EPISODE 1379

[INTRODUCTION]

[00:00:00] KP: Modern businesses run on the cloud, and increasingly so, they run on multi cloud infrastructure. As any growing company can tell you, cloud costs can easily run far out of control. Today's enterprises are trying to deliver new products and services at a fast pace that needs to be done in a cost effective, ideally cloud agnostic way.

In this interview, I speak with Jake Reichert, VP of Engineering at Yotascale. We discuss Yotascale's solution and how they've helped companies like Zoom, navigate the challenges of rapid growth. Yotascale scale offers a comprehensive spectrum of cloud cost management solutions, and we get into how modern enterprises are using these tools for optimization, governance, and more.

[INTERVIEW]

[00:00:45] KP: Jake, welcome to Software Engineering Daily.

[00:00:47] JR: Thank you. Glad to be here.

[00:00:49] KP: So, I'd love to start with where your journey begins as a software engineer. What was the hello world for you?

[00:00:56] JR: Yeah, so I started my career as an engineer, probably about 20-ish years ago. I guess, with a lot of people in the Bay Area, I was in school the time that the dot com boom happened. There's a lot of things about it that were appealing to me. My background was not in computer science, but it was in physics and mathematics. So, pretty adjacent field. I pretty quickly moved into engineering leadership positions after that. Part of that, for me was wanting to execute my vision at a larger level. Part of it was also just realizing that I felt like had a real vested interest in improving the quality of life for all of the engineers on my team and wanting to learn the skills around doing that.

So, sort of straddle the line between going back and forth between more chief architect, technical co-founder roles, and then hopping back to the other side of the line to do more people management, VP engineering roles. So, I've got a pretty good set of experience in both of those. Most recently, I came from a senior leadership position at OneLogin, which is a company that makes a single sign on software, in similar vein to something like Auth0 or Okta, and then prior to that, I was also in a senior leadership position at Amazon, working on Amazon Music.

Yeah, one of the things that got me interested in where I'm at now is, just looking at my past, and thinking that as long as I've had infrastructure budget to manage, looking at the spend that our company had, the number one item was always headcount. Number two item was always infrastructure, which of course now is 99% of that is cloud spend at this point. So yeah, so looking at the Yotascale seem like a pretty obvious product to me of kind of that lightbulb, aha moment of, really excited to be looking at solving those problems for people that engineers like me have had in the past.

[00:02:40] KP: Well, knowing that info is such a big line item for me, and that 90% of it's in the cloud, I certainly want to take a good detailed look at how I might be able to drive that down. But at the same time, I got to run my business, where can savings come from?

[00:02:56] JR: Yeah, that's a good question. I think this is where a lot of CIOs or CTOs end up just sort of stopping is with exactly that question. It's sometimes hard to look at the data that you have coming out of your cloud infrastructure and determine how to make those changes. There's a couple of different problems there. So, one of them is cost attribution. That side of things is really just understanding, where's my spending going to, right? It's hard to make good decisions, if you don't really know how to break down your cloud spend along lines that make sense to your business. That doesn't necessarily mean there's like this cluster that's called this or that cluster over there that's in this region.

Now, that might make sense to some people through one lens, but not to other people through another lens. So, part of it is really having good tools that make it easy for you to drill down and sort of this business intelligence way that people are used to, using some tools like that, to sort of generate these questions in curiosity, like I said. This is interesting, I see that this thing has really spiked over here. Let me drill down and double click on that and go into it and see what

that is made up of. And that could be subdivided along lines of different departments at your company. It could be subdivided along the cost that you attribute to different customers that you have. It could be R&D versus production ready environments that you have.

So, there's different ways you can slice that. But all of them come back to answer this question of how do I take my mental map of the business and apply that with the tools that are available to the actual cloud expenditures that we have. So, that's one half of it. The other half of it is getting good recommendations on that Intel. And that's where we're at a point in history where there's a lot of these tools that companies like ourselves are building that are ML based, basically using big data approach to to map that, not just take this linear approach to saying. Well, it looks like costs are going up on this service at about X per month. Maybe you should jump up to this sort of reserved instance at the size at Amazon that's going to better serve your needs.

That's something you don't need a tool like us to tell you, but most workloads aren't that nice and predictable, as any experienced CIO can tell you. They fluctuate, they go up and down, they're seasonal, they're hourly. And so, it's helpful to have a tool that can really look at those and say, based on those and map those to what's available in your particular cloud infrastructure, how do you actually want to make use of those tools to right size them. And that's something that you really can't get by just looking at the tools that come directly out of the cloud native tools like your cloud watch, and AWS, or something like that. They're just never going to give you quite that level of granularity.

[00:05:31] KP: Those services offer me some niceties, like I know, I can tag all my resources, and maybe if I tag them very intelligently, I can sum them up in ways that are useful to me. Do I have to have done my homework in that regard, and all the chores of tagging everything in order to use a solution like Yotascale?

[00:05:50] JR: Yes and no. I mean, I think that as with any tool, any business intelligence tool, the more you put into it, the more you can get out of it. But we also know that part of the value add that we deliver to people is that we are doing a lot of the heavy lifting so that others don't necessarily have to. There's a couple of ways that we can do that. So, one is through a set of tag normalizations that we have. We have algorithms where you can take tags that might have

been developed by different parts of the business and look similar, but maybe people like use singular in some place plural and other parts of the business. That's one very simple example. But ways to sort of consolidate those into one bundle, they look like the same thing.

Another way that we can do it is we can do allocation that's based more on what's going on inside of your virtual private cloud, right? So, it could be that you have things that are tagged to have some particular semantic meaning for business line. But you might say that only gets you so far, because then there's all the stuff we didn't get around to tagging. So, for those, that's something we can bundle together and say, “Okay, well, let's split that along some other dimension.” Maybe let's look at that based on the type of application it is, whether it is, we're looking at AWS, whether it's an EC2 instance, or whether it's a database, or it's something to your that your Aurora database versus your columnar database that you have set up. And that's another way that we can then subdivide it.

So, there are ways like clever tricks that we can use in our tool to drill down further and get some higher fidelity information even in places where data isn't tagged. Finally, we have some clever tricks we do with things like tag promotion, right? So, one of the things about tagging is that it's not just you tag everything, and it's all static at one level. We have a very complex or rather than the ability to make things as complex as you wish in terms of the hierarchy, instead of a business unit that might have smaller business units, which have projects, which have R&D versus production ready code, and you might want to have attribution of all of those different levels.

We have some clever things in there, where if you have things tagged at a very granular level for some areas, but not for others, we can do things like promote some of those tags up that chain to say, “Okay, well, something that's further down, if we don't have something at a level further up, let's just propagate those up and treat those all as like the top level tags.” So, everything gets nice and normalized, you don't have one giant bucket that just shows it's unallocated, because we can introspect further on the stuff that's further down and get more granularity out of that stuff that's further down.

So, those are kind of the two halves of where we add the value. The first half is on that side of cost visibility. The second side is on actually making the recommendations for how to turn those into actionable practices in terms of making different buying decisions with your cloud providers.

[00:08:26] KP: I've been in a couple of situations in my career where a company does something very good. They let's say double their customers, but in the time, they double the customers, they did a 4x on their cloud costs. So, something's wrong, got to be addressed, good problem to have, but they need some early wins. What are some common early wins people get when adopting your solution?

[00:08:46] JR: Yeah, that's a great question. So, when I look at a couple of our really successful customers, we're one of the key parts of Zoom's infrastructure. This is a good example, where I think, as we're all aware, March of last year, they just went bonkers in terms of the need, that they had to scale their infrastructure massively, is everybody moved to working from home, and they did a fantastic job of scaling up that infrastructure. But then, they also had a lot of concerns about their cloud spend, right? That didn't come for free.

So, this is an example of where we were able to apply our software to some of their problems. They have a massive amount of spent, half a billion dollars, and we were basically able to substantially improve their ROI during that period of time in which they work. They were scaling very, very rapidly. Another good example that we have of that, of one of our company customers is Hulu, they were kind of in a similar boat where their business model is, it's clear that you pay a fixed fee as a customer, but you basically stream all you want. And so, it's important for them to be thinking internally at the engineering level, not just the business level about how do we build our tools in such a way that we're being really cost conscious and instill that is a value that engineers have they're building their products.

This is an example where we were able to give that same visibility to another customer with, again, a really large expenditure. And in this case, we're able to very quickly deliver a six time return on investment to them. So, for every dollar that was spent on on Yotascale, they got $6 back in return for that. That didn't take very long. So, you can see how this is something where if you put these tools into play, and you start building them into the consciousness of the

engineers on your engineering team, you very quickly see a return on that investment of controlling that bottom line of your cloud expenditures.

[00:10:34] KP: More and more, I know a lot of large enterprise, even small medium, I suppose. But especially large, they want to have redundancy, they want to have multi cloud setup, actually not just for redundancy, but for, I guess, leverage and cost optimization and things like that. That's certainly a much more complex world to have two or three clouds in play. How does your solution bridge these gaps?

[00:10:55] JR: Yeah. So, one of the things that I found really interesting when I started working at Yotascale was looking at the multi cloud problem and saying how big of a problem is this, really? And it turns out, it's actually quite a big problem. Something like north of 75% of enterprises today are already multi cloud. And Gartner proves it predicts that by 2025, can be over 90%. So, I think that that really reflects a growing recognition among enterprises that – well, really, have a couple of things. One is that you do need to have, like you said, redundancy. You don't want to have all your eggs in one basket, one provider. You also don't want to get so invested with one provider that you're sort of beholden to them, it leaves you more options if you're multi cloud. And finally, there are just some tools that are available at one cloud provider that simply aren't available and another one or aren't available anywhere close to the same price point. And you want to have the flexibility as a seasoned CIO to be able to manage that and distribute your workloads wherever it's going to make the most sense for your business.

So, I think that this is why our launching this multi cloud product has been really, really key to our long-term success and long-term success of our customers, because we're basically just following that trend line of ensuring that we have tools that customers can use to prioritize their workloads across not just one cloud, but across multiple clouds.

As of today, we are doing that across both AWS as well as Azure. Early next year, we will also expand that to be going across GCP. And right there between those three current trajectories, that puts us on track to cover about 70% of the cloud infrastructure that's out there today, we'll expand other things in the future as well. The nice thing about this is gives you a single pane unified view across your multi cloud environment of where your spend is going.

So, I'll just give you one example. Let's say that you've built a new product that is in R&D, that's something that you want to distribute that workload across Azure and EC2 instances, and it's something that you want to be able to get some comparison of under real world usage as you're scaling it up, how is it costing you money across those different platforms? So, rather than having to go into whatever your cost explorer tools that are cloud native, that are given to you by the cloud providers, we actually have the ability to give that all to you in a single viewport, where you can look and say, if your instances are tagged a particular way o'clock across these cloud environments, you can then do an apples to apples comparison and say, “Okay, how much is this costing me in total, across my whole cloud build out, regardless of where it lives?” And then further break it down and drill in and say, “Okay, well, how much is it costing me over in AWS? How much does it cost me in Azure? Am I getting more bang for my buck on one of those platforms than the other?” And also getting recommendations based on those.

So, you can see, based on the trend lines that are that are occurring there, which might be messy and take some of our, you know, machine learning smarts to help you figure out, but that's something that we can then use that to say, “Okay, given the growth of the product in these over time, how do we expect you to have to right size your workloads in those environments, maybe you want to shift more of the workload from one provider to another.” Maybe for that particular workload, it makes more sense just to run it in AWS, or just to run it in Azure. Our tools will help you to figure that out in a way that's not necessarily easy to do if you don't have sort of a neutral third-party arbiter that's able to view your expenditure across your whole cloud build out, as opposed to just being very narrowly focused on looking at the problem set of one particular cloud provider.

[00:14:30] KP: Many of the services in particular, I'm thinking of cloud storage, S3, as your storage, that sort of thing. There's a pretty good parody across the clouds. A developer who's worked with one. I mean, the interface is slightly different. There's different bells and whistles, but you can kind of switch between them, all the abstractions are there. I wouldn't say the same about the cost exploration stuff. That's an area where the clouds are all kind of particularly unique in the offering they give you and I don't think there's any sort of open schema kind of thing that I'm about of. How do you blend these three heterogeneous data sources?

[00:15:05] JR: Yeah, it's an active area of exploration for us now. I'm literally having conversations with my engineers about it this week. It is an interesting problem as we start getting further and further into the multi cloud space, because on the one hand, you want to be able to give people a sense of how they can pair like, applications across clouds, right? So, let's take an example of something like, let's say something like an EC2 instance in Amazon. And the equivalent thing is Azure virtual machines.

So, on Amazon, you have EC2 instances. On Azure, you have virtual machines. They're essentially the same thing and they're categorized similarly. They have capabilities based around RAM, CPU, storage, et cetera. So those ones, you can pretty easily come up with some sort of a data dictionary to compare like elements to like elements. You're absolutely right that it does get more complex, though, when you get into these higher-level services. Like, one of the things that we're actively looking at now is Amazon Fargate, which is kind of made up of a mix of compute resources, storage resources, orchestration resources, that's something that doesn't necessarily have as simple of an equivalent to in other cloud infrastructures.

We do want to build these in such a way that we're not running this risk of kind of overfitting the problem and trying to jam too many things together in the interface that aren't really similar, but at the same time, giving you enough commonality that you can see what are the trends and factors between different cloud infrastructures. And getting, in that sense, kind of as much of a realistic view of the world where it's like, “Okay, here's the things that are broadly speaking the same across AWS, Azure, GCP. Let's compare those things as like to like.” With slight differences, we're going to treat them that way. And then maybe there's resources that we don't treat them that way, that it's something that we put them into sort of a special item inside of the viewport. A lot of greenfield development to do in this area, a lot of problems still for us to solve, but that's kind of the approach that we're taking is trying to think about, what are the areas that are common? And where are the areas that truly desperate enough that we should just call that out and treat that as its own sort of new resources unique to that particular platform?

[00:17:13] KP: So, when a new client signs up for Yotascale, someone asked to be, I guess the operator, right? Log in, look at the recommendations, consider them, and maybe execute on them in some way. Are there common job titles that take on that responsibility? Who's really filling the role?

[00:17:32] JR: It's been an interesting mix, depending on the particular company. Generally, it's somebody from the operations team. Sometimes it's a DevOps engineer, sometimes it's a platform engineer, somebody who's maybe a little more code focused. Sometimes even been site reliability engineers. And to some degree, we even have people who are in more finance roles who are involved in this process. Somebody who's maybe like a finance manager, or even a director of finance, who has a vested interest in understand these costs a deeper level. Clearly, they're not going to be as involved in going in and working on things like Terraform templates to add tagging to instances that are getting deployed. But they will provide that sort of business intelligence that we need to say, “How do we categorize these things in a way that's meaningful?”

One thing I will to say on that point is, one of the abilities that we have that's that's pretty great is the ability to view our data slice from multiple lenses. This is something that is probably familiar to people with a bi background, but you might have a different way that you want to see your data divided up depending on what job function you have, or which problem you're trying to solve. As an engineering manager, or let's say as a CIO, you might have more of a vested interest in understanding which one of my clusters is the one that is performing the least well? I've got older development versus newer development. Theoretically, this newer build out that I put in place is supposed to save money, is that actually true? So, you might be looking through, out through this more sort of technical lens.

As a product manager, you might be looking at your older product lines versus your newer product lines, which may cut across those in a very different way, and you want to make cost attribution done through that lens instead. So, one of the things we do with our tool is make it so you can create these multiple lenses, and you can actually see things, the data sliced in those different views. So, somebody who is a CFO might be able to see data in a different way than somebody who's a CIO, and the platform has equal facility and be able to give you a view into it through both of those lenses, which might answer two different sets of questions.

But yeah, because of that, we basically see people from multiple job functions, fulfilling these roles, but typically, it falls into those areas, either finance on the one side, or more typically on more like a DevOps SRE platform role on the other side.

[00:19:45] KP: And when that person goes into the recommendation system, what are some of the popular or common recommendations that you see people getting issued?

[00:19:54] JR: So, a lot of the recommendations that we see, well, there's kind of two almost tiers of recommendation I can see. This gets a little bit into some of the types of visibility that we give. So, the visibility that we give is not just on the instance level, right? So, in Amazon, we're not just looking at your EC2 instances and tagging those, and giving a recommendation on those. That's one thing we do. But the other thing that we do is we also can apply that same strategy to tagging Kubernetes pods, or ECS pods, or basically any containerized workloads. So, that then you can see how those containers are being deployed onto those resources and how much of those resources they're using.

Now, the reason that that latter part matters is that what we've seen is a lot of companies end up over provisioning resources when they move into a containerized environment. It's not as straightforward of a problem, it's just saying, Well, we have an EC2 instance that we've reserved for this workload, we've got it tagged, we know we've got three of them running, and we know that they're running at about 75% utilization. So okay, we're happy with that. We feel like we're not over provisioned at this point. It gets harder when you're talking about containerized workloads, because those pods can be deployed across multiple clusters in multiple data centers in multiple availability zones.

So, it's hard to know first of all, how do you combine all those together and see the total cost of that thing, but separately, they are at some point running on a real computer, right? I might have four different workloads running in containers that are deployed to a particular instance of an EC2 instance. And I might want to know, “Okay, well how much of the total capacity of that EC2 instance got used by some containerized workload, and then further, which containerized workloads inside of that ended up eating up that bandwidth? And then finally, what was left over at the end?” That last piece of like combining those two together, sort of more the macro level of what's going on with my EC2 instances, and the micro level of how are my Kubernetes containers, actually making use of that capacity, that's then marrying those two together gives you a much better picture of where am I overspending?

To get back to your question of what kind of recommendations are we giving you, it's really kind of two different buckets there, right? One is based on an overall level, where are you basically having workloads that are simply not taking up the amount of resources you thought, or maybe they've dipped over time, because of technical improvements your team has made. You can actually dial back some of your resources to account for that. Some of it is going to be where can you get more leverage by using some of the means of saving money at cloud providers?

So, as an example, Amazon, they can have two ways of doing that, right? They have reserved instances, and they have savings plan. Both with slightly different characteristics, but basically allow you to pay some amount of money upfront, that's going to be a guaranteed amount of spend. And that's going to give you a discount on any of your services that you're using there. We can help you right size those. I can tell you that I was talking to a CIO the other day, who let me know that they know that they're over provisioned, they know that they're spending too much money on their capacity. But the reason that they are not taking action on that today is because it's simply too hard to know how much they should reserve and they don't want to over reserve capacity.

So, they're kind of a little shy about pulling the trigger on that simply because they don't have the fidelity of information that they need to be able to make those decisions. Tools like ours, I think are really nice, because they don't require you to go in and do that heavy lifting, we'll just give you that recommendation based on what we see ourselves. We can actually give you those recommendations about both how to right size your workloads, and not be over provisioned, as well as how you can make use of some of these savings plans, these cloud providers to pay a little more money up front and know that you're then not going to overspend because our tool is able to tell you that you're going to need at minimum that amount of capacity, and maybe have some floating amount of capacity on top that you pay a little bit more for, but you don't over reserve and therefore over provision that capacity.

[00:24:04] KP: Yeah, when it comes to reserved instances, I see a lot of companies in a similar position, they aren't taking advantage of them and there's some hesitation. Do you have a sense of like how much money is left on the table per year or something along those lines? Is this common a problem, seems to be to me?

[00:24:20] JR: Yeah, I wish I knew. I don't think there's any great way to estimate that in the real world, because we don't know what capacity goes unused. There's not any great metrics for that. What I can tell you is that we've seen from our customers, pretty significant percentage of their capacity that has gone unused and were able to dial back down. I don't I don't have any specific numbers in front of me. But what I can tell you is that for some of these bigger customers, it's not like a onetime thing where they instrument our tool. We give them some recommendations about their capacity reservations, they get those and then everything's good. We actually find month over month, quarter over quarter, continues delivering that value, and it ends up telling them on a repeated basis. Okay, now you've made these improvements to your own internal code or processes or ways you process jobs or moving workloads from non-containerized, containerized. Let's bring that cost down further and further, by getting this continual feed of recommendations for how you can continue to buy more reserved instances for jobs, once we feel confident that that capacity is really necessary.

[00:25:28] KP: I don't follow the day to day changes about cloud costs very much. But I will see some announcements every once in a while, that this provider lowered this machine by a penny or something along those lines. Is the market volatile enough that you have to be adaptive to it?

[00:25:45] JR: Yeah, there's kind of two different types of volatility really. One is internal, one is external. So, the one that you're talking about is really more external. That's the fact that cloud costs do change. And they change in multiple ways. They change based on both what are the tools and the pricing plans that are available from your provider or cloud providers, and how can you adapt your spend based on that. Good example of that is that for quite a while now, Amazon has had reserved instances versus on demand instances, but then they've introduced their savings plan as well. So, that's a different way of sort of splitting the difference between those in a sense, those two modes. And so that's something that you have to then as a CIO, think about your expenditure of like, “Well, is that something that I should entertain? How to essentially kind of like the slush fund that I'm using for buying instances? I know, I'm probably going to use it at some point, but I'm going to put at least enough in there, so that I feel like I'm getting value out of it. Well, not putting so much in there that I may be putting money away that I don't need for that purpose.”

So, that's just one example. There are many, many examples like that. But there's also the internal volatility. As a CIO, you've probably got pretty good instrumentation, using your monitoring tools that are going to tell you when utilization is going up or down on certain of your services, when you're just getting hit on more of your services that you built out than others. And that's something that you really need to get ahead of, right? I mean, if you look at something like the case, when Zoom really needed to scale up in response to everybody working from home last year, that's something that was volatility they couldn't really plan for. If they had not had a tool like ours, I think it would have been potentially much more challenging to figure out what were the right triggers to pull to right size their spend in a really sort of extremely volatile environment to know that they're making the right decisions, right?

As we've seen, the amount of usage of Zoom has gone up and it stayed up, I don't know if it's come back down from the peak last year or not. But I think that everybody, what we can say for sure is that looking at it from the position of let's say, July of last year, nobody knew where this was going to go. Nobody knew if this was going to be – everybody's working home for a year, people are going to go back to work in six weeks, it's going to be the next five years like this, it's going to be some proportion of that people working from home, some proportion not. So, I think tools like this really helped people to have that sense of comfort that whether the volatility is internal, based on their sort of own growth or trajectory as a company or external, based on the tools that are available from the cloud providers that are out there. In either case, having a tool like this is going to give you the confidence that you're making those right decisions without having to go and dedicate a whole team on your side to do the research and figure those out, and then just pray that you made the right recommendations at the end of it.

[00:28:32] JR: For a business that's running pretty consistent, let's say 15% growth year over year, stable software, not too much R&D, it's no surprise, you can help them with forecasting and allocation stuff. But when it comes to a more real-world use case like Zoom that's going to hit this hockey stick, how can you make recommendations? Are there any specific challenges there when yesterday's data might be a little out of date?

[00:28:56] JR: Yeah, I think this is where it's really important that your tools are using kind of some of the best of breed tools and thinking out there around machine learning to make these decisions. You're not going to be able to make these linear straight-line recommendations to

people in those cases, which I'll also just mention that even for those companies that have that stable year over year growth, you still really do have a lot of these challenges of cost attribution, maybe the recommendations part isn't as relevant in those cases. But you might find cases where people don't have a lot of visibility into what it is, they are actually spending that money on. And consequently, where there are opportunities for savings.

So, the recommendations even for them can, maybe it'll be different classroom of recommendations, but they might get better fidelity information there. Like, “Hey, your growth growth is going up year over year, and you kind of have the savings plan set up. But now it's probably a good time to carve that out into some spend on reserved instances based on growth.” But yeah, I 100% agree with what you said that if you do have these companies that's going through like a hockey stick, accelerated growth, or maybe there's just some sort of a big bulge in there that’s going to go up and maybe even come back down, really you do have to rely on on machine learning there.

Probably a great conversation for another day about getting into more of the nuts and bolts of machine learning. But really there, it's about just understanding what are the tools out there at your disposal, you know, are using state batch machines or are using neural networks. So, you're just using the – people who know how to use those effectively, I would say are my experience in my professional career, the number one problem that you end up having with machine learning algorithms or faulty is not asking the right question in the first place. You either make it too broad or too nonspecific, and then it's a little bit garbage in garbage out. If you're asking a question that's not pointed, it's probably not going to give you a very good answer. But if you're able to ask a really good question about something like, should I get a savings plan? Should I get reserved instances? What size reserves instances should I be getting based on my current trajectory of workload? That's something that I think that machine learning is pretty adaptive giving you good answers to.

[00:31:00] KP: So, it sounds like I might find some low hanging fruit when I get started. What is my experience look like 12 months later?

[00:31:07] JR: Yeah, you're definitely going to find the low hanging fruit first, as with any tool, and I think that one of the things that we really strive for, as a business is like, well, how do we

make sure that the software is not just valuable on day one, but it continues being valuable? And I think there's a couple ways that happens. One of them is, like I said, the cost attribution. I think, is really, really important. This is sort of like any monitoring tool that you have, right? You're going to notice things right off, you're like, this is fundamentally broken, I'm so glad that I rolled out this tool, and then I found this issue. But then on an ongoing basis, it's going to keep alerting you to those blips. And maybe there's going to be some periods of time where you don't get as much information and there's going to be times where it goes through the roof, right? Most companies are not building software deploying at once, and then it just sits there. They're continually tweaking it and adding to it. They're changing features or adding new product lines.

And as you do that, you can lean a little bit on your learnings of the past to make an educated guess about like, “Okay, well, what kind of cluster should I set up to run this in the first place? What do we think is going to be the right auto scaling rules that are going to get me set up to have the right capacity for what I know I'm going to need?” But that only gets you so far. This is where the recommendations come in of like, , it's a workload that might be similar to what you've had in the past, but it's probably not going to be 100% identical, which means it's going to have different characteristics.

Also, as you mentioned earlier in the discussion, the landscape keeps changing. The tools that are available at and if any of these cloud providers ends up getting more and more sophisticated, they're competing with each other on price, on availability, on the regions that they're deployed in. So, one of the things that I like to think the continual value add of our tool, is that it stays on top of all those changes so that you don't have to. You're still going to need to know at some level so that you can make the right architectural decisions as an engineer, but you kind of want to get out of the business of having to like run all these calculations yourself and figure out like, “Well, should I use this instance type? Should I use this one?” And that's something that we can just sort of hand off to you.

So, you deploy a new workload, or maybe you're moving a workload from a classic environment to a containerized environment, that's another use case we see a lot. You want to get good information about how does that play out in real life? How am I taking those learnings from the past and applying them to the future in kind of a methodical way? And finally, how do you actually make use of those new tools that maybe you don't have any experience with, right? If

you're in an environment where maybe you had some reserved instances, you had some on demand instances, but you didn't have a savings plan set up with Amazon, that's something where you might want to look at and then say, like, “Okay, well, I know this is a thing. I know, it can save me potentially even more money than I'm saving today. But it just got introduced, I don't have the time to research it. I'd like to think that if we're doing our job correctly, we're going to give you recommendations as those tools get rolled out. And we get a good sense of familiar with them ourselves, and how to right size them, we can then pass those recommendations on to our customers about how to make use of those tools in their own environment, and maybe ways that they're not thinking of today.”

[00:34:08] KP: So, I have a side project that I hope one day blows up that's running in AWS right now and the bill is pretty small. You might be able to help me get it down a little bit, but not enough that I'm concerned at this point. But if it does take off, what is a milestone or a rule of thumb, I should consider for when it's really time to put Yotascale peek at my project?

[00:34:30] JR: Yeah, great question. I think that, for a lot of people starting out with a side project, or something smaller, typically speaking, in a case like that, you're probably going to be only running on one cloud, you're probably not going to be multi cloud at that point. You're probably also working on something that has pretty defined usage characteristics, you maybe have a couple of friends testing it out, or maybe in a couple of real customers using it. I think at that point, you're going to notice that you're pretty happy with the cloud native tools. You're going to get whatever data you need out of CloudWatch. It's going to tell you what you need to know, which is essentially is my cloud spend under control? I'm not going to get hit with like a surprise bill at the end of the month.

Where, I think, you want to start looking at Yotascale is something like the case where you start to add some meaningful number of customers, well, there could be a couple of triggers for this. One could be, you either start to add a significant amount of usage to an existing product, and you don't necessarily know where that's going. You know it's going up, but you don't know how far off it's going or at what velocity. The other case where you might really care about something like this is when the usage patterns start to differ a lot. So, what I mean by that is, maybe you start to move from a US base operation where you have this sort of neat curve around US business hours was getting used, you start going global with your product. That could be

another use case where you don't necessarily know how that's going to change your needs over time, because it's not just the total capacity that's going up, it's the burstable capacity is also going up potentially, at times you haven't planned for and made purchasing decisions around that.

And then finally, if you're thinking about moving from a non-containerized, to a containerized workload, that's absolutely a place where a tool like this is critical. It's very, very hard to predict how your costs change when you're moving from a classic environment to a containerized environment. And tool like Yotascale can really help you get at the heart of what are the different cost drivers in there. So, you don't have to be worried that you're either over provisioning or under provisioning and then paying a penalty for spending a bunch of money for on demand instances that maybe you could have with better fidelity information upfront, decided to reserve some things without getting hit by a really large bill for it at the at the end of your first couple of months of operations on that product.

[00:36:53] KP: Moving from non-container to container-based infrastructure, to me feels like the obvious choice, a good step for a company. Are there mistakes people are making? Or why isn't that just a smooth and easy transition?

[00:37:07] JR: I think there's a few reasons for that, and I'm speaking just from my experience of doing that accompanies. You start with that process, and you think it's going to be let's say a six-month project, and then six months and you realize you barely scratched the surface of that. I think part of that is due to a lot of the DevOps work that goes into it, right? The the deployment process for your code in those environments can be very, very different. Thinking about how you map out your network architecture ends up being very different. Also, moving things along the pipeline ends up being a challenge, right? When you're trying to move things from like a, let's say, your dev environment to a staging environment to a production environment, that process can also look very different in a containerized environment versus one that's more of a classical setup.

So, those are all kind of things that makes it challenging to have that mental map of what does this process look like end to end. So, layering on top of that, the cost stuff is like if you can't even come up with a good mental model of how does that deployment process look of rolling it

out, thinking about the cost is like an added vector to that, that becomes really challenging. At one place that I worked, we were moving from this classical environment to Kubernetes setup. And the VP of Engineering at that company was totally on board with the project. He had the full support behind him of the platform team to say like, “I've got your back guys, go ahead and do this.” And it still ended up being a really, really long project, even with a lot of internal support. A lot of that really did come down to the fact that was very difficult to just show how much capacity should we reserve there, right?

We didn't have enough of a baseline about what the performance characteristics were going to look like when you moved from the setup where you had like a load balancer with three machines behind it to these containerized pods that you're just sort of moving out into almost like a metered utility of compute power that you had and saying, “At what point do I hit the threshold where I need to start adding more to my total compute capacity, as opposed to thinking do I have enough machines in this cluster or this cluster, or this cluster?” So, the one factor there is, of course, spinning up more instances to handle that workload.

But then, of course, the other one is like, well, how much is that going to cost you? Right? That's something that if you're looking at deploying that yourself on let's say, a cluster of these two instances, you need to know how many EC2 instances to reserve. But we also have support for looking at things like ECS or EKS in Amazon. Like their containerized infrastructure, of giving you recommendations for how to use that. Likewise, in Azure, we have something for their containerized infrastructure.

So, there are different ways to slice it. It's not necessarily you're just setting up a bunch of EC2 instances, deploying Kubernetes there yourself and managing it. A lot of companies are now also going this route of using the containerized infrastructure at the cloud provider, or again, maybe even a combination of those two for various reasons. We've seen that as well. So you want to get visibility into how much your workloads are costing you across, potentially all of those environments, right? You could have a classical environment without any sort of containerized workloads plus a containerized environment, where you're deploying it to bare metal yourself, or a containerized environment that's fully managed by your cloud provider.

When you start adding that together, trying to get a good picture of what that looks like, not just the start of the process, and the end of the process, but how do you actually carry things through from the beginning of the process to the end becomes really challenging, and you want to make sure you don't put a foot wrong there, and end up getting hit with a massive bill, because you did a good job of envisioning what the start of the process looked like, and what the end goal was, but not necessarily the path to get there, and you ended up making some decisions along the way that could have cost you a lot of money.

[00:40:48] KP: If I'm on a relatively Greenfield project, I've got a lot of choices here for my container system. I think there's some version of managed Kubernetes on every cloud, there's things like EKS that are custom, where, as you'd mentioned, I can spin up my own EC2 instances and run my own Kubernetes. A lot of choices here. How can Yotascale help me decide which direction to go in?

[00:41:10] JR: Yeah, again, I think that gets back to just the two kind of hallmarks of our platform. The cost visibility and the recommendations, right? Recommendations, I think that's pretty straightforward. We can give you the recommendations about how you can get your workloads into the right places, and at the right capacity to service for what you need, maybe a little bit of burstable capacity, not too much. And on the visibility side, I think that's also in a sense, we're giving you the visibility to make recommendations in ways that no machine learning tool is going to be able to even tell you.

What I mean by that is that if you have really high-fidelity information telling you about where your expenditures are going, a seasoned engineer is going to be able to look at it and say, “Okay, well, here's the thing that's costing me 50% of the spend of our whole cloud infrastructure. What can I do architecturally like at the code level, to actually change things in such a way that I'm not spending as much money, right?” Good example is that. Or maybe there's places where you're dumping a bunch of log files onto S3, and that's something that you're then processing those log files, ingesting them, and then saving them there as a backup. Maybe it turns out, that's not the best way to do it. Maybe you have so many data files to aggregate, that once you aggregate them and get them in some normalized fashion into your SQL database, or your data lake, maybe you don't need to keep those around anymore. Maybe it turns out, you never need to reprocess those files, and you can get rid of them.

At Yotascale, we actually use Yotascale ourselves for managing our own operations and this is a real-world example for us, we found places where there are data files that we had extra layers of redundancy that we simply didn't need and the tool was able to tell us that. Rather, the tool could not tell us directly, “Hey, you ought to go and remove these files.” Because the tool has no knowledge about what those files are for, if we do need to keep them around or not. But the cost visibility piece of it, really let us drill down and see, “Wow, for the last three months, here's this big expenditure.” And I was able to then go to my engineering team and say, “Hey, guys, I see that we have all these S3 buckets, that have a lot of stuff in them, but I don't quite know what it is. And they're not tagged in a way where we can even know what it is. Can one of you poke around in there and see what's in there?” And that's where we discovered, “Oh, these are actually a bunch of these files that we thought, because of some architectural decisions we made, a year back, we might need to keep around in case we need to reprocess them.” But because of subsequent architectural decisions, they literally were never going to be processed again, and we're able to just cut it out entirely.

So, that's something where in one fell swoop, that visibility gave us the ability to look at some places where we could cut out tens of thousands of dollars per month and expenditures for ourselves and our customers see those same kinds of returns.

[00:43:45] KP: So, by adding support for Azure, you've effectively – I don’t know if consider doubled, but certainly expanded the reach of how you can help companies make decisions. I'm curious if you saw after that a change in adoption, was there a strong market demand for a dual cloud system that wasn't available? So, you saw adoption there or maybe this was an opportunity for existing customers to start exploring their multi cloud options. What's the state of maturity in general?

[00:44:12] JR: I think what we've seen so far is customers who had very little visibility into especially their Azure workloads that now have that. So, it's basically a big gap in their knowledge that we had, or that they had about their Azure spend. And they didn't have a good way of pairing that up with their, their AWS expenditure, and figuring out like, “Well, is this really the right way to be spending our money? Or how does it compare in the environment where, a lot of times we have customers who maybe they have their own customers who depending on

what those customers need, they have a requirement to be on Azure versus AWS." It's not something that is necessarily to our customers control.

So, in those cases, it allows them to better forecast like, well, how are we doing with customers who have Azure workloads versus AWS workloads? Does one of them cost us more or less than the other one? That gives them more information for their future sales cycles of like, should we go out and get one type of customer more than another one, based on their how much they're going to cost us to run their operations? So, I think that's one of the places that we've seen benefit is being able to let our customers forecast further down their value stream of where are places that they should go to seek out some of those kinds of savings for things that are maybe even not even so much engineering more into even like marketing or sales.

I think in the future, what we're going to see a lot more of the product is us being able to give you better cross cloud recommendations, right? That one's a little trickier. I'll be honest, because there are differences in some of the performance capabilities across clouds, or there's work involved at the engineering level to even make that switch. As opposed to saying, "You should get this as a reserved instance, which requires pretty much no developer work to do." Making recommendations for somebody to move their workloads from one cloud to another is something that we certainly want to examine for the future. But that is something that does come with more sort of partnership required of the customer to take the time to do that for themselves. So, at this point, really, we found most of the value to be more in giving that really deep cross cloud visibility, where in a single view pane, you can see what is your cloud expenditure for a given product or a given team look like even if it spans multiple clouds.

[00:46:29] KP: So, you must have pretty deep purview into a lot of cost data and a strong understanding of how the different clouds are charging. I'm curious if you have an opinion, are the cloud platforms truly competing on price at this point?

[00:46:43] JR: I think they are competing on price. But I don't think there's a race to the bottom. I mean, I think that you are starting to see more parody in those costs. I think that where I have seen the trend moving is in cloud trying to provide deeper and richer tool sets to handle the types of jobs that are hard for their customers to handle themselves. What I mean by that is almost like the move from having vacuum tubes to integrated circuits. I think you're seeing that

same shift in cloud providers, where they're starting to bundle functionality together into higher order products that maybe solve more complex products than simply saying, “I need a computer in the cloud.”

So, I think that's one of the trends that I'm seeing there. I think there is to some degree, though, competition on cost as well. I mean, one of the things that there's been news about recently is data egress costs, like network egress costs. What I mean by that is the cost to get your data out of a particular cloud. Some providers charge more for that, some charge less, and I think you've seen a lot of things in the news, if you if you look at it, of companies sort of pressing on that note about their competitors around data portability. So, I think there still is some of that. But I think a lot of it really does come down to really the choices that companies are making about how do they want to build out their workloads in a way that matches their own business's growth curve. And doing the way that essentially makes sure that they don't end up with really slim margins are worse being upside down on the deal. I think that's something that is less about what the given cloud providers are charging vis-a-vis each other, and more about how you're best leveraging the tools at any given cloud provider to get the most value out of it for the least amount of money.

[00:48:30] KP: The cloud providers are definitely offering this just said higher level services. In theory, they could start to offer competitive cost optimization solutions. And if I took a super cynical approach, I think a cloud provider might say, “You guys are helping our customers spend less with us. It could feel very adversarial in a way, if you frame it that way.” What's your perception of the relationship you have with the cloud providers?

[00:48:55] JR: You know, it's generally been pretty positive. And I think that this has been kind of baked into the cloud space from early on, even if you're taking an example of within one particular cloud provider, let's say in AWS. They provided CloudWatch since, I think, since EC2 has been around, and it's because they have a vested interest in making sure that their customers know where their expenditure is going. So, that visibility piece is is really important and I think that all of these major providers are smart enough to know that you don't keep customers around by trying to sell them on products that they don't need or locking them into long term agreements for things that they don't need as well.

I think that they've been successful in replacing on prem infrastructures, largely because they haven't taken that approach. It's because they realize that if they are doing their job, the best that they can, customers are more incentivized to keep going back to the well and keep spending money as their businesses grow. So, kind of those boats are getting lifted together. So, I don't think we've seen that necessarily that sort of adversarial relationship that you might expect. I think what we've seen really true partnership with these cloud providers, we're partners with Azure and with Amazon, part of their partner networks, and I think that they're happy to have us there, because quite to the contrary of keeping people from using those services, or spending less on them, what it's doing is helping customers to not overspend in areas that they don't need to so that they can then spend more money in the future with them for the services that they surely do need. And I think that us giving them those insights, really, it's the same thing as the relationship between the cloud provider and their direct customers themselves is we're helping to give a richer data set to those customers to feel better and better about their cloud expenditures.

And so, they don't just decide like, “Well, I don't know. We could grow this, but let's not because we don't have any visibility and how much is this going to cost us. It's good enough for now and we don't want to get to the point where we're upside down on the deal and then it doesn't make any sense for us to even have this product anymore.” So, tools like this, I think help give people the confidence, they're not going to end up in a real bind like that, and just continue to invest more and more in those cloud providers as their businesses grow.

[00:51:17] KP: Is there anything you think I should have asked you about we didn't get to yet?

[00:51:22] JR: Yeah, I think one thing that we didn't touch on too much was just the competitive landscape out there for these types of tools. The market for this is red hot right now. There are tons of investment going into this area. And I think it's because my situation of seeing my cloud expenditure being the number two item in my company's entire budget after headcount is not unique. I think that's very common these days. I'm sure from my experience at Yotascale, as well as my experience before Yotascale, a lot of that money is spent for quite literally nothing. You've over provisioned capacity, or gotten instances that you didn't need. You left instances running when they should have been turned off. You had things that you should have bought on a metered basis versus not a metered basis.

So, I think that there's a growing recognition that that's the case, and that we can do better. As an engineering community, we can create better tools that give better information to help people confidently make those decisions. I think that you've seen this kind of rollout in multiple generations, the software that people have created for this purpose. The first few sets of tools, these we call them gen one tools that were around, they really focused on non-containerized workloads. They gave you cost visibility on the instance level, not really so much with the recommendations, and they didn't get in anything, it was containerized. You now have a lot of our competitors are also in the space of just giving you the visibility into your containerized workloads and how they're performing.

One of the things that Yotascale does that none of our competitors do is that we do both of those things. We'll give you the visibility into what's going on at the container level. We'll also give you visibility into what's going on at the instance level, and will allow you to marry those two viewpoints together. This is especially important when you're migrating over some of your workloads from non-containerized to containerized workloads. Because what you'll see happening is that, let's say that you have your Acme product, and you've got some percentage of your Acme product is running your classic environments, some is running in your containerized environment. And that's going to be the case for the next two years. You might move the needle from the first one further and further over to the second one as you're doing that migration, but they're running in both. And your finance department is definitely going to want to know how much you're spending on your Acme product as a whole, they're not really going to care about if it's running in a containerized or non-containerized environment. They just want to know what's the cost of operating that business line.

So, we can actually give you that visibility by stitching together the data for both the instance level as well as the container level, and what's your total cost of operation for a given product line or given customer that you're supporting? And that's something that just none of our competitors do today.

[00:54:08] KP: Well, Jake, where can people learn more about the company online?

[00:54:11] JR: Yeah, so they can go to our website. We're at yotascale.com. That's yotascale.com. We've got plenty of case studies out there from Zoom, from Hulu, from some of our other customers as well, that really show how we've made an impact on their business. And yeah, happy to field any questions from our excellent support and sales staff, if anybody has further questions about it.

[00:54:28] KP: Well, Jake, thank you so much for coming to Software Engineering Daily.

[00:54:33] JR: Thank you. It's great talking to you today.

[END]