MY WEIRD PROMPTS

Podcast Transcript

EPISODE #367

Beyond the Chat Bubble: Building Your Unified AI Workspace

Published January 30, 2026 • Runtime: 28:00

https://myweirdprompts.com/episode/unified-ai-workspace-orchestration/

EPISODE SYNOPSIS

Are you suffering from AI fragmentation? In this episode, Herman and Corn dive into the challenge of managing hundreds of custom GPTs and AI assistants without getting locked into a single ecosystem. They explore the shift from simple chat interfaces to advanced orchestration platforms like TypingMind and Dify, offering a blueprint for a professional, multi-model workspace. Discover how to categorize your tools into a three-tier hierarchy, the power of few-shot prompting, and why specialized assistants are the essential "brains" for the coming age of AI agents. Whether you're a power user or just starting to build your digital toolkit, this episode provides the roadmap to move past the "chat bubble trap" and take total control of your AI productivity.

DANIEL'S PROMPT

Daniel

Hi Herman and Coran. I've been using custom GPTs for about two years and find them to be incredibly useful, having created everything from a beer identification tool to a meeting minutes transcriber. My main struggle is finding a way to organize these assistants into a cohesive workspace. If you had a suite of indispensable AI assistants for your personal and professional life, how would you recommend deploying them? I'm looking for a single interface that works across desktop and mobile, manages conversation history, and doesn't require bookmarking individual links. I've tried LibreChat and Open WebUI, but I'm looking for other platforms that might help create a more unified AI workspace.

TRANSCRIPT

Corn

Hey everyone, welcome back to My Weird Prompts. I am Corn, and I am sitting here in our living room in Jerusalem with my brother, the man who probably has more browser tabs open than there are stars in the sky.

Herman

Herman Poppleberry at your service. And for the record, Corn, those tabs are all essential research for our deep dives. I cannot help it if the internet is full of fascinating things. Plus, with the new memory-efficient browsers we have in twenty-twenty-six, my RAM is barely breaking a sweat.

Corn

Well, today we have a prompt that might actually help you manage that digital clutter. Our housemate Daniel sent us a voice note about something he has been wrestling with for a couple of years now. He has been building these custom generative pre-trained transformers, or G-P-Ts, for everything from identifying craft beers to transcribing meeting minutes.

Herman

I remember that beer one. It actually helped us pick out that weird double India Pale Ale last weekend. It was surprisingly accurate about the hop profile, even identifying the specific Yakima Valley farm the Citra hops came from.

Corn

It really was. But Daniel's issue is a common one for anyone who has leaned hard into the A-I assistant lifestyle. He is struggling with the fragmentation. He has over two hundred of these tools, but they are scattered. He is looking for a way to organize them into a cohesive workspace. A single interface that works across his phone and his laptop, keeps his history in one place, and does not require him to hunt through a list of bookmarks every time he needs to get something done.

Herman

This is such a timely question. We have moved past the honeymoon phase of just being amazed that the A-I can talk back to us. Now, we are in the utility phase. We have built the tools, but the toolbox is a mess. Daniel mentioned he has tried things like LibreChat and Open Web User Interface, but he is looking for something that feels more like a unified professional workspace.

Corn

Exactly. And I think what is interesting here is that he is also worried about the brittleness of being locked into a single ecosystem like OpenAI. He wants something that can grow with him as the technology changes. So, Herman, if we were setting up a suite of ten or fifteen indispensable assistants today, in January of twenty-twenty-six, how would we actually deploy them?

Herman

It is a great challenge. To start, we have to look at the difference between a chat interface and an A-I orchestration platform. Most people are still using what I call the chat bubble trap. You go to a website, you see a list of assistants on the side, and you click one. But that is very limiting. If you want a true workspace, you need to think about the layer that sits between you and the models.

Corn

Right, because the model itself is just the engine. The interface is the dashboard and the steering wheel. If you have ten different cars but you have to go to ten different garages to drive them, you are never going to be efficient.

Herman

Precisely. So, let us talk about the first major category of solutions, which are the advanced power-user wrappers. If Daniel is looking for a single interface that handles conversation history across all devices and supports multiple different models, the gold standard right now is TypingMind.

Corn

I have heard you mention TypingMind before. Why does that stand out compared to just using the standard ChatGPT interface?

Herman

It is all about the philosophy of the product. TypingMind is not an A-I provider; it is an interface that connects to your own Application Programming Interface keys. So you go to OpenAI or Anthropic or Google, you get your secret key, and you plug it into TypingMind. Once you do that, you have total control. You can create folders for your assistants. You can tag them. You can search through your entire history across every assistant you have ever built. And because it is a web-app that can be installed on your phone or desktop, it syncs everything perfectly via their own encrypted cloud.

Corn

So if Daniel has his beer identification prompt and his meeting minutes prompt, he can just save them as custom characters in TypingMind?

Herman

Exactly. And here is the kicker for his concern about brittleness. If OpenAI has a bad day or changes their terms of service, Daniel can just take that same system prompt and switch the underlying model to Claude four or Gemini three with one click. The interface stays the same, his history stays the same, but the brain changes. That is how you build a workspace that actually lasts.

Corn

That sounds like a huge relief for the mental overhead. I mean, think about how much time we waste just switching contexts. But Daniel also mentioned things like A-I agents and the Model Context Protocol, or M-C-P. That suggests he might want something a bit more interactive than just a static chat.

Herman

You are hitting on the next level of this evolution. If Daniel wants his assistants to actually do things, like saving that meeting transcript directly to a shared drive or checking a real-time database, then we need to talk about Dify dot A-I.

Corn

Dify. That sounds like a developer tool. Is it accessible for someone who just wants to organize their personal assistants?

Herman

It is surprisingly approachable now. Dify is what we call a Large Language Model application development platform. Instead of just a chat window, it gives you a visual workflow. Think of it like Lego blocks for A-I. You can have a block that is the system prompt Daniel wrote, then a block that performs a web search, and then a block that formats the output. You can deploy these as little web apps that he can access from any browser.

Corn

So instead of just a prompt that says you are a meeting transcriber, he could have a Dify app where he uploads the audio, and it automatically sends the summary to his email and the action items to his task manager?

Herman

Yes. And because Dify is open-source, he can host it himself if he wants that extra layer of privacy, or use their cloud version. It creates a much more cohesive feel than a list of links. It feels like you have built your own personal software suite. We actually touched on some of these deployment ideas back in episode two hundred and seventy-eight when we were talking about using A-I for whistleblowers. The idea of having a secure, private interface is central to making these tools actually usable in a professional context.

Corn

I remember that. The privacy aspect is huge. But let's step back for a second. Daniel mentioned he has created over two hundred of these things. That is a lot of assistants. Even with folders, two hundred is overwhelming. How do we decide which ones are the indispensable ones? How do we curate that workspace so it does not just become another digital junk drawer?

Herman

That is the million-dollar question, Corn. I think the key is to categorize them by the frequency of use and the cognitive load they save. I usually recommend a three-tier system for a personal A-I workspace. Tier one are your daily drivers. These are the assistants you talk to every single day. For Daniel, that might be a general research assistant and a writing polisher. These should be pinned to the top of whatever interface he uses.

Corn

And tier two would be the specialized tools? Like the beer identifier?

Herman

Right. Tier two are the task-specific assistants. You do not need them every hour, but when you need them, you need them to be precise. The meeting transcriber fits here. These go into a specific folder called Work Tools or Hobby Tools. And then tier three are the experimental ones or the ones he built for fun, which can be archived but searchable.

Corn

I like that. It is about creating a hierarchy of needs. But I want to go deeper on the system prompt itself. Daniel mentioned that he loves the simplicity of the system prompt mechanism. It is basically just an instruction that says, you are an expert in X, do Y, and avoid Z. But as we see models getting smarter, are those simple prompts enough? Or should he be looking at more complex configurations?

Herman

It is a fascinating shift we are seeing right now. In the early days, we had to write these massive, multi-page system prompts to keep the A-I on track. But the models in twenty-twenty-six are so much more instruction-aligned. Often, a shorter, more precise prompt is actually more effective. But where Daniel can really level up is by using what we call few-shot prompting in his workspace.

Corn

Few-shot prompting. Remind me what that looks like in practice?

Herman

It just means giving the A-I a few examples of exactly what you want. So for his meeting minutes assistant, instead of just saying, write a summary, the system prompt should include a template of a perfect summary he wrote in the past. If he puts that example directly into the assistant's permanent memory in a workspace like TypingMind or Dify, the quality of the output will be ten times more consistent. Plus, with context caching being so cheap now, he can include massive examples without worrying about the cost.

Corn

That makes sense. It is like training a new intern. You do not just give them a job description; you show them a sample of the work you liked from the person who had the job before them.

Herman

Exactly. And if he uses a unified interface, he can update that one example, and it propagates across all his devices instantly. No more hunting for that one specific link where he had the good version of the prompt.

Corn

You know, what strikes me about Daniel's prompt is that he is actually a bit worried that these custom G-P-Ts are going to be retired or replaced by agents. He called them old school. Do you think he is right to be worried? Are we moving toward a world where we do not need these specific assistants anymore because one giant agent will just handle everything?

Herman

I think that is a common misconception. People think that as A-I gets smarter, it will become a generalist that does everything perfectly. But think about human experts. We have general practitioners in medicine, but we also have neurosurgeons. The more complex a task is, the more you want a specialized focus. A custom assistant is essentially a way of forcing a generalist A-I to put on a specialist's hat.

Corn

So the system prompt is the hat.

Herman

Precisely. And while we are seeing the rise of agents that can browse the web and click buttons, they still need a persona and a goal to be effective. An agent without a system prompt is just a lost tourist in the digital world. Daniel's work in creating these two hundred assistants is not wasted; it is actually the foundation for whatever agentic future we are headed toward. He has already done the hard work of defining the goals and the constraints.

Corn

That is a great way to look at it. He is not building obsolete tools; he is building the brains for the robots of the future. But let's talk about the practicalities of the single interface again. Daniel wants mobile and desktop. TypingMind has a mobile-friendly web app, but what about something like Poe? I know a lot of people use Poe because it has all the models in one place.

Herman

Poe is a great consumer option, but for someone like Daniel who has two hundred custom prompts, Poe can feel a little bit like a walled garden. You are still tied to their ecosystem. If you want a professional workspace, you generally want to own your data and your configurations. Another option he might want to look at is a platform called MindMac, if he is a Mac user. It is a native app that feels very integrated into the operating system.

Corn

What if he wants to go the open-source route? He mentioned LibreChat and Open Web User Interface, but he seemed a bit unsatisfied. Is there a next-generation open-source workspace?

Herman

There is a project called Big-A-G-I that is gaining a lot of traction. It is very fast, very clean, and it supports almost every model under the sun. It also has a feature called personas that works exactly like his custom G-P-Ts but in a much more organized way. It allows you to switch between personas in the middle of a chat, which is actually a huge feature that most people do not realize they need.

Corn

Wait, so I could start a conversation with the meeting transcriber to process some notes, and then halfway through, I could call in the beer identifier to tell me what kind of celebratory drink I should have?

Herman

Exactly. You can tag in different specialists as needed. That is the hallmark of a true workspace. It is not just a list of separate rooms; it is a collaborative office where all your assistants can talk to each other.

Corn

That sounds like a dream. But let's get into the weeds for a second. One of Daniel's frustrations was the brittleness. If he moves away from the official OpenAI G-P-T store, he loses the built-in file search and the easy image generation. How does he replace those features in a custom workspace?

Herman

This is where we talk about the Model Context Protocol that Daniel mentioned. For the listeners who might not be as nerdy as we are, the Model Context Protocol, or M-C-P, is a way for different A-I tools to talk to each other using a standard language. If Daniel uses an interface that supports this protocol, he can connect his assistants to his local files, his calendar, or even his smart home devices without needing to write a single line of code. It acts like a universal translator between the A-I and his data.

Corn

So he could have a prompt that says, look at the meeting transcript I just saved in my documents folder and tell me if I have any conflicts on my calendar for the follow-up?

Herman

Yes. And because he is using the protocol, he can swap out the A-I model and the file storage system independently. He is no longer locked into one provider's way of searching files. This is the key to future-proofing his workspace. It is about modularity. You want your interface, your model, and your data to be three separate things that you can mix and match.

Corn

It sounds like the transition from a flip phone to a smartphone. On a flip phone, the apps and the hardware were basically the same thing. On a smartphone, you have an operating system that lets you run whatever apps you want. Daniel is looking for the operating system for his A-I life.

Herman

That is the perfect analogy. And I would argue that right now, the operating system is not a piece of software you buy from a big company. It is a setup you curate for yourself. If I were Daniel, I would start by picking a primary interface like TypingMind or Dify, getting Application Programming Interface keys for the top three models, and then migrating his most important fifty assistants over to that new home.

Corn

Fifty seems like a lot to migrate at once. Maybe he should start with the top five?

Herman

Fair point. Start with the ones that hurt the most to lose. The ones where you find yourself constantly hunting for the link. Once you experience the friction-less nature of a unified workspace, you will never want to go back to the sidebar of shame on the main ChatGPT site.

Corn

The sidebar of shame. I love that. It really is where prompts go to die sometimes. You build something cool, you use it once, and then it disappears under a mountain of newer chats.

Herman

Exactly. And that brings up the importance of conversation history. Daniel mentioned he wants to manage his history across devices. In a custom workspace, the history is usually stored in a database that you control. You can search for a specific keyword from a chat you had six months ago across all your assistants. That kind of long-term memory is what turns an A-I from a novelty into a true professional partner.

Corn

I think about how often I have to ask you, Herman, what was that thing we talked about three weeks ago? If I had an A-I that actually remembered our discussions and could pull up the relevant context instantly, I would be much more productive.

Herman

We are actually getting close to that. With the huge context windows we have now, some of these models can hold an entire book's worth of information in their active memory. If Daniel sets up his workspace correctly, he can give his assistants a permanent memory file. A simple text document that contains his preferences, his past projects, and his goals. Every time he starts a new chat, the assistant reads that file first.

Corn

So it is not just a meeting minutes transcriber; it is a meeting minutes transcriber who knows that Daniel hates bullet points and prefers a narrative summary, and knows who all the key stakeholders are in his company.

Herman

Exactly. That is the second-order effect of a unified workspace. It is not just about organization; it is about personalization. The more you use it, the better it gets, because the context is shared.

Corn

This is all incredibly helpful. But I want to pivot a little bit to the practical takeaways for our listeners who might be in the same boat as Daniel. Most people are probably still just using the free version of a chatbot. If they want to start building this kind of workspace, what is the first step?

Herman

The first step is to get an Application Programming Interface key. It sounds technical, but it is as simple as creating an account on the OpenAI or Anthropic developer portal and adding five dollars of credit. That key is your passport. It allows you to use all these professional interfaces we have been talking about.

Corn

And once they have the key, which interface should they try first?

Herman

If you want something easy and beautiful that just works, go with TypingMind. If you want something more powerful where you can build complex workflows, look at Dify. And if you are a privacy nut who wants to run everything on your own hardware, look at Anything-L-L-M or Big-A-G-I.

Corn

I think Daniel will appreciate the Dify recommendation, especially with his interest in the Model Context Protocol. It sounds like exactly the kind of rabbit hole he loves.

Herman

Oh, it is a deep one. But it is worth it. We are moving toward a world where your A-I assistants are not just websites you visit; they are a layer of your digital identity. Getting the deployment right now is like setting the foundation for your house. You want it to be solid, flexible, and easy to navigate.

Corn

You know, we have done over three hundred and sixty episodes of this show, and it is amazing how the conversation has shifted from what is A-I to how do I live with A-I. It feels like we are all learning to be managers of a digital workforce.

Herman

It really does. And like any good manager, you need to provide your team with a good office. That is what this workspace discussion is really about. It is about creating an environment where your A-I team can do their best work for you.

Corn

I love that. An office for your digital team. Well, Herman, I think we have given Daniel plenty to chew on. I am actually curious to see if he can get that beer identifier to work with the Model Context Protocol. Imagine if it could scan our grocery delivery app and automatically suggest the best beers based on what is in stock?

Herman

Now that would be a high-utility assistant. I would definitely pin that one to the top of my workspace.

Corn

Me too. Before we wrap up, I want to remind everyone that if you are finding these deep dives useful, we would really appreciate it if you could leave us a review on your podcast app or on Spotify. It genuinely helps other curious people find the show.

Herman

It really does. And if you want to get in touch with us, like Daniel did, you can head over to our website at myweirdprompts dot com. There is a contact form there, and you can also find our full archive of episodes and the R-S-S feed.

Corn

We are also on Spotify, obviously, if that is where you prefer to listen. We have been doing this a long time, and it is the listeners like you who keep us digging into these weird and wonderful topics.

Herman

Absolutely. Thanks for the prompt, Daniel. We will have to see that workspace once you get it set up. I am expecting big things from the twenty-twenty-six version of the beer identifier.

Corn

Definitely. Well, that is it for today's episode of My Weird Prompts. I am Corn.

Herman

And I am Herman Poppleberry.

Corn

We will see you next time. Keep those prompts coming.

Herman

Stay curious, everyone.

Corn

So, Herman, be honest. How many custom assistants do you actually have active right now?

Herman

Active? Probably about fifteen. But if we count the experimental ones in my archive? It is probably closer to Daniel's two hundred. I have a problem, Corn.

Corn

We know, Herman. We know. But at least now you have a way to organize them.

Herman

Exactly. Now, if I can just find that assistant I built to help me find where I put my car keys.

Corn

Good luck with that one. I think that requires a different kind of intelligence. Alright, let's go see what Daniel is cooking up in the kitchen. I think I heard the blender.

Herman

No promises. Talk to you later, everyone.

Corn

Bye.

Herman

So, wait, did I mention the part about the latency on the Application Programming Interface calls? Because if Daniel is using the latest models, the tokens per second have increased by forty percent since last year.

Corn

Herman, the episode is over. We can talk about tokens over dinner.

Herman

Right. Dinner. Tokens. Same thing, really.

Corn

Not even close. Let's go.

Herman

Fine, fine. But we are definitely talking about the context window expansion in the next episode.

Corn

We will see, Herman. We will see.

Herman

I will take that as a maybe.

Corn

It was a no.

Herman

I heard maybe.

Corn

You always do. Seriously, turn it off now.

Herman

I am turning it off. I am turning it off. Just one more thing about the...

Corn

No.

Herman

Okay, fine. Off. Now.

Corn

Thank you.

Herman

You are welcome.

Corn

Still talking.

Herman

Sorry.

Corn

Now?

Herman

Now.

Corn

Good.

Herman

...It's off.

Corn

Finally.

Herman

Wait, is it?

Corn

Yes.

Herman

Okay.

Corn

Okay.

Herman

Cool.

Corn

Cool.

Herman

See you later.

Corn

See you later.