EPISODE 526

[INTRODUCTION]

[0:00:00.3] JM: Steve Herrod led engineering at VMware as the company scaled from 30 engineers to 3,000 engineers. After 11 years, he left to become a managing director for General Catalyst, a venture capital firm. Since Steve has both operating experience and a wide view of the technology landscape as an investor, he is well-equipped to discuss a topic that we have been covering on Software Engineering Daily, which is the integration of cloud and edge computing.

Today, we think of the cloud as a network of large data centers operated by big players like Google, Amazon and Microsoft. The cloud is where most of the computation across the world takes place. My smartphone and my laptop are edge devices. They are lightweight computers that don't perform much complex processing.

I would not be able to run a large production database, or a 3 terabyte MapReduce job on my laptop. The current division of labor makes sense in this world of smart clouds and low power, low bandwidth devices. The devices are getting cheaper and smarter and more proliferate. Cars and drones and security cameras and sensors and other devices can serve as points of computation that are geographically between the edge devices and the cloud.

With more devices between your edge device and the cloud, there's an opportunity to put computation on these devices. Instead of imagining your device simply at the edge that requests data from this super smart cloud, you can imagine a mesh of different devices geographically in between your device and the cloud. Shouldn't those devices that are between you and the cloud, shouldn't they be handling some of the computation?

This is big loss of productivity, because all those devices in between you and the cloud, they're wasting compute cycles. Theoretically, they should be able to utilize those excess compute cycles to do some of the work that is required by either your device or the cloud. There's an opportunity to put computation on those devices, so everyone knows that cloud and edge computing are going to become inter-mingled in the coming years. We know this, because it just makes economic sense.

Predicting how it will play out is nearly impossible. As an investor, if you bet on something too early, you get the result as someone who was wrong altogether. A good analogy for how you can be too early and get something wrong that's related to this is the smart home. Everyone knows that the smart home is coming eventually, but it's very hard to tell how long it will be before smart home systems are in widespread use. It's an open question, how to invest in the space? If you would've invested in the technology companies coming out five or 10 years earlier and they were smart home companies, five or 10 years earlier there were smart home platforms that people did invest in, you probably lost a lot of money.

This is a pretty interesting episode, very high-level. I can't wait to dive into more of these cloud and edge stuff as it comes to fruition, but we have dive into some of the stuff in past episodes that you can certainly find.

I wanted to announce that summer internship applications to Software Engineering Daily are being accepted. If you're interested in working with us on the Software Engineering Daily open source project fulltime remotely this summer, send an application to internships@softwareengineeringdaily.com. We would love to hear from you. We'd love to get some fantastic interns to take this project to a new level.

If you haven't seen what we're building, check out softwaredaily.com, or download the Software Engineering Daily app for iOS or Android. These have all of our episodes, all 650 episodes in a searchable format. We've got discussions, related links, lots of different material around those episodes. It's all open source. If you want to check it out, it's at github.com/softwareengineeringdaily. I hope you do and I hope you ping us about it.

[SPONSOR MESSAGE]

[0:04:20.7] JM: If you are on-call and you get paged at 2 AM, are you sure you have all the data you need at your fingertips? Are you worried that you're going to be surprised by things that you missed, errors, or even security vulnerabilities because you don't have the right visibility into your application? You shouldn't be worried. You have worked hard to build an amazing modern application for your customers. You've been worrying over the details and dotting every I and crossing every T.

You deserve an analytics tool that was built to those same standards, an analytics tool that will be there for you when you need it the most. Sumo Logic is a cloud-native machine data analytics service that helps you run and secure your modern application.

If you are feeling the pain of managing your own log event and performance metrics data, check out sumologic.com/sedaily. Even if you have your tools already, it's worth checking out Sumo Logic and seeing if you can leverage your data even more effectively with real-time dashboards and monitoring and improved observability.

To improve the uptime of your application and keep your day-to-day runtime more secure, check out sumologic.com/sedaily for a free 30-day trial of Sumo Logic. Find out how Sumo Logic can improve your productivity in your application observability whenever you run your applications. That's sumologic.com/sedaily.

Thank you to Sumo Logic for being a sponsor of Software Engineering Daily.

[INTERVIEW]

[0:06:04.9] JM: I'm here with Steve Herrod, the Managing Director of General Catalyst. Steve, welcome to Software Engineering Daily.

[0:06:09.7] SH: Thanks for having me.

[0:06:10.8] JM: Today, we're talking a little bit about your history, a little bit about the future. We'll start with the history. You were at VMware from 2001 to 2013. VMware was successful because of both fundamental computer science breakthroughs, but also specific insights about how engineering organizations were working back when it got started and how they might be able to work more productively. What were the key reasons why VMware was successful?

[0:06:41.1] SH: Yeah. Especially now as an investor, I realized there is always a good preparation, but there is always a little bit of luck and things happening outside at the same time that work out so well. If you rewind yourself all the way back to those of you that were around the very start of 2000 or 1999, a few things were happening.

It was a time when Unix was big. It was a time when there are all these different CPUs being made by people. There were MIPS processors and Alpha and these things that something you might have heard about. Intel was not known to be the killer company at that point. Then there is the emergence of different operating systems and certainly there was Microsoft Windows, it might've been like Windows Millennium or something at the time.

Linux was getting some headway as well. A lot of things happened at the same time. Intel processors were getting fast enough that you could use them to actually – they had excess capacity. Also people cared about developing for the X86 processor for the Windows environment and for Linux.

VMware actually started – originally, it was the idea, "Hey, I want to try out this new Linux thing, but I don't want to have a dedicated machine for it. It would be cool if I could just run Windows and Linux on a single machine. That's why we got a lot of developers interested, a lot of QA testing people interested. We certainly had a long-term vision for where this would go. That was really the origin story.

[0:07:56.8] JM: You were there as the company scaled from 30 to 3,000 people. What were the key points along that scaling, where you had to change the management – I did a show recently with somebody from Fox and they were saying like, you go from having an individual person who's in charge of something to a team in charge of something, to an entire department in charge of something. It sounds like that was, provided at least some structural schema for how they thought about scaling.

[0:08:26.6] SH: Yeah. To be clear, that was our engineering department –

[0:08:30.3] JM: Just engineering section.

[0:08:30.9] SH: Yeah. The company itself is I think around 14,000 people at the time I left. I spend a lot of time actually doing this on the investor side is trying to tip to people about what the scale will look like as you get to the next level.

If you talk to almost anyone who's been through it will say that your job is entirely different at each order of magnitude. When you go from 1 to 10, from 10 to a 100, from a 100 to a 1,000, and largely it gets harder or more difficult to make things happen.

Really what I keyed in on the most was how you communicate as you go through each phase. When it's just a couple people around a pizza sitting at a table, you can have high bandwidth conversations, people get it. You're interacting and course correcting all the time. Each order of magnitude is almost the number of times you have to repeat saying the exact same thing to each different type of media, for things to really set in.

I would say that's the number one thing is how do you get everyone aligned on the mission, on the priorities on how we're going to do things? You would do it by meeting in person and you'd actually maybe at the time there were podcast quite as much, but you do a podcast, you send out the e-mail. You had to make sure that people could absolutely understand things according to the way they like to listen to them.

The other thing is you will find very, very few companies that have gone to that scale at one site. There is a natural – as you always talk about, there is a natural challenge when you go from having everyone in one building to when you have them in multiple sites as well, which definitely hits on the notion of communication and how you get that across to them, but there's so many other things too around how you organize yourself across offices, which we could talk about. Just even how do you make sure that it's very easy to – we used to call it throwing grenades when you don't really know the people at the other site, you can blame them for everything and vice versa.

[0:10:16.2] JM: Yeah. What about key moments in that evolution in terms of how the company was evolving. Was there anything product developments, or releases that were extremely hard to finish or get out, where the organization was really strained, things that align there?

[0:10:35.0] SH: Lots of things. I think that's really a great question to go into. You always hear about the – some people call it second system syndrome. I think it's third system syndrome is the one where you really get in trouble. This is when you've shipped some kind of product, you've got some good customers, you got things working pretty well.

I typically find in companies that the second version of the product there were so many glaring holes that you had to fix to enable the next set of sales. These are especially in the enterprise world. These are things like manageability, or high availability, or like there are some base-level security that has to be in place.

All these enterprising stuff that are so glaringly missing from your first big product release that you just have to do it. We got into challenges and certainly you heard about a lot of companies on the third version of a product. It's when you're feeling pretty good about yourself and you've got an idea for all the different things you could do now that you have this core platform.

We had this problem and is definitely a big chunk of those. My responsibility for trying to do too much in that third version of the product and you end up just in the release hell, where you're trying to get everything ready and you've got too much confidence. Everything takes a little bit longer than you hoped and you end up having to cut features late and you're late on the product. That 3.0 product for us was one of our growing up moments for me personally, but also for the whole engineering department.

[0:11:55.8] JM: Did you learn anything about decision-making in that time, where you – because if you had eventually cut features from the product that probably made you maybe reflect on, "Oh, we should've cut that earlier. We should've reconsider that earlier."

With also the knowledge as you said earlier that things were moving slower at that time so there's probably more of a penalty to taking longer to make a decision, because that compounded and compounded. Did it change your framework for decision-making around software development?

[0:12:29.9] SH: Absolutely. It hits for all the things you talked about in the show quite often too. You would think that when you have more people, you can move faster and do more stuff and the same level productivity holds. In actuality as you can imagine, it’s all dependencies that you didn’t have before that you weren’t aware of.

As you are getting to this third version of the product, the number of dependencies that happen between systems is larger than ever. It’s a classic project management where one thing can be holding up the next thing, which is holding up the next thing. If you don’t get the ordering exactly right, you start building on a house of cards a little bit. Like the fundamental core APIs you’re dealing with, or the core services just have to be there before you have other people piling on top of it.

I would just say that the dependencies in making sure that the most important thing is the foundation that everyone else is building on, doing that first and making sure it’s solid would’ve been a big deal.

The other thing you have, especially when you have a ambitious set of developers is every month to ship new features and capabilities. There’s a big morale thing around, “Hey, sorry. Your particular feature is not going to make it.” You see this a lot in the companies I work with now too is there might be some group, especially as you fragment up your department that’s responsible for something; a new file system, a new capability.

That’s only part of what it means to ship the whole product. You not only have to have this capability, but it needs to be exposed through your user interface, or through your APIs and it has to go through the full realm of the QA department or system testing. It has to be documented by the way. It would be nice if marketing could actually talk about it in a product release.

What used to feel as a simple thing, I’m just going to develop this new feature, has six, seven, eight other groups that have to do something for it ultimately to get out into a enterprise product. The lesson for me was you have to make the decision even earlier and manage the team that’s unhappy about this.

It’s all these other teams that are impacted that are not going to get it out the door. In many cases, it causes more problem if you think you can make it and then there is all these other groups that can't deliver. As you can imagine, you then have the original group saying these clowns and documentation can't get their job done and held up my cool feature. It's as much of a people dependency and morale management, as it is like what is fundamentally going to keep us from getting out the door.

[0:14:47.5] JM: When you're talking about the penalty for going too long, or having to cut features late in the game on the engineering side of things and how that can have ramifications downstream, like in marketing teams, or documentation teams, or sales teams. That can be really impactful and it makes me wonder about the whole go-to market process, because VMware was novel at the time.

You had to sell the idea of virtualization. My history is not very well developed in this area, but it almost sounds like what AWS has had to do, because they have had to evangelize the idea of the cloud and say, "No, this is like a really good way of doing things." It's not just like a crazy thing for just startups. It's something that pretty much everybody should be doing. When you've seen AWS mature and lays the path for the cloud, does that remind you of the evangelizing that VMware had to do?

[0:15:49.0] SH: It does on a few fronts. VMware was started as this tool that everyone and developers and QA people really loved. That's great and that's a more forgiving audience for sure than when you're running your crown jewel applications in a data center on top of that.

I would say the path that Amazon and VMware have gone through have been similar. You can really crank things up more quickly when it's only developers and testers using it. They're getting productivity gains and if it crashes, they restart it and life goes on. When you're trying to run your core e-mail, or your P system, or whatever it would be, that's just not acceptable. It's a business critical function.

I would say in phase one, a lot of the evangelism is to developers and say like, "Here is a better way to do things." They'll take it on. Phase two is ironically is when you slow down a fair amount by having to show that you have robustness and that you've been through the certifications and that you had your security audit, and that if you talk to the Amazon team they've gone through exactly that, which is early adoption by development test and then just a ton of work to get those mission critical application supported. Again, part of it is technical, part of it is everything else that goes on there.

We spend a ton of time early on convincing the big application developers that it was safe to run on virtualization. They wouldn't support say an Oracle database when run on top of VMware. You've seen Amazon, they'll do the same thing too where they, again you have to convince so many constituencies that it's okay in here.

[0:17:14.4] JM: Yeah. AWS launched in 2006. That was about in the middle of your tenure at VMware. As AWS is becoming popular, what was the reaction side of VMware? Did you have ideas for "Hey, maybe we should be doing something similar to this? Or maybe we should be leveraging it somehow?" What was the VMware response to the rise of AWS?

[0:17:36.7] SH: Yeah. It will be really fun to look at this in hindsight. I'd say there are a lot of reactions. One was to approach them and say, "Hey, how about you use VMware software to power your environment?" Their cost model and goals were such that they wanted these free and open source to be up to scale at the time.

[0:17:52.4] JM: You use Zen, right?

[0:17:53.9] SH: They use a very modified version of Zen. But yeah, at the time Zen was hot and everyone was interested there. They forked it off now and it's a very different version. Anyway, at one point it was, "Hey, try to use our software." That would be cool. Then we both have a common goal.

The second phase was this was at the time when adoption of the cloud was fairly slow and it was – you could make whatever number it was. Most of VMware customers were not even – they were poo-pooing the Amazon cloud as not safe, or like we know best how to run a data center. We're going to use our VMware software and our cool stuff.

You get a business model where we're doing extremely well selling to on-premises, data center and data center teams. It was this thing on the side that we were watching out for, but doing really, really well without having a story there.

Then phase three is the one where we really thought we had a great solution and part of it, because we're doubt for sure, which was, okay there is going to be ton of clouds out there, public cloud services out there. There is going to be regional cloud services for each country. There might be a healthcare cloud and a federal cloud. There is a State of Illinois Cloud that we got involved with.

Then our path became, "Hey, there is going to be a plethora of clouds out there." Let's give them all the software. That actually worked really well. They needed some way to compete with this emerging Amazon thing and VMware software is considered very enterprise ready.

We had this great coalition of the willing that we're doing, their clouds based on our software. At this point, it's very clear that Amazon, Google, Microsoft, the level of capital expenditures, as well as technology that they've put in to their own public cloud offerings are so much larger than a lot of these other ones that you just have to have a story with them in addition to these other stories.

[0:19:35.6] JM: How does that – I don't know if you can answer this, but how does that impact the strategy of the other cloud providers, if you're somebody like Digital – Digital Ocean might be a bad example. Or VMware today, or Linode. There's a bunch of smaller cloud providers. What strategy would you recommend?

[0:19:57.2] SH: Yeah, it's tricky. Again on the venture capital side now, you can't meet a single startup without asking two questions. One is what is your open source-related strategy? The other is what is our Amazon, or Google, or Microsoft-related strategy? It does permeate everything, just the sheer scale they have to build up these systems.

Every one of the ones you mentioned, you can build a really great business by having some niche that you're strong at. In the case of Digital Ocean, the developers love them and they made the simplest possible way to just get going. They've kept their developer love best they can tell and really that's been a calling card.

It's very clear, Amazon has them in their targets in trying to keep getting better at their own developer friendliness. I think the areas we're going to definitely see diversity is when it's a very vertical-specific public cloud offering. The rules around the financial markets, or around healthcare, or around retail can be such that maybe do you have these cloud services that are very well-poised for providing not only a cloud that approved and certified for whatever regulations are there. It might have value-add services that are very specific to things you do in that industry.

Then the other thing is everyone has noted, as we grow even more global, there is just a fair amount of data restrictions and more all the time now such that people want to keep their data within their boundaries of some kind. There is pretty much every country on earth has its own – you see a Telco company providing a public cloud service.

I still think we're going to have a nice mix of clouds that are out there, many of which have their own niches that are still very good businesses, but there are still these looming BMFs on top of that that are going to be ubiquitous and very heavily used.

[SPONSOR MESSAGE]

[INTERVIEW CONTINUED]

[0:22:55.2] JM: Off the companies that you see, how many of them are multi-cloud? Do you see a lot of multi-cloud companies these days?

[0:23:00.3] SH: We do. It's something everyone talks about. I think almost every customer I spend a lot of time with bigger CIOs of companies, and every one of them on earth has – it's funny, because we went to this at VMware as well. Every one of them has been through the process of some vendor get super strong and then has – makes them feel nervous from a pricing, or from a – even just a risk management.

I mean, this was IBM in the 60s and 70s, Oracle have this mantle for a long time. Sun became super strong in so many accounts that people were looking for a way to knock them down a little bit. VMware absolutely, people were worried about too much going to us and too much pricing pressure, then now on the Amazon site as well.

You always hear customers saying, "I really want the option, or at least the price negotiation time. I want the chance to say I have a viable option somewhere else." They'll put some amount of engineering effort, or business effort into keeping something alive on one of the other clouds in order to do this.

We see a lot of customers that will put their data in S3, but they'll also put it say in the Google cloud. At least with that, they can have a bit of a hedge in saying, "Hey, I could move over here if I need to." I invest in a lot of companies either on the systems management side or on the security side, or even on the developer side, where the calling card will be, "We will work across multiple clouds."

It's easy to say that and it's easy to say on multi-cloud we enable choice, etc. The problem you get with almost any company is these clouds move so fast as you're saying the engineering velocity of new capabilities is so fast that you now have say a startup trying to bring out the latest AWS service, or trying to enable the cool new AI thing that is enabled.

Unfortunately, you too often end up with a lowest common denominator, or at least certainly lagging behind the innovation speed of these clouds. It's very hard to be multi-cloud and bring the most you can out of each of them. The core concept of give me optionality, or give me some kind of leverage for moving across them is alive and well.

[0:25:04.1] JM: You're seeing people doing multi-cloud a little bit more defensively rather than – because I've talked to some companies where like, they use Google because they have access to BigQuery for example. It's like, "Oh, this is really high-leverage service and there is no equivalent on AWS."

It makes me wonder how much – is that what we're going to see in the future, like more exotic differentiated services where you actually want to have multi-cloud – you want to be ready to be able to use services like exotic services that spin up on Google, or spin up on AWS and those are like totally disjoint opportunities.

[0:25:41.6] SH: Yeah. I think it's going to be fun to watch. There is again, these core capabilities, the amount of time that's going in to the latest, greatest machine learning or image recognition, or NLP. These services are moving at rocket speed across all of the platforms. I think in a short-term, you'll definitely be – this is the better cloud to be on for the latest, greatest something.

I think we're also seeing the early stages of emerging strengths for at least the big three clouds that are out there. I think most people would say at this moment, AWS is more attractive to developers coming out the door and they've really evangelized that well. You'd hear Microsoft is probably better at partnering with traditional enterprise companies and they work close to your Office 365.

They have a lot of their traditional benefits. Google most people would say that their ability to run at scale and to do special AI related technologies is probably a bit ahead of everyone else. Plus you have massive amounts of dark fiber for interesting internet connectivity. You see these emerging moats, or at least offensive areas for each of them, but –

[0:26:45.8] JM: Would the dark fiber, what does that enable?

[0:26:48.1] SH: Certainly, Google has invested in – actually all the clouds are doing this too, but investing in their own dedicated connectivity between their own data centers, so that the speed of moving data across data centers is faster than you could do over some more public network. Increasingly, you're seeing companies that find a way to get directly locked in a very direct network connection into these clouds from their own data centers using services from companies like Equinox or something.

They're able to get really low-latency, high-bandwidth connectivity into parts of the public clouds. I think that's a really offensive move to just make sure that you can move tons and tons of data into them efficiently and have a really good experience there.

You couple that with just massive capital outlays; every month you hear about the newest data center opening in a different spot. That is about, certainly about extra redundancy, but hey if I have a business that has a low-latency service, I'm going to choose the one closest to me, because it's going to be a better customer experience. I think you're going to see these battlegrounds that are just fighting out left and right and then some of the emerging strengths from each of them.

[0:27:53.0] JM: Yeah. I think we're going to get into a little bit of edge-related discussion, which I think would be related to this conversation. I want to talk a little bit about containerization, because with containerization we saw that the deployable unit got even smaller than the VM. Although, I guess with VMware – I don't know if you were thinking about these VMs as deployable units, as much as gork stations maybe. You can correct me if I'm wrong, but when you're at VMware, did it occur to you that the ways that people would be running applications could get deployed to something even smaller than a VM?

[0:28:31.6] SH: Yeah. It was certainly a topic all the time. If you go all the way to the top level, whether you're a cloud service or you're a infrastructure software service, sure you think about number of license and sold, or how much usage there. At the top, top level, you want as many applications on earth as possible to be running on top of your stuff.

We used to have this discussion all the time. What are today's applications? What are tomorrow's applications? How do we make sure that they would run on top of VMware as well as possible?

An application that runs, like you used to have [inaudible 0:29:04.2] and now it's running as a salesforce in some public cloud. That's never going to run on top of your own data center software. That's an application that will never end up on top of some software that you have there.

You're looking at things that move this as from on-premises as a set of things that move away. You're absolutely looking at what's the next generation, like what are the cool kids building that are cloud-native applications as you would call them now. How can we make sure that we add value when people are trying to create those applications and run them?

We watched containers where they – these user-level, the Linux containers early on that were less glitzy and cool. There was certainly a way that people were running software within a virtual machine. I think what the core concept and forgetting the company, the core concept of a container that allows you to move your application and run it anywhere. We've heard that a lot. I mean, that was Java's promise, that was VMware and that's a container.

It is about finding the right level of interfaces that let you spin up things and move them and share them. We launched something at the time at VMware that's so fairly popular in infrastructure software, which was a virtual machine as the distribution package. They're called virtual appliances. It's a way to distribute software that's fully configured, a full stack of software and anywhere where there is this hypervisor running. You can just get it up and running it.

I would just say the containers have done a great job of doing that too. I would just say they have less of the full stack in them. They're counting on a common version of Linux below the covers. It's the same concept of portability. I would just say it's lighter weight than a virtual machine, so it's smaller and you can probably run a few more of them in the same place.

The core notion of portability is obviously big there. I think when you couple this lightweight containers with the other modernization of software, which is the notion of microservices and decomposing a big hunkin' app into a bunch of little components, containers nicely fit with that as well. You have these things that are encapsulated talking over some network connectivity.

I think you – in the particular case of containers, you had nice efficiency and distribution mechanism coupled with a decomposition of software that fit fairly well. Now we're going into the next thing, which is serverless, which is you add another unit of computation, that I would argue has more potential benefits than virtual machines or containers in many cases in terms of security or actual total usage and efficiency. You're going to see these next generation encapsulation of applications, hopefully until the end of time. It's just a modernization of the software process.

[0:31:39.1] JM: What makes you say that about serverless?

[0:31:41.2] SH: I think it's fascinating. I think it's a bigger jump from VMs and containers to serverless, and it was from like a VM to a container. That's because for me, I spend a lot of time in cyber security investing, for instance. One of the biggest challenges with cyber security is that you have all these servers that are always on listening on course, and they're sitting there with often old software waiting for someone to come in and get into them.
Whereas, serverless is not running unless it's running. At least, it's not this idle infection-ridden thing that's sitting out there waiting for attack. I think that's pretty fascinating. I think the other fascinating thing too is because it's not taking up any cycles when it's not running, you can squeeze so many more of these functions onto a given piece of hardware. I think that's – it's just driving the cost of computation down even further than ever.

It's really interesting to watch how that plays out; still lots of challenges. If we thought containers were to get ready for mainstream, serverless is even behind that. Just a lot of the core development work to make it ready for prime time. It feels like a very good technology.

[0:32:44.8] JM: How does it affect your – does it affect investing thoughts at all? Or is this mostly like the edge of your periphery, just seeing Amazon and Google release serverless stuff. How does it affect startups that you've seen?

[0:33:02.3] SH: Yeah. I'm a enterprise investor and I do infrastructure. I do pretty low-level stuff and it's absolutely relevant. I spend most of my time making sure I'm aware of what the latest stuff is. However, my big focus is how do you sell to the Fortune 2000, the 5000 companies? My mantra, the main thing I always look for is this has to be something that's taking you to the future and it has to be – you want to tell a story for companies on once you use this software, you'll be better able to take advantage of this Nirvana that's at the end.

It also has to have some awareness of what the world is like today. The whole reason VMware did well is it was a bridging technology from all your applications today to the same applications running in a more efficient, or faster-moving way.

I would just say, I always look at the startups from the concept of great, it's an end state that people want to be in, but does it help with customer get there from this mess of stuff they have today? That's really the difference between looking at a company that is really built for and starting with startups and green field development as we'd say, versus one that wants to go and sell into an existing enterprise.

[0:34:10.5] JM: Yeah. I actually do want to come back to security, because I know that's one of your areas of expertise. You have a bunch of investments in security. I've got some questions about that, but I just want to get to discussing edge computing, because I have been some shows on this. I know you're thinking a lot about this. From your perspective, why is there an increased importance of computation at the edge?

[0:34:33.9] SH: I love the edge area right now. It has so many interesting implications on software development that I think is funny. If you look at the history of computer science, to some extent we've had this movement to centralize and then decentralize and centralize and decentralize. Just when you think the public clouds are centralizing so much calculations and computation, you end up with all things out in the wild that need to at minimum collect data, and at maximum do something with the data.

When you really start to think about it, you're trying to run – it might be AI models. A lot of your shows have covered things you want to do at the edge. I think you can just overlay some sort of formula on how you want to think about what computation should go where.

I've been trying to actually create a mathematical formula of some kind, but there are a bunch of factors and edge and IoT. It is so many disparate markets you have to be really specific about which one you're after. The traits you tend to look at is how far out are these edge pieces? Are they connected by a decent network, or a crappy network?

Are they so cheap that you can't afford to put a proper processor on them? Things you talk about a lot are do they have to be able to respond in a real-time amount, and then make some decision within a certain amount of time that would preclude them from sending stuff back to some central cloud.

I think you have to really take the traits of the use case that you're doing here. That dictates what sort of hardware do you need on these edge points, what sort of network connectivity do you need and again what kind of processing can you do there.

I've been thinking about almost to the same extent we talk about multi-cloud. You can think about the edge as being another cloud in some cases. The same challenges, how do I secure something whether it's running in a centralized public cloud, or in the edge, how do I provision and update software, whether it's on one or the other? How do I do performance management?

For me, it always comes back to those things around quality of service and are you actually getting the job done? Then you have to overlay that with the stuff is out in the wild, which it's hard enough for us to manage the desktop PCs that are actually on our desks in the office. How do you start managing cars that are driving all around, or these sensors that are behind traffic lights? I think it's just a much harder problem even in the ones we've done so far.

[0:36:47.2] JM: What I would love to hear your perspective on is like, is it too early to start thinking about this stuff? Because when I think about the places where okay, edge computing really matters. We're sitting in a high-tech room basically in a venture capital firm and I'm looking at a phone, I'm looking at a TV. There's a video camera over here. I'm looking at my computer, some recording devices, a lot of things, that maybe they would want to have some learning capabilities. Maybe these want-to-be devices at the edge, but we probably don't need them to be edge devices anytime soon.

The context that I've tried to explore are things like smart agriculture, or oil refineries where you do have a lot of devices or shipyards, where you do have a ton of devices we actually really do want real-time updates and machine learning at the edge today. You would want that stuff today. I mean, I don't know how – I guess, that's a big market, but it's also really hard to sell to. I don't know if they're ready to buy it. I don't know if the hardware devices are ready for this stuff.

That's a mismatch of thoughts. I mean, obviously when we get the drones flying around that side, we get the self-driving cars and we've got – we really have high-power edge devices that we also want to be deploying machine learning models to everywhere, then it makes complete sense to think about how are you investing in the space for sure. I guess, what I'm wondering about is like, is it too early today to start investing in that stuff?

[0:38:15.4] SH: It's a great question, that mismatches in my head all the time, because it is the best job on earth, because we meet all the startups that are trying to do everything from the latest Lidar-driven car. Decision-making too; I had an agricultural company in yesterday who was going to do weeding based on machine vision, all the way in.

I think it's pretty fascinating. I think as an investor, what we have to really look at is will the solution be at least 10 times better than the alternative? Because there is a pain for doing all the stuff. It's new development and again, depending on the industry the cycles of adding new things takes a long – that it just has to be such a big difference to make it worthwhile.

I think, again as an investor, also you need to look at where you can put things where real spending is happening. There are a lot of existing IoT businesses that have been around for 10s and 20s of years right without being called edge devices, or IoT, and whether that's Telematics for fleets. To some extent, these phones are all IoT and we've had to do all sorts of paying to manage them and keep to keep them secure.

I would just say, I personally would look at where is the impact big enough and where is there a lot of spending that you can actually get here. That's where I would focus the startups. We see certainly a lot of, we can change the world presentations. Those are awesome. If you don't have the steps to get there, it's probably not a great investment, and it's really what we spend our time with.

[0:39:36.2] JM: Are there specific areas where you think the timing is right today? Are you talking about IoT devices in a shipyard or something? Are those companies willing to purchase that stuff? I think that's where you want to get a foothold today is you could see that this is going to happen at some point in the future. If you put 10 million dollars into a company today and they don't have – they can't build any kind of business where they can start to get a foothold, start to experiment, start to build some technology and get a little flywheel going, they're never going to stick around for long enough to get to the place where we have connected cars and drones and stuff.

[0:40:11.4] SH: I think that's a really important question, I'll answer two ways. One is there is the famous word platform is something you hear. I think a lot of startups have been trained to say, "We're going to become the platform for this." I don't believe there is a platform for IoT, or for the edge. I think it is really going to be a number of different platforms and a number of different solutions, because the needs are very different based on the domain.

I would say a couple things, at least personally that I don't get excited by is when someone says we're going to build this platform and they will come and use it. I think you build a platform by having a killer use case for a specific domain. Then from there, you generalize it to use elsewhere. I'd say point one is start with the market in how you're going to fix that as opposed to a platform that could be used anywhere. That would be step one.

I do think, you do have to have a vision for, "Okay, maybe this is my insertion point," as we call it; the first market we go after, and maybe it's not that big. I would look for a startup to say, "Okay, I'm only going to start with whatever. It could be a shipyard sensor that actually has thermal controls or something."

I would look for why this is the best starting point. It's a market that's hungry for it. I have another proprietary relationship with someone who buys. Then I can leverage my learnings as you said from there to a different market. You like to see it more generally applicable than some very specific niche, but you also like to see here is some reason I think I can be successful in a shorter amount of time.

That comes down to how big is the need, who do they know, what is the development cycle of whatever device that team is pitching. Those are kind of things that we'll spend the most time in that case.

[SPONSOR MESSAGE]

[0:41:50.6] JM: LiveRamp is one of the fastest-growing companies in data connectivity in the Bay area. They're looking for senior level talent to join their team. LiveRamp helps the world's largest brands activate their data to improve customer interactions on any channel or device.

The infrastructure is at a tremendous scale. A 500 billion node identity graph generated from over a 1,000 data sources, running a 85 petabyte Hadoop cluster and application servers that process over 20 billion HTTP requests per day. The LiveRamp team thrives on mind-bending technical challenges. LiveRamp members value entrepreneurship, humility and constant personal growth.

If this sounds like a fit for you, check out softwareengineeringdaily.com/liveramp. That's softwareengineeringdaily.com/liveramp. Thanks to LiveRamp for being a sponsor of Software Engineering Daily.

[INTERVIEW CONTINUED]

[0:42:58.8] JM: Yeah. I have this friend, who he did sales for like a number of different IoT companies over the last decade. He writes a lot on Quora just about his terrible experiences in doing IoT too early.

[0:43:14.5] SH: Again, it depends where you apply it. We see plenty of people who have developed platforms that are like, "Hey, we want everything to be connected." You end up getting a couple of problems all the time.

You might be going to an appliance company and trying to retrofit these whatever, refrigerator or a hot tub with connectivity. You hit a couple of problems real quickly. At least a big chunk of industries that makes sense to be connected, don't have software developers at the level that you might need to push these forward.

The second thing you see all the time is that the cycle for creating the next appliance is five years. It is something that takes a long time. You got to really think beyond the obvious idea of, "Hey, this thing should be connected to like who's going to do the work? What does it do with their cost model? How quick is their development cycle?"

Then ultimately, let's say I make the killer solution for connecting dishwashers. How many connected dishwashers should there be in the world? That's when reality hits a really cool thing you'd like to see happen.

[0:44:11.3] JM: Yeah. Are you starting to get any convincing picture of what this is going to look like? Are we going to start to see more storage devices at the edge? Yeah, I don't know. What kinds of hardware developments do you think we'll see?

[0:44:29.0] SH: Yeah, it's fun for me. My first job before VMware was actually in a semiconductor company building low-powered ships. I do think one thing that's planning out now that we haven't seen for quite a while is a reemergence of semiconductor creation and interesting new chips you see. Plenty of pitches of people who are doing some aspect of the machine learning training stages all the way through deployment.

I do think we'll end up with some, either from the big players or other players a new set of chips that are commonly used for some of the inferences and things you need to do on the edge. Whether a startup does that, or whether Nvidia continues to go forward, or whether Intel gets even more involved. We'll see how that plays out.

I do think there will be a calculation on the endpoint for folks that they want to do something with the data, but can't afford to ship it all the way back. There will some emergence of chip there. The data story is, again I think you labeled the data problem exactly as we're talking about before.

There is the amount of volume that's coming to this edge device, how many sensors, how many pieces of data. Whether you want this data to be persistent or not, a lot of times it's useful data just for a minute as you bring it together or summarize it, and then you can throw it away. Then lastly, I think is how secret is the data.

You don't want to be sending over clear wire ton of data that matters. You want to probably have it encrypted on location with your device, so that's truly something that needs to be persistent and that needs to be sensitive. I think data is just going to have all these labels, and I think there will be a deployment pattern that makes sense for each of them.

Certainly, you are going to see more and more edge devices and you'll see little systems on a chip that have some storage and they have some processing capability and then they have – maybe they have 5G and Bluetooth and Wi-Fi connectivity. That will be something that's put into a big chunk of devices, so you'll see the common stuff and then you'll see the real, more esoteric thing for a specific use case.

[0:46:17.9] JM: Do you know much about mesh networking?

[0:46:20.5] SH: I certainly have it running in my house, sort of where you can do hop to hop network, so that you can get better coverage through them. We are seeing that as a emerging category for home usage, as well as for how do you get these devices out in the wild to get closer and closer to that.

[0:46:36.4] JM: Do you have any idea how good, or how reliable that is? If I throw a bunch of devices across my 30-square foot cornfield – I'm sorry, 30-square mile cornfield and the only way they have to get data – let's say, we've got a centralized server in the farmhouse. Can those devices scatter throughout the cornfield, communicate data between each other and get the data to the farmhouse?

[0:46:58.5] SH: Yeah. I mean, there is basic physics involved in all the stuff. There are a bunch of ways to get reach from a network ranging from project balloon, where there is like a hot air balloon collecting stuff, to just including 3G or 4G network connectivity, it's increasingly cheap just to connect a device with its own sim card effectively to using – certainly to using mesh networks, which we see in a number of cases as well.

Mesh networks just as you would expect, the further you are, the more hops there are, the less quality the network connectivity is. For a low-data sharing environment, I think it’s fine. If you need something as maybe much more real-time, low-latency and connectivity you’ll see the further out that it gets harder and harder to satisfy that.

We see a lot of a – this is a little off topic, but on this notion of internet of things that everyone likes to think about the consumer use cases and they want to think about everything you do in your house from like, we see a lot of dog connectivity solutions, like put your dog on the internet and be able to trace them. You see people trying to track water usage in their yard.

You hit a wall really quickly, which is I don’t have Wi-Fi coverage across my entire yard, or wherever these things are. Then that quickly leads into these other need for solutions like this. I’d say there have been some good early mesh networks that have general applicability, but it remains a – I think there can be a lot more done on that front still.

[0:48:21.4] JM: What about security in the IoT space? I think one problem is you deploy a bunch of devices and then some vulnerability is discovered and then you realize, “I got all these devices and I don’t know how to update all of them with security patches, for example.”

[0:48:37.8] SH: Have you ever tried to patch your – like you probably have at least a home modem or router?

[0:48:42.8] JM: Yes. Oh, yeah.

[0:48:44.2] SH: It usually installs an old version of Java on Windows XP. An hour later, maybe we get you patched to some firmware, a thing you download from a webpage. It’s the ability to patch these things even if you’re talented is nasty, because they haven’t built that in as a first-class citizen. I think for me –

[0:49:02.3] JM: It sounds like you’ve tried to do that.

[0:49:03.9] SH: Yeah. I mean, I've certainly been doing computer stuff forever. It's terrible. I mean, anyone can go try and do that. They're getting better. I think for me, one of the big watershed moments was one of the – I forgot which one it was, but one of the big recent denial of service attacks was someone who had basically infiltrated a lot of the routers and other –

[0:49:22.5] JM: Default. Default passwords.

[0:49:23.9] SH: That's right. Default passwords are certainly one. Or just hits its firmware that's been – maybe it has open SSL in there and that's been compromised like four years ago, and yet you've never patched this thing.

I think the security challenges in these emerging internet of things world is harder than ever, partly due to out-of-date software that's hard to update, partly due to just a less controlled environment where it's out in the wild. You can imagine hacking the stuff in ways that haven't been thought about so far. I'm not a ultra-paranoid person, but I just think that is going to be an area where some pretty big solutions need to come about.

[0:49:59.1] JM: Do you have any ideas, like what's the dream solution that you're waiting to walk through the door that you can invest in for IoT security?

[0:50:07.0] SH: Well, it's across the board, because I do think security is a very broad topic and there are a lot of different things you have to solve. I do think the notion that – I mean, I think like your Apple phone, or your Google phone, the way they do updates now is pretty much best of breed. It can happen automatically overnight. It tends to not break stuff. You have to have that working everywhere. The notion that you need to even stop and manually do something to update is just ridiculous. That will be one thing that I actually think that might be a place where the government needs to put some regulations in place and say if you're –

[0:50:38.5] JM: That's a Bruce [inaudible 0:50:38.7].

[0:50:40.4] SH: Good. I agree with Bruce. I do think it's going to cause enough problems, where it actually is going to mandate that companies are maybe stagy on how they do software, just have to meet some level of compliance here. It is a real risk. Then I think there's so many other areas of security where like all the data coming from these devices, however it's getting to the data center, it needs to be encrypted and there needs to be different forms of authentication involved. There will be a lot of solutions for that that come through. I don't think it's going to be one elixir that fixes all of that.

[0:51:10.9] JM: Yeah. Just like the basics.

[0:51:12.3] SH: Just the basics. I think folks like Amazon are doing a really good job of preaching their end-to-end edge device into their cloud. Now they have a biased approach, which is everything should end up in Amazon for the most part. I think there are going to be some really big vendors and really big software development shops that are able to think across the whole continuum of the edge all the way into the cloud.

[0:51:33.7] JM: You got into investing in a lot of different security companies. You sit on the board of several companies that focus on different aspects of security. Was that accidental? Did you stumble into that, or did you always had an interest in computer security?

[0:51:48.1] SH: No, it's pretty intentional. When I left VMware about five years and sort of put together if I were an investor what areas do I think are going to be the growing areas that have the most opportunity for big changes? Cyber security was absolutely one of them.

It gets back to your original question too. For me, most of the time is thinking about what is this brand new world that we want to get to and what does it look like, but how do we get there? The number one challenge for most companies in moving to a public cloud offering is security. Right or wrong, they're worried about who's going to look at my data? Can I get audited? How will I pass these different certifications, etc.?

A lot of what I've been looking at is if we assume a world in the future where half of the companies' applications are running in a public cloud and half of them might be still running in their own data centers. What is the security solution need to look like? What are the changes as containers come in and make the density so much higher? Do you have to have protection between containers on a single machine?

You just go through all these different areas and you look at accommodation of where the attacks are coming from, coupled with what can we now do to protect them. That's where a lot of these investments come from.

[0:52:59.7] JM: What is difficult about building security software, or building the security software company, sales cycle and –

[0:53:07.5] SH: Here is the number one problem. If anyone goes to – like RSA is one of the big conferences for security. There are 700 startups that you can see on the pavilion floor. Each of them are preaching some way to do something better, to protect you from something that might be causing problems right now.

I actually think the largest problem – certainly there are technical issues you need to solve and to do things well and at speed. The first problem is how can you be so much better that you can get above the noise and you become a priority for a security department? In that sense, it's sort of marketing, but it's also picking the right space that is where the puck is going.

Quick example, it used to be that we all have virus checkers on our machines and we are most worried about downloading some rogue .exe file or something. Most of the attacks these days are about 80% are coming through e-mail, through attachments and through web browsing. You need to move the Windows-based solutions and the things that are protecting the browser, or that are really built into the mail sub-system, as an example.

I think, you look at where things are headed and you make sure that you're in the right spot. I think that's one. I think the other top level challenge that I spend a lot of time on for cyber security is if you look for the most in demand job position right now, job role, it's often security. It is completely a lack of professionals that can go to all the positions that are open today.

Kids out there if you want to assure a job, go in to cyber security. What that means for a startup, is that you cannot create extra work. If you're going into a company, you can't require even more time from the overworked security staff. You need to not cry wolf, which all these things tend to do as they worry about stuff, or they raised your attention on something that might be a problem.

If you've now taken security and already work overworked security team's attention away to look at this false claim, they're not going to like you. You've actually probably harmed things overall. I think the other thing you look at is in a world where there is fewer professionals, how do I fit in to their current workflow? How do I only raise attention to things that are true problems? Then how do I tell people that I actually protected them?

All the world is now recurring businesses of SAS software. If they don't know – you want not just protect people, but you want them to know that you actually caught some stuff, because you want them to keep using your service. These are the things you really need to think about when you're creating as security business these days.

[0:55:27.8] JM: There's so many different security companies out there. There is so many different providers that solve different problems. I think a large enterprise, like a big insurance company is probably going to be buying from a lot of different vendors, like multiple different vendors. Then one of those vendors, once they get into an enterprise, they might try to upsell, like they might have a – they might get in a foothold and then they try to sell other products to that same enterprise.

Then you have different vendors that are selling to the same company and then you might have competing vendors that are really trying to sell the same thing and they're overlapping. Just one thing I'm curious about, is this is niche, but what are the dynamics of the relationships between a large enterprise and the multitude of security vendors that it might have? Are they bidding with each other? Is there some kind of – what does that relationship –

[0:56:23.8] SH: That's actually a really good question. I think security is weird in the software space. There has not been that many, what we call roll-ups. There is not many companies that provide this end-to-end suite of everything you could want. Certainly, semantic and folks like that try to aim that direction. It's a business that has more I'd say silo products and most of the industries that we're in right now.

You do end up – there are different parts of the organization that are buying different things. Your e-mail team might be buying an e-mail protection solution. Your core networking team is buying for the future of firewalls. I would say it's not as tightly rolled up to the chief information security officer as you might think. There is a lot of lower-level people there empowered to protect their unique environment the most they can.

Zero question that as a startup, you are often competing against maybe a bigger company that says, "We have something like that," and just buy from one vendor and you'll be fine. That's a very common move that they'll make. It's up to the security startup to show why their solution is not just better than this one, but 10X better to warrant having another vendor and another user interface, another training module.

[0:57:30.8] JM: You're at General Catalyst. Do you focus on specific stage, or is it stage agnostic?

[0:57:37.3] SH: Now we tend to be an earlier stage firm. We're a bit unique in that. We're multi-national, we're across the nations. We have a New York, Boston and Silicon Valley offices. Then we do primarily series A investments, but we've done a little bit later as we see really great companies. We've gotten involved with a lot of good ones out here from Airbnb to Stripe to a lot of the emerging ones as well.

Being across multiple stages is really fun, because you're solving different problems at different times in the company. Series A, or even seed investment, it's like how do you hire your first five people and how you build even a alpha prototype of some piece of software. Whereas, where you get to series C and beyond, it's how do I go international? How do I build up a full sales team? How do I do strategic partnerships for reselling? I think being multi-stage really lets you exercise different parts of the business cycle. I think having seen early and late, you can help the other one interestingly.

[0:58:32.5] JM: Are there any canonical problems that you're seeing companies encounter today that maybe they didn't encounter five or 10 years ago?
[0:58:42.0] SH: Everyone has moved, I would say 90% of companies have moved to at least having a cloud option for deploying things. Most of them have moved to a recurring revenue model for software, something people going at these days are used to. That's a big switch from

where things used to be, where it was something that you would install on premises and it was a license model and that sort of thing.

There has been business model evolution that affects everybody. How do I do this? To the point on security, I think recurring revenue models of subscription are fantastic. It's really good for the customer, because you have to prove your value every single month, or else they can literally turn off the switch and stop paying.

I think it's forced companies to really think about what is the direct value and how do I keep tight with a customer. Every startup has a customer success organization now, which is around business software actually deployed and is it being useful. That's a real mindset change from the shelf days of old, where you sold the software and then you really didn't care after that.

[0:59:41.2] JM: Yeah, because there's a little more culpability there for quality. Okay, so to close off, do you have any thoughts on where cloud computing is going that I might not hear anywhere else? Like some crazy thoughts that have just crossed your mind, that you're afraid to tell people, that how things are going to change in the next 10 or 20 years.

[1:00:04.2] SH: I think it's going to be great. I think there's a top-level understanding that certainly there is these really strong cloud offerings. I think people underestimate just how big on premises and data centers are right now and will continue to be for some class of applications.

I'm fairly certain that especially for larger organizations, we're going to end up with some sort of homeostasis around what types of applications and what percentages are running different places. I do think with a lot of the toolings that we are putting in place, the ability to choose that location will get easier, there will be less friction between moving things to a public cloud offering or moving them back, or moving them between them. There is just a lot of people investing and making that better.

That ultimately, if you do have more choice then it's going to be certainly gives you some nice cost advantages if someone gets better. I think it will ultimately allow you to adapt to something unique about your application. It's certainly the case where some applications can be run in your own data center better.

I just love this idea of a world where you can more seamlessly choose where something runs based on the unique business needs that you have in any time. Again, I think the more regulated and the older an industry is, then the less stuff that will be out in the crazy cloud world and vice versa. The newer company is with less sensitive data. It would be a 100% in the cloud. I love the idea of just watching all these balances play out across industries and company sizes.

[1:01:24.1] JM: Okay. Actually last question, if you're like me, you're trying to figure out how to learn about cryptocurrencies, or to what degree you should be learning about them. Do you have a strategy for learning about them, or learning about their implications at least? Or is there somebody at the firm that you hired to specialize in doing that?

[1:01:45.7] SH: There is two things. There is cryptocurrency and then there is blockchain, I've seen you cover before too. I think I personally am not super interested in cryptocurrency, although we have several people here trying to make sure we're well on top of what the implications are.

Blockchain itself I think is pretty fascinating as a way to distribute technology and really apply to these distributed systems worlds in a way that is pretty interesting. I'd like to think about all the stuff we have in IT today that is highly centralized and could it benefit from being decentralized.

To your question though, what we do all the time is try to be around the smartest people and meet as many companies and people as we can, and hopefully contribute to their own strategy and bring them together in dinners and communities. We do a lot of dinners, where we just bring together the smartest people that are hopefully helpful to each other in honing in their own ideas. That's a great way to learn.

[1:02:35.1] JM: Cool. All right. Well, Steve thanks for coming on Software Engineering Daily. It's been great talking.

[1:02:38.3] SH: Thank you.

[END OF INTERVIEW]

[1:02:41.3] JM: If you are building a product for software engineers, or you are hiring software engineers, Software Engineering Daily is accepting sponsorships for 2018. Send me an e-mail jeff@softwareengineeringdaily.com if you're interested.

With 23,000 people listening Monday through Friday and the content being fairly selective for a technical listener, Software Engineering Daily is a great way to reach top engineers. I know that the listeners of Software Engineering Daily are great engineers, because I talk to them all the time. I hear from CTOs, CEOs, Directors of engineering who listen to the show regularly. I also hear about many newer, hungry software engineers who are looking to level up quickly and prove themselves.

To find out more about sponsoring the show, you can send me an e-mail or tell your marketing director to send me an e-mail jeff@softwareengineeringdaily.com. If you're a listener to the show, thank you so much for supporting it through your audienceship. That is quite enough, but if you're interested in taking your support of the show to the next level, then look at sponsoring the show through your company.

Send me an e-mail at jeff@softwareengineeringdaily.com. Thank you.

[END]