Shruti Koparkar (00:00):

Hello everyone and welcome back to the official AWS podcast. I am Shruti Koparkar, your host for today's episode. And we have some really exciting guests today. They've worked in the media and entertainment industry on movies such as the Jungle Book and the Blade Runner. This is going to be one exciting conversations. They now are here at AWS and going to talk to us about a new service, AWS Deadline Cloud that simplifies 2D and 3D rendering and access to render forms. So, let me welcome my guest today, Pauline and Sean. And maybe the two of you can introduce yourselves. Pauline, you go first.

Pauline Koh (00:42):

Okay. Well, long-time listener, first-time caller. My name is Pauline Koh. I am a senior product manager over here at AWS, and I look after the ThinkBox, Deadline, and Deadline Cloud products. My background is in visual effects and feature animation, so I've worked at a few studios such as [inaudible 00:01:03], and Technicolor, and I've also worked in a little bit in games, such as on Stadia at Google. Thanks for having me.

Shruti Koparkar (01:09):

Great, welcome. Yeah, welcome.

Sean Wallitsch (01:12):

My name is Sean Wallitsch. I am a senior specialist solutions architect here at AWS on the visual compute team. The visual compute team is comprised of former visual effects and animation professionals who have now come to AWS to help customers make pretty pixels in the cloud. And we do that by way of rendering, or virtual workstations, or just helping them with their basic infrastructure and translating how their traditional workflows now work on AWS.

Shruti Koparkar (01:39):

Awesome. Well, welcome again, both of you, Pauline and Sean. Now, I do know that you have actual career experience in some of the use cases that we are going to talk about 2D and 3D rendering. You've worked on some really exciting movies. I think this is the first time I'm actually speaking to someone on this podcast who has an IMDB profile. So, tell us a little bit about that. Tell us about your background, what type of work you had done. I'm sure listeners would like to learn about the movies you worked on. And then, how did you find yourself going from that to AWS?

Pauline Koh (02:16):

I'll hop in first. So I had a great time working on the first Lego movie, and also the first Guardians of the Galaxy, and as you mentioned before with the Jungle Book, that taught me a lot, so it holds a special place for me. I think, my transition basically looked a little bit like, I quickly realized that I was much better at coordinating systems, and large-scale compute, and rendering than I was at taking creative notes. So, I quickly made that transition.

Shruti Koparkar (02:47):

I'm glad that you are part of AWS. How about you, Sean? What were your journey like?

Sean Wallitsch (02:52):

A very long time ago, I was actually an artist. So, I was a compositor, and then a lead compositor eventually. And, places like Lola Visual Effects and Rhythm and Hues. And while there, I worked on movies like Blade Runner the Final Cut, which was so incredible to work on. Cabin in the Woods at Rhythm and Hues on a very small focus team doing compositing by the seat of our pants trying to get this movie out and trying to try all these different looks for some of those effects. And then, at Dreamworks Animation, I actually took a swing and moved over to develop a role, because I realized that I liked building tools for the people who were sitting next to me.

(03:28):

And so, as an artist and as a lead, I was constantly building tools for the other artists on my show. And, at Visual Effects and Animation, there's actually a role for that. So I became a department TD and then a lead department TD. And while at Dreamworks, I actually started building pipelines, digital asset management. And, because it was cool, I started working in things like containers, and microservices, and making sure that the new stuff we were building was very cloud aware. Well, that sounds good to places like AWS. And so, I've been at AWS for about two years now.

Shruti Koparkar (04:00):

Excellent. Again, I'm glad you are part of our team. So, given that you guys have so much experience in the field, before we talk about this service, let's just talk a little bit about what is exactly 2D, 3D rendering? What is its place in movies and bringing these awesome ideas to life? And then, how is it all powered? We've heard the term render farms, what's going on with that? Is it a farm? What are we talking about? So, either of you take that question and make what you want with it.

Pauline Koh (04:34):

Yeah, I'm going to kick this off and then Sean can correct me when I get this wrong. So, the way I like to think about it is there's two levels to think about this. At a very foundational layer, it's about letting artists use their creative software of their choice to create instructions that turn into, in terms of 2D, flat images. So, a movie is made of many, many frames or images, and these frames tend to need to be rendered, because trying to do so on a workstation for example, can take a very long time and constrain the resources of that workstation. So, they tend to use what we call render farms to farm out that workload to be delivered on some other machines, and then the output is returned in packs of frames.

(05:26):

The other level I like to think about this is, it's like a compile process. So if you think about the artist is creating some instructions but they won't see the final result until the compiler is done with it, you can abstract the render farm away into a compile process, and then the output of that compile is all your finished frames.

Sean Wallitsch (05:49):

It's so true, because in the original way you used to be able to do visual facts and animation was you literally wrote scripts and code to create the actual final image that would appear in a film, is you had to be both a programmer and an artist to do that work. And, we've, thankfully, abstracted much of that away into artist applications now, so they no longer need to be coders, but those instructions still take a high amount of processing to actually result in a final thing that we put on screen for audiences.

(06:21):

In the old days, all those render farms used to be fixed size on-prem installations. And so, your farm would compose of maybe 400 machines in your data center, and those machines would deprecate over a long period of time. You would move old artist workstations into there. Once your artists get new workstations, you would just fill it up as tightly as you can pack it and as hot as you can make it to use the just massive amounts of compute power needed to render these films. Over time, companies started getting tighter and tighter deadlines for getting these films out, and so you started realizing the need for burst rendering around the deadlines. And so, in the old days, you would actually rent a semi truck full of render blades that you could use to power the crunch time of these renders.

Pauline Koh (07:12):

Wow.

Sean Wallitsch (07:13):

And so, the last two weeks of a film, you would have a bunch of semi trucks in the parking lot running off of generators with these thick cables heading into your data center, and suddenly your render farm would spike up twice as large as it was before just to finish the film on time, because the deadline is set and the amount of space in your building is fixed. But, there are ways getting around the physical space in your building and that's where AWS fits in, right? AWS allows you to use the power and scalability of the cloud to render without having to park semi trucks in your parking.

Shruti Koparkar (07:46):

Awesome. That is definitely also value prop of AWS, where you can basically do compute render on-demand. Pauline, it looks like maybe you wanted to add something.

Pauline Koh (08:00):

Yeah, I just wanted to add that something that is a little bit unique about render farms for creative applications is there's also a creative opinion that is part of that decision making process. And so, for a lot of other high performance compute applications, it's just, the result is the result that will be analyzed later as part of a report or some data analysis. However, for creative applications, a creative director or a supervisor would want to take a look and that can cause a lot of prioritization changes between the jobs at any given time. And so, a render farm in particular needs to be able to quickly between jobs that are more important now or just don't matter anymore. But yeah, otherwise, the 3D aspect versus the 2D aspect is just, what output comes out of the farm?

Sean Wallitsch (08:55):

And to follow up on the last bit, they're not literal farms, but we do call the employees who manage the farm render, wranglers, because they are wrangling the farm.

Shruti Koparkar (09:04):

Got it. Got it. Okay. So, you talked about how AWS can help. And, we actually did, right? ThinkBox Deadline, which was first released in 2004, and then became a part of the AWS family in 2017. So we have this product, ThinkBox Deadline. How did that evolve into now AWS Deadline Cloud? What made us want to go build another product, another service? And would love to understand what are some of the motivations there? What were the challenges we were looking to solve?

Pauline Koh (09:40):

Yeah. Shruti, as you pointed out, the team has decades worth of experience building these tools, but we also have a lot of customers that we can listen to or we hear feedback from. And overwhelmingly, what customers have told us is that they really liked the Deadline interfaces, but they also really wanted something that behaved more like a web service. So, in 2021, we started to take a real hard look at what we had and what we could do with fresh eyes to integrate all of the feedback that we were getting, especially after the team joined AWS.

Shruti Koparkar (10:18):

Right, right. And so, what is it that you heard from our customers? Because that's what we do best here at Amazon, we work back from the customer needs and it's great to know that you had all these customers who were using ThinkBox deadlines. So what did we hear from them?

Pauline Koh (10:34):

I think one of the clearer signals that we heard from customers was that there was a very steep cost consideration to consider a new solution, let alone, migrating to a whole different platform. And sometimes, that was just too difficult to consider when there were multiple projects going on and it was difficult to schedule any downtime. And so, that was a clear pain point we heard.

Sean Wallitsch (11:00):

And then, once they got on AWS, cost prediction was also really difficult for them. It was hard for customers to see how much they were spending and how much the next project would cost them. That tooling just wasn't there for Deadline 10.

Shruti Koparkar (11:17):

Right, right. So, because Deadline 10 came from the 2004 timeline, a lot of what was built was more from the perspective of on-prem, even though it was integrated within the AWS services, the whole design and the approach was more from that on-premises side. And, sounds like customers really wanted that elasticity, like quickly being able to get the compute they want only when they want it, but also, being able to better predict what their spend is going to be like and better control it.

Sean Wallitsch (11:53):

Exactly that, Shruti. ThinkBox Deadline 10 application was built with the idea that you owned and were deprecating all of the compute used to do those renders, and therefore the cost to operate, it wasn't really a factor. It's there, it exists, and you would then prioritize how you wanted to use that compute. And that was the primary focus. And what we heard from customers is that as they move to the cloud, cost becomes a much larger focus point for them, because now they have to pay for every hour of compute they use, and how can they control those costs? How can they explore those costs? And how can they budget and predict their cost going forward?

Pauline Koh (12:33):

I'll give you an example too. I've heard from a customer where they told us that for any project that is considering cloud rendering, they need to actually go through an approval process that involves their CFO, that's how much they care about managing the operational insights and visibility.

Shruti Koparkar (12:57):

I mean, certainly, cost is an important factor. The quick scalability or elasticity of the cloud is important as well. I might be wrong here, but complexity has got to be one of the things as well. I mean, all of these render pipelines can be too simple. Can you double click on that a little bit? What is the complexity level involved in creating this render pipeline? And then, maybe we can get into how Deadline Cloud can help with it. But yeah, I first want to understand what is it? How complex is this?

Sean Wallitsch (13:33):

Generally, the complexity involved in a feature film pipeline for visual effects or animation is pretty high. And it's something that every studio working on that stuff needs to do at some level. Once you get beyond 30 artists, you stop being able to communicate by yelling across the room where you saved a file. And so, files start to get organized. You start developing a structure. You start working with one custom tool to make it easy to do one process that you're doing over and over again. Then you have another, and before you know it, you've got a full-blown digital asset management pipeline in the middle of your studio.

(14:04):

So, studios start building those. And the large studios have thousands, and thousands, and thousands of lines of code devoted to managing their assets, moving things from one part of the production schedule to another part of the production schedule and for work to be done on it. And then, at the same time, all that stuff needs to head to the render farm in some way, shape, or form. And so, managing the connection between the pipeline and the render farm is an additional area of complexity that results in studios becoming very tied to one specific render farm solution that they have.

Pauline Koh (14:41):

It's important to recognize the realities that these infrastructure teams are dealing with as well from a production or a business perspective, asking for time to take systems down, not impossible to schedule, because it's just when would you get a break from actually trying to deliver these shows and movies. I think, one of the things that is key to focus on is trying to remove unnecessary complexity by trying to improve modularity and interoperability at the tools layer within the pipeline. I think a lot of our customers have a long way to go before they get there, but hopefully we can help.

Shruti Koparkar (15:24):

Excellent. Okay. So we've talked about some of the challenges that customers face, right? We've talked about the cost control that they desire, the compute elasticity and scalability that would be really welcome, as well as trying to simplify some of this complexity in terms of setting up the pipelines and working on them. So now, tell us about Deadline Cloud. How did all these challenges turn into this solution? What does it do? How does it help customers?

Pauline Koh (15:56):

So I think, one of the things I like to talk about the most is comparing just the simplicity of it and the ease of getting started. If you know what you're doing, Deadline 10 is pretty straightforward to get set up and running to set up a farm. But, with Deadline Cloud, we've got that process down to creating new render farms that can accept real jobs and start rendering within about five minutes. Now, I had asked the team for 15 minutes and they've really overachieved here. So, I'm very proud of that. And then, there are a few other things, like I think some of the production tracking and the usage tracking.

Shruti Koparkar (16:32):

Okay. So I think I heard Pauline say that you can literally get started with Deadline Cloud in five minutes. Is it really five minutes? How quickly can someone build a render farm using Deadline Cloud?

Pauline Koh (16:47):

I'm so glad you asked. So, really actually, you can get a render back in under an hour if you're just speed running it. The part for setting up a farm through the AWS console to process a real job, that can be as quick as five minutes. Then after that, you would typically download some of the digital content creation tool plugins that we offer, and then also install the Deadline Cloud monitor in the same step, and that would make it really easy to submit your jobs to your new farm, and that should take 15 minutes tops, and then the rest of it is probably going to be getting the renders back.

Shruti Koparkar (17:24):

Which will take longer, right? So the setup is five minutes, as quick as five minutes, and then depending on how much and what volume you're trying to render tells you when you get it back.

Sean Wallitsch (17:33):

If you're like Pauline and I, you're going to spend a lot of time agonizing over the farm names and the queue names, because you have to make sure those names are perfect. And then, you want to make sure that all your settings are correct, and that'll net you closer to a couple hours to get renders back. Then if you're trying to do something on the complexity of traditional AWS ThinkBox Deadline 10 Farm and all of the configurability that entails, and making customer managed fleets, and controlling your AMIs, and stuff like that, that's realistically a few days of configuration, testing, iteration. But, for ThinkBox Deadline 10, that used to be over a month for most of our customers.

Shruti Koparkar (18:06):

Wow, that's some serious time savings.

Sean Wallitsch (18:11):

So, for me, just hearing all these challenges from customers, one of the big things that we always get is that when customers are using the cloud to do any peak rendering for them as they head into these deadlines, they want to scale up and back down to zero when the job is done and they don't want to have to pay for that infrastructure between jobs and Deadline Cloud makes that easy to do, because when you scale up, it scales up for you. Make the fleet as big as you want to have it, and complete those jobs as fast as you need to get them completed. And when you're done, it'll scale back down to zero elastically, without you having to pay anything more going forward.

Pauline Koh (18:48):

Another great one, I think, is the team has worked with partners to also build out managed solutions here as well. So, licenses are pay as you go. And also, we have a growing selection of software and creative applications that can be used in our service managed flee straight out the box.

Sean Wallitsch (19:09):

We were talking about complexity earlier, and this is one area where we developed an open source specification called open job description to try and start tackling this complexity. And, we know the joke about standards, right? You have five standards in the market, no one's using the specific one, so you made a sixth one, and now you have six standards that exist. But, we have tried to work with the industry and tried to get feedback from everyone including ourselves as to how we could make this problem of render farm integration complexity much easier. So, open job description is a specification in a set of tooling that exists and Deadline Cloud is powered by it. All of the job submissions that you make to Deadline Cloud go through open job description or done with open job description, and it allows you to describe units of work that Deadline Cloud will accomplish. And, the best part is, it's completely open. So, we hope to build and support AWS ThinkBox Deadline 10 to use open job description as well, so you can use it to describe units of work for both Deadline 10 or Deadline Cloud.

Shruti Koparkar (20:12):

Awesome, awesome. So, we've talked about Deadline Cloud being really fast. You can get started with it really quickly, and build and deploy. You've talked about being able to scale up and down depending on your needs and only pay for the one that you use. We've talked about the ability to track costs and control them better. And, through some of these integration, it sounds like, there is a lot of customization that customers can do as well. Is there anything to double click on there in terms of customizations? What functionality have we built in to help customers make it their own? Because, I imagine everyone has their own unique set of needs.

Pauline Koh (20:58):

I think, one of the really interesting things that we do on the team is we use Deadline to build new versions of Deadline. And so, if there's one thing that I think the team is really excited for is to see what cool new things people can do with Deadline Cloud. And, really I think that we're inspired by rendering and render farms, but this isn't only for rendering and render farms. And so, some of these cases might be game builds or software build pipelines where you can customize exactly how to integrate and automate Deadline, but we also provide some pretty snazzy new interfaces as well.

Sean Wallitsch (21:42):

We were talking earlier about the broad range of complexity that exists at visual effects and animation studios. And, we think that Deadline Cloud addresses that broad range of complexity by offering, again, a broad range of configurability back to our customers and what they can do. Deadline Cloud offers a service manage fleet option where the AWS Deadline Cloud service manages all the workers and the compute, and it comes with a variety of directly support applications. So, you can just say, "Oh, I've got this job. It's directly supported. It supports usage based licensing." You can submit that job to AWS Deadline Cloud, have it go on into service managed fleet, and you don't need to worry about managing AMIs, or licensed servers, or anything like that.

(22:26):

For our customers who need more complexity and more configurability, we also offer the ability to upload your own packages to the managed service portion of AWS Deadline Cloud. And so, you could upload your own pipeline tools, or if you have a software package that is custom and bespoke to your studio, you could go ahead and upload that as a package to AWS Deadline Cloud and have it run that package on its managed workers. And finally, for those who need total control over everything, there's customer managed fleets, which are where Deadline Cloud still handles the database on its end, still handles the scheduling on its end, and will send your infrastructure scaling signals, among other things, that will tell you, "Hey, you've got work coming in. You might want to scale up that spot fleet." For example. And then, you manage the EC2, you manage the machine images, you manage what software gets installed on those boxes, and you control everything except Deadline Cloud takes care of the scheduling and the scaling signals for you.

Shruti Koparkar (23:28):

Awesome. Well, this has been so much fun. I think, this is a really cool product. Hopefully, customers want to try it out right away. What are some of the other alternatives though? If Deadline Cloud is not just quite the right fit for them, what are some of the other options on AWS that customers can use?

Pauline Koh (23:49):

Of course we have a Deadline Cloud and we also have Deadline 10, but also, there's some interesting services such as AWS Batch and offerings from high-performance compute such as pCluster and Slurm. They offer different benefits and different use case automations, which can be quite compelling. And so, really encourage customers to take a look, and compare, and see what the controls are like, and what they prefer.

Shruti Koparkar (24:16):

Excellent. And, I think we covered this, but I'd still like to touch upon it one more time, is what are some of the use cases that people can start to think about when it comes to Deadline Cloud? What is Deadline Cloud best suited for?

Sean Wallitsch (24:33):

We've talked a lot this podcast about visual effects studios and animation studios, and certainly, that's where we come from, and that's a lot of the customers that we talk to. But, we also talk to and help make pretty pixels in the cloud, customers in spaces like industrial design, making product visualizations or catalogs, customers doing architectural visualization when they need to very dynamically spin up a whole bunch of renders for a new set of construction buildings or a new housing development. We've also talked a lot with game companies making game cinematics, or real-time rendering, or slightly less than real-time rendering using real-time render engines. And all of these customers can use Deadline Cloud to fulfill their needs, because Deadline Cloud is meant to be easy for them to use and to speak their language in the same way that it helps visual effects and animation customers do their renders.

(25:26):

Over time, we've seen customers in those spaces become more and more aligned to visual effects and animation workflows. And the reason for that is because visual effects and animation is for a long time taken 3D geometry and made 2D images out of it. And as those companies in other spaces have developed those needs, they've started taking visual effects and animation, both software and the professionals who make those pictures, into their fold. And so, they're starting to use these workflows a lot more and we want to make it so that it's easy for them to continue to do that.

Shruti Koparkar (26:00):

Great. Any other use cases you want to cover, Pauline?

Pauline Koh (26:03):

I would call out medical simulation and medical visualizations, as well as data processing for things like digital twins for remote facilities. So things covered there would be LIDAR scan processing and complex geometry decimation, all of that good stuff where the output needs to be high fidelity and looks a little bit like visual compute processing we can deliver. Ultimately, to get started with any type of rendering or any type of compute, it's any instruction that can be run on a headless CLI or scripted to be run via a CLI can be run on Deadline Cloud. So, the sky's the limit there. So, while we do provide some awesome interfaces, ones that we think will be good enough for artists as well as producers to operate, all of these interfaces are built on top of our public APIs. There's more than 100 of them, and really, people can develop using them to deeply customize the interfaces that they want to see or automate the way that they want to work. So, this customization basically uses the same developer tools that we used to deliver some of the managed services niceties that we offer.

Sean Wallitsch (27:23):

In fact, any piece of data that you see on any of the Deadline Cloud dashboards is available via those APIs, right? There's no private APIs in use here that would make it so that we have access to data that you wouldn't use in those same APIs?

Pauline Koh (27:35):

Absolutely.

Shruti Koparkar (27:36):

That's pretty cool. So there are a standard set of dashboards that if customers get good value out of them, they can just stick with them. But, if they want to do more, customize, they can just use the APIs and create visualizations that benefit them.

Sean Wallitsch (27:51):

You could create a Shruti render farm, given enough work and using those APIs, you could have it up and running.

Shruti Koparkar (27:57):

Oh, wow. Yeah. You know what? This sounds like a service that I need to get my hands on. I have this idea to build a little three-minute animated story for my 2-year-old child around all the cool places in our town that he likes to go to. I don't even know where to begin, but it sounds like I would need something like this or could benefit from it. But it's just three minutes, so maybe not. Who knows? Okay. So, any advice for folks who want to get started with Deadline Cloud? Where do they go? How do they get started?

Pauline Koh (28:32):

I would say, one of the most important things is to choose the right tool for the job at hand. And there's no better way to figuring that out than giving it a try. We invested a lot of time and energy to make the getting started process very, very easy. I spoke before about how it's a five click farm creation process, and you can get started with that just to kick the tires and see if that suits your needs. I think we can definitely help with that three-minute little render as well.

Shruti Koparkar (29:05):

Okay. Well, maybe I have to catch both of you right after this podcast and tell me how to make it happen. I think I'll be his hero basically if I make that happen. Well, thank you so much to both of you, Pauline and Sean, for joining the podcast. And, thank you for the audience for tuning in. And, until next time, keep on building.