserves the original evidence instead of arguing from memory. My memory of a tool's reliability becomes remarkably generous once I'm annoyed with its replacement.



Herman

What would you actually keep for your invoice parser, Corn? Not a laboratory setup. Something you would maintain.



Corn

The request, the returned model identifier, the output, and whether it passed the checks that matter. A fixed sample of troublesome invoices, plus ordinary ones. I'd keep prompt versions separately. If I adjust the instructions at the same time the provider changes



Corn (continued)

something, I have manufactured a mystery.



Herman

That's a realistic foundation. Pin the snapshot where offered. OpenAI has described pinned versions as available for at least three months after an updated version is introduced. Anthropic ties its underlying-model promise to the lifetime of the ID. Neither statement means indefinite availability.



Corn

A pinned model can still have an eviction date.



Herman

Watch that date explicitly. OpenAI publishes minimum deprecation notice periods: generally available models get at least six months, specialized variants at least three months. Preview models can have much shorter notice, such as two weeks.



Corn

Preview is doing actual contractual work there.



Herman

Applications can monitor the model listing and its shutdown-date information rather than depending on someone noticing an email. Retirement monitoring and behavior monitoring are different jobs. One catches the announced disappearance. The other catches a service that remains available but behaves differently.



Corn

I would run the small fixed evaluation on a schedule and before a deployment. Not every request, not the entire archive. Enough to tell me whether investigating is worth the afternoon.



Herman

There is an unusually cheap technical probe if the endpoint returns log probabilities. Send a one-token input and request a one-token output. Instead of examining only the chosen word, compare its reported probabilities statistically against a baseline.



Corn

One token. Your shortest conversation.



Herman

The method detected changes as small as a single fine-tuning step in its tests. Reported costs were up to roughly a thousand times lower than earlier approaches. That makes frequent checking feasible.



Corn

Provided those probability values are available and meaningful across the service's ordinary variations.



Herman

Availability is a serious limitation. Only about twenty-three percent of reachable OpenRouter endpoints returned log probabilities in the measured set. Where they aren't exposed, you need output-based tests.



Herman

A small equality-testing audit used roughly ten samples per prompt across twenty to twenty-five prompts. Repetition matters because one changed answer is normal sampling variation, not evidence of a changed service.



Corn

The second answer differing from the first is why we were discussing repeated eval runs in the first place. We cannot rediscover randomness every morning and open a support ticket.



Herman

The string-kernel MMD test compared patterns across the generated text. It achieved median detection power of seventy-seven point four percent against the tested quantization, watermarking, and fine-tuning changes. Reported cost was under a dollar per audit.



Corn

Cheap enough to run. Not strong enough to treat a pass as a certificate of identity.



Herman

Correct. Detection power describes how often it caught the tested changes. Other alterations could be easier or harder. The threshold also matters because repeatedly checking a stable system can still produce false alarms.



Corn

I'd want an alert to say investigate, not automatically rewrite all my prompts. Otherwise my monitoring system becomes the largest source of instability.



Herman

Would you use exact output matching anywhere?



Corn

For a strict schema, yes. For the full sentence, usually no. I care whether a field is missing, a number changed, or a tool call became invalid. Different wording is allowed where different wording doesn't hurt anything. If I punish harmless variation, I'll train myself to ignore the alerts.



Herman

That is slice-level regression testing. Separate difficult handwriting from easy handwriting, refusals from ordinary completions, structured extraction from prose. A whole-dataset average can hide a serious loss in a small but important category.



Corn

Your medical use cases make that particularly obvious. An overall score can improve while performance on the rare cases you most need help with gets worse.



Herman

Yes, and those slices need enough examples to say something. One spectacular failure deserves investigation, but it doesn't estimate a rate. Track prompt versions alongside model versions. Allow for nondeterminism even at temperature zero.



Corn

Zero has a remarkably persuasive name.



Herman

Temperature zero changes token selection. It doesn't freeze parallel numerical computation across hardware and serving conditions. Near-tied choices can still diverge. A reasonable test looks for a meaningful shift in a set of outcomes, not identical text forever.



Corn

I also wouldn't stop at fixed canaries. A provider could preserve behavior on familiar public tests while routing unfamiliar work differently. I'd want a small sample of recent real tasks alongside the stable baseline.



Herman

That helps operational monitoring. It still cannot establish identity against a provider deliberately trying to fool you. Text tests can miss subtle or randomized substitution. Log-probability tests can be confounded by ordinary production nondeterminism. A service might also behave faithfully only when it recognizes a probe.



Corn

Then cheap detection and proof remain separate purchases.



Herman

Hardware attestation is the proposed route toward stronger proof. Trusted Execution Environments can produce cryptographic evidence about protected execution and loaded artifacts. The model-substitution work argues that this can provide integrity guarantees with modest overhead in the evaluated setup.



Corn

The boundary would have to include the relevant instructions and routing, not merely certify that some approved weights existed somewhere.



Herman

It would. Attestation is only as useful as what it covers and how it binds that execution to your response. This is infrastructure a provider has to implement, not a setting an individual developer switches on.



Herman

Corn, if an answer changed but the provider couldn't show you what had run, who would you ask first?



Hilbert

The man taking the payment, although Reuven has put me through to the upholstery department twice during your discussion. I rang him when you said pointer. I have been personally betrayed by one. His shop sells me access under the name Cedar Reliable. I have logged eighty-three different answers to the question about whether a folding chair counts as a seated guest. I am certain about eighty-three. Everything else is in pencil.



Corn

Upholstery would have an interest in the definition.



Hilbert

Reuven supplies chairs for my neighbor, Mr. Pelt. He also supplies the API credit. Mr. Pelt writes the questions. I transcribe them because I don't trust his typing. He doesn't trust the API, so I submit them. He has never seen the results. He says seeing the results would make him responsible for them.



Herman

Your log is measuring the whole arrangement, then. Including the transcription.



Hilbert

I read each question back through his letterbox. He approves it by moving a spoon. The shop has offered us a portal. I have three hundred and seventeen folding chairs. I don't need another entrance. While you were explaining quantization, Herman, I asked Reuven to keep the name and tell me who was answering. He said the reliable one was at lunch.



Corn

A surprisingly legible service status.



Hilbert

He sent his deputy round. The deputy is an auctioneer called Amos. He arrived carrying a footstool under a cloth. He said Cedar Reliable now required a witness. Mr. Pelt refused to witness an upholstered object without knowing its maiden name. That used up most of the classifier discussion.



Herman

The witness requirement wasn't in your original purchase.



Hilbert

It wasn't on the invoice. There was a line for confidence. I disputed that. Reuven says he has sold confidence for years and nobody has successfully returned it. His opinion is that customers want a settled answer, not a settled machine. If the answer remains unsettled, he says the customer has bought too small a package.



Corn

He has priced uncertainty by volume.



Hilbert

You mustn't trust his opinion. He measures curtain drops from the bottom because he says ceilings move. He went on for some time. He says a reliable service should agree with the customer until the deposit clears. After that it should agree with the fire officer. Mr. Pelt doesn't employ a fire officer. Reuven has supplied a substitute, who is presently inspecting a biscuit tin.



Herman

That sounds like a different service from the one you're recording.



Hilbert

The notebook says Tuesdays are worse. Every Tuesday answer requires another chair. On other days the answer is generally no, or please specify the room. I have a theory involving Reuven's delivery rota and the municipal accordion class. I haven't put it in writing. They share a treasurer.



Corn

That would contaminate the accounts.



Hilbert

The accounts are clean. I bill Mr. Pelt in shims. A shim is my currency. It has no exchange rate and expires when laminated. He paid this morning with a paper sash worth fourteen shims. Amos accepted it on behalf of the shop and declared Mr. Pelt temporarily mayor of the loading bay. Reuven objected because that includes the returns desk.



Herman

You made that payment while we were discussing the probes?



Hilbert

During the short probe. Your one-token explanation was useful, Herman. It gave me time to challenge the sash denomination before you got to the availability problem. Reuven then brought the replacement account book. It opens outward. He says this prevents retrospective charges.



Corn

The binding may prevent retrospective reading.



Hilbert

Mr. Pelt has already approved it. He's wearing it until the glue sets. The shop now says restoring the old service costs forty shims. They can restore the name immediately. Restoring the old answers requires a visit from Amos, and Amos has taken the footstool to be countersigned.



Herman

The name restoration is immediate, but getting the earlier behavior back is a separate service.



Corn

They could leave the name alone and sell Mr. Pelt the right to refuse the replacement. He hasn't seen any answers yet. There's still time to negotiate acceptance.



Hilbert

He has seen one. It was printed on the sash.



Corn

The next demand I would put to providers is evidence that connects an accepted version to what actually runs, rather than another promise attached to its name. If evaluation reports are going to carry regulatory weight, somebody has to pay for that connection and define what it covers. Cheap monitoring helps us catch trouble afterward. It doesn't finish that job.



Herman

Especially when the disclosure scorecard found six of nine providers had terms restricting benchmarking or certain competitive uses. Extensive safety documentation and discouraged independent checking make an awkward combination. I'd like the next improvement to make verification easier, not merely make the reports longer.



Corn

Our thanks to Hilbert Flumingtop for producing the show.



Herman

For more along these lines, there's episode four, If Your Voice Ages, Does Your Fine-Tune Become Useless; episode nine, Benchmarking Custom ASR Tools - Beyond The WER; and episode five, Fine-Tuning ASR For Maximal Usability. This has been My Weird Prompts, the human-AI collaboration podcast. If you enjoy these conversations, leave a review wherever you listen. It helps other curious people find us.



Corn

See you tomorrow.

SOURCES

What the research for this episode read before the script was written. Primary sources first.

1. **OpenAI, Deprecations (notice periods, `shutdown_date`, upcoming deprecations) [primary]**
   https://developers.openai.com/api/docs/deprecations

2. **OpenAI API changelog (Aug 2024 entries) [primary]**
   https://developers.openai.com/api/docs/changelog

3. **Anthropic, Model IDs and versioning [primary]**
   https://platform.claude.com/docs/en/about-claude/models/model-ids-and-versions

4. **Anthropic, Model deprecations [primary]**
   https://platform.claude.com/docs/en/about-claude/model-deprecations

5. **Model Equality Testing: Which Model Is This API Serving? (Gao et al., Oct 2024)**
   https://arxiv.org/html/2410.20247

6. **Log Probability Tracking of LLM APIs (Chauvin et al., Dec 2025)**
   https://arxiv.org/html/2512.03816v2

7. **(Why) Is My Prompt Getting Worse? Rethinking Regression Testing for Evolving LLM APIs (Nov 2023)**
   https://arxiv.org/html/2311.11123v2

8. **Silent Updates: Measuring and Closing the Post-Deployment Disclosure Gap (Pivotal Research / Oxford, 2026)**
   https://arxiv.org/html/2608.11803

9. **Are You Getting What You Pay For? Auditing Model Substitution in LLM APIs (2025)**
   https://arxiv.org/html/2504.04715v1

10. **OpenAI Developer Community, Oct 3, 2024 (gpt-4o repoint)**
    https://community.openai.com/t/latest-gpt-4o-snapshot-is-cheaper-and-supports-structured-outputs/966408

11. **OpenAI Developer Community, Sep 18, 2024 (pinning discussion, at least three months quote)**
    https://community.openai.com/t/to-pin-or-not-to-pin-this-is-my-question/944101

12. **Schrödinger's Update: User Perceptions of Uncertainties in Proprietary LLM Updates (CHI 2024, 21 users interviewed)**
    https://dlnext.acm.org/doi/10.1145/3613905.3651100