

[Transcript] FYI - For Your Innovation / Unlocking the Power of AI in Gaming with Unity's Marc Whitten

Welcome to FYI, the four-year innovation podcast.

This show offers an intellectual discussion on technologically-enabled disruption, because investing in innovation starts with understanding it.

To learn more, visit arc-invest.com.

Arc Invest is a registered investment advisor focused on investing in disruptive innovation.

This podcast is for informational purposes only and should not be relied upon as a basis for investment decisions.

It does not constitute either explicitly or implicitly any provision of services or products by Arc.

All statements regarding companies or securities are strictly beliefs and points of view held by Arc or podcast guests, and are not endorsements or recommendations by Arc to buy, sell, or hold any security.

Clients of Arc investment management may maintain positions in the securities discussed in this podcast.

Hi everyone, welcome to another episode of For Your Innovation by Arc Invest.

A podcast on all things related to disruptive technologies.

I'm Andrew Kim, research associate covering consumer internet and fintech, and I'm joined by Nick Groose, associate portfolio manager.

Today we have the great privilege of speaking with Mark Whitten, president of Create at Unity.

Hi Mark, thanks so much for your time today.

Yeah, for sure, happy to be here.

And for our listeners, we think it'd be great if you can just start with an introduction and how you ended up where you are today.

So I'm Mark Whitten, and as you said, I'm the president of Create at Unity.

What Create is, is all the tools and services that we provide for creators to build in real time 3D, whether that's the Unity engine and editor, whether it's services like Unity gaming services, whether it's tools like Weta Digital, so that people can create games and other experiences in real time 3D.

I've been at the company about two and a half years now, and prior to that, I was most recently at Amazon as a vice president of a bunch of the devices work, but I spent the majority of my career at Microsoft, whereas one of the first employees on Xbox and spent 15 years building the original Xbox, Xbox 360, Xbox Live, and Xbox One on the platform side, sort of running a big chunk of those businesses.

So I spent a lot of time around games, I've spent a lot of time around 3D and developers and creators, and as far as what brought me to the company, I knew our CEO quite well for my time in Xbox, and we had about a 20 second conversation on the mission of Unity, which is we believe the world is better with more creators in it, and this opportunity that they saw, obvious to me, something that I care super, super deeply about.

But then as you got closer to the company, got to meet the extraordinary people inside of there, is like one of those things where I went from like, yeah, yeah, I'm happy to have a conversation, John, to I'm at the wrong place.

These are my people, and I need to get here as fast as possible, and I haven't looked

back.

That's awesome.

Thank you.

And obviously, Unity has been at the forefront of AI in the past couple of months.

That's obviously in the minds of many, many people, AI everywhere.

Are people talking about AI a lot?

They say, supposedly.

Yeah, it might be a big thing here.

And you guys talked about, I guess, Muse and Sentis as like the primary AI products, and as for our listeners, I think it'd be great if you can just provide us with an introduction to let's start with Muse, you know, what does it do today, and what use cases do you aim to support in the future?

Yeah, well, let me give you like a top liner introduction to both because they're related, and then we can talk about Muse in a little bit more detail.

Because frankly, it starts with the opportunity and the need.

It seems clear to us that over time, everything in every game, every quest, every boss, every object, every pixel, every system is going to be touched by AI at create time, creators using it deeply inside of their workflow, but also is going to be touched at runtime.

Actual inference happening on every pixel being displayed in every frame of a game or other real-time 3D experience.

And so as we started thinking about that, it became clear to us that what we needed to deliver was a platform approach for each half of that.

So how do we enable great AI workflows at create time to make it creators more productive, to give them more capabilities, and to make it easier for more creators to create?

And then also, how do we unlock the real power of AI at runtime when the game's actually being played?

So Unity Muse is our platform for creation side AI, and Unity Centus is our platform for runtime side AI.

When you look at the two segments, where are the most challenges when you say, you know, we're at X point today in this journey of AI being a part of every aspect of a game, which of the two segments poses the most challenge?

And also, where are we in that journey?

Like AI has had this huge run up in the last six, eight months, it feels like it's starting to seep into every sector, gaming being one of those, and there's so much innovation happening so fast.

It almost feels like, hey, we've made so much striding, but I'm curious, where are we in this journey, and then where are the challenges ahead?

Yeah, those are great questions.

And it's funny, I was sitting there saying, well, I don't know, which one is actually the bigger challenge?

Because they're kind of an orthogonal set of challenges, and I'm not sure if I could figure out which one's the right magnitude, because there's some really, really key problems that you're trying to solve that are a little different on each side of that, and they're

both important.

I think there's a lot more intention being spent on the create side.

When people talk about generative AI today, they generally are talking about how to use AI tools to create assets or help with coding or whatever those things are.

And I think, at least when we talk about this, people, when we talk about Unity Muse, people are like, oh yeah, yeah, that's exactly what I expect that Unity would be doing.

When we talk about Unity Sentence, they're like, oh my god, I didn't even think about this is what's going to be the implication of artificial intelligence as it applies to games and other real-time 3D experiences.

As for where we are, I think we're, first thing I'd say is AI has been used in a lot of ways.

We've used AI inside of Unity for a lot of our tools for many, many years.

Obviously there's been this sea change because of the availability of extremely large language models of techniques like diffusion and the availability of those in a lot of broad class that really brought generative AI to the forefront.

But in another sense, especially on the create time, this is kind of like a continuum from manual placing of pixels through procedural tools through full inference-based creative tools.

It's a little more analog than digital in going between those states.

A good example of a non-what people think of as AI tool today at Unity that serves a lot of the same purpose is something like SpeedTree.

I like to joke that actually most video games are just SpeedTree simulators because if you see vegetation in a video game, there's a really good chance that it was done by using SpeedTree as a tool, also in a bunch of film and other things as well.

Well, okay, no one's sitting around drawing individual trees and they're not even picking from a class of trees or whatever.

Instead what they're doing is they're defining what the environment is that they're trying to fill with vegetation and then allowing the smarts of SpeedTree to create variation, so that you walk through a forest that's been created inside of it and it looks real and it looks varied enough to not feel like you're in that old style of seeing sort of the same object over and over again.

That's procedural though.

It's not an AI tool.

Now you're moving more into where you're using these generative and inference-based techniques, but they've been heavily used for many years by artists, programmers, developers in real-time 3D.

Now, there's a bunch of challenges to now take it to the next level, which is what we're spending time on.

So, I guess the generation of environments becomes a lot easier by combining aspects of text to 3D models and also procedural generation as you've just mentioned, but I guess with respect to more complex dynamic malleable assets, like a human for example, obviously you can't just generate a holistic human.

You want it to move in the way a human would, and I'm just wondering, what are the puts

and takes with respect to training such a complex AI model that has to reflect how it should behave in any given environment, if that makes sense?

It totally makes sense, and there's several challenges there, and there's not one system of training that solves all problems.

In fact, part of the reason we think of Muse as a platform is in some level, what it is now and what it will become even more so over the future is really orchestration of AI techniques. What you want to do is unlock natural input, whether it's language or sketching or anything like that, to give input into what you're trying to accomplish and then allow a variety of AI, or in some cases procedural techniques, to then take over and help get you further along where you're trying to get.

In large environments, in very complex environments, there's a couple of big problems that we're spending time on.

The first is, there's actually a lot of research that's going on in, as you talked about, text to 3D, but one problem that often comes from a model is models can start to become better and better about generating a 3D mesh, but we call it this triangle suit, because what it is is a sort of undifferentiated set of triangles.

Where does the ground end and the chest begin?

Can you open the chest?

Typically today, that's a much harder problem, because you're actually forward-generating what the behavior of an actual object is in the environment, not just the generation of the triangle.

We're spending a lot of time on how do you think about problems like that.

The other thing is, what you're going to see is a lot of expert-trained AI systems for specific characteristics.

You talked about humans, which is a really, really great example.

We spent a lot of time, and in fact acquired a company called Zebra Dynamics that's really focused on ML and AI-based training to solve this problem.

The old way of creating a high-quality character was to create it and then you hand-rig every point that's potentially animatable and then hand-move them through every particular scene. It's extremely time-consuming, which is one reason why the quality of characters that you often see, especially in real time, has not increased at the same level that the performance capabilities of GPUs or your hardware has.

We take a different approach with Zebra.

What we've done is we've trained an AI model that understands a huge number of data points and tens of thousands of different emotional states around a character's facial animation and deformation.

Then we run it through an advanced deformation solver so that you can move to instantaneous inference of what the animation should be versus having to hand-rig it.

Something that might take a team of artists three months suddenly can take a server run about 10 minutes to get you back something that you can use in real time and interactively.

It's not pre-canned animations.

It's not pre-canned rigging or anything about that.

The fundamental is still going to come back to the same thing, which is there's not one

AI technique or AI system that's going to apply to all of these because they're all really, really complex behaviors.

Our focus is instead, how do you create orchestration in a platform layer so that we can plug in all of the expert AI systems in for a particular area?

You mentioned a really interesting stat there that three months to 10 minutes.

I want to zoom out a bit here and talk just about productivity gains at large.

When you look at the video gaming space, obviously, AAA games take years and years to develop.

I'm curious, what do you think are the areas to focus on when you talk about productivity?

Where are you hearing stories from customers already saying, we are just rocking and rolling now with all of these services you're putting out there, this is taking us half the time.

You mentioned three months to 10 minutes on this animation styling.

I'm curious, where are the other areas where you're seeing these productivity gains?

First off, we're still early in some of these tools rolling out.

We're learning with our customers and we'll continue to move them forward, but I'll step back a little bit of the curvature of the earth and then talk about that.

The truth is that we have a crisis of content creation.

It's not possible to make the amount of 3D content and assets that video games and other real-time 3D experiences demand.

When everybody was so hyped about things like the metaverse, one thing that many of us were talking about is like, this is going to be a bunch of really empty worlds because there's not enough content in content time, a thing that you often see in video games in particular, especially from larger games, is if you look at the announced video of a game and then the final game, they often ended up not necessarily achieving their vision.

Well, that's not about the tech.

It's actually about the content creation and the assets.

It's one of the things that takes the most time.

One of the reasons the games are delayed or become less ambitious or fail to reach their goals.

There's a desire for much more of that content.

We have to solve the productivity.

The interesting point about the three months to ten minutes is it's actually zero to ten minutes because everybody looks at the three months and say, we can't afford it, so we don't do it, or we're going to use a much worse model and it looks worse or whatever.

We're able to change the quality bar and also give them productivity gains.

On the asset side, we see this as two ways.

Number one, what we want to do is make sure that these tools enable a broader class of creators to have access to technology and become creators.

We believe the world is better with more creators than it is our core mission.

Number two, to create productivity gains for professionals so that they get their job done faster and they feel more in control of the end outcome.

This isn't about generate some art for me.

It's about give me a generation and then all of the knobs and the nodes that I need in order to turn that into something truly special for the game that I'm trying to build.

In the end, I think it will mean a significant more high quality content in significant more games.

I actually think it's going to mean more net creators and sort of be expansive for the industry, but also allow people to make bigger bets on the types of games that they want to be willing to try because they can get there a lot faster in prototyping to understand what the challenges are and these other areas because they have these techniques.

Wow.

Yeah, that's really fascinating to hear it.

It's an expanding market.

Embedding AI can actually expand the market.

You're talking about potentially even new content types as well because-

I strongly believe that's what's actually going to happen.

It doesn't mean there won't be changes and that people will have to learn new tools.

I actually think it will be expansive of the types of things that people are trying to accomplish as they go create and especially in real time in 3D.

So I guess you've iterated more content and more high quality content, but I guess looking at the other side of the equation just on the consumer front, given that people have limited time or limited resources, how do you think the market will evolve as it may look more competitive going forward for developers in general to get the game in front of consumers?

Yeah.

Gaming is already extraordinarily competitive to find and keep your audience.

It's also extraordinarily lucrative and so with or without AI, it is only going to continue to increase in the level of competition as people look at it and go try to do great things.

Now, to be clear, that's one of the longest uses of AI that we've had.

If you look at the ad monetization in Unity, the ad business is really about how do you help people find their customers to keep those customers engaged and then obviously be able to monetize their game so that they can be successful.

That's all based on using AI models and neural networks in order to better target so you find the specific player that would most like their game is most likely to stick around for day 7 or day 30 or whatever your retention metrics that you're looking at.

So AI will play a continuous role in doing that.

However, there's another trend.

It actually sort of leads into the other half of the AI stuff that Unity sent us that we've been mentioning, but I'll start it by saying the other fundamental part of that is all of these games are becoming what we call live games or live ops games.

It's not about the initial bits that you shipped, but it's about did I build this community and am I maintaining the game that I've built for them to sort of continue to be engaging and robust so that I hold on to them for a long period of time.

And I'm a big Marvel snap player as an example and every time a patch comes out, I'm sitting there saying, oh, I wonder how I'm going to change my deck and this is going to shake up the meta and let me keep playing it.

Well, the core reason they're doing that is they want to keep the game engaging for their

existing players as well as try to drive new players in.

One of the big opportunities in AI at runtime is to create deeper, richer living worlds that actually feel much more alive to the players inside of them.

And there's some key examples that we've been talking about.

If you saw a video of our alien that we did with the Sentis one, you can see a little bit of this of like an NPC that you can truly have a conversation with and sort of use that to sort of shape the game experience in a way that's very different than everybody else.

If you think of Skyrim, one of the best games ever, in my opinion, there's things like you'll go talk to someone and they'll say, I used to be an adventurer like you, but then I took an arrow to the knee.

And it's a total meme.

And the reason it's a total meme is because millions of gamers have heard it dozens of times because they don't, you know, going back to the content creation thing, you can't have every, you can't go create content for every guard to say something different to everybody every time.

Well, when you switch that out where you're still in control of the game world and its lore, but I might have an extremely different experience than you had.

In fact, maybe I talked this guard into like coming along with me or giving me some different information, then you can suddenly do like it's going to feel a lot more alive than anything that we have today.

And that's an example with NPCs, but that plays out actually in a lot of the systems that you'll think about inside of your games.

I guess just following up on intelligent NPCs and like how far are we from a fully interactive Skyrim and like what, like how big do the models have to be to be able to run that locally?

Yeah, it's a really good question.

And I think one of the things there's there's kind of an economic question and then there's a capability question.

I don't think we're very far away.

In fact, you can see lots of examples, including people hacking Skyrim to plug in LLMs and try them out and getting cool results.

Hacking, it's not not sort of the way that you think about a finished game yet.

But the first problem you have to solve is running a bunch of cloud inference can be very expensive.

It's one thing if it's at create time, because it's a relatively small number of people for for a defined period of time doing the creation.

If instead you said I have to pay every time you talk to an NPC as a cloud inference times number of concurrent users that I have and you have a successful game, well, that could actually the cost of that could actually scale in a way that's not just not manageable.

And so part of what we're trying to do a sentence is to make it easier for you to imagine using neural network based features at runtime with deterministic cost.

The other problem that so then you're like, oh, I'm just going to run it locally.

Well, the problem with running it locally is number one, the tech stack for most neural networks and the languages behind them, whether it's, you know, TensorFlow or PyTorch and

things like Onyx and things like that is not is designed for a very specific architecture.

Typically, it's designed for cloud based PCs.

And so the capabilities and both the hardware language support and all of those sorts of things don't exist broadly.

And so now you have to say only people on this particular device can play the game.

Well, game creators say that's too small of a market.

If I start, you know, sort of willing it down.

So what we're trying to do a sentence is to make it so that you can take any standard model, assuming it fits in, you know, has to fit in local memory, you can make one that's too big.

And we will make sure that works anywhere the unity runtime runs.

So we will take care to sort of transpile the PyTorch into whatever we need to locally to run on the GPU or CPU or the best possible performance that you can get.

So that suddenly says, as my design palette, I can imagine the model running locally.

So now I can start to design for it.

Then you get into the size of models.

There's a lot of research going on that says that you started specifically about intelligent NPCs. It's unclear that you need massively large models to make that happen.

You might have small models that work for most people.

And then maybe you go to the cloud for like some, you know, boss or the head wizard or whatever that you need more, you know, deeper inference from.

But the other thing I would say is it's not just about those types of features.

There's I believe that we're going to enter this golden age of inference based techniques used at runtime.

And many of those are going to be very small models because the other fundamental thing it's actually been interesting talking about AI runtime with game creators is there's one hard truth in game development.

You have 16 milliseconds to get your frame done and then it's time for your next frame.

You know, that's not much time.

So that often means what am I able to do?

Oh, I want to do this advanced lighting thing that I want to do.

Oh, I can't because I'm on a piece of hardware where I don't have enough time to run the math to render that lighting technique in my 16 mil, however many of the 16 milliseconds that I can spare for that particular lighting technique, because I have to do everything for the game, the whole simulation input, the entire thing and every 16 millisecond turns.

Well, inference means that you can take some of those problems and turn them into inference instead of instead of mathematical generation or it's different, it's different math, I guess, but use a use a model instead of doing direct calculation.

And what we're hearing is that people are super interested in having all sorts of models, models that do lighting, you know, to use that specific example.

Instead of calculating the physics, think about ray tracing.

That's a very high performance physics calculation to generate how does a photon get to my eyeball to turn many types of techniques into I'm just going to do a bunch of training.

And now with a very small model, because it's very purpose based, I can do extremely fast

inferences and actually create it as a feature that I can run across every platform that I'm trying to deliver.

Gotcha.

And do you think as we enter this like golden age of AI, as you've said, like, does this accelerate like growth in memory, do you think?

Like, because like if you think about the like iPhone, you know, unified memory, it's been growing for like the past decade, but then starting in the, I would say the mid 2010s, it kind of slowed down, right? And that, I assume, presents a bottleneck for more performant models running locally.

And I'm just wondering, like, is that like a meaningful constraint in your opinion in the longer term?

I think on a core RAM, probably, it probably will end up sort of helping drive additional increases there. The thing that is super clear is that there's going to be a lot of energy put into continued evolution of AI specific silicon, that is, whether it's a module, I mean, you know, use a GPU for a lot of the silicon, they're becoming more and more tailor made to run an inference versus run a render pass. And I think that will continue. And that will be on every SOC, you know, everybody's going to be thinking about like, what's their kind of inference engine that they're embedding in a piece of silicon. And that will then probably drive this cycle about like, okay, well, what does that change in terms of how we think about hardware cache or RAM size or other things to get the most out of that, just like GPU evolution has core GPU evolution for 3D graphics has for the last, you know, 20 years.

Mark, I want to change gears here a bit. And first, just congratulate Unity on the partnership with Apple. But my congratulations is also the question. And, you know, could you give us a bit of background

on the on this partnership with Apple? And, you know, what's happening now that they have the Apple Vision Pro out in the market, or, you know, it'll be coming later this year, early next year? Yeah, you know, we've collaborated with Apple for a very long period of time across many things. You might remember that when they originally launched their M1 silicon, Unity was there, not just the runtime capable of running on it, but the full editor already ready to go on M1 silicon. And obviously, we've worked very, very closely on iOS and iPhone for many years, huge, huge part of the gaming market. And so, you know, when we first heard, you know, what they were thinking about doing, we were really excited to see where they were going to go. They've really, I think, gone super deep into how do you think about spatial computing and what they've built inside of Vision OS. And it's been really exciting working with them on it. There were two things that we were really excited to do. The first one is, I'll call it Unity standard platform approach. Our goal is to make sure as much of Unity just works for you as you target whatever platform matters, so that we want to make it as easy as possible for the way that you think about building your game, your workflows and all that sort of stuff for you, then say, I want to now target platform X, Y or Z. So that's standard, make sure it's just easy to use Unity on day one of the launch. But the other thing that I think is very unique about Apple's approach to spatial computing and the Vision OS is shared mode in the experiences. And that meant that we needed to work quite closely with Apple in order to think about how to render, how to build and deliver multiple Unity experiences that can be resident and running at

the same time, whether they ever knew about each other or not, and that those experiences could be running next to an Apple experience or a Reality Kit experience. And that's different than what we've

done in any other platform. And I think it really, we're excited to see what creators do with that, because I think it creates this opportunity. You can imagine there's this crazy game or experience that I've built. And next to it is another experience that a video screen where you and I are chatting or sort of can see each other as we're doing it. And then maybe I have something off to the side that I'm just kind of slightly paying attention to. That's cool. And then what's also cool is you can start to imagine those experiences being able to interact with each other over time. And what we

tend to find is whenever there's this type of paradigm shift that game creators and real time 3D creators were like this tip of the spear of like figuring out kind of like what new sort of modalities or interesting ways that you could use that technology to create something fun or novel or different than what exists elsewhere. So yeah, we're super, super excited about it.

Yeah, it sounds amazing. I'm so excited for this collaboration. I have a follow up question and this ties it back to what we were originally discussing with Muse. Do you think as Muse scales, it will be possible for developers to deploy on a device like Vision Pro, even if they didn't design for that platform originally?

Well, I mean, that's always our goal to make that as easy, regardless of whether you're using AI or not, we're going to do everything we can to make it easy. I think there's two parts about that. And I think AI likely makes part of it easier. There's the technical part is my game capable of running on a particular platform. But then there's a second part, which is game design, which encompasses a lot, because it just turns out that different types of platforms lend themselves to different ways that you think about presenting information and taking input from the user. Think about a PC game or PC application, it's extremely information dense, because you have a large screen, you know, in one of things people learned when they first started doing things like game streaming of, let's say console games on a phone is, you know, it's really cool to play Assassin's Creed on a phone. There's a lot of text, like your screen is like covered with text, because it was designed to be on a 50 inch display, suddenly it's on a five inch screen you hold in your hand. Twin stick is different than mouse and keyboard is different than touch. And those aren't just how do you translate between one versus the other, they often can go really deeply into how you design the gameplay loop to be fun, because you could translate between mouse and keyboard to twin stick, but that doesn't mean that it feels great on

twin stick or that it's the right mechanic for you. And so that part AI will help because it will help you explore the space and it will help you get farther. But you will still have to fundamentally understand the type of game or experience that you're trying to build and how a user is going to interact with it.

That makes sense. Very excited to see what gets built for the Apple Vision Pro.

Yeah, me too. It's going to be really fun to watch.

I think your point on like shared experiences for the Vision Pro is really interesting, right, because at least from what I gathered from the Vision Pro demo, right, and it seems like the Quest Pro, right, sorry, the Quest is taking a very different approach to gaming compared to the Vision Pro, right, because

you have 2D in a spatial environment versus like full immersive 3D for the Quest. And I'm just wondering,

do you think those two experiences will like service different markets? Do you think like Vision Pro

eventually will kind of also collide with like that kind of full 3D use case? Like how do you think about like developers developing for either device?

What's exciting about Unity is that we end up, you know, working with a lot of different platforms. And

for us, we're always looking at how does that create more opportunity for creators? Because we think all

those spaces will get filled. If there's a high quality device and a company that's really out there sort of

building really cool stuff for that, then game creators will come in and fill out interesting use cases inside of it. And so I think that, you know, whether you're talking about a device like the Quest, or you're talking about Vision Pro, or you're talking about Xbox or a PC, all of them are going to have really, really robust ecosystems, and people are going to explore them. Now, if we talk about Vision Pro,

particularly, it's extraordinarily powerful device. And it will definitely have really rich 3D, full 3D experiences. It's more than capable of doing it. One of the things that was exciting in working with

Apple is just the amount of time that they spent on thinking about whether it's the sensors, the display

quality, or, you know, the power of their in silicon to drive, you know, really amazing results. So I think you're going to see some some very interesting stuff. And again, what will be exciting is when people start to use that kind of palette of toys that they have, I always think of it as how many crayons do they have in their crayon box to start, you know, building something, they're going to go do

something that none of us think that the device is actually built for. One of my favorite first examples on the iPhone when they launched the App Store was from a company called Smule, who built

this app called Ocarina, which was essentially, your iPhone's a flute, and you blow into the microphone, and then you use your fingers to fret. And the thing that's interesting about that is you know, you know, no one at Apple, no one at Apple thought that they had developed the Woodwind API.

And developers didn't eat weren't thinking about that. But suddenly they came out and did it. And you

could see everybody at Apple go, Oh, and you can see every other game creator go, Oh, and it's that way

that the space gets broader and broader, because people, it's actually, you know, probably telling a little excited about this kind of stuff. Like, I've been doing platforms for a long time. And I always like the questions when people ask me, what do you think people are going to really do with this? Like, I can tell you that whatever idea I have is so low on the creativity scale. Because the beauty of being a platform creator is you unleash the creativity of so many brilliant people to sort of really, you know, tease away and people do amazing things. There were people on the original PlayStation one, PlayStation one was out forever, it lasted a long time. They were people using the audio chip to run extra graphics, because they just wanted to do these crazy things. I mean, creators are inventive, they're brilliant, and they're going to do amazing stuff. If you give them new

techniques and new tools to play with.

And that's certainly possible now with the Apple Vision Pro and the Unity collaboration. And I think we'd be remiss to say, but the beta just opened, right?

That's right. We just started our first closed beta. We've had extraordinary interest since the WWDC announcement and continuing with people signing up, both in games and non games. And on Wednesday of

this week, we started putting our first closed beta. And now, obviously, our goal is to expand as soon as

possible to make it easy for everybody that's interested to target it. But it's really exciting to get started with our first creators on it.

Very exciting. Everything that Unity and Apple are doing in this collaboration. We want to end with just

one last question. We've talked a lot about gaming. Unity has exposure to non gaming segments. It's being used in many different industries. How do you think that this is going to continue to evolve?

How

do you think Unity, especially with AI now, is going to be used in some of these different industries that you're currently being used in?

Frankly, one of the main reasons that I joined was because of the massive opportunity and the continued growth of real time and 3D in non gaming use cases and industrial use cases and digital twins. It's a, I believe deeply that essentially every industry is going to be rebuilt around real time and 3D. And for a variety of really important reasons, we are 3D and real time people. And so whether it's medical or manufacturing or the way that I interact with my house or devices or anything else like that, 3D is a richer, deeper, more information full way for me to do some interaction that I'm trying to interact. And we're very, very focused on how do you make sure these tools that were pioneered in gaming and continue to be extremely robust in gaming are as useful and

that we expand through all the different types of use cases. AI, really, really critical. Obviously, inside of it, you go through that content creation problem we talked about. It's even more so outside of gaming where there's not maybe the same concentration of artists and 3D technical creators in a particular industry or a particular enterprise that they have to go through. So we're working very, very hard to make it so that it's easy for everyone to pick up and use. And frankly, it's the fastest growing part. You know, it continues to see extremely robust growth, whether it's in manufacturing, commerce, medical, construction, you know, sort of end to end. The last thing I'll just leave you with on that one, since you asked the question. One of the things I think is the most exciting opportunity in a place that Unity plays specifically well, we're not trying to go... Well, actually, let me step back. Every industry has a whole bunch of 3D data. Like a manufacturing company builds

their stuff in 3D CAD tools. They have tons and tons of 3D data. We're not trying to change that. We're

not trying to say, oh, you shouldn't use this super high-end CAD tool that you've always used to create

the very specific stuff that you need to do before you go to tooling and manufacturing. What we're trying to do is free that 3D data, because today that 3D data is captive on high-end desktops in the

R&D

department of one part of the company. And yet the use cases, if you take that 3D data and you say, okay, well, actually, what I really want to do is I want to be able to do an unlimited amount of photos so that I can do for commerce or for any other thing that I need to do it. Well, the way that works today is you wait to get a real object and you actually have to go create photos of it and all that sort of stuff. There's no reason for that. You should be able to literally have not the R&D department, but the marketing department have access to the same 3D data, transformed to be real time and interactive, and transformed so that it can run on every device so that it can get both throughout the enterprise, but also to all of the customers and partners of a particular company. And so that's our goal. Our goal is to actually take the 3D that they already have, but truly free it so that it adds value across everything that they're trying to do in their enterprise from, you know, core sales to support to, frankly, even internal logistics and operations. And that's why we see such an extraordinary pull from all of these industries into Unity.

Amazing. Really exciting times ahead.

Awesome. Yeah, this is great. It's really, really great speaking with you guys. So yeah, I've enjoyed it. Well, thank you so much for your time, Mark. And for all our developer listeners out there, make sure to check out Muse, Sentis, and the Unity VisionOS Open Beta. And also make sure to check out Arc's blog on generative AI's impact on gaming. Thank you.

Unity gets a nice shout out in that.

Thank you. All right. See you guys.

Arc believes that the information presented is accurate and was obtained from sources that Arc believes to be reliable. However, Arc does not guarantee the accuracy or completeness of any information. And such information may be subject to change without notice from Arc. Historical results are not indications of future results. Certain of the statements contained in this podcast may be statements of future expectations and other forward looking statements that are based on Arc's current views and assumptions and involve known and unknown risks and uncertainties that could cause actual results, performance or events to differ materially from those expressed or implied in such statements.