# The perfect swirl: Gen AI meets traditional for better outcomes (Ep. 52)

[00:00:07 Weston Morris]

Welcome to the Digital Workplace Deep Dive, gen AI edition. I'm your host, Weston Morris. In preparation for this podcast, I did a Copilot search on how to make the best use of generative AI to improve my business. Then I did a follow-on about how many failures there have been with generative AI. I got a mix of answers. One was an article about dealing with not getting an expected ROI. A Gartner article says a third of all gen AI projects will be abandoned.

[00:00:26 Weston Morris]

On the other side, there was a Harvard Business Review article saying boost your productivity with gen AI and 101 tips in another article. So, lots of confusion about this. Our second podcast was on AI last year, we did our first on gen AI and we're really interested in what's happening in terms of return on investment. That's why I'm glad our guest Alan Shen, vice president of Solution Management for Digital Workplace Solutions here at Unisys, has carved out some time to speak with us today. Welcome, Alan.

[00:01:14 Alan Shen]

Hey, nice to meet you, Weston. Glad to be here.

[00:01:16 Weston Morris]

Alan, I know you've said — maybe I'm paraphrasing here — but in the past you've said that what's missing in generative AI is AI. I want to figure out what you mean by that — how AI plus generative AI are better together. I know we could quickly go down a rabbit hole here in having a very technical discussion. One of the other things you said is that enterprises should really be focused on outcome-based AI versus just doing it because it's the cool thing. So, let's talk about generative AI from that point of view, maybe look at some real-world examples, real problems and use that to see how gen AI might fit in and how AI might complement it. One example might be: imagine you are the CEO of a fast-food chain, and in every one of your thousands of restaurants is an ice cream machine. The crew is there, they're busy and there's a problem with the ice cream machine. How do they figure out the answer? Can they just go to Alexa or ChatGPT or Copilot? How does that work?

[00:02:21 Alan Shen]

That's probably a common thought. There's a bit of hype on generative AI that can answer everything and be the be-all and end-all, but the thing to remember is that a lot of what you experience in the consumer ChatGPT world was trained on public internet data coming from YouTube, Wikipedia, that sort of thing. So, if you ask a question about how to fix an ice cream machine, it's probably going to look at your department store ice cream machine and give you that answer, and that's not going to be what's needed to fix an enterprise ice cream machine, which is very specific. You could get a wrong answer and probably be a bit frustrated.

[00:03:04 Weston Morris]

OK. I was kind of anticipating where you might go with that, so I did a test to see if you were right. I asked the public Copilot that question: how do I repair a commercial ice cream machine? It's giving me quite a few different answers here. One, kind of obvious stuff, check error codes, read the manual. Love that. Try turning it on and off again. Another good one. Call a technician. That wasn't super helpful and there's this: providing a link to a YouTube video, showing some funny stuff about broken ice cream machines. So not super helpful. What is gen AI missing here to really get an answer to a question like this?

[00:03:44 Alan Shen]

In theory, what you might initially think is, if it's missing that data about that enterprise machine, the logical next step would be to train the LLM so that it has that information as part of its model. But as you expand this concept into other enterprise scenarios, it's pretty clear that as a corporate CEO, I don't want my company to have all that data and ask questions about my corporate proprietary ice cream machines. We really don't want to be training a public LLM with that data that's proprietary to my business.

[00:04:24 Weston Morris]

That makes sense. So maybe I'm going to say, how about this: what if I purchase or rent the LLM, put it in my own private instance maybe in the cloud, still a bit of private area, and then train it on how to fix an ice cream machine? Is that the right model?

[00:04:42 Alan Shen]

If money was absolutely no object, possibly. The thing is, training these LLMs to do a private island just for me is incredibly expensive. I think the stat is that for ChatGPT 3.5, which has been around for about a year now, OpenAI spent something like $100 million to

train that large language model. That's a lot of money to drop down to troubleshoot an ice cream machine. So probably not very workable.

[00:05:13 Weston Morris]

You're kind of limiting my choices here. I'm thinking maybe the next choice if I don't retrain the LLM to have this knowledge about ice cream machines. How do I get that knowledge associated with the LLM so I can ask questions about it?

[00:05:31 Alan Shen]

The good thing is that about nine months ago, it became clear that using LLMs in an enterprise space had a unique need of needing to fold in enterprise-specific data as part of that question-answer process. There's a mechanism called retrieval augmented generation where, as I ask a question, the process allows the engine to first retrieve a set of documents that are relevant to the topic I'm asking about. Hence the retrieval portion. And then augment the LLM just for that one question-answer session, just for that transaction, to say, "For this question, I want you to answer this question that the user asked but include these set of documents as part of your large language model data set and generate the answer." That way we get a bit of the best of both worlds. We don't have to retrain the whole LLM, but we can augment it with corporate-specific data and get an answer that is that humanized response without populating the data into the LLM and disclosing it to the wide public. That's the way to do it.

[00:06:48 Weston Morris]

OK, so you're combining the knowledge without putting it into the LLM but somehow linking it to it. I'd like to get into the details about how that works, maybe a little bit more in just a minute, but let's go back to what you said: outcome is the most important thing. So, the outcome here would be with the crew at your fast-food restaurant. They've got a problem with this ice cream machine. Now what is their experience like? If we've done what you just said, you take their knowledge, use RAG and integrate it with the LLM. What is their experience like?

[00:07:21 Alan Shen]

In that case, it's critical to make sure you have the right documents in the back end. For instance, if the enterprise has used version five firmware, but the documents are version four firmware, the LLM and the retrieval augmentation has no sense of right or wrong, it just knows this is the best answer. So, populating back-end data with the most up-to-date

correct information is important. The great thing about the LLM experience is it's a far more humanized experience. For instance, I may say, "My ice cream machine is not serving." The LLM may say, "In my whole set of restaurants, there are several different models of ice cream machines. " They may say, "I don't exactly know which model you have; you tell me," and the user might say, "Sure, it's model ABC." The cool thing about LLMs today compared to legacy chatbots is that when I say model ABC, it knows from the first question that I'm talking about an ice cream machine. When we present that to the LLM, we're presenting that in context with the first question and that's more how humans talk. There's a sense of context. There's a sense of, "Oh, yeah, you asked about that. Now I'm adding more detail." I don't have to have an experience where every question has to be absolutely, perfectly formed to get that absolute right response.

[00:08:56 Alan Shen]

Humans aren't necessarily good at entering every single detail. So, if you have that sort of experience, what ends up happening is the LLM — once it gets to the point where it has all the different parameters it needs — will actually return the exact set of steps needed to resolve the problem rather than having to read through the whole document. It might say, "OK, I've pinpointed. Do this step one, step two, step three." Maybe a picture is included in there. That's just a way better experience for the end user because they don't have to search articles and read an entire document.

[00:09:35 Weston Morris]

So, maybe to recap what I'm — to see if I'm following along Alan here. If I just start with an out-of-the-box LLM, it may have some idea about ice cream machines in general, but not the details that are needed for your restaurant and maybe even more specifically the fact that you've got a restaurant in Bogota, Colombia that has a certain brand ice cream and a certain model. And then you're in Seattle, so an ice cream machine there in a Seattle restaurant might have a totally different model and a different set of instructions. This LLM now is able to do a conversational, contextual based conversation to draw out exactly what information is needed so then it can search and then you've exposed this additional data about all these different models, that wasn't part of the LLM. Did I get that?

[00:10:25 Alan Shen]

That's right, absolutely. I'll add to that. It could be, for instance, that as a user, you didn't provide every single detail about the error code, and so maybe it gives you a set of instructions you try and you say, "Oh, that didn't work. It's giving you this error code," which

you didn't present to the LLM in the first place because you thought maybe the LLM didn't need that. Once it sees that error code, it says, "Oh, actually, now I found a different set of instructions, maybe in a different article," and presents that back to the user because you give it more detail and context. That flow, that interaction very much mimics what a user would have if they were talking with a live agent. You give it more detail and it gives you more information as you refine, and that is the superpower of LLMs and why it's so applicable to this sort of troubleshooting scenario.

[00:11:21 Weston Morris]

You said at the very beginning, retraining an LLM from scratch to get this ice cream machine data in there is not cost-effective. You leave it separate but somehow, you're connecting it. But that feels like a manual labor-intensive process. Can you explain a little bit more about how that works to get that knowledge associated with the gen AI?

[00:11:41 Alan Shen]

Without going into too many details, we have to remember that these large language models don't understand meaning in the same way you and I do. What ends up happening is a process called vectorization and embeddings, where you're basically taking all the data both within the large language model, but also that back-end data. When you retrieve those documents, there's a process where the best I could explain is: You know that scene in "The Matrix" where you have reality and all these green things, these numbers going down that represent reality? It's almost like that — you're basically digitizing all the different factors of different words in these articles. But decomposing it in a way that talks about location, time, color, and all these different dimensions. And saying, "Hey, this is what the meaning of this is." And we present that to the LLM. You're almost then talking in the language of the large language model so that it can incorporate that into that response retrieval, just like in "The Matrix." You have a digitized version of the real world. That's a very simplified version of what's happening in the background.

[00:13:02 Weston Morris]

I think we're kind of getting close to what you said at the very beginning when I was partially quoting you that gen AI is missing AI. AI plus gen AI together, that's really going to make them better together. Can you give me any other examples of how you might use AI to help gen AI provide a better answer over time?

[00:13:22 Alan Shen]

Sure. A couple come to mind. One, as I mentioned before, the retrieval augmented accuracy and that experience is only as good as the back-end documents that are provided that you're searching against. So, it's really important — this is where AI can come into play. This user is not the first user to encounter an ice cream machine issue, so one thing you can do is use AI to say, "Let me look at all the previous incidents where people asked about ice cream machines and make sure that the documents we're retrieving and presenting in that retrieval augmented data set are the right documents." And maybe we slough off the old documents that are for old versions. The more you can constrain that data set to the juicy articles that will give the best responses, the less likely it is you'll get what you've probably heard of, which is hallucinations where you're pulling in multiple articles, some of which are basically distractors.

[00:14:28 Alan Shen]

You're presenting that to the LLM, and it's going to just confuse it because now it has to disambiguate from the good articles and bad articles. Now, imagine you present both version four and version five articles all of a sudden. Now, that's something that it has to obtain from the user as an attribute. And that's just an added step of confusion that you could create even though version four is no longer in service. So that's a good example of how the back end can help make that low accuracy improve.

[00:15:01 Weston Morris]

You know, Alan, when you describe interacting with the service desk, I think there's a perception in service desk that it's just based on how people are measured. Service desk agents where they are often looking to get that initial fix, the quick fix which might just be a temporary fix, but, really, they haven't gotten to the root problem. Can AI help with that type of scenario?

[00:15:28 Alan Shen]

Absolutely. You made a really good point. As good as the experience is, and it is important to restore service, we have to remember that while this ice cream machine is down, the crew members are distracted and have to troubleshoot this rather than servicing customers. You have customers that have maybe placed an order that are waiting in line and holding up the queue. So, there's a revenue impact to that restaurant. This is an impacting issue, and the faster that we can resolve it, the quicker we can restore operation. So, there is an important benefit to the speed of an LLM accessing and getting to the right answer.

[00:16:05 Alan Shen]

But to your point, it'd be far better if we could avoid the issue in the first place, and that's where traditional or predictive AI really comes into play. For instance, if we take that same scenario, there's a way to look at the telemetry coming from the ice cream machine, just like it might with your PC. We've all had our PC get to a point where it's running really slow — it needs a reboot, or maybe it needs some driver updates, whatever it may be.

[00:16:35 Alan Shen]

The firmware on the actual machine may be exactly the same, whether it's the compressor, maybe it's getting some dust caught in it, it's running very hot. So that's where gathering telemetry from these devices and leveraging AI to say, "Hey, what's the point where we haven't hit a problem yet, but we're at the point where we're at risk?" We're witnessing the temperature of the engine go high. We're seeing some error codes, some warning codes come in from the system. That might be a good time to proactively call some services in or maybe run through some preventative maintenance when the crew is having some downtime to avoid that incident from happening during the busy lunch hour. And that's a great example of preventative AI matching with generative AI.

[00:17:26 Weston Morris]

I'm smiling just thinking about it. The more we can get to the root cause of a problem, that's more of a long-term fix. Rather than having that same simple fix being done every day, that's a much better resolution. I love that. So, we've talked here so far, Alan, about how we can use AI to improve gen AI. Is the reverse possible? Can we take gen AI as a way to enhance AI?

[00:17:52 Alan Shen]

Absolutely. If we stick with the existing example of an ice cream machine, let's walk through a realistic scenario where the restaurant is maybe not having too much traffic, and they report that there's an issue with the ice cream machine and it's a mechanical issue. It's not something you can just fix by doing a reset. One of the questions is about dispatching somebody. How urgent is it to get somebody on-site to fix that machine?

[00:18:23 Alan Shen]

This really depends not only on the fact that they don't have many customers now but also on whether or not there will be customers coming in soon. For instance, one of the use cases that you might think about is if this restaurant is located near a sports stadium here

in Seattle. Let's say here in Seattle. Let's say there's a big football or soccer game that's about to let out. It may be the case where that's actually a really urgent issue, not because you have customers immediately, but because there could be a crush of customers coming in 30 minutes or an hour. So that's actually a high priority incident. We just don't know about it yet. And this is where gen AI can really play an interesting role.

[00:19:05 Alan Shen]

You can imagine a public feed of information, whether it's news, sport events, traffic data, weather data coming into play, and we can use gen AI. Gen AI is really great at, because of the large language models, processing this open-ended stream of textual data. We think about it mainly in the GPT chatbot scenario, but it really has a lot broader applicability than just in the chatbot.

[00:19:34 Alan Shen]

So, it can consume all this freeform text and basically answer the business question of, “What's the priority of this incident?” by predicting through that text prediction whether there could be a crush of customers coming to that restaurant. From there, we could say, "Oh, actually, this is a P1 incident because of that sports game." They can get an agent dispatched at the tech on site, get that machine fixed as an urgent issue, and you're good to go.

[00:20:07 Weston Morris]

I think anything you do to prevent a Seahawks fan from not being able to get their ice cream is a good thing. We don't want that in the news.

[00:20:15 Weston Morris]

Now, Alan, the examples we've used here, obviously in the fast-food industry, devices that are specific to that, and great application of both AI and gen AI together that are going to make life better. Can you see this being used in other industries?

[00:20:33 Alan Shen]

Absolutely. If we abstract the notion of this actual machine being in the quick-serve space to any sort of device, whether it's in the health care system, transportation, or aviation, it can even be translated into PCs and compute devices. The reality is that for many customers, what they want and care about is the consistent, ongoing operations of their business. They don't want to be wrestling with unexpected events that are coming and disrupting their flow of business. So, we're seeing huge amounts of applicability of this gen

AI and AI, the pairings you sort of heard, where it can be predictive, where it can be reactive and where it augments the reactive accuracy.

[00:21:32 Alan Shen]

That has huge applications, and I think the exciting thing here is that from what I see, all of the experience and know-how we've seen in the information worker space is really transferable into line-of-business applications. We've seen that with this case of the ice cream machine, but it's really applicable to lots of other lines of business, too.

[00:22:02 Weston Morris]

I'm just going to wrap up with some takeaways here and see if I've got this right. Clearly, there's a need for and a desire for conversational access to dynamic knowledge we might say, right? And being able to get that in a friendly way and an accurate way and without hallucinations.

[00:22:18 Weston Morris]

Clearly, in an enterprise, it's not going to be expecting a public LLM to be able to give them these answers; we've got business-specific knowledge that only my business knows about. What you're recommending here is we marry that business-specific knowledge, tokenize it, expose it to the LLM, and then on top of that, you've said that the missing ingredient has been AI in gen AI, meaning that we can use AI to process that knowledge, to be able to access it, to get even predictive — add a predictive nature to the knowledge that we're looking at here to be able to solve these problems.

[00:22:50 Weston Morris]

I really like that. I think hopefully this has been helpful for our listeners and that's what you're saying is available today. But looking at the future, what might we still be missing, Alan?

[00:23:06 Alan Shen]

This area is evolving so fast. One of the topics that is being discussed a lot is — we've heard LLMs, which are large language models — but there's also this concept of SLMs, which are small language models, which you can imagine obviously have a much smaller data set that can be run more locally on, say a PC or an edge compute device. I think there's an interesting area that is going to be the next iteration of this that says, well, are there certain use cases where the right thing to do is actually build a small language model and invest — and certainly because it's a small language model, you're not talking about spending $100

million because you don't need to scour the whole internet. You have a much more focused problem: you need to build an LLM that can be run on a more local compute-type situation that doesn't have the cloud compute-type constraints you might have for other situations.

[00:24:27 Alan Shen]

There are interesting tradeoffs because one of the pros of a small language model is that you have much more control over the data set that you're going to code into that small language model. So, when you're doing your prompt engineering to make that response augmented generation, there's less noise that you have to steer the LLM to get the right answer because you've already, in some ways, filtered it to just the relevant data that you need. That's one of the pros of that. On the flip side, because it hasn't been programmed with the entire set of YouTube videos and Shakespeare and all that sort of stuff, its ability to have that humanized experience, that colloquial experience, may be reduced because it isn't armed with the entire volume of English language.

[00:25:10 Alan Shen]

So, I think those SLMs are probably more geared toward vertical applications that aren't necessarily directly exposed to end users because you have a situation. Maybe you need to do some genome programming, or in a medical scenario, you really want to have a very targeted use case. You don't need to expose this humanized experience. Rather, you know the exact question you want to ask, and you're just looking for it to crank out and focus on getting the best answer possible. That's where I think SLMs will start to be quite interesting.

[00:25:43 Weston Morris]

I'm going to pause here and write something down: Future podcast, SLMs versus LLMs, and when to choose each.

[00:25:54 Weston Morris]

Hopefully we'll be able to have a future podcast and invite you. Every time we talk, Alan, I think I get a little bit smarter. And I know our audience does as well. I really appreciate you giving me some time here today, as well as our listeners.

[00:25:57 Alan Shen]

Sounds exciting. Thanks so much. Been great to chat with you, Weston.

[00:26:12 Weston Morris]

Well, I've been chatting here with Alan Shen, vice president of Solution Management for Digital Workplace Solutions here at Unisys. This is the Unisys Deep Dive: AI edition. Thanks for listening.