00:10 Noah Kravitz

Hello and welcome to the NVIDIA AI Podcast. I'm your host Noah Kravitz.

Today we're talking about the future of collaboration in 3D. Universal Scene description, Open USD, is revolutionizing 3D graphics and simulation, especially when you combine it with the latest in physical AI. The technology is transforming industries from manufacturing to robots.

And here to explain what OpenUSD is and why it works so well together with AI is NVIDIA's Aaron Luk.

Aaron is a director of product management for NVIDIA simulation technology, leading universal scene description ecosystem development. Aaron, welcome to the AI Podcast.

00:50 Aaron Luk

Hi, Noah, good to be here.

00:51 Noah Kravitz

Great to have you. Thanks for taking the time to join us. So let's start kind of at the at the beginning and work our way up, if you will.

What is Open USD and why does it matter so much?

01:02 Aaron Luk

That's right. So as you mentioned, Open USD, the USD stands for Universal Scene Description. It's a project that was open sourced by Pixar Animation Studios in 2016.

But it's the result of the evolution of decades of data engineering at Pixar around, you know, basically 3D world building. 3D world building among all the sort of disciplines that it requires for filmmaking, but it generalizes quite beautifully to world building in the industrial world and in the real world as well, too.

So it's an open source project that also now is under the governance of the Alliance for Open Universal Scene Description, the AOUSD, in which we are formalizing USD as industry standards with a lot of great partners.

01:42 Noah Kravitz

Fabulous. So what are some of the benefits? I mean, obviously having an open source standardized framework for describing and working with 3D worlds is great in and of itself. But what are some of the particulars about OpenUSD that make it really great to work with?

01:57 Aaron Luk

So the really interesting thing about USD is that it's designed to bring lots of different types of data sources together. In particular, it's called composition within USD and every document in USD is called a layer. So when you bring all these things together, you have these networks of layer stacks within USD that presents itself as a holistic composed scene graph. And every object in that scene graph is like an object in 3D that you can use for movie making, but also for industrial layout and design.

And the power of USD is all of that is abstracted from the actual data source and the actual data serialization, the actual formats. And this was a boon within Pixar because like I said, every type of artist within Pixar, whether they're doing modeling, animation, effects animation with physics and other simulation, lighting, all that kind of stuff, they might have different tools. They might have different ways of working with things. And what Pixar did was they kind of unified them all around these common data models in USD, so they present themselves as schemas in USD where every object in USD has a typed schema. For example, a mesh for the shapes that you define in USD. And you can also add applied schemas onto those geometries like physics APIs to imbue them with collision properties and things like that.

03:12 Noah Kravitz

Oh, fantastic.

03:13 Aaron Luk

Yeah, but it's all abstracted in USD and it's all this data can live in separate layers.

And those layers, they can live on disk as files or they could be in the cloud as files, or they can be populated from databases or even dynamically generated, right? So you can see where I'm going here, where this makes it a really nice fit for the sheer volume of not just the amount of data, but the types of data that are flowing into industrial digital twins. It's really, really exciting to see.

Certainly even within filmmaking, like you're already bringing lots of different types of tools together, but in the industrial world, that's even more expansive between all the CAD tools that can feed USD. And then as you expand that out into those kinds of industrial workflows, like product lifecycle management, facility design and planning, all the way up into operational twins, right? In which you actually have a physical facility and also the digital version of it that's tracking all the things that are happening with the equipment and the robots that are in the facility and so on.

04:12 Noah Kravitz

Yeah, the little bit of exposure I've had to OpenUSD has been through industrial projects, right? So that whole world of just operating these, you know, physically accurate and pixel-perfect simulations, digital twins of a factory industrial site is just amazing.

One of the things to me that's really cool about OpenUSD as I understand it is that we can be working on different layers, collaborating, but sort of working without getting in each other's way. And it's non-destructive as I understand it.

04:40 Aaron Luk

Yeah, that's right. So again, let's take the filmmaking analogy, right? Where multiple artists might be working on the same shot, maybe not at the same time, but certainly in different layers of that shot. And that way, right, a layout artist can make like the basic layout of where the characters start in a shot. And then a character animator can then add, you know, all the, all the expressiveness on top of that layout and so on and so forth, right?

And those same principles apply to industrial design and layout, right? Where a layout person, a planner might do some initial factory layout, but then someone who is really specializing in a particular work cell within that factory might iterate on top of, like, you know, the base layout of the entire factory in there. And then the person who's working on the robot arm within that work cell might specially, like, you know, add more details on top.

And so what we mean by non-destructive is that everyone who's working on that, their work is still preserved somewhere in the layer stack. And so you're not, you're overriding each other's work and adding to it and tweaking it accordingly, but everyone's work is preserved so you can always sort of look exactly at what someone did and what kind of changes were made on top and that kind of thing. So it's like really quite a boon to industrial workflows where everyone has a part to play and everyone's adding something.

05:52 Noah Kravitz

Right, absolutely. So how does OpenUSD work with AI, and in particular, how can it accelerate the development of physical AI?

06:01 Aaron Luk

Yeah, I think OpenUSD is a really good fit for physical AI because OpenUSD has already, you know, the native 3D paradigms, but because of the nature of its flexibility of the data model and the composability of different data sources, right? It makes it a great way to describe, you know, the worlds in which physical AIs are operating in particular for training robots. So with USD, right, you could have your robot that's described in USD, and maybe it's translated from URDF or NJCF, any number of robotic formats that can be mapped to USD via schemas. But also the world that the robot is in is also within USD and then that can come from CAD or it could be made in-house and so on and so forth. What USD does is it gives you this unified holistic way of simulating the world for physical AI to create those environments for robots to learn how to navigate, how to respond to different scenarios.

And then it plays really nicely with technologies like NVIDIA Cosmos as well, right? Where you have your baseline scenarios in USD, you can start some synthetic data generation to vary different objects in the scene and vary different scenarios with them. But then you can vary even more conditions in Cosmos, like time of day and weather conditions and things like that. So before you know it, you basically have just a rich, comprehensive set of scenarios that the robot can learn from, right?

And this is all being rendered through something like Sensor RTX in Omniverse, right? So that you get, like you said, it's pixel-perfect, but it's pixel-perfect for sensors, right? So it's physically accurate in simulating what the physical sensor on the robot would see in the real world. And that robot is effectively seeing vast amounts of scenarios to be able to learn from and before it's ever deployed into the physical world.

07:55 Noah Kravitz

So when we're talking about robots, autonomous vehicles, things that we're simulating to sort of prepare to deploy in the real world, right? Can you explain what the SIM-to-real gap is and how OpenUSD plays into helping solve that for physical AI training?

08:11 Aaron Luk

Yeah, sure. So the SIM-to-real gap is basically what I just mentioned, right? Where what the robot sees in the virtual world should match what it would see in the real world. And so the big aspects there are the physics simulations. So you need really good solvers to run through all the rigid bodies and all the sort of things that happen in the real world. And then you also need to visualize it in the same way that a sensor would perceive it as well.

So those are some of the key aspects to fill in. Obviously there's always lots of great physics research that's going on. Even outside of the computer graphics community, right? Physicists are always learning more about how the world works. But the cool thing is like AI is also learning that as well too, right? And that's another place where Cosmos comes in, right? Where you can like, if what you're trying to do is simulate what the robot is seeing, right? Like AIs can sort of recognize the patterns and again, like vary things accordingly that you don't even have to then simulate in 3D anymore.

09:11 Noah Kravitz

When we’re talking about generating simulations like this, are capturing and replicating physical — the way physics works — Is that one of the trickier aspects or the trickiest aspect of it? Or what are some of the big hurdles that have to be cleared? Or perhaps that, you know, the growth of OpenUSD has helped us clear recently?

09:28 Aaron Luk

I think what OpenUSD is doing is, it's more giving folks a common place to unify their data models, right? So every physics solver is going to have different behaviors, different characteristics for the kind of performance characteristics they're trying to hit. But what USD is helping with is like, can we standardize the inputs to those solvers, right? Such that you can run even multiple solvers for multiple aspects of your scene for co-simulation. That way you can have multiple physics solvers and engines all operating on different parts of your scene, just like in the real world, you wouldn't have the same physics solver for locomotion as you do for grasping things necessarily. And certainly for all the things that robots are doing and what other machines are doing within the factory, right?

There's all sorts of different things and what USD is doing is providing this framework around like, can we converge on the inputs that are common to these kinds of physical operations like rigid body collisions, like soft body collisions, and that kind of stuff.

10:36 Noah Kravitz

Right. We mentioned earlier, I mean, we've been mentioning throughout, but specifically talked a little bit about digital twins and industrial AI earlier. And I referenced actually, we had Siemens on the podcast not too long ago talking about this. Obviously lots of NVIDIA customers, Lowe's is another one, that are building digital twins, these replicas of factories or cities, or perhaps even retail sites, using OpenUSD, using AI to, as you've been talking about, simulate scenarios, but to optimize operations and do these other things that are driving innovation right across industrial use cases. You talked a little bit, and I think you referred to this, I was thinking about synthetic data when you were talking about simulations. But how does OpenUSD specifically play a role in creating digital twins and generating synthetic data for these industrial use cases?

11:26 Aaron Luk

Yeah, so because again, USD already has all these 3D capabilities for describing virtual worlds, right? It's a good fit to very, you know, existing USD objects to produce synthetic variations of baseline objects, like the things that you're manufacturing in a factory or the things that you're selling in a retail space and so on and so forth. And it's all about like, because USD can kind of adapt to any data or any data model can be adapted to USD, right? Kind of anything that you want to vary, certainly if it manifests itself in sort of physical appearance and shape and things like that is a natural fit in USD. But even beyond that, right? There might be other things you want to vary, like I said, like weather conditions and things like that, right?

Where I'm sure there's simulations that we probably haven't even thought of yet. But I know that they'll be expressible in USD because it is, you know, you'll be able to define a schema around which those simulations can be described and the inputs to those simulations, that kind of thing. So again, it's that kind of extensibility of USD that makes it a really nice fit for, you know, synthetic data, like variations that we haven't thought of.

12:35 Noah Kravitz

Right. And along those lines, I would imagine it's got to be beneficial when you're building or even scaling out pipelines to deal with synthetic data. And even in these situations, like you said, we haven't imagined the thing that we want to simulate down the road, but you know it'll be expressible.

12:50 Aaron Luk

Yeah.

12:52 Noah Kravitz

So, Aaron, you talked a little bit earlier about some standards emerging out of OpenUSD that are making it easier for people to work on different projects in different places. Just standards just make things easier for folks to work. Are there standards, similar standards emerging out of OpenUSD, specific to working in physical AI that are making things easier, maybe across industries?

13:13 Aaron Luk

Yeah, I kind of think all of the standardization that's happening in the USD will eventually funnel towards some sort of physical AI use case. Standards are particularly important because I think they're the bridge to what I was mentioning earlier where USD is great because it's so flexible, right? And so adaptable to lots of different domains, lots of different use cases. And that's why we're seeing such large adoption around it.

But the flip side of flexibility is ambiguity. And what standards really do is empowers you with the flexibility, but sort of removes the ambiguity such that we're all sort of like rolling in the same direction. And so USD is very open in how you express things and it's great for that. But where standards come in are, for example, USD allows you to express transforms in any arbitrary number of ways, which is very powerful. But if you want to use it for physical AI, right, you might want to simplify the transform stack that your USD object has so that your physical AI kernels have less complexity to reason about and things like that.

So you can envision USD as itself a stack of multi-part specification. And at the core of it is the core specification. So in the AOUSD, I'm serving as chair of the core specification working group. And that is really where we are normatively specifying the most novel aspects of USD in this foundation, like the ability to compose data together. So, you know, what are the specific data models that feed the composition engine? What's the algorithm for composition? And then how do you take that composed scene graph and issue predictable queries on like, say when you traverse that scene graph, you're able to have predictable queries on like, what are all the objects and what all the properties of that object? Everything beyond that, you can kind of think of USD as a standard of standards. So there's already quite a lot of standards in the industrial space for CAD, for product lifecycle management, for geometry, and all those kinds of things.

In the operational space, there's OPC UA and Web of Things. And what I described before as USD schemas, you can think of those as like mappings of those existing data models into USD. And so as we build out this stack of standards, it's about mapping other standards into USD, right? So that USD is speaking all these other data models that exist, but presenting them to you in this holistic way.

And that's sort of what physical AI needs in particular, because you need to be able to describe everything that's happening in the real world. And the real world does have a lot of these standards that exist as well for physical objects, but in particularly around equipment in your facilities and things like that. There are already specs for that equipment and that kind of stuff. So a lot of the standards work is like mapping those existing standards into USD.

15:58 Noah Kravitz

Into USD, right?

15:59 Aaron Luk

Yes.

15:59 Noah Kravitz

You mentioned the working group that you're part of. You were at Pixar previous to joining NVIDIA?

16:05 Aaron Luk

Yeah. That's right.

16:06 Noah Kravitz

And were you working on the development of USD back then?

16:10 Aaron Luk

Yeah, so I was actually one of the original two developers on USD. It started off as a pair programming project, taking some existing technologies at Pixar, particularly the composition engine from the animation package, as well as the scene cache format that was being used to move data between departments, between tools at Pixar and sort of marrying them into a single paradigm.

16:32 Noah Kravitz

Very cool, yeah. And how long ago did it get started? When was it OpenUSD for the first time?

16:37 Aaron Luk

So the actual USD project, I think started in 2012-ish or so, kind of right after, around that time.

But the technologies underpinning it have been dating back to Pixar since probably a bugs life. know, pretty much right after they wrapped Toy Story 1, they were already thinking about how, you know, how can we better organize this data across our departments? Yeah, so the composition engine started, I think probably around 2005 or so.

Certainly it's sort of the referencing and like the non-destructive workflows type stuff have been around at Pixar for decades.

17:17 Noah Kravitz

Yeah, yeah, neat. I don't know, looking back on, you know, almost 15 years now, I guess, of USD, how has it evolved? How have you seen it change? Or there are things that you, you know, I don't know, maybe you didn't think of when it first got going that now you're like, wow, I'm so glad that, you know, that came to be?

17:32 Aaron Luk

Yeah, it's been evolving quite quickly. I certainly didn't envision all of this industrial adoption at the time because what excited me at the time was more seeing how many of those concepts mapped to what other movie studios, both in visual effects animation, were doing. Right? Certainly going to SIGGRAPH at the time, I would attend pipeline talks and hear about similar concepts, right? So it's great now to be on calls with ISVs and customers and hearing about their ways of working and really like showing them how it maps to this USD way of working, of like, kind of having the data really travel all along your workflows and really rethinking what we mean when we say pipeline. I think in the industrial world, it's a little bit more the source data kind of gets very hard exported between disciplines and things like that.

Then the original still kind of exists, but you've kind of lost that link to it over the course of how that data travels. USD kind of allows that data to travel and you're adding to it as you go along, just like you do in a real assembly line. When I was working on USD, we didn't have this notion of API seamless, which I think are super powerful. So that's how, like I said, you can add additional annotative properties to existing objects. That's how your shapes also become physically simulatable objects for rigid bodies and other simulations.

This is how we've also added semantic labels onto objects as well, which is really key for machine learning and segmentation of the scene, those kinds of things.

19:02 Noah Kravitz

Very cool. So we've talked about USD, OpenUSD, and you know, I was going to say all of the upsides, we've only hit some of the upsides, but it's vast in all of these different industries and situations we've been talking about the power of having digital twins and collaborative 3D simulations and such. How do you get started?

I'm listening to the podcast. I'm listening to you, Aaron, talk about this. I'm like, wow, this sounds exactly like we need, but how do we, I don't know where to begin USD. What do I do? What does it even mean to get started with USD and how would one go about that?

19:32 Aaron Luk

Yeah, sure. So just like you can get started with AI on NVIDIA's Deep Learning Institute, we also have Learn OpenUSD on NVIDIA's Deep Learning Institute as well. So that's a growing curriculum of hands-on, self-paced courses that start with really the basic foundational principles of OpenUSD, and we're always adding more courses to it over time. And that's a really good way to get yourself grounded and really learn the skills that you need to contribute to USD and develop these pipelines that are so key to physical AI, to moving data around into the unified worlds that physical AI needs. And that path to that leads you to a new USD certification program for which this DLI curriculum is designed to get folks certified just so you can get certified as an AI developer as well. And that's way you can really distinguish yourself and get hands on and learn USD for any number of domains and use cases.

20:33 Noah Kravitz

And so somebody could go to Nvidia DLI, Deep Learning Institute, and get started learning open USD?

20:39 Aaron Luk

Yep.

20:39 Noah Kravitz

Fantastic.

20:40 Aaron Luk

Yeah, and then of course, USD is open source as well. So it's got a GitHub repository, which has its own set of issues and lots of great folks in the community have been labeling issues as they triage them as good first issues. And that's a way to, as a new developer, to get hands on with USD and contribute to it directly by fixing a bug or improving documentation and that kind of thing.

21:04 Noah Kravitz

Fantastic. And for someone who's more versed in the 3D space, a designer, developer, but not necessarily a coder, do you need a coding background to get going with USD?

21:14 Aaron Luk

Not necessarily. Especially now too, where you can use co-pilots to issue prompts to be like, please write me a Python script in USD to create a grid of nine boxes in a factory or something like that. These are things that you can try, especially with Omniverse technologies where I know some partners have integrated things like that into their experiences.

Yeah, I think even without a coding background, just like the world of coding is evolving in general, so is the world of coding for USD. And coding may mean refining prompts accordingly.

21:51 Noah Kravitz

Right. Everything's changing. Aaron Luk, this has been a fascinating conversation. And for the little bit — I mentioned I had exposure to USD beforehand, I've certainly learned a ton. And that idea of, you know, the standard with the standards inside of it, and you can, it's portable and you can annotate and it just, it all makes sense. And it's, I can see why it's so popular and so powerful. Thanks for taking the time to join the podcast, to talk about it.

We mentioned the certification program and DLI. Anywhere else you would direct a listener who wants to learn more about USD, about the work Nvidia is doing with it, the work that you and your teams are doing? Anywhere else they might go online?

22:29 Aaron Luk

Yeah, sure. AOUSD.org is the entry point for AOUSD. There's also forums there, forums.AOUSD.org. For NVIDIA, I highly recommend build.nvidia.com. I know that's come up on other podcasts as well. There's blueprints there around Digital Twins in which USD is involved. There's always going to be new or expanded blueprints around that. And certainly docs.omniverse.nvidia.com is a good place to go. There's dedicated USD learning paths there as well that complement the learn OpenUSD material as well like things like you know workflow guides on sort of using USD to to assemble industrial scenes that kind of thing.

23:12 Noah Kravitz

Perfect. We'll leave it there. Listeners have a whole bunch of places to go dig in, get hands on with USD and OpenUSD. And again, Aaron, thank you for taking the time, let alone all the contributions you've made to USD in the industry over the years. It was a pleasure talking and let's do it again sometime.

23:28 Aaron Luk

Alright, thank you Noah.