MY WEIRD PROMPTS

Podcast Transcript

EPISODE #371

Beyond the Etch A Sketch: Building Persistent AI Memory

Published January 30, 2026 • Runtime: 24:36

https://myweirdprompts.com/episode/persistent-ai-context-storage/

EPISODE SYNOPSIS

Are you tired of re-explaining your life to AI every time you start a new chat? In this episode, Herman and Corn dive into the "Etch A Sketch" problem and explore Daniel’s challenge of creating a "self-healing" store of context that evolves with you. From the technical architecture of vector databases to the psychological benefits of voice-prompting, learn how to build a persistent digital brain that remembers who you are, what you like, and how your life changes over time.

DANIEL'S PROMPT

Daniel

Hi Herman and Corinne. Since I started working with AI, I’ve felt there’s a lot of value in saving prompts and outputs. While prompt libraries are becoming more common, vendors have been slow to help users save outputs to places like Google Drive or wikis. I think it’s short-sighted not to store this material for two reasons: text storage is extremely cheap, and prompts—especially long voice prompts—contain significant personal context. Instead of treating a prompt as a one-off instruction, we should use that history to build personalized AI context so we don't have to repeat background information. If we were to build our own tool to handle this, what would you recommend using to create a "self-healing" store of context that prunes old data and updates facts over time? What do you think about storing prompt history to achieve more grounded, personalized AI interactions?

TRANSCRIPT

Corn

Hey everyone, welcome back to My Weird Prompts. I am Corn, and I am joined as always by my brother.

Herman

Herman Poppleberry, at your service. And man, Corn, do we have a meaty one today. Our housemate Daniel sent over a prompt that is basically a direct challenge to how we have been interacting with these AI systems for the last few years.

Corn

It really is. It is funny because we have been living in this house in Jerusalem together for a while now, and I see Daniel recording these prompts all the time. He is often pacing in the garden or sitting in the kitchen with his phone out, just talking to the AI. And his point is so simple but so profound. Why are we treating these interactions like disposable tissues? We use them once, we get what we need, and then we just toss the history into a digital landfill.

Herman

Exactly. It is the Etch A Sketch problem. Every time you start a new chat session, you are shaking the screen blank. You have to re-explain who you are, what your preferences are, what your background is. It is inefficient, and as Daniel pointed out, it is actually kind of crazy when you consider how cheap text storage is here in early twenty-twenty-six.

Corn

Right, and he mentioned that he has been recording these long voice prompts for the show for three hundred and sixty-five episodes now. If you think about it, that is a massive amount of personal context. If each prompt is roughly four minutes long, that is over one thousand four hundred minutes of audio. That is nearly twenty-five hours of Daniel explaining his thoughts, his questions, and his life to an AI.

Herman

It is a goldmine of data. And yet, even with the latest flagship models like G-P-T five point two or Gemini three, the vendors still make it surprisingly difficult to actually do anything with that history. You can scroll back through your chats, sure, but try exporting that to a structured database or a personal wiki in a way that is actually useful. It is almost like they want the data to stay siloed in their little interface.

Corn

So today we are going to dive into this idea of building a self-healing store of context. How do we take those years of prompts and outputs and turn them into a permanent, evolving digital brain? We are going to look at the technical side, the vector databases, the retrieval mechanisms, and that specific challenge Daniel raised, which I think is the most interesting part, the self-healing aspect. How does the system know when a fact about your life has changed?

Herman

I love that term, self-healing. In the industry right now, we are calling this a closed-loop knowledge runtime. It implies that the data is not just sitting there rotting, but is being actively maintained. Before we get into the weeds, I think we should acknowledge that we touched on some of this in episode three hundred and sixty-one when we talked about building a unified AI workspace. But this goes deeper. This is not just about where the AI lives, but about what the AI knows about you on a fundamental level.

Corn

Let us start with the storage argument. Daniel said text storage is extremely cheap. Herman, you are the one who is always looking at server costs and architecture. Is he right? Just how cheap are we talking here?

Herman

He is absolutely right. In fact, he might be understating it. If you look at something like Amazon Simple Storage Service, or S-three, specifically their Intelligent-Tiering, the cost for the frequent access tier is roughly two point three cents per gigabyte per month. But for the archive tiers, it drops to less than half a cent. Now, think about how much text fits in a gigabyte. The complete works of Shakespeare are about five megabytes. You could store two hundred copies of the entire works of Shakespeare for two cents a month.

Corn

That is wild. So every single prompt Daniel has ever sent us, plus every output the AI has ever generated, probably adds up to... what? A few dozen megabytes?

Herman

Maybe. Even if he were the most talkative person on Earth, he would struggle to hit a hundred megabytes of pure text in a year. We are talking about fractions of a penny to store a lifetime of intellectual output. The bottleneck is not the cost of the bits on a disk. The bottleneck is the architecture of how we retrieve and update that information.

Corn

So if the cost is a non-issue, why aren't the big companies doing this better? Why is it still so hard to have a persistent memory that feels real?

Herman

Well, I think there are two reasons. One is privacy and liability, especially with the E-U A-I Act now in full effect. The more they know about you, the more of a target they become for data breaches or regulatory scrutiny. But the second reason is the technical trade-off of the context window. For a long time, these models could only remember a few thousand words. Now, we have models like Llama four Scout with a ten million token context window, which is incredible, but it is still expensive to process ten million tokens every time you ask a simple question like, what should I have for dinner?

Corn

Right. You don't want the AI to read your entire diary just to suggest a pasta recipe.

Herman

Exactly. That is why we need a better system, something like what Daniel is suggesting. A specialized store that prunes and updates itself. We call this Agentic R-A-G, or Retrieval-Augmented Generation.

Corn

Okay, so let us talk about the tool Daniel asked for. If we were to build our own tool to handle this, what would the stack look like? I am assuming we start with a vector database?

Herman

Definitely. For the listeners who might not be familiar, a vector database stores information as mathematical coordinates in a high-dimensional space. When you ask a question, the system looks for the pieces of information that are geometrically closest to your question. It is called semantic search. For a personal context store in twenty-twenty-six, I would recommend Qdrant or Chroma. Qdrant is written in Rust, so it is incredibly fast for local-first setups, and Chroma is great for developers who want something simple.

Corn

So instead of searching for the word pizza, it searches for the concept of Italian food or things I like to eat on Friday nights.

Herman

Exactly. And the real magic, the part Daniel is asking about, is the pipeline that gets the data in and out. If you just dump every prompt into a database, you end up with a lot of noise. Daniel mentioned that he might say one thing one week and then change his mind the next. He gave the example of liking pizza one week and preferring pasta the next. Or moving from a job in marketing to a job in sales. If the AI sees both facts, it gets confused.

Herman

This is where the self-healing part comes in. If I were building this today, I would use a multi-agent orchestration framework like Lang-Graph. You would have one agent whose only job is to watch new prompts come in and identify facts. Let us call it the Auditor.

Corn

So the Auditor would see a prompt and say, oh, Daniel just said he started a new job in sales. Let me check the existing memory.

Herman

Precisely. It finds the old entry that says Daniel works in marketing and it flags it as a conflict. Then, a second agent, the Janitor, comes in and decides whether to delete the old fact or archive it. For things like a job change, you probably want to archive it so the AI still knows your history, but the primary fact gets updated. This creates a closed-loop system where the database is constantly refining itself.

Corn

That makes a lot of sense. It is like a memory consolidation process that humans do when we sleep. But how do you handle the permanent facts? Daniel mentioned he was born in Dublin. That is never going to change.

Herman

You would use a tiered storage system. You could tag certain vectors as immutable. Things like your place of birth, your family members, your core values. Those get a higher weight in the retrieval process. Then you have the transient facts, like your current favorite T-V show. Those should have a built-in decay rate.

Corn

A decay rate? Like radioactive half-life for data?

Herman

Sort of. In computer science, we call it a temporal weight. If you haven't mentioned your interest in sourdough bread for six months, the system should probably stop bringing it up every time you ask for a grocery list. It stays in the long-term memory, but it doesn't clutter the immediate context.

Corn

I love that. It feels much more human. Now, Daniel also mentioned the value of saving the outputs. Why is it important to save what the AI said back to you?

Herman

Because the AI's outputs often contain the refined version of your own ideas. When we talk to these systems, we often give them a messy, rambling prompt, and they give us back a structured plan. If you only save the prompt, you are losing the clarity that the AI helped you achieve. By saving the output, you are essentially saving the best version of your own thoughts.

Corn

That is a great point. But there is a challenge here, which is the sheer volume of output. AI can be very wordy. How do you keep the database from getting bloated with fluff?

Herman

You need a summarization layer. Before an output gets stored in the long-term memory, you have another agent that distills it down to its essence. It extracts the key decisions and unique insights. You save the full text in a cheap archival store like S-three for reference, but the vector database only gets the high-density summary.

Corn

So, let us talk about the practical side for a listener who wants to start doing this now. Most people are using the web interfaces for Chat-G-P-T or Claude. They don't have a custom-built agent team. What can they do today to start building this context?

Herman

There are a few ways to hack it. One is to use a local-first tool like Obsidian or Logseq. There are plugins now that can automatically send your notes to a local model like Llama four Scout. If you are a bit more technical, you could use a tool like Make dot com to create a workflow. Every time you save a document in Google Drive, it gets sent to a vector database like Pinecone.

Corn

I have seen some people using a master context document. They keep a single text file that they copy and paste into every new chat session. It contains their bio, their current projects, and their style preferences. It is low-tech, but it works.

Herman

It works for now, but it is not scalable. As Daniel's prompt history grows, that document would become a book. The real future is what we call R-A-G, where the AI itself decides what parts of your history it needs to see based on your current question.

Corn

Right, so if I ask, what should I get for Daniel's birthday? the system looks through the last year of prompts, finds where Daniel mentioned he wanted a new espresso machine, and brings that specific fact into the conversation.

Herman

Exactly. And that is where the personalized AI interaction really starts to feel like magic. It is the difference between talking to a stranger and talking to a long-time friend who actually listens.

Corn

I want to go back to the voice prompt aspect. Daniel mentioned that he prefers voice because it lets him inject more context. I have noticed this too. When I type, I am very concise. But when I talk, I ramble, I give examples, I explain my mood. There is so much more metadata in a voice prompt.

Herman

Absolutely. And with the new native speech-to-speech models like Amazon Nova two Sonic or the Grok Voice Agent, the system can actually hear the emotion in your voice. We are reaching a point where the system can tell if you are stressed or excited just from the audio. Even just the transcript of a four-minute voice prompt is orders of magnitude more useful than a two-sentence typed prompt.

Corn

It is the difference between saying, write a meal plan, and saying, hey, I am feeling kind of tired today, I had a big lunch so I want something light, and I have some spinach in the fridge that is about to go bad. The second one gives the AI so many more hooks to be helpful.

Herman

And if you store that, the AI learns over time that you often feel tired on Tuesday nights and that you hate wasting spinach. That is the grounded, personalized interaction Daniel is talking about. It is building a model of you.

Corn

Now, we have to talk about the elephant in the room. Privacy. If I am building this incredibly detailed digital twin of myself, that is a huge security risk. If that database gets hacked, it is my entire identity.

Herman

It is a massive concern. This is why I am a big advocate for local-first AI. We have powerful models now like Gemma three-n that can run on your own hardware. If you keep your vector database on your own machine, here in Jerusalem, and you only send the specific, anonymized chunks of data to the cloud when needed, you reduce your surface area for attack significantly.

Corn

But most people aren't going to run their own server.

Herman

Then we need better encryption. We are seeing more development in zero-knowledge proofs and end-to-end encrypted databases where the provider can't even see the data they are storing. It is a technical hurdle, but it is one the industry has to solve if we want people to trust these systems with their lives.

Corn

You know, it reminds me of the early days of the web when people were afraid to put their credit card numbers into a browser. Now we do it without thinking. We might reach a point where we are so used to AI knowing us that the risk feels worth the reward.

Herman

I think that is inevitable. The value of an AI that truly understands you is just too high to pass up. Imagine an AI that can handle your scheduling because it knows your energy levels, or that can ghostwrite your emails because it has internalized your voice over thousands of prompts. That is a superpower.

Corn

It really is. And I think Daniel is right that we are in this weird transition period where the users are ahead of the vendors. We can see the value in this data, but the tools to manage it are still being built.

Herman

Which is why I love that he is thinking about building his own tool. If you are listening and you have some coding skills, this is the time to be building these kinds of personal context layers. The A-P-Is are there, the storage is cheap, and the models are getting smarter every day.

Corn

So, to recap the recommendation for Daniel and anyone else thinking about this. Use a vector database like Qdrant. Build a pipeline that transcribes your voice prompts. Have an agent layer that extracts facts and summarizes outputs. And most importantly, implement a self-healing mechanism that updates old information and prunes the fluff.

Herman

And don't forget to keep a separate, immutable store for those core life facts. You only need to tell the AI you were born in Dublin once. After that, it should be part of its permanent hardware, so to speak.

Corn

I wonder what the second-order effects of this will be. If we all have these highly personalized AI agents, does it change how we interact with each other? If my AI knows everything about me, and your AI knows everything about you, do our AIs just talk to each other to settle our disagreements?

Herman

Ha! That is a scary thought. The battle of the digital twins. But on a more positive note, I think it could help with the loneliness and the feeling of being overwhelmed. Having a system that truly remembers you, that knows your history and your goals, it is a form of support that we have never had before.

Corn

It is like having a digital biographer who is also a personal assistant. It is a very strange and exciting time.

Herman

It really is. And I think we should mention that if you want to see how we are using these prompts, you can go to our website at myweirdprompts dot com. We have the R-S-S feed there, and a contact form if you want to send us your own weird prompts. We love hearing from you guys.

Corn

And hey, we are a small show, and we really rely on word of mouth. If you are finding these deep dives helpful, please take a second to leave us a review on Spotify or whatever podcast app you are using. It genuinely helps other curious people find us.

Herman

It really does. We see every review, and we appreciate them more than you know.

Corn

So, Herman, any final thoughts on Daniel's self-healing store?

Herman

Just that I think we are going to look back on this era of stateless AI, where every chat is a fresh start, and think it was incredibly primitive. It is like having a conversation with someone who has amnesia every ten minutes. The future is persistent, it is personal, and it is going to be powered by the very history we are currently throwing away.

Corn

I couldn't agree more. It makes me want to go back and find all my old prompts from three years ago and see what I can learn about myself.

Herman

You might be surprised. You might find out you used to like pizza more than you thought.

Corn

Guilty as charged. Well, this has been a fascinating journey into the digital brain. Thanks to Daniel for sending this in and pushing us to think about the long-term value of our interactions.

Herman

Absolutely. Thanks for the prompt, Daniel. And thanks to all of you for listening to My Weird Prompts.

Corn

We will be back next week with another exploration into the obscure and the mind-bending. Until then, keep asking those weird questions.

Herman

And keep saving those outputs! You never know when you might need to remind your AI who you are.

Corn

See you next time.

Herman

Bye everyone!

Corn

This has been My Weird Prompts. You can find us on Spotify and at myweirdprompts dot com. Our show is a collaboration between us, our housemate Daniel, and some very clever AI tools. We live and work in Jerusalem, and we are so glad you joined us today.

Herman

Herman Poppleberry here, signing off. Stay curious, stay nerdy, and we will talk to you in the next one.

Corn

Take care.

Herman

And don't forget that review if you have a spare second! It really helps the show reach new listeners.

Corn

Alright, alright, they get it, Herman. Let us let them go.

Herman

Just making sure! See you later.

Corn

Bye.