EPISODE 667

[INTRODUCTION]

[0:00:00.3] JM: Engineers need to have an awareness of the business model that allows their company to succeed. When a software company is going to market, the engineers need to work closely with these sales and marketing team to formulate a strategy for building and selling that product. This is especially true in highly technical products, such as database as a service, or platform as a service companies.

An engineer at a Hadoop as a service product needs to work with the sales and marketing team to explain why a customer might want a data platform. An engineer at a SaaS company needs to understand how the cost to provide a service might scale so that the sales team can decide on appropriate pricing for that service.

Mitch Ferguson has been developing businesses at software companies since the 90s. He help build out SpringSource and arranged the acquisition of SpringSource by VMware. This was an acquisition that later enabled the creation of pivotal software. Mitch then joined Hortonworks as an early member of the team bringing their Hadoop platform to market.

Today, Mitch works as a cofounder of AccelG2M, an organization that helps brings technology companies to market. He helps build out their sales, their marketing, their product and their organizational strategies, and it was a great pleasure to talk to him. He had so much knowledge from engineering, to product, to sales, to psychology.

Before we get started, I want to mention that if you are hiring engineers, particularly niche engineers, Software Engineering Daily is a reasonable place to advertise your job openings. If you are interested in working with us and advertising some of those postings, you can send me an email, jeff@softwareengineeringdaily.com.

There have been a number of people recently who have told me they've hired particularly niche, highly qualified engineers because the listenership is somewhat selective on the show, particularly if you're looking for or hiring a particular role and you want to advertise on a show

that relates to that role. So it would be a great way to support the show and I'd love to hear from you. You can send me an email, jeff@softwareengineeringdaily.com.

[SPONSOR MESSAGE]

[00:02:24] JM: Cloud computing can get expensive. If you're spending too much money on your cloud infrastructure, check out DoIt International. DoIt International helps startups optimize the cost of their workloads across Google Cloud and AWS so that the startups can spend more time building their new software and less time reducing their cost.

DoIt international helps clients optimize their costs, and if your cloud bill is over $10,000 per month, you can get a free cost optimization assessment by going to D-O-I-T-I-N-T-L.com/sedaily. That's a D-O-I-T-I-N-T-L.com/sedaily. This assessment will show you how you can save money on your cloud, and DoIt International is offering it to our listeners for free. They normally charge $5,000 for this assessment, but DoIt International is offering it free to listeners of the show with more than $10,000 in monthly spend. If you don't know whether or not you're spending $10,000, if your company is that big, there's a good chance you're spending $10,000. So maybe go ask somebody else in the finance department.

DoIt International is a company that's made up of experts in cloud engineering and optimization. They can help you run your infrastructure more efficiently by helping you use commitments, spot instances, rightsizing and unique purchasing techniques. This to me sounds extremely domain-specific. So it makes sense to me from that perspective to hire a team of people who can help you figure out how to implement these techniques.

DoIt International can help you write more efficient code. They can help you build more efficient infrastructure. They also have their own custom software that they've written, which is a complete cost optimization platform for Google cloud, and that's available at reoptimize.io as a free service if you want check out what DoIT International is capable of building.

DoIt International art experts in cloud cost optimization, and if you're spending more than $10,000, you can get a free assessment by going to D-O-I-T-I-N-T-L.com/sedaily and see how much money you can save on your cloud deployment.

[INTERVIEW]

[00:04:48] JM: Mitch Ferguson, you are the cofounder of AccelG2M. Welcome to Software Engineering Daily.

[00:04:53] MF: Thank you, Jeffrey. Glad to be here.

[00:04:55] JM: You are a little bit different in history and profession than a lot of the people that come on this show. Most people that come on this show are software engineers. You are more of a go-to-market technical specialist – How would you describe yourself?

[00:05:12] MF: So I describe it this way. I am not a technical specialist. My background, my education is not technical in nature, but it's business. But I've worked with a number of startups in very early stage, where there's product, there there's part of a product and the company is needing to figure out how do we take this to market? How do we position it and who are we positioning it for or to? What are the value props to those people? How do we need to price it? What are the best ways to articulate? What our technology does for whom? Why it does it? How it relates to other offerings in the market? That's generally where I will get involved in a company.

So very early is how we are coordinating go-to-market with partners or with our own sales organization. So it's very much combination of working with engineers at the company with marketing, and then also with sales, on trying to line those three so that they are all working very closely together and their capabilities are well-aligned so that from an execution perspective, whether it's engineering executing your marketing, executing your sales executing, they're all executing at the same level.

[00:06:28] JM: Give an overview of some of the technology companies that you've been a part of.

[00:06:31] MF: Sure. Primarily enterprise software. I would say one of the early ones – Well, I was in minicomputers a long time ago before client server and before PCs, and that had its own

challenges, and it's interesting you kind of saw minicomputers come back, and a lot of the technology comes back overtime. Client servers come back. Now with cloud computing or software as a service, it used to be called service bureaus.

So some of the business models resurface. It's just technology changes how they get executed. So some of the companies that I've worked at, one was called Arbor Software, which was in the early days of the OLAP space and leading OLAP engineer. Company went public and eventually bought Hyperion, and then Oracle took out Hyperion.

Worked in the early days of application integration at a company called Vitria Technology. Worked in federated queries at a company called Composite Software, that Cisco bought. Was in the open source application development space with a technology called Spring, and SpringSource was the name of the company and grew that and then VMware bought the company. Most recently, I was with a company called Hortonworks right from the beginning, and they were one of the – And/or one of the leading players in the Hadoop market.

[00:07:50] JM: I want to talk about some of these companies, specifically your experience at Hortonworks and SpringSource, because I'm a little bit more – and VMware after that, because I'm a little bit more familiar with that era of things, but I think you bring a lot of historical context to software companies, especially relative to a lot of the people that I've had on the show, and you come at it from a different angle since you're coming from the business development side of things. So I'm really looking forward to having this conversation.

To set the table for some of the engineers that are listening, a go-to-market strategy affects everyone in the organization and it's important for engineers to know how it affects them, because they're going to be interacting with the business development people or the sales organization. The engineers are going to have to be explaining certain things to the sales organization. Now, sales organization got to explain certain things to the engineers. What does an engineer need to understand about their companies go-to-market strategy?

[00:08:52] MF: I think that's an excellent question, Jeffrey. I think if I explained it, maybe summarize it at a high level. So a company is – Let's just talk about enterprise technology just to use that as an example, and that could be whether it's a SaaS based offering or something is

used on-prem. The alignment between engineering and what engineering is developing and building and providing and what sales or the sales organization or company is selling, the more closely those are aligned, the higher likelihood of success for the company.

What I mean by that, if the technology isn't quite – It doesn't have the best product market fit. Meaning what the company is building isn't resonating with the market. It lacks features. It may have features that the market really doesn't care about in general.

The sales organization is outselling that. They may be aligned in that my ability to execute as a sales organization is great. My ability to execute as an engineering organization, we've got all the great engineers and product management people and technical support people and we're all executing to the best of our capabilities. We just aren't building the right product for the market. So it's important I think for any young company listen to the customer.

So the customer is always right, because fi we think we know what is better for the customer and we want to keep pushing that rock up the hill, we might be successful. But most of the time, the market will buy what it believes is – And certainly early adopters. Will buy what they believe is going to be compelling for their business, because it's going to give them a competitive advantage. They're in a competitive disadvantage, or with a young technology. They aren't already [inaudible 00:10:46] the market. So that's where you'll get more laggards who will buy a product when it's already more established in the market.

When I talk about that alignment, if sales, if my sales capacity outstrips my ability to develop a product for the market, I've got incongruence between those two parts of my organization. If my product development capacity isn't quite meeting the needs of my sales capacity, then I still I'm going to be at a disadvantage of not going to be executing as well as a company. So I go back to how well engineering is aligned with marketing, is aligned with sales and how well those three are intertwined and executing at the same level, that will have a great impact on how effective the company is in its go-to-market strategy.

So go-to-market strategy really is how am I – What is my product? Who am I positioning it to? How am I articulating what that product means for that customer? So many times with early stage technology companies, and I've been involved in this. Here's what my product does. We'll

be great at explaining what the technology does. Then what we're doing is asking the person we're explaining it to for them to try to connect the dots on what this technology means to them. How they're going to use it. It is just human nature. It is more effective if I can explain to someone, "Here is what my technology does and this is what that means to you, you specific individual."

So that's where we talk about product market fit is not only what the technology does, how we articulate what it does, but the value that that will provide to the person who we are articulating that to. The higher that value proposition resonates with that person, the greater likelihood they will want to use my product.

So go-to-market is aligning my product market fit with how I'm articulating the value props around that and then how am I taking those messages in that product-to-market directly as a service, through partners, so that I know who I should be articulating, what my product does and why it's a value. I know to whom I should be articulating that to. How do I get that message out? Those three things are all related to a go-to-market strategy.

[00:13:15] JM: We can examine this relationship between product and engineering and sales/ marketing through the lens of some of the different companies that you've worked with in the past and I also eventually want to talk about the present and the future with things like Kubernetes and cloud. But going back to Spring, Spring, if I recall, I've used it at a couple of companies. It's a framework for Java. So basically for Java developers, if you're using Spring, you can be more productive. You can develop faster. It helps you design an engineer web applications, and this was a really big deal for a lot of companies. It continues to be a big deal.

It's interesting, because it's a very technical product. If you're an engineer and you're not working with Java or you're not designing web applications, you probably have no idea why Spring is useful to you. Furthermore, if you're not an engineer, you have no idea why Spring is useful to you at a baseline. So this is a perfect example of a company where the communication loop between product and engineering and sales needs to be pretty tight.

Can you tell me a little bit about your experience at Spring leading up to the acquisition. How did you develop a go-to-market strategy?

[00:14:32] MF: Sure. What was – And Spring is still a very prevalent technology. What was Spring back in I think 2007 or so when I joined the company? A Java development framework that was wildly successful. What I mean by wildly successful, 50% of the Java applications written that were running on WebSphere and WebLogic servers were Spring-based applications. So I call that wildly popular, because Java was probably the most popular programming language certainly for client server in the 2000s, late 2000s for sure.

So who was our audience? Developers. We weren't trying to sell to CIOs. We weren't trying to articulate our value propositions to CIOs. It was more to developers and to application architects, because those were the people that were charged with developing the applications that the company/business needed and how could they meet the needs of their business constituents inside their companies as quickly as possible and as optimally as possible. So that's who were effectively selling to.

Now, the interesting thing about Spring was that it's an open source technology, Apache software license, super friendly license. I'm a very strong proponent of open source. The beauty of open source from a business model's perspective, and it's evolved a lot in the last 10 years. There are issues today that we didn't have to deal with back in 2007, 8 or 9.

But open source technology can be very viral. Meaning it is very easy for someone to access that software and bring it into the organization. They don't need a license from or permission from ETUs, my open source technology, especially with a license like Apache. So Spring became heavily used by developers, and our go-to-market strategy then, the initial one with SpringSource was help the Spring developers use Spring better. So educate them on how they can use the technology better than how they thought they could originally use it.

So there was very much of outbound in nurturing the community, which is a phrase that I didn't coin it. It came from people that I've worked with at Spring that were very early in JBoss. Nurturing the community, meaning helping your open source community leverage your technology to the best of their capabilities, and that's the relationship that is golden for any sort of open source technology, the value of the community and the vibrancy of the company.

So our go-to-market strategy first was help make sure that the Spring developers have the best experience with their technology as possible. Now, the challenges with open source is how do you monetize it? While we had all the primary developers for Spring and related Spring technologies on staff and we were paying these people as we needed to, as they were employees of the company. It's important for any open source company, if it wants to sustain its viability, you've got to build business around it.

So at Spring, we initially started selling support for the Spring framework and we were, again, had a direct sales force targeting those buyers who I mentioned earlier, primarily application architects and development leads, selling them support where we could provide enterprise class support. This was very structured just like a proprietary software model, but one hour turnaround and the types of support that an enterprise expects.

Quite honestly, that wasn't enough to sustain the business. Then we looked at what other technologies could we add to our portfolio to drive additional revenue and more importantly value to the Spring community. We acquired a company that was very instrumental in the Tomcat application server. We acquired a company called Hyperic that was very instrumental in Java application performance monitoring. We have then a stack of open source technologies that we were taking out to market, and we're able to grow an interesting business around that.

At the same time, I reached out to VMware. So this probably 2008, and I thought with any go-to-market strategy that I look at with a technology. Given for any of the engineers or developers on your podcast, the pace of technology, it moves faster every day. It's moving faster today than it was 5 years ago, than it was 10 years ago, than it was 20 years ago. Trust me. I think they know that. So the pace is moving so fast. It is critically important for a young technology to try to establish its relevance as quickly as possible.

The challenge if they don't do that, they get disintermediated by something else. The best technology, we know this, the best technology doesn't always win. So it's the technology that many times is most recognized or perceived to be the best that will win.

So I looked at SpringSource, how can we elevate – While Spring was wildly popular, how can we elevate our market traction? How can we elevate the brand of our company SringSource as

quickly as possible? I reached out to VMware and I thought, here's a company that's trying to sell application virtualization. If SpringSource was the single largest doorway into the largest community of Java developers, and what I meant by that, SpringSource was the most influential company behind Spring. So through that single doorway, we, SpringSource, could give VMware access or provide VMware access to all these Spring developers, because they were coming to SpringSource to, again, understand how to use the technology better.

That resonated with the CEO of VMware at the time, Paul Maritz, and that ended up leading to VMware acquiring SpringSource. If you fast-forward a bit more, I touched on earlier not only the Spring technology and related technologies around Spring, the application server, monitoring technology. Well, at VMware, we acquired a messaging layer as well. Those were the foundational components. Not all, but the foundational components for what is now pivotal software.

So that's how VMware used the technology to then elevate its influence in the application development world and effectively cloud computing world, taking advantage of the vSphere and then writing, and then moving up the stack so to speak. So hopefully that answers your questions about some of the go-to-market strategies we used at SpringSource not only how we monetize open source technology, and we tried to build commercial products around that, but then also how we leveraged third-parties in this case, VMware, to help accelerate our go-to-market, which ended up being an acquisition of the company.

[00:21:52] JM: It's an epic story of this set of technologies snowballing together, starting with Spring and saying, "Okay, developers love this. They're really productive with it. We can't build a support business around it. Let's acquire a couple other additive technologies and then we start talking to VMware. Okay, we talked to VMware." I'm not exactly sure what the synergy between VMware and Spring was. Is it that if you – Like let's say you're a developer, you're building your application in Spring. Your company is starting to grow and you realize we need to scale. In order to scale our infrastructure, we need to start deploying our applications across VMs. Was that the additive element?

[00:22:31] MF: Yeah. Effectively, yes, but I'll answer that specific question. I think what we saw when you mentioned the epic story of snowballing with the technologies, what we saw at

SpringSource was very tight alignment between Rod Johnson, who's the father of Spring, and Rod's core engineering team, and the business team that was led by Rob Bearden, who's CEO of SpringSource, and now is CEO of Hortonworks. That alignment between here's a technology with a business vision led to adding these other components to our core assets of SpringSrouce that not only provided more technology, but also business value props that we could then roll in to a very effective go-to-market strategy. So that snowballing was both snowballing of technology and business aligned to create a much more valuable company.

So the VMware angle was – And it was really kind of this is more of my philosophy when I look at doing strategic relationships with, A, if I'm at a young company, which typically is who I'm working with, or have worked at, but I'm approaching a large company like VMware. I need to explore, "Here's what my technology does. Here is how I think that may be of benefit to you. I don't know. You tell me if that's the case to help VMware." So at VMware was they're trying to virtualizations of applications as fast as possible. The more apps that were virtualized, still to this day, the more vSphere licenses VMware sells.

So if we, SpringSource, could provide them access to a million+ Java developers and provide them with a concerted access, a concerted voice through this single company called SpringSource to these million+ Java developers, that would give VMware an opportunity to articulate the value of use Spring developers, you're building your apps in Spring, these Java apps. Here's how they could run better in a virtualized world in our world of vSphere, and that resonated.

Paul Maritz had the vision for going more up the stack to getting to the application developers as a way to drive more usage of their fundamental technology, called vSphere. It's not unlike why did Satya Nadella recently buy GitHub. You're seeing Microsoft – They certainly were going after developers with .net. They're extending that even more as trying to evolve that developer experience. Why? Because they want more and more applications to run on Azure. Not dissimilar to Paul's philosophy. I wouldn't say it's directly related, but it's not dissimilar to Paul Maritz's philosophy of VMware. How can we get more of these new applications running on our platform because I'll sell more of my stuff? In that case, virtualized software.

[SPONSOR MESSAGE]

[00:25:43] JM: DigitalOcean is a reliable, easy to use cloud provider. I've used DigitalOcean for years whenever I want to get an application off the ground quickly, and I've always loved the focus on user experience, the great documentation and the simple user interface. More and more people are finding out about DigitalOcean and realizing that DigitalOcean is perfect for their application workloads.

This year, DigitalOcean is making that even easier with new node types. A $15 flexible droplet that can mix and match different configurations of CPU and RAM to get the perfect amount of resources for your application. There are also CPU optimized droplets, perfect for highly active frontend servers or CICD workloads, and running on the cloud can get expensive, which is why DigitalOcean makes it easy to choose the right size instance. The prices on standard instances have gone down too. You can check out all their new deals by going to do.co/sedaily, and as a bonus to our listeners, you will get $100 in credit to use over 60 days. That's a lot of money to experiment with. You can make a hundred dollars go pretty far on DigitalOcean. You can use the credit for hosting, or infrastructure, and that includes load balancers, object storage. DigitalOcean Spaces is a great new product that provides object storage, of course, computation.

Get your free $100 credit at do.co/sedaily, and thanks to DigitalOcean for being a sponsor. The cofounder of DigitalOcean, Moisey Uretsky, was one of the first people I interviewed, and his interview was really inspirational for me. So I've always thought of DigitalOcean as a pretty inspirational company. So thank you, DigitalOcean.

[INTERVIEW CONTINUED]

[00:27:50] JM: I want to ask a little bit about the acquisition. What went on during the acquisition talks? Was there a lot of back and forth negotiating, or are there any tricks that you can use to shortcut? If you can say, "Okay. We are going to be win-win. This is going to 1+1=3. This is a great synergistic acquisition. Let's do it. But we're not sure what the upside is going to be. It could be massive. It could be fairly straightforward." What kind of arrangement can you orchestrate to make sure that the windfalls of the acquisition is reflective of what happens after the acquisition?

[00:28:27] MF: So a couple of things. I wouldn't call them tricks. That almost sounds deceiving. Not that I'm at all insinuating that's what you meant.

[00:28:37] JM: I don't mean them that way. I mean as a team we are trying to figure out what's the trick to making this alignment work. I agree, bad choice of words.

[00:28:44] MF: Here's what I think is critically important to articulate, and this is the trick, if I was going to say trick, but it's not deceiving. I touched on it earlier. It is not uncommon for a technologies, and I say this with complete respect. It is not uncommon for a technologist to be very good about articulating what their technology does. It's harder for a technologies, and this could be an individual or a company, to articulate why that is important to the market.

So I will use the phrase, as I'm working with VMware or similar to a strategy that we executed with Microsoft at Hortonworks, helping a company understand. So if I'm a young technology company, helping to have a much larger company understand the art of the possible. How working with my technology will enable them to do X, Y, Z and what is the value of X, Y, Z to them, that's the trick. If they don't understand the art of the possible or the art of the possible isn't that compelling to them or they don't understand what the art of possible means to them, they can articulate the value of the art of the possible, then they aren't going to want to do any strategic with that company.

VMware understood the art of the possible with SpringSource. We were effective as a company in articulating that, and they felt that given the art of the possible, it was important for them to own it, and that's why they acquired the company.

[00:30:19] JM: Did you stay at VMware long enough to be involved in that development of Cloud Foundry?

[00:30:26] MF: It's interesting. So I stayed at VMware for a year and a half, and I think VMware is an excellent company. My passion is in very young technology companies. So after the acquisition, I did not intend to stay at VMware a long time and I made that clear to VMware. But I stayed to help with the integration. I actually got involved in another strategic initiative between

VMware and salesforce.com that was announced, but then it fell apart for a number of different reasons on both parties sides primarily because of competing competitive issues at the company. That was very interesting.

So I was involved in helping VMware leverage the spring technology for some additional strategic initiatives, and I started to get involved in a little bit with the phase 1, I should say, or version 1 of Cloud Foundry, but I left before that was rolled out. Then I joined Hortonworks.

[00:31:25] JM: With Salesforce kind of trying to figure out where exactly the boundaries around their business work, because they acquired – I think they acquired Heroku around this time and they were like, "Maybe we're a cloud company." I mean, Facebook did the same thing. They acquired Parse and were like, "Maybe we're a cloud company." They're like, "No. Let's stick to our knitting."

[00:31:43] MF: That's an interesting question, because you're right, you've definitely done your homework. The Heroku acquisition came after the joint announcement. I think it was called VMforce, if I remember it correctly, between VMware and salesforce.com. So when Salesforce announced the acquisition of Heroku, it really made VMforce irrelevant. So both companies didn't need to work on that anymore and it died a quiet death to the best of my knowledge.

But if you think about it, VM – Where I touched on earlier, what VMware was doing, Paul Maritz's vision and strategy of going after application developers to help utilize more of the VMware assets, core assets. Salesforce is doing the same thing with Heroku. Now, I don't know how well that's taken off, but Salesforce certainly is and their business is growing is beyond just Salesforce automation software or CRM software. So they are providing more of a platform and will continue. I fully expect them to provide, continually provide more of a platform that enterprises can use for applications written by third-parties or applications that they write themselves, and that was really the Genesis of the Heroku acquisition.

[00:32:57] JM: Yeah. I think Heroku continues to be a really interesting asset for them, and I think what I hear, I've heard before that a one problem with Heroku as a cloud provider kind of business is that when the companies get big, they move off of Heroku and go to AWS or Google Cloud or something. But, as you said, you can imagine a future where Salesforce like gradually

moves down the stack or Heroku moves up the stack and there's some kind of unexpected thing in the future where, "Oh! Now these businesses fit together, like perfectly aligned puzzle pieces."

[00:33:34] MF: Yeah, or up or down and they provide additional capabilities where they further mitigate the migration of any apps and they start in Salesforce that could move to an alternative cloud. I don't know what their strategy is around that. I haven't studied it, but you can see certainly the company is wildly successful and they're continuing to make acquisitions. Look at the recent Mulesoft acquisition. They're continuing to make acquisitions to make it easier for enterprises to integrate things and connect things with Salesforce.

[00:34:07] JM: So you were then at Hortonworks for five years.

[00:34:10] MF: Yup.

[00:34:11] JM: When we first met, we talked a little bit about this early experience of being in the midst of the Hadoop vendor wars. Tell me what it was like to be in those Hadoop vendor wars.

[00:34:25] MF: It was a lot of fun. We ran as fast as we could possibly run. So I actually join Hortonworks probably in April of 2011, and the company was founded in June 28th, 2011. That's when we got the term sheet from Benchmark Capital. So here're a couple of data points. Benchmark Capital funded was the initial investor in SpringSource. Benchmark Capital is the initial investor in Hortonworks. Rob Bearden was the COO of SpringSource. Rob Bearden is the CEO of Hortonworks. I was at SpringSource running business development. I joined Hortonworks running business development. The point I'm making there is – Our VP of strategy was VP of strategy at SpringSource and ended up being VP of strategy at Hortonworks. Our head of sales at SpringSource ultimately was the head of sales that Hortonworks.

There were people that knew how to do their jobs. From day one at Hortonworks, we had the leading Hadoop engineers. Cloudera might argue with us, but leading Hadoop engineers that came from Yahoo. Yahoo was an investor in Hortonworks, and became part of Hortonworks. Day one, June 28, 2011, we had 27 employees. We're approximately 27. 24 of them were

engineers, leading Hadoop engineers, and then three business people; myself, the CEO and an office manager.

So this was 2011. In 2009 after the spring source acquisition, actually had met with Cloudera. So they were already at in the market a year prior since 2008. So we started Hortonworks in 2011. We were at a three year disadvantage to Cloudera. So we needed to really establish ourselves quickly in the market, because we didn't have the brand. While we had great engineers, we did not have a brand.

So one of the things we did to quickly establish ourselves in the market is say, "Our Hadoop distribution is going to be 100% open source." People scratch their heads, but that was a move that we had to make and that we wanted to make, because we believed with the leading Hadoop distro, it's effectively – Or distribution. It's effectively 20+ unique products all integrated together to provide a complete solution, and someone will use some combination of those 22 to 26 open source projects.

So we believe the fastest way to make the Hadoop market function, and what I mean by that, make it evolve as quickly as possible so that enterprises could use the software faster, more easily and the ecosystem around Hadoop could work with Hadoop faster and more easily. What I mean by the ecosystem, Hadoop itself – Think about it as this big old file system. There are – I remember a meeting with J.P. Morgan at the time, and J.P. Morgan said, "We love all these new technologies. This may be something for your earlier stage technologist. We love all this new technology." Everyone says you should use my product, and we're really interested in using new technologies.

But we've made significant investment in other technologies. We need to help understand how this new technology will work with our existing technology. That's one of the reasons we were firmly behind doing everything in open source with Hadoop so that the Tableaus and the other databases and various TL technologies or whatever that needed to work with Hadoop could more easily integrate with it, because that's what the enterprises were expecting. If I'm going to bring Hadoop in, I need to work with my existing ETL products, my BI products, whatever it was. So that was core to our strategy on why did everything in open source.

[00:38:26] JM: So this is like a stark contrast between the early – You've mentioned, with the early world of Spring before when you are trying to do a support business. Support business works a little – And I'm not sure if you would call it a support business for Hortonworks, but it sounds like it was kind of an integration and support business, because there's so much more stuff to integrate with and it's a multinode system and it gets really complicated really fast.

[00:38:54] MF: It does get complicated really fast. But what we ended up selling, and it was probably after a year and half, a year and a half after the company started, we released our first release of the Hortonworks data platform, and that was our distribution. So a hardened distribution of these length of time, I think at the time, 22 unique Apache projects all related around Hadoop that we put together in a single installed offering called HPP, Hortonworks data platform. Our product was not HTTP. Our product was selling enterprise support around that. So we sold subscriptions. People can argue while Hortonworks is a services based company, that's not how we accounted for it. It's not how we managed ourselves. It's not how Wall Street looked at us as well. It's very much was a product and people could use our distribution freely, but if they wanted to be able to get enterprise class support, they needed to have a subscription. That was our go-to-market strategy and were – As Barclays mentioned, we were the fastest company to go from zero to a hundred million dollars of any technology company.

So that was a credit to a number of things. I think the market appetite to a technology like Hadoop, the quality of our engineers and the quality of our go-to-market team, both sales and marketing. We executed really well as a company.

So I want to go back to that earlier comment, your earlier question. Like, early stage, what was it like in this kind of Hadoop wars. I think one of our competitors was trying to compete with Hadoop with their own proprietary file system, and I think in a young or – I shouldn't say young. In a fairly popular open source world, meaning where the technology is fairly popular. In 2011, Hadoop was pretty popular. It wasn't used by probably more than 20, 25% may be Fortune 500 companies. But early adopters were definitely looking at the technology. That was very prevalent, because they were looking at ways they could take advantage of all of the majority of data being created by their businesses was coming from machines, websites, other things like that, and it was falling on the floor. They didn't have a cost-effective way to capture it, to begin to look at correlations or trends across that data. So there was a large appetite for innovative

companies to look at using Hadoop to see if they could drive their businesses better, more cost-effectively, or give them competitive advantages.

So trying to compete with an open source technology where there's a lot of popularity with the proprietary technology, if you're a young company, is very hard. That would be a lesson that I would give to anyone. If there's a very common and popular open source technology, the viral effect of that, that train is barreling down the tracks. It's hard to stop it if you were a young proprietary software company.

This has nothing to do with; is open source better than proprietary? That's not my point. The point is if an open source technology is very very popular, trying to compete in your young proprietary technology company is a daunting task. So coming back to the early days of Hortonworks, we made that decision to do everything in open source to take advantage of the interest in Hadoop and to help the market evolve and function as quickly as possible both for the enterprises as well as the ecosystem.

It also was important, as I touched on earlier, for us to build a brand for the company. So one of the things that we did was we were concerned that companies using enterprise data warehouses who thought that's where they were managing all their data, would they perceive Hadoop as competitive? Give or take whether they do now is not the point.

So we reached out to Teradata. It was at the time one of largest enterprise data warehouses, vendors and they still are. The CTO of Teradata had the vision that there was a coexistence strategy between Hadoop and Teradata enterprise data warehouse. The type of data managed in an enterprise data warehouse was complementary and many times different from the type of data that would cost-effectively sit in a Hadoop system.

We articulated the art of the possible for both companies. We formed a relationship with Teradata. That helped shine a bright shining light on Hortonworks as a new vendor and also showing how Hadoop can complement an existing enterprise data warehouse.

The second key relationship that we established was with Microsoft. They were looking at potentially building a proprietary offering competitive to Hadoop. Had the vision to why not take

advantage of the momentum of open source and given the unique relationship that Hadoop had, that Hortonworks had with the Hadoop community, we formed a very strategic relationship with Microsoft. That helped transform Hortonworks. It truly sent a global message, that and Teradata that Hortonworks is a leading dupe provider in the market even though we had come in the market three years after Cloudera. Those two events were really instrumental in helping to build the Hortonworks brand early on, and then the rest is history where we just executed really well from a technology and sales perspective to build the business.

[00:44:36] JM: I'm seeing a pattern of your value at a company and it often involves – I'm sure there's lots of blocking and tackling you're doing that you have really mentioned, but seeing these different chess pieces on the different companies that are in the landscape, like being able to identify that it makes sense for Hortonworks to work closely with Teradata or seeing that VMware and SpringSource would make a lot of sense together. It seems like a valuable skill that you have in your set. I want to get to the Kubernetes chessboard eventually and talk a little bit about that and how you think that market is going to unfold.

While we're on the topic of data engineering, a few companies that I've spent some time covering are Confluent and Databricks, and I don't know if you are willing to talk about these companies in specific detail. But if you are at each of these, how do you see the landscape? Because these are both companies in the – These are big companies in the data engineering space. They're kind of nascent. Confluent is like the Kafka company. Databricks is like the Spark company. Do you have any strategic evaluations or what would you be doing if you were at either of these companies?

[00:45:56] MF: So I'm familiar with both companies, and I spoke with both companies probably a year and a half ago last, and it was about how I personally might be able to help them out. I think they're both great companies. I just elected to pursue something different, and that was the company that we have now, AccelG2M.

So here's how I would – And Jay at Confluent, or Ali at Databricks could disagree with me, and I completely respect that. So here's my view. When I touched on earlier, it's very hard to compete against an open source technology that has a lot of momentum in the market if you're proprietary vendor. There is an example of Kafka.

So I think Jay and the team did a great job really developing the Kafka community, and that is so wildly used. A company that is trying to – A young technology company trying to out Kafka Kafka today, that is really hard, because Kafka is just so widely used.

I think to Confluent's credit, they realized that they need to – That they can leverage Kafka, since that's a – I don't mean this in a negative way, a control point for them. They're the main Kafka company, but they could leverage that and continue to nurture that open source community, but provide interesting technology on top of it where they could really build a scalable business. So take advantage of the platform, Kafka, that's wildly popular and leverage that given our knowledge of Kafka and vision as a company to build technology on top of it, proprietary or open source, and drive that value into the market. I think Confluent's done a very good job at that.

Companies trying to compete against young technologies, trying to compete against Kafka, it's really hard for them to do right now. So I think Jay and team at Confluent are doing very good job and I'm very bullish on that company.

I think the Databricks – I think has evolved very nicely. What I mean by that is when I was at Hortonworks, Spark was, and still is really popular. But Spark was super popular. It was important for us, Hortonworks, to work with Spark and provide an enterprise class experience for our customers around Spark.

I think that Databricks had an opportunity to do even more with Spark early on to monetize it, and whether some of your listeners may not care about monetizing software, I'm usually hired by companies and venture capitalists to help monetize software so that companies can become profitable and they can hire more people, versus just burn cash.

So I think Databricks actually missed out early on in an opportunity to monetize Spring, because they truly had a unique position in the market. Eventually, their control point around Spring got diluted. You had companies like Cloudera, and Hortonworks, and IBM and others, IBM specially, that develop their own Spring expertise and they were able to work with Spring on their own. I'm sorry. Work with Spark on their own.

What I think Ali and team have done is really evolved their business model leveraging spark, much like IBM and Hortonworks, and other companies are leveraging Spark, but providing a unique offering cloud-based based that takes advantage of Spark, but also underlying infrastructure to provide a really interesting application experience for companies that want to use Spark in cloud-based apps.

They seem to be doing very well around that. They've gotten great valuation and more and more companies are using their technology, but I think while they're the Spark company, they were the spark company early on and they kind of lost that control point, and I don't mean that in a negative way, but they are doing very well executing because of their Spark expertise with additional vision and technology on top of Spark to build a really interesting business.

So there's a company that I think is also executing really well both from a technology perspective and a business perspective, just like Jay and team are over at Confluent.

[00:50:26] JM: Yeah. Yeah, I think we could go a lot deeper on these two companies and talk about their alliances and their relationship to the cloud providers are potentially competing with the cloud providers. But I want to get to Kubernetes related stuff, because I feel like that's the market that I really want to ask you about.

I've been to a couple of the KubeCon, Kubernetes conferences, and I spent a lot of time at these conferences walking around the Expo Hall. In fact, I probably should spend more time in the sessions that are like discussing software engineering related technical things, but I find myself just walking around the Expo Hall and kind of thinking, "Huh? Why is this company here? Why did they buy the super large booth with the 80 people handing out t-shirts? Why is that other company have a small booth? How is this evolving?" Do you have a thesis that you can condense the market of Kubernetes related businesses into and explain to me?

[00:51:29] MF: I do. I don't know if I would use the term thesis. That sounds too intellectual for me. I would say it this way. So AccelG2M was intimately involved in working with CoreOS around their acquisition by Red Hat. So we saw a lot that happened in January of 2018. so this

year. We started working with them in the fall of 2017. Here is how I would describe the Kubernetes market today and then what that means to a young technology company.

The Kubernetes market today is primarily dominated by large vendors; Red Hat, with OpenShift, Google, Amazon and Microsoft. If you think about if I'm trying to stand up a Kubernetes service and allow anyone who wants to run their Kubernetes, their container apps in Kubernetes, it's going to be very hard for me to compete with those vendors.

I would say all of those vendors, especially the cloud vendors, are driving the price points of a Kubernetes service to close to zero, and they can afford to do that, because they've got other services that add value to Kubernetes-based apps that they can sell to the market. So they just want new applications, always new apps that are going to be containerized, run them on our Kubernetes service. Which is why you're seeing a lot of interest from Azure, from AWS, from Google and OpenShift and providing core-based Kubernetes services for the enterprises. It's going to be hard for a young technology to compete directly against them for core base Kubernetes services.

So then I think if I'm a young technology company, how do I then monetize, or where is their opportunity for me in the Kubernetes space? I think that they're going to have to move up the stack so to speak. They're going to – You will see no doubt on the cloud vendors continue to add more functionality on top of their Kubernetes service, whether it's to help with monitoring, whether it's to help with alerting, or whatever, and this is beyond Istio, for those container-based apps.

So how can I as a young technology company make sure that there's still open space for me to monetize in the Kubernetes space? There's going to be places for that. It'll be higher up the Stack. It could be around management and monitoring that take advantage of how I could manage Kubernetes apps across platforms, on-prem, in cloud, across cloud. We're seeing companies like HashiCorp, where I've got good friends over there from Hortonworks and they're executing extremely well as a company that originally were competing with Kubernetes, with Nomad, but now very much embracing, still supporting Nomad, but very much embracing Kubernetes, but providing a platform that is not about running your Kubernetes app. It's about

how can you move it around. How can you secure it? How can you monitor it wherever you want to be running Kubernetes-based applications?

Those are things that good technologists listening to this, those of the problems they should be looking at focusing. Not, "Here, run your Kubernetes app on my platform." Instead it's, "How can I help you run it better? Whether it's more securely, whether it's easier, whether it's making it more transportable, making it easier for people to deploy them, to change the apps, whatever. That's where I think there will be newer term opportunities for technology companies.

[00:55:13] JM: One thing I thought was interesting at Google's recent conference, they had a couple technologies around their Kubernetes – Well, I guess it was around some monitoring technology, but Google seems to be open to a multi-cloud vision even though they are a cloud provider. But I guess if you have a "multi-cloud strategy", you would still rather bet on HashiCorp technologies rather than Google Cloud technologies.

[00:55:47] MF: Possibly. Again, I said, the people that I like at Hashi. I think with all three of the cloud vendors, we will continue to see them evolve their solutions for on-prem and cloud. Early on, you saw Amazon do it with things like Snowball and others to let me help you suck your data on-prem faster into the cloud. There are going to be applications that there's no way they're going to, or I shouldn't say no way. Possibly no way, but it will take time for a company to migrate them to a Kubernetes service or even running it its own dedicated cloud service.

Those are opportunities for these cloud vendors to take advantage of to manage or control the voice of the customer. I think Microsoft is in a great position to that. Look at how extensive their enterprise salesforce is. Look at how much on-prem usage of their enterprise software there is in the market. Azure is doing really well. I think they're in an interesting position I think with Satya and team executing really well to take advantage or implement, successfully implement a hybrid cloud strategy.

I think Amazon, an obviously, they're just such a powerhouse. They're in a great position. I think Google is too. There are just other things Google needs to do. My opinion to catch up to Microsoft or to Amazon, and I think Google, with some of their partnerships, with Pivotal, and

Red Hat and others, are using these partnerships as a way to give them a competitive differentiation in a hybrid cloud play, meaning on-prem and with GCP.

[SPONSOR MESSAGE]

[00:57:39] JM: Leap.ai matches you with high quality career opportunities. You are more than just your skills, and a job description, and a resume. These things can't fully capture who you are. Leap.ai looks beyond these details to attempt to match you with just the right opportunities. You can see it for yourself at leap.ai/sedaily.

Searching for a job is frustrating, and Leap tries to reduce the job search from an endless amount of hours, days, weeks to as little as 30 seconds trying to get you matched to a job instantly by signing up based on your interests, your skills and your passions.

Leap works with top companies in the Bay Area, unicorns and baby unicorns. Just to name a few; Zoom, Uber, pony.ai, Cloudera, Malwarebytes and Evernote. With Leap, you are guaranteed a high response and interview rates. You can save time by getting direct referrals and guaranteed interviews through leap.ai. Get matched to jobs based on your interests, your skills and your passions instantly when you sign up.

Try it today by going to leap.ai/sedaily. You would also support Software Engineering Daily while looking for a job. Go to leap.ai/se daily. Thank you to leap.ai.

[INTERVIEW CONTINUED]

[00:59:18] JM: Let's fast-forward five years. Most large enterprises will have a Kubernetes cluster. They're going to know how to operate it. Maybe it's via OpenShift or Platform9, or Mesosphere, or raw AKS, or EKS or Google's cluster, Google higher level Kubernetes thing. What kinds of businesses? What kinds of startup opportunities will present themselves in that world?

[00:59:49] MF: Five years is a long time. It's hard for me to think five years out in the technology space. I would say when you're looking at something like Kubernetes, Kubernetes is a platform.

I'm going to need to provide some sort of solution on top of it. It could be a solution that could be used across companies or it could be solutions specific to industries that are going to help them use, help them develop or utilize container-based apps better.

I think I'll make another correlation, blockchain. Blockchain really has a lot of buzz around blockchain today. For blockchain to really take off, it's going to have to be solution-based, and you're really seeing some of that already. Here's a young company we're building blockchain apps for the insurance industry, or we're building blockchain apps for the manufacturing industry. Versus, we have a blockchain platform. You market decide how how you want to use that for a young company will be very difficult to pull off.

There are some companies doing that, most notably IBM, trying to build a blockchain platform, and you'll see them do more around solution-based marketing. I'm sure on blockchain, you're seeing digital asset holdings do similar thing, but they're very focused at least at my impression when I last looked at them, very focused around trading exchanges. Kind of that's where they see blockchain taking off.

So I see that similar analogy with really any core infrastructure technology. Kubernetes is a core infrastructure technology. As a core infrastructure technology becomes more prevalent, if am going to build – I'm a young company, I'm going to build something related to that, it's likely going to have to be solution focused.

[01:01:44] JM: That's so interesting. So you could imagine we're the Kubernetes company for your assembly lines. We're the Kubernetes company for your oil rigs, something like that.

[01:01:56] MF: Yeah. I don't know if I would say we're the Kubernetes company, but your applications are on container-based for the oil, and it's just a very – Not to use a play on words, very crude example. You are developing all your new apps around pipelines and whatever in that industry. I'm not as familiar with it.

So you're building all of those in containers and running them on Kubernetes. We have technology that helps your Kubernetes apps better understand the data coming from sensors of this type that are widely deployed across your pipelines. You saw that with other companies,

and SAP is partnered with some that are dealing with pulling data coming from sensors off of various manufacturing platform systems into SAP apps so they can do analytics on it.

Let's go back to my crude example. Just because my app is a Kubernetes-based app, as Kubernetes is prevalent, is less meaningful, versus how am I helping that industry or that customer use Kubernetes better for their applications. Well, they might need to get – More easily get data coming from other systems into their Kubernetes apps. I'm helping with that. That's the solution with the valuable that I'm providing.

[01:03:17] JM: You could be doing a lot of different things. You could be an investor. You could be an executive at Google. You started a unique type of business. Why did you decide to start AccelG2M?

[01:03:28] MF: So I don't know if I could be an executive at Google, because I've generally worked at young companies, but maybe Google would want to hire me, and it's a great company I think. My preference is working with young companies, whether it was working for them, and I've done that number of times and really enjoyed that. Very much enjoyed the all-consuming nature of a startup. So whenever I am doing a startup, I am totally addicted to adrenaline and it is seven days a week. That's just the mindset that I have.

Do I appreciate work-life balance? Yes. But when your future is uncertain as a young technology company and you're burning cash and you have a window of opportunity to make or break a successful business, I'm going to do everything possible to focus on capitalizing on that window of opportunity, and that adrenaline rush comes with it. I enjoyed that. But I've done that a number of different times.

I've learned a lot of things I've learned from people that I've worked with. I've learned a lot from people that I work with. At least where I am right now, and who knows if I'll go back to another startup, I've got the scars. I've seen what works and what doesn't work. I want to continue to learn, and I think there's a lot that we at AccelG2M can provide in value to assistance to a first-time CEO, a technologist CEO, a young technology company, whether it has no revenue, whether they've got $40 to $50 million in revenue, to help them optimize how they are going to market because things continue to change.

I think when we started Hortonworks in 2011. So what is that? It's seven years ago. We didn't have to think about a cloud-based offer. Today I would say if you are a young technology enterprise software company, it's imperative that you have some sort of service-based offering. I shouldn't say for all. It's not a black-and-white answer. But for many, it's imperative that they have some sort of cloud-based solution service, whatever you want to call it, because there is so much momentum. Just look at every quarterly earnings from Amazon, Azure, Microsoft and Google, the slipstream that is pulling apps and data to the cloud is not something we've seen in a very long time.

If you're a young technology company, you have to take advantage of that slipstream or you'll get left in the dust. So those are things that we're helping with these companies. I touched on earlier, whether it was SpringSource or when I said a company called Composite Software, or early days of Spring, we didn't have anything to sell. We had technology and we thought we were selling services or something, and we had to come up with, "Hey, this is what we're going to sell and this is who we're going to sell it to, and we think these are the value propositions to the market. We're going to listen to those customers and see what resonates. Those that resonate, great, we're going to continually evolve those messages. We're going to add more fuel to the go-to-market strategy and grow faster.

I think with what we've seen with a number of different younger technology companies today, and this is not at all a criticism. A first-time CEO or a technical CEO, they're building their technology and they believe that this is the value of their technology. If it's not selling, they think they have a sales problem, and they may or may not have a sales problem. But the first thing we will look at is what is your technology do, and to whom is that of value to and why is that of value to them? That's product market fit.

Generally we'll see a bit of a mismatch in product market fit. I might be selling to the wrong people. I might be not articulating the right value props, and that's influencing my go-to-market momentum. So we have to address that first and then we also look at, "So, with the go-to-market, the sales and marketing strategy, how effectively are they targeting how they should be taking this technology to market?" Those are generally areas that need to be tuned. So that's where I go back to early on in the podcast. The alignment between what I'm building for whom,

how I'm articulating it, and why that should be of value to them and how I am communicating or interacting with those people; product, sales, or product, marketing and sales, the alignment of those. How closely aligned and moving at the same pace will dictate how effectively the companies go-to-market. That's what we're doing. We're helping those companies with that with AccelG2M.

[01:08:27] JM: Well, I hope there's some people listening out there that find that appealing and maybe reach out to you. I wouldn't be surprised.

[01:08:34] MF: Sure. Www.accelg2m.com.

[01:08:41] JM: Awesome. Open sourced based business models, do you feel like those have change? I was talking to somebody at dinner recently and they were just outlining the fact that you start an open source project and then a cloud provider takes it off-the-shelf and rebrands it as a cloud provider product and then monetizes it. It's pretty hard to compete with that. What are the open source based business models that work today?

[01:09:11] MF: This is an excellent question. When I touched on earlier with our company, AccelG2M, product market fit. Part of that product market fit is product packaging as well as licensing model, especially for young companies. We just finished an engagement with two companies that are very prevalent in open source technologies, helping them with their packaging and licensing. So you're right. That is a challenge for some companies with some of these big cloud vendors where they could disintermediate their market opportunity by just taking it, because license allows them to. The technology just stand it up as a service and really create a – I wouldn't say a barrier to entry, but certainly momentum against the young technologist company's ability to build a success [inaudible 01:10:01] viable business.

But I want to first stage, a vibrant open source community is critical for any open source project. What I mean by vibrant, the bigger the community the more people that are interacting with it, that are contributing to it. I believe the better a company has an opportunity to monetize that. A small community, if I only have a hundred users of my open source technology, or a thousand, or 10,000, it's going to be harder for me to figure out a way that I can monetize that open source tech.

So, again, I'm a strong believer of open source technology, because it provides – There's a lot of interest in enterprises. It provides a value to the market. From a business perspective, I can provide a very effective way to get my technology in the hands of lots of people. So I love that model, but it's changed a lot. It's changed a lot since the JBoss days and the SpringsSouce days and the Hortonworks days.

Hortonworks business model is 100% Apache software licensed, and at least for [inaudible 01:11:11] platform and support around that. That would not be a monologue I would implement today, because I think the market, it'd be too difficult for a young technology company today to do that. Hortonworks is very successful doing that and they've got a strong presence in the market and position in the market where they're able to do that today.

To your point, I think we saw with Elastic, and we've seen it with some other technologies where some large players take that product and then they stand, they'll open source project and they stand it up on their own. So there are more unique licensing schemes today that an open source technology company can implement. That not only helps them maintain their relationship with the open source community and nurture that open source community with an open source license. But also leverage a licensing model that protects their intellectual capital and additional intellectual property in a way that would necessitate a larger player to need to have a relationship with that young technology company if they wanted to capitalize or leverage that open source software. Those are some things that we help these companies with.

So they are additional licensing schemes that take advantage of open source as well as maybe third-party open-source and proprietary technologies that an open source vendor may need to include in their offering in the market in a licensed model that makes it easier for an enterprise to adopt, because you have to think about even if I am an open source technology company today and am trying to sell to an enterprise, whether as a cloud service or on-prem. That what I'm selling, if I'm asking for money, is going to have to go through a buying process. It's going to likely have to go through a legal or procurement process.

There are certain things that companies need to do today to make it easier for their open source offering to more cleanly go through an enterprise procurement process. If you think about very

early, like early days of SpringSource, and certainly with JBoss, JBoss initially was selling indemnification for their open source offering. That was their monetization initially, because these enterprises were saying, "Hold it! I can't bring this open-source technology in. It's going to poison all my other software, and then I'm going to lose control." So they were selling indemnification.

Now, if you think about it, are there enterprises that still ask for indemnification? Sure. But enterprises are so much more educated on what open source licenses are, what they mean, viral effect of open source. How it may or may not be harmful to their offering. They're so much more educated. So it's less of you have – The kind of licensing models from open source 3 to 10 years ago today are different given the understanding of open source in the market, the competitive nature in the market, and therefore it necessitates young open-source vendors to use more innovative licensing schemes to provide to capitalize on their open source, but also give them a viable opportunity to build a business around their open source tech.

[01:14:27] JM: Well, we're planning to do some shows around licensing in the near future. There's obviously – As you've seen in the recent news, there's a lot of hunger to better understand those questions.

[01:14:38] MF: Happy to help. We've got someone on our team that would be very well-versed in that, from the JBoss days, SpringSource days, Hortonworks days.

[01:14:47] JM: I'd love a referral. Yeah, send them my way.

[01:14:49] MF: I will do that.

[01:14:49] JM: Okay, last question. We've covered most of the the buzz-wordy things, which I think are material buzzwords, but you've even touched on the blockchain related infrastructure.

[01:15:03] MF: I didn't say crypto.

[01:15:04] JM: You didn't say crypto. That's true. We didn't say machine learning, but we said data engineering. So I think that's close enough.

[01:15:11] MF: It's interesting. There's another example, machine learning. Machine learning blockchain, they're words. Machine learning, those companies that are doing well in machine learning are providing solutions specific to markets, versus technology and asking the customer, "Figure out how to use my machine learning stuff."

[01:15:29] JM: That's for sure. Okay. But the last couple things that I did want to ask you about, do you see any places in serverless or edge computing that cannot be captured by cloud providers, or will be captured by businesses outside of the conventional cloud providers?

[01:15:44] MF: Give me 90 days, and I will give you a more educated answer, because we're likely going to be working with a serverless company very shortly.

[01:15:56] JM: Okay.

[01:15:56] MF: There's an example of how do we need to capitalize on a vibrant community? What should our packaging strategy be? Product market fit. How do we play with the cloud vendors? Work around them? Work with them? That's important. But serverless right now is really you. Kubernetes is mature relatively service-based in my opinion.

[01:16:20] JM: Right. Completely agree. Edge computing?

[01:16:22] MF: I can't comment on that only from my kind of days at Hortonworks, with IoT, and I think that's still a market, a young market.

[01:16:29] JM: Yeah. Okay.

[01:16:31] MF: I'm just giving you an educated answer at this point.

[01:16:33] JM: Yeah, totally fine. Mitch, it's been really great talking to you. I really appreciate the breadth with which you're able to approach the software landscape, and it's been a real pleasure and education for me.

[01:16:44] MF: Jeffrey, happy to be on and thanks. For some of your listeners probably think I'm a suit, happy to be a suit on your podcast.

[01:16:53] JM: Okay, great.

[01:16:54] MF: Thanks.

[END OF INTERVIEW]

[01:16:58] JM: Azure Container Service simplifies the deployment, management and operations of Kubernetes. Eliminate the complicated planning and deployment of fully orchestrated containerized applications with Kubernetes. You can quickly provision clusters to be up and running in no time while simplifying your monitoring and cluster management through auto upgrades and a built-in operations console. Avoid being locked into any one vendor or resource. You can continue to work with the tools that you already know, such as Helm and move applications to any Kubernetes deployment.

Integrate with your choice of container registry, including Azure container registry. Also, quickly and efficiently scale to maximize your resource utilization without having to take your applications offline. Isolate your application from infrastructure failures and transparently scale the underlying infrastructure to meet growing demands, all while increasing the security, reliability and availability of critical business workloads with Azure.

To learn more about Azure Container Service and other Azure services as well as receive a free e-book by Brendan Burns, go to aka.ms/sedaily. Brendan Burns is the creator of Kubernetes and his e-book is about some of the distributed systems design lessons that he has learned building Kubernetes. That e-book is available at aka.ms/sedaily.

[END]