EPISODE 724

[INTRODUCTION]

[00:00:00] JM: Cloud computing changed the economics of running a software company. Before the cloud, a software company had to purchase physical machines, which often required thousands of dollars paid up front. The cloud allowed developers to deploy their applications for free to operate a business for cheap and to scale without hiring a dedicated team to manage those servers. Building in the cloud is cheap, but scaling in the cloud can get expensive. A growing company can often save money by changing which cloud instances and services they use. They can reduce the number of server instances. They can change the size of compute instances and they can change the rules around auto scaling by using monitoring, dashboards and regular analysis of where money is spent. A business can find thousands of dollars of money to be saved per month.

There are also broad strategic decisions around cost. One area to study is the use of managed services, like Amazon DynamoDB, Google BIgQuery and Amazon Lambda. These services are proprietary and can lead to lock-in. Sometimes they can be quite expensive, but they can save developers hours of time, because they're easy to use and they provide high-uptime guarantees.

Ran Rothschild works at DoIT International, a company that helps business figure out how to save money on their cloud infrastructure. Ran joins the show to discuss the places where the most money is wasted and how startups can manage their infrastructure in a cost-effective manner. He also tells some stories about significant overspend.

Full disclosure; DoIT International is a sponsor of Software Engineering Daily.

[SPONSOR MESSAGE]

[00:01:50] JM: Today's episode of Software Engineering Daily is sponsored by Datadog; a monitoring platform for cloud scale infrastructure and applications. Datadog provides dashboarding, alerting, application performance monitoring and log management in one tightly integrated platform so that you can get end-to-end visibility quickly and it integrates seamlessly with AWS so you can start monitoring EC2, RDS, ECS and all your other AWS services in minutes. Visualize key metrics, set alerts to identify anomalies and collaborate with your team to troubleshoot and fix issues fast. Try it yourself by starting a free 14-day trial today. Listeners of this podcast will also receive a free Datadog t-shirt. Go to softwareengineeringdaily.com/datadog to get that fuzzy, comfortable t-shirt. That's softwareengineeringdaily.com/datadog.

[INTERVIEW]

[00:02:57] JM: Ran Rothschild, welcome to Software Engineering Daily.

[00:03:00] RR: Thank you very much. Happy to be here.

[00:03:03] JM: You are the chief business officer of DoIT International, and a lot of what you do is around cloud cost optimization. So I want to talk about cloud spend. A company, when it's getting off the ground, cloud spend is usually not an issue. I've got some EC2 instances, or maybe I'm standing up my cloud on Heroku, or on Firebase. There's a bunch of options, but it's basically free for a startup. When you have no traffic, very few users, it's super cheap. At what stage in a company's growth does the expense of cloud computing start to become important?

[00:03:44] RR: We believe that actually this is something that specially new customers should take into account very early in the process, because it helps not only in predicting your future cost once you're going to being production and have an X-amount of customers. But also to help you better reach a decision on which cloud provider is the right fit for you. Sometimes it can be technical reasons, but in many cases, especially for early stage companies, startups more particularly, it has a very big implication on their business and this can save them a lot of money if they opt for the cheaper solution, which also answers their technical requirements and challenges.

We will normally recommend for new customers, and this is by the way something that we see that early stage companies that are making their first steps on the cloud are considering what the cost is going to be like once they go to production. Once the company becomes more mature, it moved into production, you the spend going up, you see the utilization of additional services coming into play. It's something that this is the second phase that the customer should analyze their cost in to make sure that they are optimal and optimizes much as possible using the right tools and services, because making changes and sometimes it can be architectural changes, sometimes it requires changes to the application and so on. It will be more difficult to do at a later stage. It's something that is also very relevant to the early stage and then to the first phase once you start production and bigger deployments.

[00:05:23] JM: Okay. Let's say I have started a company. It's a photo-sharing app. You could share photos and videos and it's all cat pictures. You can only share photos and videos of cats, and my social network gets super popular, it's getting lots of traffic and I've gotten myself into a situation where I'm spending tons of money on infrastructure. If that kind of thing happens to me, what's the typical waste? Where is the waste in the kind of infrastructure that has been built up as that company has scaled?

[00:05:58] RR: Normally, the problem here is with the way that you actually use the various storage options that the different cloud vendors have. All cloud vendors have more than one storage solution, and what you need to do is to make sure that you store the data, the cat images in this case, on the right storage type if you need it accessible right away. So that's one type of storage. But for images that are less likely to be accessed, I will move them to a different kind of storage which will be cheaper and much more suitable to the use case at hand. If you build a lifecycle policy around the data that you manage, this will yield optimizations as far as cost is concerned.

[00:06:46] JM: If we're talking about AWS, is that typically like I've stored all of my images and videos in S3 and the once that are rarely accessed I should put them in – I don't know, block storage or something?

[00:07:00] RR: Yeah. Normally, on AWS, you start with the EBS volumes for the data that is required to have very superfast access. You cannot afford the latency for going to S3. Then on S3, you'll have frequent access and infrequent sequence. Then you have the other option of glacier for archived images. If I have, for example, let's say images that the chances of something requesting them are only 10% of the time. Even in those cases I can wait a few seconds longer until I provide the image. Then it will make sense to move the data to the infrequent access storage.

You have to create and think about what kind of information am I storing? What is the probability that someone will request certain type of images and so on? It's something that you have to learn not only on your existing dataset, but also as you increase the dataset, think about the probability of access and what will make the optimal fit for those kind of images.

[00:08:07] JM: When I was in school, we learned about the memory hierarchy. I think that's what it's called. You have disk, which is the cheapest in terms of expense, but it's the slowest access time. Then you have, I think, in-memory or RAM, which is faster access time, but it's more expensive. It's more scarce, and then there's this set of tradeoffs between those types – There's a gradient of different storage mediums between those types. That was kind of a single machine notion, and now we're in this world where you have pretty much the same idea where you have this tradeoff of cost versus access time. But it's moved to distributed systems. They're kind of opaque. We don't really understand how they work, or do we understand – If you talk about EBS, I think that's elastic block storage, right? Talk about EBS versus S3, do we even understand how these systems work or do we just understand that there's some SLA and there's some cost associated with them?

[00:09:12] RR: Regarding your first observation, even today you have on AWS, for example, Instance Store, which is similar to the cache that you had before. You store images that are – You need to retrieve them superfast on that, but that's rarely the case especially when you're talking about images, for example. Then you to move it to EBS, and unlike before when disk were the cheapest storage medium, here, EBS actually is the most expensive, and then you move to the different types of S3. Even also on EBS, you can go magnetic, you can go SSD disk and so on. If you are really keen on getting the knowledge, you can get it, you can understand how it works. AWS do not disclose everything, but you can certainly get a better understanding of what's the optimal solution for you based on how it works. Once you understand that, it's easy for you to choose. Mind you, you can always benchmark and test. It's pretty simple to move data from one storage type to another and then run the same test again and see the latency and see the impact on sole price.

[00:10:23] JM: You're talking about cost tradeoffs for storage. I hear a lot about cost tradeoffs for compute. So when you have these – I hear these words like spot instance or reserved instance, and I think this represent different pricing options for servers that you can buy on AWS or I think on Google Cloud. I think Google Cloud has a kind of market like that. Can you tell me about what are the kinds of applications that get into a situation where they need to be conscious of the costs of these different compute instance types and how do they choose between these different compute instance types?

[00:11:06] RR: Great. Great question. Basically on AWS, for example, today you have three tiers of compute, you have on the demand, which is the most expensive, which is basically there is no commitment from any side or from AWSs nor from the customers, and I pay the highest price for that specific instance. Then on top of that I can take on a commitment the reserved instances, which is the same instance, but AWS allow me to commit for using a specific family type in a specific region, on a specific operating system. In return, I get a discount. Then you have a totally different kind of machines, which are spot instances, which is something pretty genius that AWS came up with several years back already, which is basically a pool of resources that are not utilized by anyone. AWS basically gives the option to bid for these resources. Like the stock exchange, you bid. You can reach discount level of about 90%, 80% from the on-demand prices. The challenge here is just knowing how to bid and how to play with those mechanisms.

You can either do it yourself and optimize your spot fleet instances or you will have several third-party solutions that can do that for you. Now, of course between the on-demand and reserved instances, there are no implications to your application. It's just basically a license for commitment that you take on that specific compute. Then on the spot instances, that, of course, changes the whole way to look at things, because it really depends on the type of application that you are running. Because when you're bidding for resources and getting them, someone can outbid you. If you set a threshold, a maximum level of threshold that you will not pay above that price and someone outbids you who will take that resource. You have thirty seconds notification that this resource is going to be taken.

Now, let's say that your application is stateless. So that's a great option, because you can terminate the session and launch it on a different spot instances. If you know to manage a stateless application on a spot fleet of resources, that can save you huge amounts of money especially when compute is one of the more expensive services on the cloud today. You can certainly optimize your spend and also the way that you work using spot instances.

Google Cloud on the other hand have a different way to do that. They have what is called sustained use discount, which is basically automatic. Any VM that is running, for example, a full month will enjoy an automatic 30% discount on the list price. On top of that, you can take what is known as CUD, committed use discount, which is the equivalent of reserved instances on AWS. You have the equivalent for spot instances, you have the pre-emptible VMS, which basically the main difference between that and AWSs spot instances is the fact that any VM, you can own it only up to 24 hours. Regardless if someone outbids you or anything like that, the resource will be taken. Worst case scenario, after 24 hours.

[00:14:33] JM: Tell me more about the kinds of applications where the application developer is going to get into a situation where their server costs are too much, because you talked about the kinds of applications, the storage cost, like a photo-sharing app. But when am I going to get into a situation where my server costs are too expensive and I need to have these spot instances or choose between spot and reserved instances?

[00:14:58] RR: There isn't a simple answer for that, because of course there is a correlation between the type of instance that you're launching, the more memory and CPU that you need means that you'll pay more. Of course, operating system is a big factor in this, Windows, VMs that cost much more than any type of Linux machines. Of course, when developing applications, make sure that they, from a cost perspective aspect, should run on Linux rather on Microsoft, but it's just too expensive. Then as far as managing the sessions and the state of the application, it's also a factor because – Of course, microservices is a huge factor, especially with the huge boost and interest that containers have shown over the past couple of years and all the orchestration solutions managing the containers. These are all factors that you need to take into place when developing new applications. The type of operating system, the state of the application. Am I running on a monolithic application or microservices? I think that those are the three main considerations, but there are others of course.

[SPONSOR MESSAGE]

[00:16:23] JM: Managed cloud services save developers time and effort. Why would you build your own logging platform, or CMS, or authentication service yourself when a managed tool or API can solve the problem for you? But how do you find the right services to integrate? How do you learn to stich them together? How do you manage credentials within your teams or your products?

Manifold makes your life easier by providing a single workflow to organize your services, connect your integrations and share them with your team. You can discover the best services for your projects in the manifold marketplace or bring your own and manage them all in one dashboard. With services covering authentication, messaging, monitoring, CMS and more, Manifold will keep you on the cutting-edge so you can focus on building your project rather than focusing on problems that have already been solved. I'm a fan of Manifold because it pushes the developer to a higher level of abstraction, which I think can be really productive for allowing you to build and leverage your creativity faster.

Once you have the services that you need, you can delivery your configuration to any environment, you can deploy on any cloud, and Manifold is completely free to use. If you head over to manifold.co/sedaily, you will get a coupon code for $10, which you can use to try out any service on the Manifold marketplace.

Thanks to Manifold for being a sponsor of Software Engineering Daily, and check out manifold.co/sedaily. Get your $10 credit, shop around, look for cool services that you can use in your next product, or project. There is a lot of stuff there, and $10 can take you a long way to trying a lot of different services. Go to manifold.co/sedaily and shop around for tools to be creative.

Thanks again to Manifold.

[INTERVIEW CONTINUED]

[00:18:38] JM: Another issue that can emerge with too much cost is when a company, if I'm building my cat photo-sharing company, cat photo-sharing app company, and I use too many managed services. Let's say I use one of these expensive hosted databases that takes care of all my auto scaling and elasticity and it makes things really fast, but I get to a point where, "Oh! This is just way too expensive." That's just one example. There're lots of managed services that people use, and I love managed services, because generally they speed up your development time. I use Firebase a lot, and Firebase is the perfect example of a managed service that takes care of so many different dimensions of engineering for you. But I believe that if you build your app on Firebase, if it gets really big, the economics can get kind of dangerous and you can be spending a lot on it because it's sort of this – It takes care of things for you kind of database/hosting system.

Do you have any advice on managed services and when companies get into trouble with managed services?

[00:19:50] RR: Yeah. Again, great question. Normally when you're any new company, startups, normally opt for managed services, they don't have the men power nor the resources, nor the interest to continue and managing database services, for example, or others by themselves. Remember that companies running on the cloud – Cloud is not their core business. As you grow, it becomes a huge part of what you do, maintaining that, analyzing, cost of initiation and all that.

New customers, new companies start with many services rightfully so. What happens when you grow and continue to use managed services rightfully so. What happens when you grow and continue to use managed services, you need to basically analyze the pros and cons of using that and it's not only on the direct cost aspect that companies should focus on, which is normally the case unfortunately, because the alternative of running a managed database – Okay, I launched, let's say, a minimum of three VMs. I build my cluster and my monitoring VM, and I have a database, but what is my operational cost managing that? Do I need now to hire a DBA or do I need someone from my ops team to be fully dedicated on managing that database or other service?

It's something that you need to factor in is the operational overhead as well. Again, going back to my previous comment, remember that cloud is not your business. It's a derivative of what you do. You should ask yourself the question is; should I manage that solution on my own? Is it something that I want to do? Is it something that is important for me or is it a value? It can be financial, of course, but it has also operational aspects to that. Don't forget to look at the operational cost, and it's a premium service. That's the way I look at it, and you always pay more for a premium service. Someone else is managing that for you. That's where the added cost lies. Is it worthwhile? That's for each company to decide on their own. But for the most part, you rarely encounter companies that want to go back in time and manage their own databases.

Normally it come – By the way, just to complete that point. When customers opt to managed services on their own, it's coming due to a lack of functionality, which is missing in the managed service. Not necessarily from the cost aspect.

[00:22:24] JM: One thing I have found about running a business, and I run a podcast business. So it's not exactly an infrastructure-intensive business, but we pay for things like podcast editing, posting of our podcast episodes, which are on a CDN that's a little expensive. We pay for hosting of our WordPress site, which we way overpay for, but it always stays up. So maybe it's fine to pay for that. One thing that I've realized is what's kind of interesting about business and what's fun about business or at least the kind of business that I'm in is – First of all, when you start a business, especially a business like a podcast business, you have no idea what the inputs and outputs are going to be. You just don't know how much it's going to cost to run it. You don't know how much money you're going to make. You don't know what the unit economics are going to be. You don't know what the operational costs are going to be.

But overtime, maybe it takes one month. Maybe it takes six months. Maybe it takes two years. You figure out kind of the ins and outs. With some obviously degree of variation, maybe the economy spikes and you get a huge windfall, or the economy crashes and the entire economics of your business start to slow to a crawl, but you can figure out the inputs and outputs of a business. Once you start to figure out that sort of equation, you can almost draw the equation on a chalkboard if you wanted to.

Then you start to see, "Okay. If I just pay $5 for this thing, it will solve a problem that is costing me $10," and that kind of math makes business incredibly profitable. If they can figure out the places where they can spend money to make money. That's why a business can be so interesting and why the choice of vendors can be really important. The choice of services can be really important. The choice of consultants that you go to can be extremely important.

I just say all these to motivate the conversation of once a company starts to get to a point where they understand the inputs and outputs of their company and they can start to say, “Okay, I can pay for X and I get X + Y in returns,” they can start to make these savvy economics decisions. I have heard this call – As it relates to computing, I have heard it called finops. That’s a term that you may be familiar with. Do you know what finops is? Have you heard that term?

[00:24:50] RR: Oh, yeah. Absolutely. Finops is financial optimizations, of course, or financial operations. I heard different versions. But finops basically is the science behind financial optimizations of your spend on the cloud, on-prem, whatever, but it’s the science behind that. How do I optimize? How do I make sure that I do not have wasted money or waste resources, underutilization and so on in my environment?

[00:25:23] JM: If I have my podcast business and I ‘ve got some services that I pay for, or we could say it’s the cat photo-sharing business, because that’s a better example. Cat photo-sharing probably has a ton of infrastructure. You’ve got different managed services for different image recognition things going on to make sure all the images are actually cats. I’ve got enough going on in my company that if I allocate resources to finops, it’s going to take some resources away from the main things that I’m doing, building software, perhaps finding advertisers, perhaps working with my community. What’s the strategy for allocating resources to the process of finops? If you’re a company with traction, when do you start to just look at your infrastructure and say, “Okay. Where can I save money?

[00:26:15] RR: Finops is becoming more and more acceptable in the industry because of the maturity of cloud usage over the recent years and customers have reached certain scales of spend on the cloud where they just have to better understand what they’re paying for and find ways to optimize their spend. There are different options. The first option for customers is that there are many third-party solutions out there in the market that provide SaaS for analyzing and recommending – Analyzing your spend and providing you with recommendations on how to save your cost. The second aspect of that is working with partners. There are partners in the industry for all cloud vendors that have made financial optimizations part of their offering and it's part of their specializations.

We have to understand that, again, not only cloud is not normally your business, but understanding the structure of cost on any cloud takes a lot of learning and understanding. There are a lot of moving parts to your cost and you have to understand them and know them in order to be able to make optimizations and recommend savings.

Either you invest in internal resources to learn, study and deploy such knowledge, and granted you choose that, you have to understand it's not a onetime thing. You don't do it once a year or once every six months. It's an ongoing work, especially when you start to increase your spend and the type of resources that you have on the cloud. You have to either invest in it internally or hire a consultant or partner that will help you do that, which is something that they're knowledgeable about and have good experience, proven good experience in optimizing customer's cost.

Now, another thing to consider is the fact that cost optimization can come, and probably does come from the most part from understanding what you're paying for. You can optimize your cost just by taking on, for example, reservations, and commitments, and choosing the right storage type, like we discussed earlier. But it can also come from changing your architecture. For example, many AWS customers don't know that traffic between availability zone has a cost. It's not just traffic out of your VPC that has cost associated with it.

You have to basically analyze if you're running on high-availability applications, on multi-AZ production and so on if it's cost-effective, if it's right for you, if it's the right time for you to do that. It has been benefits, of course, redundancy, discontinuity and so on. But it has a cost associated with that. You have to juggle between these two options and decide what's the optimal working method is for you.

[00:29:29] JM: This is what you are in the business of working on. At DoIT International, you run a cloud consulting company. Full disclosure; you're a sponsor of Software Engineering Daily. But this is a very interesting topic to me. Why did you decide to start a cloud consulting company and what do you do for engineering organizations?

[00:29:53] RR: We are basically – Actually, we stumbled across cost optimization by accident, and I'll tell you exactly how. But we're basically an engineering company and we help customers which are either new to the cloud or existing cloud customers to optimize their workloads, to be better understand how to design the application on the cloud, the architecture, deployment, support and so on. We are basically regarded as the trusted advisors of our customers and of companies that are running on the cloud.

They discuss various topics with us on maybe they want to change their application. What services to use? Is this specific service the right one for me or should I look for an alternative? How do I optimize my workload on a specific cloud provider or a specific service and so on?

Several years back, we started our business based on the partnership with Google. Started with GSuite and then cloud and then Google Maps, but in respect to Google Cloud. So we realized early on, several years back, that not only our customers are flying blind in the sense of spend, but us also as resellers, we were missing a lot of information on what our customers are doing spend-wise on the cloud. We started to invest in that in one understanding all the various moving parts that comprised your spend on Google Cloud. Then not only that, but we also developed a solution which is called reOptimize, which is free for anyone to use on Google launcher or with us. We built a solution which gives customers, specifically Google Cloud customers, three main things. The first thing that it increases visibility as to what they're spending on Google Cloud today. It's like walking into a totally dark room, and we give you a torch light to focus on specific areas of interest and better understand what you're paying for and so on.

The second thing is that machine learning running on top of that. No more bill shock anytime that we detect a financial anomaly in your spend. We alert you on that. Everybody face this probably more than once working on the cloud. Some engineer made a mistake and left a huge cluster on running for the weekend, and you will find that out probably just when the invoice comes. Yeah, it still happens and many companies are not aware of that and they do not put the right gatekeepers on that, budget management, alerts and so on. That's the second thing that we do.

The third thing, which is part of reOptimize, is a recommendation engine. Basically, giving you recommendation on how to optimize your spend and what aspect of your spend are costing you a lot of money, and maybe you should consider optimizing those. In addition to that, as far as AWS for example is concerned. So we partnered several years back with CloudHealth Technologies, purchased recently by VMWare, and they had a great solution of, again, providing customers with insights and better understanding on what they are paying for. Using that tool with our knowledge, we provide customers the right information that they need in order to optimize their spend and in many cases cut their costs down.

I can't tell you how many times I sat with customers on a one-hour meeting and you walk out of that meeting and you saved the customer like $100,000 a year just by showing them that they did not terminate their unattached EBS volumes, for examples, or things like that which are very trivial, but because you are not in the business of managing cloud, you don't always see that and you're not aware of those tricks. It's something which is we regard it as part of our service and obligation to the customers. It's not only about the engineering and the technical aspects, but it's also about cost, because if you advice your customer in a certain architecture, if it's not cost-effective, it's a bad recommendation. So it has to be also cost-effective.

[SPONSOR MESSAGE]

[00:34:30] JM: We are running an experiment to find out if Software Engineering Dailydaily listeners are above average engineers. At triplebyte.com/sedaily, you can take a quiz to help us gather data. I took the quiz and it covered a wide range of topics; general programming ability, a little security, a little system design. It was a nice short test to measure how my practical engineering skills have changed since I started this podcast. I will admit that though I've gotten better at talking about software engineering, I have definitely gotten worse at actually writing code and doing software engineering myself.

But if you want to check out that quiz yourself, you can help us gather data and take that quiz at triplebyte.com/sedaily. We have been running this experiment for a few weeks and I'm happy to report that Software Engineering Daily listeners are absolutely crushing it so far. Triplebyte has told me that everyone who has taken the test on average is three times more likely to be in their top bracket of quiz course.

[INTERVIEW CONTINUED]

[00:36:29] JM: I don’t know where you’re from, but in the United States we have these television shows where you’ll have the host of the show comes into business, like restaurants, and then he’ll tell them everything that’s wrong with their business and then he’ll show them how to – “Here’s how you rework the menu,” or “You need to open up this extra room for more guests,” or “You need to get new lightning.” The whole 30-minute episode will be about just this person coming in and showing them how to run their business a little bit more efficiently. That kind of seems like what you do for cloud infrastructure.

[00:37:08] RR: It is exactly that. Yeah, that’s a great example.

[00:37:11] JM: What’s the most egregious case where you come in and you see somebody and you’re like, “You are spending $5 million more than you need to.” What’s the most egregious mistake that you’ve seen?

[00:37:24] RR: We’ve seen some – I would call them sad stories, but one example that I can share, which is a public online that’s been published is work that we’ve done with a company called Jelly Button, which are in the gaming business.

[00:37:40] JM: Jelly Button.

[00:37:40] RR: Yeah. Jelly Button actually used a monitoring solution which is called Mixpanel, which –

[00:37:47] JM: Oh no.

[00:37:48] RR: Let’s scrub the name?

[00:37:50] JM: No. I’m saying oh no because I’ve looked at Mixpanel pricing before.

[00:37:55] RR: Exactly.

[00:37:56] JM: We don’t need to scrub the name.

[00:37:58] RR: Exactly. It’s in the public use case. Yeah, Mixpanel has a great solution, but it has, again, a price tag associated with it due to the licensing aspect than other aspects that you can consider.

[00:38:11] JM: Because, by the way, they’re the market leader.

[00:38:13] RR: Exactly. That’s okay to charge a premium when you’re the market leader, but there are alternatives. We’ve sat with Jelly Button and we actually moved them to Google’s BigQuery solution and we managed to save them $240,000 a year just by moving from Mixpanel to BigQuery. It gives them – They’re not losing any functionality or visibility or anything like that, but it just required some work –

[00:38:42] JM: Yeah, you have operational costs in the migration, but once it’s set up, you’re good.

[00:38:49] RR: Exactly, and that’s onetime cost. It’s not ongoing, which is the scariest part of any cost structure is the ongoing cost that you have. That’s what happened there. In general, customers should not be concerned, scared or intimidated by looking for other alternatives. Everybody is busy, everybody is promoting their businesses and they have their business to run and they’re getting used to paying for something which is not necessarily the right solution for that. Invest this one time. Do a discovery on your architecture, on your third-party solutions and check other alternatives. Maybe doing it yourself like moving to BigQuery, which basically means you do it yourself, will be more cost-effective, and maybe from the technical aspects fit for you. Don't be alarmed. Don't be frightened from that.

[00:39:45] JM: Do you ever look at these cloud provider business, like Google Cloud or AWS and just think, "Oh my god! This is the best business ever." They have the best business ever, and they keep building higher and higher level services that they charge more and more – We think Mixpanel. You look at Mixpanel and you're like, "Jesus! Their margins are good." But that's what AWS or Google Cloud is on a massive scale, like not just monitoring your mobile game. It's like every aspect of your infrastructure, they have really good margins on.

[00:40:21] RR: Exactly. I think the people that can, first and foremost, appreciate that, those that have been in the industry for a while and remember what it meant to acquire a server. What you had to go through until that server was in production. All the procurement process, choosing the right spec, having the shipment arrive, installing whatever you need on it and then shipping it to the datacenter and installing it there. It just took forever, and I came from that business. I managed a collocation infrastructure.

When AWS came up and I remember the first EC2 instance that I launched and I was like shocked that instead of wasting a month on getting that server installed in the datacenter, I got it like in less than hour because I had to learn. It was my first EC2 to launch. I did it super slow and better understand how to do that, and definitions and security groups and policies and all that, and it was working and accessible and installed and it was wow.

Then of course since then it's all ancient history, and today it's like everything that you need. That's the one of the targets that big cloud vendors have is to provide you with a comprehensive solution under a single provider on a single roof that will take care of your every need. It can be from your production staging, testing, operational department, finance department. Everything is there already. Yeah, it's amazing.

Just to complete that, if you look at the pace of innovation that happen over the past 6, 7, 8 years, even though AWS were here like 12 years ago and then [inaudible 00:42:15], but if you just look at the last 6, 7, 8 years. I mean, it's just mind-blowing.

[00:42:22] JM: This is why I'm so offended by the tech-lash, kind of the backlash against tech companies in the press or it's like, "Oh! Amazon is ruining work for people. These warehouse workers that are underpaid." I'm like, "Do you understand that Amazon is responsible for the tech boom that we've had since 2006?"

[00:42:46] RR: Anyone claiming that just is too scared of adopting change and being part of this wave. Any cloud vendor does not come to replace anyone, but you have to adjust. You have to learn a new skillset. You have to change the way that you work. If you are not part of that, you are out of the game. I hear that still today and, again, it's just – I can't understand why – I mean, people are scared of change. I understand why they are claiming that and acting accordingly, but I just don't understand it. It's here. It's going to stay here. You're going to be on the cloud. Most chances, you are already on the cloud even if you are not aware of that with different SaaS solutions that you are using and so on. Get your knowledge up to par. Get experience. Get your fingers dirty and start working on the cloud and you'll discover a different world which, of course, from my perspective is a better one.

[00:43:53] JM: Okay. Here's something I want you to answer for me. I've done so many shows about Kubernetes and migrating to Kubernetes and standing up your Kubernetes and so on. There's a million ways to do it, very early days. But when I look at Kubernetes I'm like, "This is – You could build your infrastructure out of managed services, high-level managed services and Lambda functions, AWS Lambda functions." There're people who are standing up their own Kubernetes or doing multi-cloud Kubernetes and they want to have this control over their infrastructure. They want to not feel locked in to proprietary solutions. I'm like, "There is a big cost to doing that." You will gain so much speed if you go the route of wiring together DynamoDB with AWS Lambda and triggering things and having event-driven, whatever. Am I crazy or is like that a more cost-efficient? Not only more operational efficient, but more cost efficient way of managing your infrastructure.

[00:45:01] RR: You are 100% right. Yeah, we do come across many customers that have this fear of vendor-locking, which you can understand but you reach customers that are running their production on a single vendor for a few years now. They have no plans to change that, but they are still refusing to enjoy the full benefits of managed services on that cloud provider. Saving cost, optimizing the way that you work, operational efficiency and all that just because conceptually there is this thing which is called vendor-locking.

It's just why not? You're not going anywhere, and even if you decide to do that, it's going to take you time to plan and check the alternatives and you can move away from any kind of solution and it's just misleading. It's the same, like we talked a lot about cost-optimizations, but it's the same as not opting for commitments in order to get discounts [inaudible 00:46:08] committed use discounts. I mean, you're running your workloads there. You're not going anywhere. Why not take full advantage of that and save cost? I mean, what are you afraid of?

That's my view on that. I understand it at the early stage deployments where, again, you are unsure about where you're going to be in six months from now. But once you become more mature and the probability of you moving from one platform to another is going down. So why not enjoy the managed services and the proprietary services that the cloud vendor has to offer? Any solution can be replaced.

[00:46:49] JM: Here's the other thing that I think gets misunderstood about the state of cloud today. You have people looking at the cloud providers like they're in a race. It's like AWS versus Google Cloud. AWS doesn't have a BigQuery. AWS has got to build a BigQuery kind of thing, or Google Cloud doesn't have a strong as integrated an offering for serverless functions or in serverless triggers and things. When are they going to catch up with each other?

I think what people miss is these are just software companies. They're not in direct competition with each other. I mean, they kind or are but it's more like you've got this blue ocean of cloud provider potential. Yeah, maybe we're kind of like in the early days and they're both building their primitives and you've got DigitalOcean over here. They're kind of building their primitives. But the direction that these things are going in is just going to be total – I think it's going to be increasingly disjoint. They're just going to have different pros and cons and they're going to have their own little benefits and it will be easy to integrate them. It will be easy to have multi-cloud, different API high-level managed services and it's going to be really cool for developers, because you're just going to be able to choose from this giant buffet of different high-level services. Tell me where I'm wrong about that thesis.

[00:48:10] RR: I don't think that you are wrong. I couldn't agree more. Yeah. I mean, again, we're living in this amazing period of time where technology is like everywhere and innovation and the competition between the cloud vendors is just working. The customers are benefiting from that. You have to – Again, everybody was running on a single cloud platform up to, I would say, a couple of years ago more or less, three years ago. But then as the work that you're doing became more complex, more diverse, your requirements are different. You have different departments that are now utilizing the cloud.

You have much more options and flexibility to choose what is the right fit for your application and whatever you are trying to run. Each cloud has great pros and cons. You have to understand that and just choose what is the right solution for you based on your specific use case. If people want to understand more about, and let's focus just for a minute on AWS and Google Cloud. If people really want to understand the differences between them, they should think about where each company started its way from.

Basically, AWS started as amazon.com and this is why you see these segregated islands, the regions, which have no connection with other regions. It's just a standalone region with amazing buffet of services and options for you to choose from, but it's isolated. In relation to that, for example, if you look at Google Cloud, which is google.com, the search engine, and to provide services on a global scale. You have the private-owned fiber network of Google, which is all the datacenters and all the regions are connected on and running on top of that network. You have to better understand where they came from, and then dive into the services and find the right fit for you, but the competition is great and pushing the innovation to brand new amazing services that we'll enjoy.

[00:50:32] JM: what do you think about edge computing and the cloud making its way to the edge. First of all, you have the oil rigs, or you have the shipyards, like sipping container yards and these are like areas where you have maybe unreliable Wi-Fi connections and you've got some devices that are on-prem. You've got servers on-prem because they need to help with predictive analytics or maybe it's just running some on-prem factory software and the cloud providers are making their way into these sort of on-prem businesses and they're injecting machine learning into the factory processes. But they're still operating like cloud providers. They have these services like Lambda at the edge, or AWS Greengrass.

Then you also have it kind of happening on mobile where you have services like – I think it's Appsync. There's some AWS services for mobile applications and it helps you make your mobile applications, build your mobile applications faster more efficiently. We've got the cloud providers reaching their tendrils into our devices, our own process, like, "This is my processor. Get away from my processor. I shouldn't have to pay for this. This isn't on the cloud. This is on my machine." But what's going on there? What do you think is going to happen at the edge?

[00:52:02] RR: If I understood the question correctly, is that you are referring to distributed workloads, or I misunderstood you.

[00:52:10] JM: I guess I'm actually curious how you think it's going to affect your business also. I mean, as these cloud providers move into the edge and they're working with the factories, they're working with the oil refineries and then they're obviously charging a lot for certain services and then I guess that's going to affect your business, because then you're going to be working with oil refineries and factories, or maybe you already are.

[00:52:35] RR: Yeah. You do see a shift in what we call born to the cloud business versus the more traditional ones. The first and I would say second phases of cloud adoption where relatively easily cloud native companies didn't see a whole other option but to run on the cloud, versus the more traditional business that are running on-premise or on collocations, on remote locations. The way that you cannot disregard the cloud now today in anything that you do, these are businesses are not immuned and we see the shift in adoption in the traditional businesses, the enterprise businesses. Everybody already had a kind of footprint on the cloud today. It does change the business a bit, because first and foremost, the cloud vendors are developing solutions that will also feed those kinds of businesses and not only the cool cutting edge high-tech startups.

You see more development [inaudible 00:53:42] going towards those sectors. For example, you saw there in the recent Google Next, Kubernetes on-prem was announced. Why did Google that? Well, I think it came up because one of the criticisms on Google and the enterprise vertical is the fact that there were almost non-existent there and they completely surprised everybody with on-prem Kubernetes, which is a game-changer.

Again, now, traditional business, large enterprises that are running collocation have a solution that will connect them to the cloud from their own infrastructure. The change is more gradual and will allow them to adopt cloud in a more suitable manner for them. It's a game changer and we see all cloud vendors tapping towards those kinds of companies and invest a lot of money and development for those.

[00:54:40] JM: All right. Well, this has been super interesting. Everybody who is listening, if you're paying a lot of money for your cloud provider, then you should reach out to DoIT International. You should go to our link. We have a promo link. There are some companies that they advertise on the show that I feel on the fence sometimes about it. DoIT International, I don't really feel conflicted about advertising at all, because it's pretty much – People will only become a client of yours if they're going to save money. I don't really feel troubled in promoting your business. I hope people check it out, and it's been really fun talking to you, Ran.

[00:55:21] RR: Jeffrey, thank you very much for this interview. Thank you for all those listening and hope to hear from you soon.

[END OF INTERVIEW]

[00:55:30] JM: OpenShift is a Kubernetes platform from Red Hat. OpenShift takes the Kubernetes container orchestration system and adds features that let you build software more quickly. OpenShift includes service discovery, CI/CD built-in monitoring and health management, and scalability. With OpenShift, you can avoid being locked into any of the particular large cloud providers. You can move your workloads easily between public and private cloud infrastructure as well as your own on-prim hardware.

OpenShift from Red Hat gives you Kubernetes without the complication. Security, log management, container networking, configuration management, you can focus on your application instead of complex Kubernetes issues.

OpenShift is open source technology built to enable everyone to launch their big ideas. Whether you're an engineer at a large enterprise, or a developer getting your startup off the ground, you can check out OpenShift from Red Hat by going to softwareengineeringdaily.com/redhat. That's softwareengineeringdaily.com/redhat.

I remember the earliest shows I did about Kubernetes and trying to understand its potential and what it was for, and I remember people saying that this is a platform for building platforms. So Kubernetes was not meant to be used from raw Kubernetes to have a platform as a service. It was meant as a lower level infrastructure piece to build platforms as a service on top of, which is why OpenShift came into manifestation.

So you could check it out by going to softwareengineeringdaily.com/redhat and find out about OpenShift.

[END]