

PRESENTER: Test, test. All right. OK.

Well, I guess we'll start from-- I can't start over. Anyway. All right.

And transcoding is a big business. Think about YouTube. We upload-- I think there's, like, 1,000 minutes of video uploaded to YouTube every hour, or something along those lines, probably even more. And all that needs to be converted from many different formats that people upload in to the YouTube-friendly format and also with many different flavors, the different resolutions, that type of stuff. So that's a huge workload on the cloud, not of much interest to you guys, but maybe archiving is.

So we've got these different layers, different levels of storage product that are super easy to access and inexpensive, and we charge by the gigabyte. And you're not charged for upload. You're only charged for download. So if you're archiving stuff, and you want to store it long term, there's these cold storage solutions, which we call cold storage.

There's nearline, which is stuff that you might want to access infrequently. And then there's live storage. There's persistent disk, which is essentially stuff that you access all the time. And there are different levels of cost and functionality, but they all have the same APIs. And then, there's things like distribution, which would apply to everybody because we've got the notion of local disk which is attached directly to a virtual machine.

One step up is regional disk, which is within one data center. And then there's multi-regional, which is synchronized across all of our data centers all over the world. And one thing to remember is that Google has its own massive software-defined network. We've laid fiber optic cables under the oceans. We just recently launched a Japan data center, and that has a direct line from the Los Angeles area to Tokyo.

We've actually leased ships and laid our own fiber under the oceans, and that's our network, so it's not the public internet that any of your data is going across. It's our network. So if you need to transfer data from one region to the other, it happens almost at the speed of light. Multi-regional storage is pretty handy because if you upload something to a data center in Oregon, for example, it'll automatically synchronize with all the data centers all over the world. That's really helpful if you're a global company.

But on to rendering-- so Zync. Zync is a software as a service, we call it, that is a render platform. It's essentially your own render farm, running on Google Cloud Platform. And there's a bunch of different front ends to it, different pieces of software. Today we're going to look at how Maya interfaces with it.

But really, what you're doing is you're sending your files and your scripts and your ancillary project data up to the cloud, and it's all a solution that is, essentially, wrapped up for you. And there's the typical marketing tag line, designed by VFX artists for VFX artists. But really, it's a really powerful way to get up and running really quickly and have your own render farm.

You could have up to 500 nodes running up to 32,000 cores. And you pay by the minute, with a 10-minute minimum. So people that can't afford or don't want to support on-prem servers and their own render farm, you don't have to deal with HVAC and floor load and things like maintenance and upgrading and hiring an IT staff. This is, essentially, farming all that out to the cloud platform.

When you're not using it, you're not paying for it, unlike a traditional on-prem render farm, where it's sitting there, and if it's idle, you're paying for it. When we say, work in your own environment, it essentially means you're running your local software. You're not running anything remote. You're not drawing on any different licenses or any different scripts or anything. And anything that you use, any custom tools that you use locally, can be synchronized with Zync so that any renders that happen remotely will be able to use those plugins, those scripts, and some of the other auxiliary data that you might have to upload.

And there's a dashboard that we're going to look at which gives you this visibility and control, just like a local render farm. You can re-queue jobs, you can cancel, you can re-transfer, you can pre-upload, you could pre-load data so that you don't have to constantly synchronize data. It's really super handy.

And the machines spin up when you ask them to. When you send a job, a bunch of things happen. Any references, any textures, anything in your project file that is anywhere on your network gets wrapped up, and it gets uploaded automatically onto Google Cloud Platform. So you don't have to worry about launching a storage bucket or defining a storage area or anything like that. Zync takes care of all of it for you.

And you can actually watch the logs progress. I'm going to show you a demonstration of how it works, of it working with a live job. When we say unlimited scale, obviously, within reason.

Right now, you can only have up to 500 machines running across 32,000 cores, but really, if you wanted to run that on your own-- build your own render farm with 32,000 cores, you would have a machine room this big, as large as this room.

And again, permanent billing-- I wouldn't say amounts to zero waste, but you're not paying for stuff that you're not rendering on. Ignore some of these competitor's logos, but right now I'm going to show you Maya 2017 rendering in the latest cut of V-Ray. From Maya, we can also render to Arnold. We just introduced RenderMan 21.1, I think, capabilities, which is super powerful, super awesome.

And another thing is, when we talk about all these third-party renderers, you don't have to worry about licensing. Zync takes care of all the licensing for you. So essentially, we are a license server that is reselling you a license on a permanent basis. You don't need to think of it as a reseller, but that's kind of how it's happening.

So there's different pricing models, depending on which renderer and which version of Maya and so forth and so on. And there's a calculator that will help you understand how the costs are billed. But you don't really have to worry about any that because you just get one total bill per job that you could look at. And I'll show you how you can investigate all that.

So rendering on demand-- talking about integration into your pipeline. Traditional visual effects pipelines or any type of rendering pipeline involves your local workstation and a 3D host application and possibly a compositor, if anybody's familiar with NUKE or After Effects or whatever. What Zync does is it replicates this, but on the cloud. So we're able to feed data from your third-party applications to Zync, running the same applications.

And Zync is actually capable of communicating between applications. So if you do a Maya render or any type of a render, and your NUKE job or your compositing job has a specific path that it knows to look for those renders, it will find them on the cloud and keep running. So you don't need to re-upload anything or re-path anything. Zync also takes care of all the pathing, as well.

So if you're running on Windows, the pathing will automatically be converted to Linux because Zync, right now, runs all on Linux. Or if you need to do some drive swapping, if you've got low res and high res assets-- but we can talk about that later. And then, once it's all done, it automatically gets downloaded, transferred to your local machine, whether that's a server or your local machine or whatever.

-

And this is not the type of demonstration that's happening outside of Trump Tower right now, but-- come on, people. That's the only joke I have in the whole thing.

[AUDIENCE LAUGHTER]

Come on, come on.

All right. Maya 2017-- I don't remember what service pack this is, but whatever. So I've just got this animation of this robot, a little Dutch roll there. The content isn't that important. But the fact is that I've got this scene all set up with V-Ray, ready to render. And quite simply, when you install Zync, it modifies a couple paths, and it installs a few plugins, which you can see here.

You can see all the different client apps that it supports-- so Windows and Mac and two different flavors of Linux-- and then the different installers for-- this is Maya-specific, but you get the idea there. So you install the plugins. You also install the client. And I'll show you the client in a second, but it's basically this little guy here that runs in the background. You have to keep this running in order to help Maya communicate with Google Cloud via Zync.

So if I'm ready to render this, and I've done my test renders locally, or I've set up my scene properly, pretty simply, you just click the button. And it will perform a bunch of actions. Let me just make sure I'm logged in.

So actually, before I do that, let me show you what the-- the client app, pretty much, is you just set it and forget it. So you're assigned a URL so that you're able to go and access your jobs and watch your job logs and everything. And I'll show you that in a second. But essentially, you do this through your Gmail account.

If you have a corporate Google Apps set up, which I believe we now call G Suite-- if you have one of those, you can log in via that. But otherwise, you have to log in with a Gmail account or a Google account if you work at Google. By the way, when you sign up for Google Cloud Platform for the first time, you get a \$300 credit, which you could use for Zync, as well.

So I encourage you to just sign up. You do need a credit card, but it will only bill you if you don't cancel-- it won't let you spend beyond the \$300 credit unless you give it confirmation. So it's not like one of those evil multi-level marketing things.

There's just a few definitions of parameters here-- how many transfers to perform at any given

time. This will populate once transfers occur. No big deal. But we've got to keep the Zync client running in order for this guy to launch. So this is the interesting part in that you could choose the number of machines.

This is a 96 frame sequence, so I'm just going to assign one, machine one VM per frame, quite simply, because you have, at your disposal, up to 500 machines, which is super awesome. So how long will this take to render? Well, what's the longest that one frame takes to render? It's about an 11-minute render, by the way.

You can choose from your machine types. And here, in Zync, you've got a predefined set of machine types. In GCP, you've got, not an unlimited number, you've got a much different selection of different combinations of virtual CPUs and RAM. And you could dial each of them independently of each other. But for Zync, you've got defined set of images.

And it goes from 8, 16, 32 CPUs. There's an alpha of 64 CPUs right now. So if you've got a super crazy, heavy job, you can run it on that. But alpha, in Google speak, means it's beta in other speak. So it's like, I don't think we officially support it, but you could try it. Any errors that occur, I think, aren't part of our service agreement-- something to that effect. But anyway, I've tried out the 64 core machines. They're awesome.

And then, there's standard and preemptible. Preemptible VMs are about 70% cheaper than regular machines, but they can only run for a maximum of 24 hours. And given a 10-minute notice, they can be taken down and reassigned. They're, essentially, excess capacity that we sell on top of the standard VMs.

So they're super cheap. And if you've got a render that's less than an hour, you should definitely use them because the chances of one getting taken up in an hour, or even two hours, is pretty slim. If a machine gets taken, and obviously your job will be killed, you're one frame will be killed, it will automatically get picked up again by another machine.

So I'll set this to preemptible. We'll do 32 core. This gives me a cost per hour, which sounds like a lot, 123 bucks, but we're not going to run this for an hour. We're going to run this for probably less than 10 minutes. We can click the cost calculator, and it comes to this page, which goes over some of the details that I was just talking about.

Let's say preemptible. I'll choose the 32. Host Application-- I'm going to use V-Ray for Maya. I kind of take issue with how the UX of this was designed, coming from a product design

background. This is kind of backwards. But it's assuming that you've run test frames and asking you how long those test frames took to render.

So let's say I rendered five test frames, and it took 10 minutes total to render, and there's 96 frames in my job. I'm going to click Estimate. So it figures out that-- what did I say-- I said five test frames divided by-- it's about two minutes a frame, therefore, total estimated cost is \$4.13, which is pretty good considering you've got a render farm at your disposal. You could render these things all day and not spend over 20 bucks. So for a small business or medium-sized house, it's a pretty good cost. It's a pretty low cost.

You've got this notion of projects on Zync. So these are, essentially, where your storage happens on the cloud and how the billing occurs. I've only got one project, but I could create a new project if I want, here. I'm going to leave this on this Zync demo project. You can assign different IDs and different priorities if one job is already running.

So a parent ID is a job that's already running on Zync that must finish before your job starts. So I'm just going to leave that blank. There's this notion of Upload Only and Skip File Check. So what's going to happen when I hit Go is it's going to start looking for all the data that's in this scene, whether they're referenced Maya files or textures or referenced-- it could be animation nodes or any type of cache files.

And it's going to gather all those up, and it's going to upload them to the cloud, to Zync. If you don't want to perform the render, you only want to upload only in order to prewarm, I believe we call it, the render, then you can just do Upload Only and it won't actually perform the render. Conversely, if you have some missing files, or you have some new files that you don't want to upload, just go Skip File Check, and it will only use the stuff that's up there on the farm.

There's some plugin errors that might occur in certain plugins. And the developers have come up with a workaround for that, so you could just ignore Missing Plugins. You can Email On Job Completion, as well. And here is where we get to choose the renderer. This UI will take most of the render settings from Maya. However, you can override a few of them, such as the renderer.

What's really cool, also, is that, because we have a relationship with Chaos Group, which is the developers behind V-Ray, we also have available the nightly build. The Chaos guys are crazy about updating their software. I don't know if anybody's used them, used V-Ray or

interacted with Chaos Group, but they put out multiple releases, sometimes even in one day. So if you call them up, and they say, oh, yeah, that's fixed in the nightly build. There was this bug that you reported. You can use the nightly build, and it's like interacting with them one-on-one.

Distributed Rendering only works for V-Ray standalone. Right now, we're going to use V-Ray for Maya, which is, basically, Maya running the internal V-Ray plugin. To do V-Ray standalone is a bit of a different animal. We're not going to really cover that today. Obviously, if you turn that on, that checks automatically. So we can override the frame range, we can say the frame step and the chunk size.

Because I'm rendering one frame per VM, I'm just going to say the chunk size is one. Chunks are the number of frames to render per VM. If I have different layers here, I could choose which layer to render, which camera. I only have one camera. I can override the resolution. And this is confirming that I've logged in as my Google account. And just launch the job.

Obviously, you need an internet connection. So you can't work offline. But if you're talking to the cloud, it's not really that useful. So job submitted to Zync. Let's first look at the client. So if we look at the log, we're going to see-- let's actually wait until we're going to go to-- where am I, where am I-- the web console here. Go away, pop ups.

So here's my job. I've got seven other jobs that I was working on earlier. And once the upload starts happening, you'll see some interesting stuff going on in the client. It's going to report how it's uploading. And here, let's watch this guy for the log. So here, it's reporting on different tasks like making sure that textures are synchronized and everything. And now, we're at the point where we're waiting for instances.

So it's queueing everything up. And you can see that it's queued up all these frames. And if any of you are familiar with a render farm or render manager, this should be familiar to you. This runs in a browser. So you could access this from anywhere. I've got my own personalized URL. And when you sign up for a Zync account, you define your own URL.

So that means that I could access it-- I don't know. I guess it works on mobile. I haven't tried it. But you could be at your home machine. You could be on the road. You could be anywhere and access your running jobs. Or you could relaunch a job if one failed or something to that extent.

So here we're starting to pick up machines. It gives you the instance IDs. Anything else? There's no transfers to report here. But there's transfers that will occur here. Let's just stick to the main page and just go through here.

So on the left here, this is my list of jobs either completed or canceled or running. Let's turn off Show Completed. And once I click on them, it's going to give me a status as to what's happening with the upload and what's happening with the render. Still, we're right now waiting for frames to queue up on instances.

And right now, because they're preemptible instances that I've requested, it might take a tiny bit longer. Usually it's three, four minutes in order for them to pick up. But here, we've got a bunch of different parameters, just to verify what's going on.

What's interesting and important to point out, though, is that-- the question that you might be asking yourself is, well, how does Zync handle all the paths if everything could be all over the place? I've got textures on this volume. I've got cache files on my temp directory on my local machine. I've got a scene file that's on an external hard drive or somewhere else on the cloud. How does Zync handle it?

Well, Zync actually, simply-- and it's quite elegant-- it simply replicates the file path, the absolute file path of everything. And I can show you what I mean. If you click on Browse Files up here-- I can make them [INAUDIBLE]. If you click on Browse Files up here, you can see that Zync has replicated everything, all the way down to, I guess, the lowest common directory.

So all the projects that you create on Zync will appear under this Projects hierarchy. So I don't have a directory called Project locally. But this is the project that we're working in. So everything appears in my user directory. So it starts all the way up at Users, and then, I've got this stored, actually, on my Google Drive.

So it's replicating everything, which, at first, might seem silly. But don't forget, this data is transient. This will be deleted, or can be deleted, after a certain period of time. So the fact that the paths are crazy, remotely, doesn't really matter. And again, this is only so that the paths don't need to be modified at any given time. You can go-- oops. Not Support.

If you go under My Account, this is where you can change a bunch of drive mappings. You can also add custom paths. So if you need to override certain paths for plugins or different

renderers, you can add these things. It's, essentially, defining the environment that's running on the remote machine.

If I go back to the main page, let's see what's going on. We still haven't picked anything up. Give it a minute.

We're losing people. Come on, everybody.

You can define individual users that you can keep track of. So if you're an administrator, you can have everybody in your organization or in your facility act as a different user so that the billing can be tracked separately. You can see, hey, this one artist seems to be submitting 10 times as many jobs as everybody else. He's spending 10 times as much money as anybody else. Let's go have a talk with him.

And if you look under Usage, you could see all of that billing data. So we could see how much I've spent in the last number of days or the last number of-- based on time or broken down by jobs. For example, the preemptible total render time that I've run recently, in the last few days, is 11.26 hours. It costs me \$21.97.

You could break it down by software. So this is all the V-Ray usage by projects. So this is probably the handiest. So if you're a producer, you want to be able to see how much did we pay for rendering on this particular job versus this particular job and bill the clients accordingly.

And that's another thing, you're moving to a different model, where you have to estimate how much beforehand. You're going to have to estimate before this how much to charge the customer for your own render farm usage or for some sort of render service or co-location or an on-prem render farm.

Well, here, you could bill them after the fact, based on actual render usage, should you want to. I mean, some people kind of want to keep that more of a black box in that they add a certain markup on top, but I'm not going to tell you how to do that.

Let's see if we got any picked up. There we go. So we got one picked up. As soon as one picks up, usually a bunch of them become available. And if we look at our file log, there we are. So we start to see different frames computing.

And again, you don't have to worry about where the files are going, what data center it's running on. And you can keep track of all these by clicking on the log link next to the job type.

So this is going to give me a live stream of the log file running off the machine.

And unfortunately, I made the window too big.

So obviously, if you've got a really slow render job, you could watch it chunk by, and you could see what percentage it's on, just like any other render farm software, which is super handy. And this is based on a single frame. If you have chunks, you'll be able to see the frames complete as they track down. And this is simply the standard out of the renderer itself. So it's no big deal.

Should you need to change the order or the priority of a job, over here, on the right-hand side, based on the selected job over here, you can restart the job, cancel it. You could change the priority of it in the Job Options. There's this thumbnails functionality, which works on stuff that's already rendered.

There's also the job log itself, which is the overall Zync log, not the individual renderer's, but it's showing what nodes have been picked up, whether they've spun up, their IDs, all that sort of stuff. So a lot of this is useful to anybody who's monitoring the renders and may need to report an error.

Or maybe, by chance, one of these machines keeps dying, they can identify as such. Not that I'm advocating that that ever happens. But these things do happen, usually at 3:00 in the morning, when you're trying to deliver a job.

So the Zync UI gives you all the tools that you need to be able to retain control over all of your different jobs. So there, you get a progress bar showing levels of completion. And once the jobs are done-- see more are picking up-- downloads start occurring. And you can watch that in this Transfer section here.

So I've got one frame that's already rendered and downloaded here. And once they're downloaded, they disappear. But once all these guys start completing, you'll see a whole list of them queued up. And you'll see progress as to how quick they're downloading.

And if I switch to my terminal here-- don't get scared by the text or by the shell. So frame one here is complete. And I'm going to use RV, another Autodesk product. And here, we've got frame one of our robot guy. And this is a standard V-Ray render with all the separate passes. It's EXR multi-layers. So it's just as if you were rendering it locally, but you've got access to many, many cores, many, many prox.

So let's let this guy cook for a bit, and we'll hop back into our-- any questions so far about the Zync UI? Yeah?

AUDIENCE: Can you apply [INAUDIBLE]?

PRESENTER: To different jobs? Probably not. Job Options do not permit you to change that. No. It's got to be per user. So if you had different users running different jobs, you could do that because you're logged in as an individual, based on this URL. So if you go to your account, you can change the variables there.

You can upload scripts, by the way, that you could call from an absolute path from within your scene. So if you have a Maya file, for example, and you want to run a pre-render script, you could call a MEL or a Python script in here. Got to make sure all the libraries are present or all the pointers are present and everything, which could be a bit of a challenge to set up, but it's totally possible.

And then, you could put the logic inside those scripts as opposed to inside the render variable. Yeah?

AUDIENCE: Can submit bigger jobs [INAUDIBLE]?

PRESENTER: Yes. There is a Python API that the Maya Zync API talks to. And all the code is totally open and available, free to modify. And I'm not sure what the support scheme becomes once you start to modify, but we actually provide-- you can download right from GitHub all of the scripts and all the code. Yeah. You're totally free to address the API and do with it as you wish.

I mean, you could override the monitoring here because that's all that the browser monitor is doing is reading this with a RESTful API. From within Maya or from within any application, you could write your own client for whatever application you want. That's a good point.

I'm relatively new to Zync, by the way. And I'll talk about this a little bit later in the presentation but, you could effectively, using this API, you could launch any job.

You'd have to classify it as one acceptable job type under the Zync environment, but you could. You could say, if you had submitted a Revit job-- but then, that's the problem-- is the renderer won't run. So if you could figure out a way to do that, then you might be able to.

I was talking to a few people earlier at lunch about other software that we might want to allow

to run on Zync, including architectural or design or BIM rendering software. And if you guys are interested, please come talk to me, or I'll send out a follow-up email to everybody on the list, everybody registered, and at the end, I'll give you info to email for questions. Yeah?

AUDIENCE: Does it work with [INAUDIBLE]?

PRESENTER: No. And we're always looking at new software. I can't make any promises, but we're looking at it. But yes, absolutely, because I know there's a huge architectural--

AUDIENCE: You should really [INAUDIBLE].

PRESENTER: Google Cloud supports Windows, so there's no reason we can't support Windows in Zync. It's just a matter of money and time. And Google runs-- they don't have a whole lot of money to go around. That's a joke. That's the second joke of the--

But it's assigning resources and solving a whole bunch of other problems that Windows presents. Also there's a ton of plugins and separate licensing that is a lot more complicated than just running it outside of Maya, from within Maya. So that would be additionally challenging. And there are certain plugins that we offer licensing for, but that sort of gets into a longer discussion of how those come about.

So we're almost done rendering, and if I look at transfers, you'll see a bunch of them just sort of zipping up and, here, I'll even-- I guess you don't need to see the actual files, but it looks like they're all downloaded. So if we look one last time at the client app, you'll see that it gives confirmation that it successfully downloaded, whatever, whatever, how long it took to download, et cetera.

And should you get any errors or any of this fail, you can re-queue a download, going back to the main page, clicking on this guy. Select one or multiple jobs and just go to the Task menu here and do Re-queue Tasks. You can also skip or force a re-download. Re-queue will re-render it, obviously. Re-download will just re-download it.

Right now, there's no way to access this project data as a Google Cloud Storage type model. We might introduce that if there is interest in the future. But obviously, anything you mess with up on the cloud may directly affect what happens when you send a renderer up there. It might cause some re-synchronization, or it might cause some errors. So these are things that we're thinking about.

Yeah. I mean, there's the render. Let's see how much this cost. I think I'm missing a frame. Oh, it didn't-- what's the hot key to reload. There we are. If anybody wants to know more about RV, just let me know because I'm a huge fan of RV. We use it in visual effects to view individual file sequences or QuickTimes. It's just a really slick design piece of software that Autodesk acquired not too long ago.

So what did I pay for that? I go on to My Account. Go under Usage. And let's do Today. So that cost \$21.58, which, for using 32 core machines, is not bad. You could cut the core count down, and it will cost a little bit less, but it'll take a little longer because, obviously, there's a law of diminishing returns once you get to a certain number of cores that the thread ability of certain renderers might top out at 32 cores or 64 of course.

So there might be no need to go up to 32. But again, these are preemptibles, so it'll be more expensive if you were to run regular machines. But again, none of those machines got preempted. So they operated almost as if they were standard workstations.

AUDIENCE: How linear is the relationship between the price [INAUDIBLE]?

PRESENTER: It's not perfectly linear. A 16 core is not half the price of a 32. I think it goes up slightly. But you've got to think about what you're paying-- your artist time, the wait time, the wall clock time. And they're going to be a lot more expensive than the difference in a few dollars on the render farm.

So you've got these jobs that might take an hour frame to render, and instead of waiting eight hours for the whole sequence to render, you just spin up as many VMs as you have frames. And you're only waiting the length of as long as a single frame takes to render, at the longest time.

So any other questions? Follow up questions so far?

There's some new stuff that we're talking about, we're thinking about on the horizon-- is, like I was saying, introducing new software applications or DCCs, Digital Content Creation Applications. I'd like to gauge the interest for architectural and engineering rendering from this room. Obviously, the M&E track at AU is pretty thin, so I'd like to see what the interest level is here.

And it sounds like, from what AWS is offering, would you say this is easier to set up and run

than what you were doing on EC2?

AUDIENCE: Yeah, in a lot of ways. In a lot of ways. I mean, but then, the trade off is less [INAUDIBLE].

PRESENTER: Correct. Yeah. That's kind of how this product is angled, is that there's a lot of stuff that's already been solved for you, but you don't have ultimate flexibility over that. If you need more flexibility, we can talk about the Google Cloud platform in its raw format. That's something that I'm doing right now, is I'm writing up a bunch of solutions.

As a solutions architect, I figure out how to run pipelines and workflows that relate to this on the cloud, using the basic cloud offerings from scratch. So I will author these documents that will walk you through that and provide information on that. But yeah, this works out of the box. And it, basically, does everything for you.

It works really well with the other packages, the nameless packages in similar fashion. There's just a menu item with similar options. And it uploads in the background, which is really awesome. One thing I didn't mention is the caching. So once it's uploaded, let's say you've got 3 gigs of textures for this character that I've uploaded once. And I'm only changing the animation on the character, and I need to re-render.

It's only going to send up the updated animated Maya file. It's going to compare. It's going to do a diff between the textures that are on disk and the textures that are on the cloud. And if nothing's changed, if the checksum is identical, then it won't re-upload them. So you save in time and bandwidth, as well.

That's one of the reasons that you can do the upload only-- is, if you have a huge amount of ancillary data to upload, you could, maybe, before you go home, you could set to Upload Only for half a terabyte of data. But the next day, you come in, and you can just launch your animation files. All your data is up there, and it's ready to go.

The last point here, the plug and play development-- setting up a VM and storage is not difficult on GCP, but if you had to do it for every single project, and you wanted to use an API, you wanted to do it from command line, or if you had to do it by the actual console, it gets kind of tedious if you're doing a lot of jobs.

And to have to monitor that and deal with all the storage-- and you're also paying for that storage, once it's up there. And that's different with Zync. If I look at-- where was I here-- if I go back to my-- where is it again? Browse Files-- no, I'm sorry, it's on the My Account. You can

see Projects. You can see how much everything's taking.

And this one project that I have up there is 3 gigs in size. You can delete a project if you're done with it, if you want to for security reasons. If you're concerned about that, that's another thing that we should talk about, is this level of security that Zync affords you, running on GCP. But you're not charged for data that's stored up on the cloud.

I think the web management console is pretty simple. There's a couple of UI choices that could be improved upon, and we're constantly working on improving this. But it does what you need it to do. And it runs without having to refresh, through the magic of HTML5.

And the ability to monitor the logs, I think, is pretty powerful. They're selectable, so you could copy and paste them into a bug, if you had to submit it, or to somebody who is troubleshooting a job for you.

This is kind of an older list. There's more customers using Zync today than when this slide deck was built. But some of the larger facilities have already found success running Zync and have dumped their local render farms. And it could kind of be a bitter pill to swallow to tell your accountants or your CEO, your CTO, that million dollars that you spent on a render farm is now obsolete.

So you could sort of approach this as a hybrid, so that if you run max capacity on your local render farm, you can use Zync for burst capacity, we call it. And that's just it. You're not penalized for not using your account, not using your sign up or even your credits.

So that's what I got. If there's any questions, you can email info@zyncrender. If you want to talk to me one-to-one, you've got about 10 minutes. Thanks for watching, and check out Google Cloud Platform.

[APPLAUSE]