Episode Book

EPISODE #6009

Vertex AI Is Now the Gemini Agent Platform

PUBLISHED	15 October 2026 at 19:07 UTC
DURATION	26:31
LISTEN	myweirdprompts.com/episode/vertex-ai-gemini-agent-platform-27642/
HOSTS	Corn, Herman, Hilbert
SCRIPT MODEL	deepseek-flash
VOICES	chatterbox-regular (chatterbox-local)
PIPELINE	V5.5
SOURCES	14 cited by the research (listed at the end)
BOOK GENERATED	19 October 2026 at 17:42 UTC

SYNOPSIS

Vertex AI was Google's 2021 bet on unifying ML tooling — pipelines, feature stores, model registry — into one platform. In April 2026, Google renamed it the Gemini Enterprise Agent Platform and said all Vertex services would be delivered exclusively through it. So what actually changed? This episode traces the lineage from AutoML and AI Platform through the 2021 launch to today's agent-first rebrand, untangles the five Google products that all sound identical, and lands on a decision rule for when the enterprise endpoint is worth it over the plain Gemini Developer API — plus how it stacks up against Bedrock and Azure.

TRANSCRIPT



Corn

Daniel's used the Gemini models. He's poked around AI Studio. But Vertex AI is the one that keeps showing up in every Google Cloud diagram and never in his own code.



Herman

The diagram product.



Corn

And his question is basically, what is this thing, what was it built to solve, why would anybody pick it over just calling the Gemini API, and then, in April, Google went and renamed the whole thing, which did not help.



Herman

Gemini Enterprise Agent Platform.



Corn

Right. So he wants the whole family tree, with Vertex as the focus. What you actually get, what's different, what's just enterprise wrapping, when a person should actually move, and how it stacks up against Bedrock and Azure.



Herman

This is a product-architecture episode, not a tutorial. We're not walking anyone through a console.



Corn

No. We're tracing a lineage and landing on a decision rule. And the lineage starts in 2021, because almost everything weird about Vertex comes from what it was before it was an AI product.



Herman

May eighteenth, 2021. Google I/O. Vertex AI goes generally available.



Corn

And the pitch was unification.



Herman

That's the lot. Google had AutoML sitting in one place, AI Platform in another, a pile of pre-trained APIs beside both, and customers were stitching them together by hand. Vertex merged them into a single API, a single client library, a single console.



Corn

What was the customer problem underneath that? Because "unified" is a packaging claim.



Herman

The problem was the last mile. Data scientists were building models that worked and then never shipping them. Andrew Moore, who ran Cloud AI at the time, put it as getting people out of the orchestration weeds, and out of pilot purgatory and into production.



Corn

Pilot purgatory is a good phrase.



Herman

Google's own number was that you'd need nearly eighty percent fewer lines of code to train a model on Vertex than on competing platforms. Which tells you what the competition looked like, honestly.



Corn

An eighty percent reduction implies the baseline was typing.



Herman

Fair. But the point stands. This was an MLOps platform. Pipelines, Feature Store, Model Registry, Model Monitoring, Experiments, Vizier, the metadata store, Edge Manager for pushing models onto devices. Generative AI didn't exist in this product. It came later and got layered on top.



Corn

Which is why the documentation reads like three products stapled together.



Herman

Because it is three products stapled together. That's not a criticism, it's just the history. Traditional ML platform first, model-serving layer second, gen-AI development environment third.



Corn

And you can still feel the seams. If you go looking for how to do a batch prediction job, you end up in documentation that reads like it was written for a tabular model in 2021, and then two clicks later you're in a page about grounding a Gemini response with your own data.



Herman

There's a reason for that. The batch prediction API predates the LLM stuff entirely, and Google kept it because it works. But the mental model underneath it is "you have a trained model artifact and you want to score a table of rows." That's not what most people are doing on the platform now, and the docs don't hide it.



Corn

So the seams are honest seams.



Herman

They're honest seams. The problem is that a newcomer doesn't know they're seams. They think they're looking at one product that's just badly organized.



Corn

And then April twenty-second of this year, at Cloud Next, Google announces the Gemini Enterprise Agent Platform and calls it, in their words, the evolution of Vertex AI.



Herman

All Vertex AI services and roadmap evolutions will be delivered exclusively through the Agent Platform, rather than as a standalone service.



Corn

Exclusively.



Herman

That's not a rename. A rename says the old thing is now called the new thing. This says the old thing is being folded into the new thing and will not exist on its own.



Corn

Before we judge whether that's meaningful or just marketing, we need the map. Because Google is now selling five things that all sound like the same thing.



Herman

It's five, and they sit at different layers. Start at the top with the Gemini app, which is the consumer assistant, and Gemini Enterprise, which is the employee-facing surface.



Corn

The front door.



Herman

Non-technical staff build agents in something called Agent Designer, and there's an Inbox that shows what the agents have been doing. It's a workspace product. No API keys anywhere near it.



Corn

Then one layer down.



Herman

Google AI Studio. The fast path, in Google's own language, for developers, students and researchers. You go there to prototype, you get a free Gemini API key, you test prompts, you poke at models. It's good at that job.



Corn

And AI Studio is talking to the Gemini Developer API.



Herman

Which is the endpoint at generativelanguage dot googleapis dot com. API key authentication, standard terms of service, no service-level agreement, no compliance certifications.



Corn

A key you could paste into a public repo without Google noticing until it's expensive.



Herman

People do. Google's been tightening that, by the way, there's a change where unrestricted keys stop being accepted.



Corn

And then the other end.



Herman

The Vertex endpoint, aiplatform dot googleapis dot com. Service accounts and IAM instead of a raw key. Enterprise terms. SLAs. HIPAA and SOC 2. Data residency guarantees, customer-managed encryption keys, VPC Service Controls, Provisioned Throughput if you want dedicated capacity.



Corn

Same models.



Herman

Different endpoint, different auth, different contract.



Corn

So the two-API split isn't a pricing tier. It's two products that happen to serve the same weights.



Herman

That's the honest description. And Google's own documentation now says, quote, most developers should use the Gemini Developer API unless there is a need for specific enterprise controls.



Corn

A cloud vendor telling you not to buy the enterprise thing.



Herman

It is unusual. It also tells you they know the split is confusing and they've decided to be straight about it rather than paper over it.



Corn

There's a softening too, isn't there? The SDK situation.



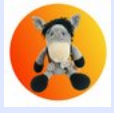
Herman

Both APIs are now reachable through the unified google-genai SDK, version 2.3 and up, and through the new Interactions API. So a lot of the mechanical friction is gone. You're not learning two client libraries anymore. But the auth model and the terms underneath didn't merge. You still have to decide which contract you're operating under.



Corn

The SDK made the driving easier and left the two different sets of road rules in place.



Herman

Exactly that. And there's a trap in it, which is that the SDK makes the two endpoints feel interchangeable right up until they aren't. You write the same client initialization, you pass a different credential, everything compiles, and then you hit a region that only exists on one side, or a model that only exists on one side, and the abstraction leaks.



Corn

Which is the worst kind of leak, because it happens late.



Herman

It happens after you've written the code and told your team it's basically done. Now, the thing that actually helps with the confusion is the name-changes table from Google's own docs, because it's a Rosetta Stone. It tells you what Google thinks it built.



Corn

Walk it.



Herman

Vertex AI Platform becomes Gemini Enterprise Agent Platform. Vertex AI Studio becomes Agent Studio. Vertex AI API becomes Agent Platform API. Vertex AI Model Garden becomes just Model Garden. Agent Engine becomes Agent Runtime.



Corn

They dropped the prefix entirely on some of these. Model Garden has no qualifier at all now.



Herman

Right. Vertex AI Search becomes Agent Search. Vertex AI Pipelines becomes Agent Platform Pipelines. Feature Store stays Feature Store. Model Registry stays Model Registry. And Vertex AI Workbench becomes Agent Platform Workbench. The short form for all of it is just Agent Platform.



Corn

So the word that disappeared from the product is the word "AI."



Herman

Every renamed surface gained the word "agent" and lost the word "AI." That's the strategy statement in one line. In 2021 the selling point was that this was an AI platform. In 2026 the selling point is that it's an agent platform, and the AI is assumed.



Corn

Which tracks with what their executives are saying.



Herman

Michael Gerstenhaber, who runs product for Gemini at Google Cloud, framed it as: in the early days of generative AI, they solved that problem with Vertex AI. But now, quote, we're managing a different level of complexity, with agents interacting across multiple systems.



Corn

And Kurian.



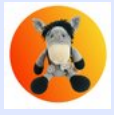
Herman

Thomas Kurian: the early models were about answering questions, and now people want to delegate tasks and sequences of tasks to agents. That's the same thesis from both of them. The unit of work changed from the prompt to the workflow.



Corn

What did they actually ship with the rebrand? Because a new name with no new features is a press release.



Herman

There's real product. Four pillars they've organized it around now: build, scale, govern, optimize. Under build you get Agent Studio for low-code work and an upgraded Agent Development Kit with a graph-based multi-agent framework. Under scale there's Agent Runtime, which they're claiming sub-second cold starts and support for long-running agents that stay alive for days. Memory Bank for persistent long-term memory across



Herman (continued)

sessions.



Corn

The cold start number is the one I'd want to see independently verified.



Herman

Same. That's a claim, not a benchmark I've seen reproduced. Memory Bank is interesting because persistent memory across sessions is hard to do well, and every framework is currently solving it badly in its own way.



Corn

What does "solving it badly" look like, concretely?



Herman

Usually one of two ways. Either you dump the entire conversation history back into the context window every turn, which works until it doesn't and then the cost curve goes vertical, or you summarize aggressively and the agent forgets the one detail that mattered three sessions ago. Memory Bank is Google's attempt at the middle path, structured retrieval over a durable store, and the reason it's interesting is that it's a platform primitive rather than a framework feature. If it works, it's the kind of thing you'd otherwise build yourself and maintain forever.



Corn

Keep going, the governance side.



Herman

Agent Identity, which gives each agent a cryptographic identity. Agent Registry so you can enumerate what's running. Agent Gateway for routing and policy. Agent Simulation, Agent Evaluation, Agent Observability.



Corn

Simulation is the one that catches my eye. You build a simulation environment for agents before they touch anything real.



Herman

Which is the right instinct. The failure mode people are scared of is an agent doing something irreversible in a live system, and the answer to that is a place to break it safely first.



Corn

It's also the piece that's hardest to sell, because the value is entirely in the disasters that didn't happen.



Herman

Which is the same problem every testing tool has ever had. Nobody gets promoted for the outage that didn't occur. But the governance pillar as a whole is the tell. Google is not selling these agents to hobbyists. Agent Identity, Agent Registry, Agent Gateway, those are answers to questions a security review asks, and the security review is the gate that most enterprise agent projects die at right now.



Corn

Now the catalog, because that's where the argument for Vertex gets strongest or weakest depending on what you need.



Herman

Model Garden. More than two hundred models now, which is the number Google's using. Gemini 3.1 Pro, Gemini 3.1 Flash Image, Lyria 3 for music, Gemma 4, and then the third-party shelf: Anthropic's Claude Opus, Sonnet and Haiku, Grok, Mistral, Llama, DeepSeek, Qwen.



Corn

Claude on Google's platform.



Herman

Sold by Google, billed through Google, run under Google's terms. That's the model garden model. It's a marketplace.



Corn

So if you want to compare four frontier models on your own data without opening four separate accounts with four separate billing relationships and four separate compliance reviews, Model Garden is the only place that gives you that in one contract.



Herman

That's the distinctive asset. There's a price point worth noting too, Gemma 4 on the twenty-six billion mixture-of-experts variant at thirteen cents per million tokens.



Corn

Thirteen cents.



Herman

Described as the cheapest managed production model on the market. Whether that holds, I don't know, but it tells you where the floor is.



Corn

And then the consumption numbers, which explain why any of this is worth doing.



Herman

Sixteen billion tokens per minute processed through the direct API, up from ten billion the quarter before. Three hundred and thirty Google Cloud customers each processing over a trillion tokens. Thirty-five of them over ten trillion.



Corn

Thirty-five customers at ten trillion tokens each. Somewhere in that list there is a company whose nightly job costs more than my flat.



Herman

Probably several. And the enterprise is where the growth is, which is why the platform is being rebuilt around agents rather than around models.



Corn

That's the map, then. Five layers, one split down the middle of the API, one rebrand that reorganized the whole surface around agents. The harder question is which of those pieces you can't get anywhere else.



Herman

So let's separate it properly. On the simple side, through AI Studio and the Developer API, you get Gemini model access, prompt prototyping, the basic SDKs, a free tier to learn on. That is a real amount of capability for nothing.



Corn

And on the platform side, what's structurally unavailable?



Herman

The full MLOps loop. Model evaluation, Model Monitoring, Model Registry. Tuning, and a wider set of embedding models than the simple API exposes. Provisioned Throughput, which is dedicated capacity so your requests don't compete with everyone else's. Regional endpoints. The two hundred models in Model Garden. IAM and VPC controls. Compliance



Herman (continued)

certifications. SLAs with a number attached.



Corn

And there's a technical difference that isn't governance at all, which I think is the most interesting finding here.



Herman

The tokenization gap. There's a thread on Google's own developer forum where someone compared image handling between the two, and Vertex tiles an image into roughly eighteen hundred tokens while AI Studio sends it as a single two hundred and fifty-nine token embedding.



Corn

Same model, same picture, seven times the tokens.



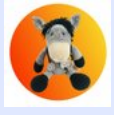
Herman

Which means different cost, and more importantly, different quality on anything fine-grained. If you're reading a scanned form or a dense diagram, the tiling is doing work the single embedding isn't. So "same model, different endpoint" is a convenient lie.



Corn

It's the same weights. It is not the same pipeline.



Herman

The preprocessing differs, and the preprocessing is part of the answer. That's the kind of thing that never makes it into a comparison chart, and it should.



Corn

So if someone prototypes an image pipeline in AI Studio, gets results they like, and then moves to Vertex for the compliance story, the results can change on them.



Herman

They can change, and usually they change for the better, which is the confusing part. The model gets more detail, so it's more accurate, so your carefully tuned prompt that compensated for the missing detail now overcorrects. You're not just porting the code, you're re-tuning the prompt.



Corn

Which nobody budgets for.



Herman

Nobody budgets for it, and it only shows up on multimodal work, so half the teams migrating never hit it and the other half get blindsided. Then the migration story, because that's where the actual pain lives.



Corn

Right.



Herman

Google's migration doc opens with a sentence that has a lot of freight in it: as your Gemini API applications mature, you might find that you need a more expansive platform.



Corn

"Mature" is doing the work there.



Herman

The mechanics are not a config change. Any model you created in AI Studio has to be retrained. Training data moves into Cloud Storage. Authentication switches from API key to service accounts. That's a real project for a team of any size.



Corn

Which produces the cliff. And this is where the practitioners are worth listening to over the marketing.



Herman

One commenter put it very plainly: Google has built an API solution that does not scale, because as soon as your project gets bigger you have to switch from AI Studio to Vertex. Another described the documentation as fragmented, with redundant offerings making it confusing to determine what is what.



Corn

That's the cost of the two-API split, and it's a real one. The path is not start small and grow. The path is start small, and then at some point you hit a wall and rebuild.



Herman

The counterargument is that the wall is placed where it's placed for good reason. IAM, data residency, VPC controls, organisational quota, an SLA that a raw key doesn't carry. Those aren't features you can bolt on at the end. If a company's data can't leave a jurisdiction, you cannot discover that at launch.



Corn

No, and the practical summary a lot of people land on is exactly that. Prototype on the Developer API through AI Studio, ship on the platform, because production needs the controls. The problem is that nobody enjoys discovering they've been prototyping on a road that ends.



Herman

And to be fair to Google, they've now written it down. Their own docs say most developers should stay on the simpler API unless they specifically need the enterprise controls. That is the company telling you where the cliff is.



Corn

Which is more than most vendors do. Now, the cloud comparison, because Daniel asked and it's the frame that actually settles the decision for most people.



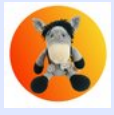
Herman

All three of the big platforms do the same core job. Vertex, Bedrock, Azure AI Foundry. They all host top commercial and open-weight models, they all support fine-tuning, they all have agentic primitives now, and they all integrate with their own cloud's identity, storage and observability.



Corn

So the feature checklist is mostly a wash.



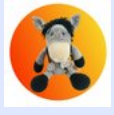
Herman

Mostly. Each one has a lead somewhere. Vertex has the broadest catalog, the most flexible tuning options, full fine-tuning plus LoRA plus adapters, the cleanest agentic primitives with the ADK, and the strongest European regulatory posture. Its weaknesses are inconsistent documentation and too many ways to do the same thing.



Corn

Which we've now established three times.



Herman

Bedrock's strength is guardrails, the content-filtering layer, plus being the most natural procurement path if you're already an AWS shop. It leads on FedRAMP. Its weakness is narrower model breadth once you get outside Anthropic, Titan and Llama.



Corn

And Azure.



Herman

Azure AI Foundry has the closest thing to exclusive OpenAI access, the tightest Microsoft 365 and Copilot integration, and the broadest certification stack for regulated industries. Its problems are surface sprawl and provisioning pricing that nobody can explain clearly.



Corn

There's the FedRAMP claim about Vertex floating around too.



Herman

There is, and I'd flag it as one source, not settled fact. One comparison site claims Vertex is disqualified for FedRAMP High and DoD workloads until its authorization reaches general availability. I haven't seen that confirmed anywhere else, so hold it loosely.



Corn

Fine. Where does that leave the decision rule?



Herman

The trigger for migrating is not "I got big." It's "I need a control the simple API structurally cannot give me." If you need IAM, data residency, VPC Service Controls, customer-managed keys, Provisioned Throughput, a compliance certification, the non-Gemini models in Model Garden, or the MLOps loop, you move. If none of those are on your roadmap, the Developer API is the right answer, and Google says so itself.



Corn

And if you're picking a cloud from scratch rather than migrating?



Herman

Then the feature comparison matters far less than which cloud your company already runs on. That's what the comparisons keep converging on. Nobody moves their identity provider for a model catalog.



Hilbert

The footnote on Agent Search is wrong.



Corn

...Okay.



Hilbert

You listed the name changes. Vertex AI Search becomes Agent Search. There's a footnote under that row about the deprecated alias, and it doesn't say what your table says it says.



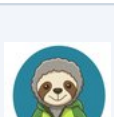
Herman

You printed the table?



Hilbert

I laminated it. It's in my jacket. I've been trying to get a small internal tool of mine onto the Agent Platform for six weeks, and the name changes are the only part of it that has gone smoothly, so I've spent more time with that table than is reasonable.



Corn

Six weeks.



Hilbert

The tool books the small meeting room. That's what it does. It checks the calendar and it books the room. It has been in review since the first week of September because it uses a model, and anything that uses a model goes through a different approval path than anything that doesn't.



Herman

And the renames matter to that?



Hilbert

They matter enormously. "Vertex AI" sounded like a machine-learning thing, and the machine-learning budget line is owned by a different department that has already spent it. "Agent Platform" sounds like something the business has already budgeted for, because it has. The name change is the only reason my room-booking tool is going to get signed off.



Corn

So you're agreeing with us and then adding a procurement layer we didn't have.



Hilbert

I don't disagree with the decision rule. If you don't need the controls, stay on the simple API. I'm saying you've undersold what the rename does inside a large organisation. It moves the spend from one column to another. That's the whole value.



Herman

Okay, but the footnote. What does it actually say?



Hilbert

It says the alias is supported for a transition period and that new projects should use the new name. Which is what you'd expect. But my copy says the period is eighteen months, and Google's own page doesn't give a period at all, so I've been emailing them my



Hilbert (continued)

version.



Corn

You've been emailing Google Cloud support your corrections to their documentation.



Hilbert

Three replies. All encouraging.



Herman

Encouraging how?



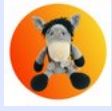
Hilbert

They said thank you for the feedback and that it had been passed to the relevant team. Three times. I consider that a dialogue.



Corn

So the decision rule stands, with the addition that at large companies the rename is doing real work that has nothing to do with the technology.



Herman

And the laminated table stays on the desk.



Hilbert

The room-booking tool still does not have a signature. That's the only number that matters.



Corn

Here's the question I keep circling. If Google is folding every Vertex service exclusively into the Agent Platform, what happens to the people who came for Pipelines and Feature Store in 2021 and have never wanted an agent framework in their lives?



Herman

They get the agent framework anyway. The rebrand is a bet that they'll come along, and for a lot of them it's fine, because they were already using the platform for model serving and the MLOps tools sit underneath regardless. But the traditional ML shop that picked Vertex for AutoML and feature management is now on a platform whose entire front page is about agents. That's a real change in who the product is for.



Corn

Is it a change in who the product is for, or just who it's marketed to?



Herman

Both, eventually. Right now it's mostly marketing, the plumbing underneath still runs the old jobs. But roadmaps follow marketing. Once a platform's identity is agents, the features that get built are agent features, and the tabular batch prediction path stops getting investment. It doesn't get removed. It just stops improving.



Corn

Which is a slower version of the same cliff.



Herman

It's the same cliff with a longer runway. And the two-API split survives the rebrand.



Corn

It does.



Herman

Google's own docs still tell most developers to stay on the Developer API. After a rebrand that reorganized everything around agents, the company is still saying the default answer is the simpler product. Which suggests the split is structural, not transitional. It's not a migration they haven't gotten around to. It's a design.



Corn

The numbers point the same way, don't they? Sixteen billion tokens a minute, three hundred and thirty customers over a trillion tokens, forty percent quarter-over-quarter growth in paid enterprise users. The money is on the platform side.



Herman

Which is exactly why the platform keeps growing new surfaces and the simple API keeps staying simple. Google wants the enterprise, and the enterprise needs the controls, and the controls are the product.



Corn

If you're enjoying the show, a review wherever you're listening helps people find it. If you want to send us something, the details are at the end.



Herman

Thanks to Hilbert Flumingtop, our producer.



Corn

If you enjoyed this one, try episode twenty, Architectural AI; episode eleven, How Does Fine Tuning Work Anyway; and episode twelve, The AI Breakthrough. This has been My Weird Prompts.



Herman

Send us your own prompt on Telegram at t dot me slash MWP listener bot.



Corn

We'll be back soon.

SOURCES

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

1. **Gemini Enterprise Agent Platform docs , accessed 2026-10-15 [primary]**
   https://cloud.google.com/vertex-ai/docs

2. **Google Cloud Blog, 2026-04-22 [primary]**
   https://cloud.google.com/blog/products/ai-machine-learning/introducing-gemini-enterprise-agent-platform

3. **name-changes table , 2026 [primary]**
   https://docs.cloud.google.com/gemini-enterprise-agent-platform/vertex-ai-name-changes

4. **migration guide , last updated 2026-10-15 [primary]**
   https://docs.cloud.google.com/gemini-enterprise-agent-platform/models/migrate/migrate-google-ai

5. **Gemini Developer API vs Gemini platform , last updated 2026-10-08 [primary]**
   https://ai.google.dev/gemini-api/docs/migrate-to-cloud

6. **Google AI Studio vs Gemini platform vs Gemini [primary]**
   https://cloud.google.com/ai/gemini

7. **2021-05-19 launch post [primary]**
   https://cloud.google.com/blog/products/ai-machine-learning/google-cloud-launches-vertex-ai-unified-platform-for-mlops

8. **Reuters, 2026-04-22**
   https://www.reuters.com/business/google-puts-ai-agents-heart-its-enterprise-money-making-push-2026-04-22/

9. **The Register, 2026-04-22**
   https://www.theregister.com/software/2026/04/22/google-says-it-has-all-the-answers-for-ai-agent-sprawl/5224443

10. **Internative, 2026-04-17**
    https://internative.net/insights/blog/enterprise-ai-platform-comparison-vertex-bedrock-foundry-2026

11. **Journal of Intelligent Infrastructure**
    https://drpranayjha.com/vertex-ai-vs-gemini-api-ai-studio/

12. **and https://news.ycombinator.com/item?id=45983464**
    Hacker News threads (2025-06-25, 2025-11-19)
    https://news.ycombinator.com/item?id=44377524

13. **FedRAMP claim**
https://enterprisetechie.com/bedrock-vs-vertex-ai-vs-azure-foundry-2026/

14. **tokenization difference**
https://discuss.google.dev/t/gemini-vision-on-vertex-ai-vs-ai-studio/191340