EPISODE 1023

[INTRODUCTION]

[00:00:00] JM: Cloud computing caused a fundamental economic shift in how software is built. Before the cloud, businesses needed to buy physical servers in order to operate. There was an upfront cost that often amounted to tens of thousands of dollars required to pay for these servers. Cloud computing changed the upfront capital expense into an ongoing operational expense with businesses increasingly shifting to Amazon Web Services, Microsoft Azure and Google Compute Platform.

Although the initial motivation for moving on to cloud providers might have been decreased cost, overtime, the cloud providers have developed unique services that makes software even easier to build than before. So there's a decrease in both cost and the barrier to entry to building great software because of these new abstractions. There's been a proliferation of new software infrastructure companies that have been built on top of these cloud providers as well. There're new databases, there're new blogging companies, there're new platform as a service products.

Danel Dayan is a venture investor with Battery Ventures and he's the coauthor of the *State of the Open Cloud 2019,* which is a report that compiles a wide set of statistics and information on how cloud computing and open source are impacting the software industry, and Danel joins the show to talk about his work as an investor as well as his previous career at Google where he worked on mergers and acquisitions. If you want to reach Danel, he is an investor and he's looking for investments. You can find his contact information in the show notes. It's ddayan@battery.com, and he also is on Twitter @DanelDayan.

[SPONSOR MESSAGE]

[00:01:49] JM: I remember the days when I went to an office. Every day, so much of my time was spent in commute. Once I was at the office, I had to spend time going to meeting rooms and walking to lunch and there were so many ways in which office work takes away your ability to be productive. That's why remote work is awesome. Remote work is more productive. It allows you to work anywhere. It allows you to be with your cats. I'm looking at my cats right now.

But there's a reason why people still work fulltime in offices. Remote work can be isolating. That's why remote workers join an organization like X-Team.

X-Team is a community for developers. When you join X-Team, you join a community that will support you while allowing you to remain independent, and X-Team will help you find work that you love for some of the top companies in the world. X-Team is trusted by companies like Twitter, Coinbase and Riot Games.

Go to x-team.com/sedaily to find out about X-Team and apply to join the company. If you use that link, X-Team that you came from listening to Software Engineering Daily, and that would mean that you listen to a podcast about software engineering in your spare time, which is a great sign, or maybe you're in office listening to Software Engineering Daily. If that's the case, maybe you should check out x-team.com/sedaily and apply to work remotely for X-Team.

At X-Team, you can work from anywhere and experience a futuristic culture. Actually, I don't even know if I should be saying you work for X-Team. It might be more like you work with X-Team, because you become part of the community rather than working for X-Team, and you work for different companies. You work for Twitter, or Coinbase, or some other top company that has an interesting engineering stack, except that you work remotely.

X-Team is a great option for someone who wants to work anywhere with top companies maintaining your independence, not tying yourself to an extremely long work engagement, which is the norm with these in-person companies, and you can check it out by going to x-team.com/sedaily.

Thanks to X-Team for being a sponsor of Software Engineering Daily.

[INTERVIEW]

[00:04:26] JM: Danel Dayan, welcome to Software Engineering Daily.

[00:04:28] DD: Thank you for having me. Pleasure to be here.

[00:04:30] JM: You’re an M&A in Google. This is sometimes known as business development. How did that prepare you for investing?

[00:04:38] DD: Yeah. A caveat there, businesses development is a little bit different from M&A and investments especially on the corp dev angle. I think at a smaller company, those tend to blend together, but businesses development is really around the commercial aspects of a deal. Very much around the business engagements and the relationships between the two parties there.

On the M&A and investment side, it's very much what the strategic angle is there and what the strategic relationship is, right? The example that most people give is how can you take one plus one to equal three? From Google's perspective, the thing that I think it set me up for success with was, one, getting exposure to an ecosystem of very, very smart, technical people within the Google ecosystem and then access to founders, investors and the startup ecosystem that I think is hard to rival.

The volume and velocity of deals that Google did, I think there're not many companies out there that can match that. Over the three years I was there, I worked on 25 acquisitions and investments from a range of different sizes as small as 2 million to as big as a billion, right? I think you just get such a broad exposure and you get a lot of reps working through these processes. It's a good foundation to understand what to look for, for businesses from a strategic angle, what will big companies like Google look for from a tech perspective, from a team perspective, from a strategic perspective, and then obviously all the business diligence that goes into it, right? Understanding who the competitors are? What the market dynamics look like? The go-to-market channels. How are they acquiring customers what their customers? What their customers say about them?

[00:06:31] JM: There’s a widely held belief that most technology acquisitions don't work. This is a phrase I've heard many times. Is that an outdated belief or is it still true?

[00:06:42] DD: There's some truth to it, but I think the way to look at it is especially for these large companies, the risk of doing a lot of acquisitions is quite low especially if depending on where they're done, right? There is a lot of scrutiny that goes into doing the multibillion-dollar deals and a lot of people involved in that decision-making process. But take Microsoft's acquisition of GitHub, for example, it was 7.2 billion or 7.5 billion. That was may be less than 1% or 1% of the market cap at that time. It’s a very small percentage of the overall business in the grand scheme of things.

Yes, on absolute dollars basis, it's large and risky, but in the overall grand scheme of things it's quite small. If that thing is successful, it can be magnitudes more impactful than 1% of the market cap. Look at Instagram, for example, when Facebook bought them versus where they are today. I think that's the type of strategic outcomes that M&A drives and that it is almost binary in some ways and there are a few examples where taking those shots have resulted in massive outcomes.

Booking.com and Expedia is another example there. They acquired them for mid-hundred million dollar valuation, and Expedia has now become bookings.com completely. They officially changed their name to that. I think M&A kind of runs in a spectrum. The outcomes I'd say are in terms binary, but it also depends on what the strategic rationale for that acquisition is. If it's a short-term and a roadmap acquisition, you're trying to fill something quickly here. It could be a team, it could be a product or a feature within an existing product. Something to augment something that's already out in market versus maybe some of these bigger acquisitions, which are new business units, new or adjacent areas that they want to get into, and they might have more risk there, but the potential upside is a lot greater.

[00:08:50] JM: When you joined Google, I imagine it feels like – How Google looks from the outside. It's this unimaginably powerful, unimaginably omnipotent entity that you are now being absorbed into, you’re being absorbed into the Borg and becoming a part of it. It feels like this invincible, impossible to understand entity. But then probably over the course of three years, some of the luster wears off. You start to see the rusty years and the messy duct tape and chicken wire that might be holding together some parts of the company. Did it feel like a more vulnerable company after three years of understanding how it works?

[00:09:35] DD: The short answer is I think it did, and the reason is really around speed, speed of execution, speed of higher, speed of just getting simple things done, and I think that was something that they thought about all the time and that's why M&A was used at such a high-scale and velocity to plug some of those holes quickly. Because acquiring a company or startup that has something in market, you can continue selling that as you co-develop it or migrate it on to a Google service.

There were a lot of instances where it felt like Google was behind or falling behind just because they couldn't execute fast enough and there's a big machine and many layers of approvals and checklists and different organizations that you had to go through in order to get things done. But I think their size also allowed them to focus on very interesting moonshot projects. Say, healthcare I think is a great example of that where there's not a lot out there of what they're doing, but if something comes out of it, it could be extremely interesting just given on their expertise in machine learning and AI that they have, the amount of data that they can collect and the things that they can do with it. That in itself can be a whole another company.

I think the other thing that surprised me being within Google is that each of these business units, regardless how small they looked on the outside, are massive in scale. Something as small as Google domains, for example, can be a multi-hundred million dollar business, and from the outside it looks like there is no effort or resources being put behind it.

The bar is very high and the pressure is always there to continue to execute quickly, and I think just by being a big company that slows down. That's I think where a lot of the kind of tape was around and how we as an M&A team look to fill those holes.

[00:11:36] JM: Did you feel like you got a sense of where things are going? Like where the future is headed in a way that you don't have as much of a handle on out? I feel like in some sense, like as a venture capitalist, yeah, you go out and you examine these companies that are kind of building the future, but to some extent, there so much momentum just in what Google is doing or what Amazon is doing or what Facebook is doing that you really actually cannot get a glimpse into the future in the way that you can on the inside at these companies if you're an outsider. Do you think that's true?

[00:12:15] DD: There's some truth to it, but I think there's a lot of different data points that you can collect by being on the outside that can help inform a decision or thesis around where the future is going that I don't think you necessarily get by being within a company, because you're a little siloed into the four walls of Google or Microsoft or an Amazon. The strategic priorities for Google might be very different from where the world is going on the outside and what normal enterprises – Maybe not normal, but what you kind of the general pool of enterprises might be needing at any given time.

I think where you can get a sense of interesting technologies and features is how they're developing things. Google being one of the first companies to kind of codify this microservice architecture. That was a challenge or a solution to a challenge of out of the scale at which they were running at, or Netflix, for example, and their kind of recommendation engine and the scale at which they have to continue to iterate and build things and how that bred into chaos engineering and some other interesting technologies that are in markets today.

I think you get interesting glimpses into big challenges and interesting challenges that these types of companies are trying to solve just because the scale is so large, the problems are so unique to them. But I think from an outside world, you get to collect a lot of those data points and stitch them together that you wouldn't necessarily get by being within the four walls of a big company.

[00:13:44] JM: The idea of Google infrastructure for everyone. This is a phrase that I started to hear getting used maybe five or six years ago, Giphy, and this manifests in some ways in how people use cloud providers. They manifests in some ways in open source tooling like Kubernetes or TensorFlow. I know you aren’t engineer inside of Google, but do you know of any technologies inside of Google that have not been productized for the wider market yet? To what extent do we have Google infrastructure for everyone today?

[00:14:22] DD: Yeah. I’d say I’m sure there are tools that they use internally that haven't been productized. I don't have a definitive answer as to what those are. I will say I think at the core, the goal of Google Cloud is to get storage into their system and to run as much computer as possible on that storage.

At the end of the day, that's kind of the end goal that they're solving for. If it means giving out Kubernetes for free so you can host things on GCP. The ultimate goal there is just to get more data, more storage around the Google Cloud ecosystem. I think the tools that they release, I think it's important to frame it into what they're actually trying to achieve with that. It’s to build the community around Google Cloud. It’s to get people comfortable using those types of solutions to standardize them on a way of developing that might be different from AWS or Azure. So you’re using more GCP. So you’re using more compute. So you’re using more BigQuery. I think that's kind of how I think about the tools that they release out to the public versus whether this is actually something super-secret or proprietary that they’re holding behind.

[00:15:39] JM: You have written this report called the *State of the Open Cloud 2019*, and the word open cloud, I don't think we would've seen this word be used perhaps without the rise of Kubernetes. Just to take people back to the container orchestration wars, I think what happened there was not only was there the container orchestration wars that were going, but the container orchestration wars happened against the backdrop of the fact that AWS was running away with the cloud market and everybody else was watching and saying, “Is there anything that can be done? Is there any point of change that can open up the market for other cloud providers to get in a meaningful way?”

It felt like there was some consolidation around the idea that with containerization, there is going to be a platform shift, and the platform shift could potentially unlock an opportunity for other players to get into the cloud market. If you are the best at running containers, perhaps you can start to funnel budget away from AWS and into your own cloud and you can start to build a money printing machine that is on par with Amazon, and Kubernetes happened to be the container orchestration framework that people consolidated around. Tell me what you mean by the open cloud.

[00:17:14] DD: Yeah. It's the symbiotic relationship between open source tools and cloud native infrastructure or developing on cloud native infrastructure, so like many of the cloud providers and the tools and solutions that they offer. It’s this understanding that the way software and applications are being built has changed. It's a combination of using open source components and cloud native infrastructure. So using one of the public cloud providers to stand up a handful of microservices, but also using Kubernetes to manage those or orchestrate them at scale. That's combination of both something that you're paying for on the compute side and an open source framework that you're leveraging essentially for free.

It's also I think taking a step back, this notion that developing software and applications is now more open and accessible than ever before. The one stat I love to talk about is how IDC Gartner. They still tout 25 million developers worldwide. But if you go to the most recent GitHub report, they claim 40 million developers on their platform. There’s a huge difference between what like these professional market actuaries are saying versus like what's happening on the ground. I think it ties really well to this notion that this cloud is much more – It’s bigger and much more open then I think we’ve even imagined. I think it plays into those two themes really, is one symbiotic relationship between developing things that are free or using open source components that are typically free as a building block to develop new applications and new software being hosted on a cloud provider and then access to actually developing these types of solutions.

[00:19:08] JM: You've been following cloud infrastructure for several years at this point. Something you had to do at Google and certainly something you’ve done as an investor today. But your work on this report, I assume you're talking to portfolio companies from Battery. You're probably talking to people leave known for a while. You're talking to other investors at Battery. Was there anything particularly novel or shocking or insightful that you came across when you were doing your research as you were looking into the “open cloud”?

[00:19:47] DD: Yeah. I think there're a handful of things to talk about there. One, understanding that there's two major shifts happening in this world of enterprise infrastructure. One, dollars are shifting from on-prem, to hybrid, to cloud. Two, this market is – Or the pie is expanding at the same time. This is kind of why I think you see the growth rates of these cloud providers actually accelerating over the last few years, is because there's just so much green field or momentum behind the usage of cloud native tools. I think that's one interesting observation. Two, of access, lower barriers to develop and the way software is built today. I think these three things are all also accelerating kind of the adaption. Microservices, the explosion of different applications or modular monoliths if you want to use the Shopify example, just increase the spread and sprawl of having to manage all these things.

I think what you're seeing is explosion of market opportunity and actually a lot of operational challenges and how to take advantage of that. I think that's what we tried to really capture here, is the market is really big. It's growing really fast and it’s moving at a pace that we never seen in the past. Companies are only now starting to realize that they have to make this transition and it's opening up a lot of opportunities for infrastructure vendors. But there're also a lot of operational challenges and how do you take advantage of that. How do you define who your customer is? When your user is not actually your buyer, how do you kind of balance that and who do you sell to? Who do you target?

Ultimately what that means is if you've done it right, you have these set of kind of metrics or goalposts that most investors look at. By executing our understanding the operational challenges, being aware of them, having a feedback loop that allows you to iterate and improve on them, you will ultimately have or be kind of in these ranges of what we think are the best of breed infrastructure or software companies. I think at the highest level, that was like a big observation for us and we can dig into any specifics that you want to go into there.

[SPONSOR MESSAGE]

[00:22:18] JM: If you are selling enterprise software, you want to be able to deliver that software to every kind of customer. Some enterprises are hosted on-prem. Some enterprises are on AWS. Some enterprises are on cloud providers that you've never heard of and every cloud provider works differently.

Gravity is a product for delivering software to any kind of potential environment or data center that your customers want to run applications in. Think of Gravity as something you can use to copy-paste entire production environments across clouds and data centers. Gravity is made by Gravitational, and Gravity works with on-premise data centers and on different cloud providers.

Gravity can get software to your biggest customers without the pain of developing individualized deployment systems for every single customer. Gravity puts a bubble of consistency around your application so that you can write it once and deploy it anywhere, and Gravity is open source so you can look into the code and understand how it works.

You can also listen to the episode I recorded with Gravitational CEO, Ev Kontsevoy. Gravity is built to solve the problem of software delivery. Gravity ensures compliance and lowers the cost of development. You don’t have to write your code to support every platform. It is as easy as copying and pasting your deployment each time.

Gravity is from Gravitational and it’s trusted by leading companies including MuleSoft and Anaconda. Go to gravitational.com/sedaily to try Gravity Enterprise free for 60 days. Gravity uses Kubernetes under the hood and the Gravitational team knows Kubernetes well. If you go to gravitational.com/sedaily, you can sign up for a free consulting session about cross-cloud Kubernetes security. This is in addition to the 60-day free Gravity enterprise trial.

If you feel like you need to get a better understanding of Kubernetes security, check out gravitational.com/sedaily for this offer of a 30-minute free Kubernetes consultation along with a 60-day free Gravity Enterprise trial.

Gravity is a system of securely delivering your applications into any environment, and you can try it free by going to gravitational.com/sedaily. Gravity Community Edition is also available on GitHub and it’s free to play with. If you are curious about how Kubernetes will change software deployments, I recommend checking out the Gravity repository, and thanks to Gravitational for being a sponsor of Software Engineering Daily.

[INTERVIEW CONTINUED]

[00:25:19] JM: First, talk about the interaction between AWS and the open source ecosystem. Kubernetes, it comes out of Google. Google kind of starts to use it as a way to incentivize people to come on to Google Cloud Infrastructure because they can say we know how to run Kubernetes better than anybody else. How has AWS responded to the rise of Kubernetes?

[00:25:45] DD: Yeah. AWS has their own hosted container engine now. They have a handful of managed services that they've started to offer around the container ecosystem. Ultimately, I think their end goal is still trying to drive compute and storage to AWS at the end of the day by any means necessary. So if that means having managed service for what have traditionally been open source tools, they'll do that. I think they take a stronger stance on playing friendly with the ecosystem than some of the other cloud providers do.

That said, I think their perspective is all around time to the value. If you can deploy something through an AWS console because it's already part of their ecosystem, it's hosted by AWS, then that's a better user experience than just having an open source project that you have to kind of stitch together with your existing infrastructure.

You can see where they're coming from. What that means for the software development ecosystem I think is very different. But ultimately it means business models are going to have to change open source. My view of it is it's a distribution model more so than a business model. I think we just have to kind of change the framing of how we think about open source relative to proprietary software, the cloud providers, and you have to kind of defend your space if you're trying to build a business around there. Now I know there'll be purists that say open source should for always be – For always be open. But if you're going to build a company, I think you have to be aware of how the cloud providers can offer that as a service themselves and protect yourself against it.

[00:27:37] JM: So in midst of these gigantic cloud providers, there have also been standalone software companies that have done very, very well providing infrastructure, a company like Datadog or I think Sumo Logic is in battery portfolio. You got Twilio, etc. What is required to build an infrastructure company that is not a cloud provider and how do you compete with the major cloud providers?

[00:28:08] DD: Yeah, that's a great question. It comes down to obviously a lot of different factors, but I think first and foremost, you got to go back to this idea that the cloud providers are really squarely focused on storage and compute. At the end of the day, they want as much data hosted on an AWS instance and they want to be able to monetize through their compute instances on that. There are many things that they will release that might be shallow or good enough, but I think, inherently, developers want best-of-breed tools, things that help them be more productive, be more efficient, help them automate certain tasks that might be redundant, and these are all opportunities for independent infrastructure companies to build into.

Then going back to that M&A point, speed I think is the one thing that, software, as you grow big, it's easier for them to acquire a strong team with deep domain experience and a solution in a market than it is to build something by staffing a bunch of engineering resources to it I think it's favorable for a company to understand some of these gaps are shallow offerings that the cloud providers have and going after them.

Now, how you operationalize that I think is very different. There're a lot of challenges around that especially if you take a community building approach where you might be building your open source community. How do you start to monetize that? How do you actually start to build a strong business on top of these cloud providers? There's a handful of ways to do that. One of our portfolio companies, Matillion, and has a strong relationship with AWS marketplace, for example, and that's been a great channel for them. Why? Because a lot of these companies that they sell into already have ELA or enterprise license agreements with AWS. So spinning up an ETL pipeline with Matillion is just one click on that AWS console. It's a procurement challenge that they've been able to overcome. That's been kind of their unique angle.

[00:30:23] JM: Like in the AWS marketplace, if I am selling my ETL tool like Matillion – We just had Matillion on the show pretty recently. So people can listen to that one if they want an example. What kind of terms does that turn out to? If I say, "Okay, look I would love to build a completely standalone software company, but ultimately I got to realize that most of my buyers are on AWS, and I'm just going to sell an AWS marketplace." What do you have to give to Amazon through that marketplace?

[00:30:55] DD: Yeah. I think one of the biggest challenges by selling through a marketplace is visibility into the customer who you're selling to. I think ultimately at the end of the day the cloud providers own that relationship. But what I think more interestingly that is happening or shifting in the market is what software buyers are getting accustomed to, right? AWS, GCP, Azure they're all on a consumption-based kind of pricing model. So we've recently seen the shift of companies embrace that as a way to align themselves with the ways developers are using AWS or some of these cloud providers today. Everything being on consumption-based pricing.

I guess like you can think about how to compete with them but also stay friends with them is kind of align yourselves to how they are selling in the market as an easier way to onboard customers that are already using their services. Ultimately, if they get big enough, you can start to own that relationship yourself. But on the onset, going through the marketplace, that's I think the biggest challenge that kind of surfaces.

[00:32:01] JM: What do you see is the competitive breakdown between the three cloud providers, the three major cloud providers; Google, AWS and Microsoft?

[00:32:11] DD: Yeah. I think Amazon has done the best job. One just being the first to market and owning that market. I think, still today, it's kind of the de facto solution when you're building from 0 to 1. They've built a $35 billion business growing at 35% annually off that. Really focused on storage and compute.

Google's positioning in my view is really around kind of these AI machine learning services and how can you funnel a lot of data into their system and run some interesting experiments on it. That's through their cloud AI platform. That's through BigQuery. Now within data visualization access with Looker. So for them, it's surfacing interesting insights. It's using that backbone of machine learning and intelligence to build insights on your datasets, and that's an interesting wedge for them to say, "Hey, actually bring more data into GCP."

Then I think Microsoft has done a good job obviously on the enterprise side, leveraging a lot of their existing channels that they've had. But building applications on top of it, right? Their RPA tool that they've kind of announced, right? That's a big deal in the kind of process automation space because they own a lot of the compute resources behind that already and you can start bundling those together. They have all of their productivity suites, suite tools that they have sitting on top of Azure. I think for them, it's more like a portfolio approach and existing channels that they've been able to leverage. But that's kind of how I'd see the three break down.

[00:33:58] JM: I keep hearing this meme, Google can't sell, like the Google Cloud people don't know how to sell. I don't believe it. First of all, I know lots of developers who use Google cloud and absolutely love it, I see it is tremendously differentiated from AWS and Microsoft. But then again, I don't know much of the sales process, the enterprise sales process, especially when it comes to a cloud provider. I don't know what kind negotiations I'm making. Am I saying like, "Hey, give me a discount on my DynamoDB. I've been a good customer. I want 20% off. I don't even know with these conversations look like. Do you have any insights into the sales process of a cloud provider and how the sales teams of these different cloud providers match?

I mean, obviously Microsoft has the longest expertise in this, but like does that even matter? I guess you could say AWS has the longest expertise because they've been in the cloud business for the longest. What is the importance of an enterprise sales team at cloud provider and do you any insights into how these different cloud providers square up?

[00:35:05] DD: Yeah. Maybe I'd be able to touch on the importance. I think just as the sales is equally important in any startup or any business, I think it's the same for any of the cloud providers. It's got a make or break their business at the end of the day and it has to be done in an efficient, scalable way.

There is definitely an enterprise sales DNA that Microsoft has that AWS built that Google is still catching up to. There is an enterprise sales DNA that Microsoft already has AWS has built and Google is still catching up to. I think this is informed by the solutions and services that Google has historically provided, very much consumer-facing, freemium and less really about building these deep relationships with companies offering variable pricing depending on relationship, strategic importance and so forth. So I think they have struggled there. But I think the way cloud providers are selling today is selling compute resources in these kind of various tranches. By X amount of resources and then if you don't use all of it, will refund you the excess, or by even more of it and you get access to some of our other services for free.

I think the sales strategies are still changing. I think the one thing that is here to stay is kind of this pay-as-you-go consumption-based pricing, and then there is obviously bunch of different levers you can pull there if you want to buy in bulk if you want to commit to certain usage for a period of time, and that I think is all helping them figure out how much they can start to charge for some of these additional add-on services.

But I think the sentiment is right, is that Google has lagged, I think, behind on the enterprise sales motion. I think they've started to build that up over the years and there are still a lot of opportunity and greenfield that I think allows them to be growing from 5 billion to – Was it 9 billion they just reported in cloud revenues? Part of it is still just learning and growing into to their size.

[00:37:24] JM: Coming to a specific vertical that of security. My sense of how cloud has affected security, it's multifaceted. So on the one hand, you have companies who have built such a sprawling cloud infrastructure that there's a total heterogeneity to their infrastructure. Maybe they've got on-prem, they've got cloud resources. It's not heterogeneously managed, and so securing everything is really hard.

On the other hand, because of the rise of cloud computing, you have a ton of new security companies getting started because the startup cost is so cheap. You have sprawl, but you also have arguably better security. He also arguably have better isolation because of Kubernetes, for example. Tell me about security companies, modern security companies. When you're looking enterprise investing opportunities for security companies, what are the areas you're looking at and what's changing?

[00:38:27] DD: Yeah. I think the one key observation there is we see security lagging a lot of these kind of waves of tech innovation, right? It was only after cloud adaption really started to pick up, and you saw a lot of these S3 buckets with open ports that like investment and explosion of these cloud security companies started to come to market, right? It takes these security breaches and events for the market to realize that there is this new vector or new aspect that can be exploited. I think that's driven a lot of interesting themes in the market today, but that's kind of one big observation that we've seen.

We talk a little bit about in our report where you can see the M&A and investment activity really pick up after these certain vectors get exposed. It was hosted, or antivirus type malware that kind of permeated the computing ecosystem. Then all of a sudden you had the AVGs, RSAs, Symantec type companies of the world come up. By the time they were "mature", it was this idea of network perimeter or the potential compromise of your network perimeter, and then you had Fortnets, the Safenets, Impervas of the world start to kind of grow into their scale. After that, it's the endpoint and sandboxing type companies of the world, and so forth and so forth. Now we're in this cloud security space and it's only now that we see Splunk as this massive security company, and that took a long time for them to get there, but it was from years and years of kind of managing logs and understanding like where these vulnerabilities are in disparate systems.

Container security, for example, I think is still nascent, but the explosion of microservices today makes it such an interesting opportunity or area to think about, and why hasn't the market kind of matured yet in that security space? That I think is maybe my one big observation as it relates to security and waves of innovation.

Sorry. What was the other question that you –

[00:40:50] JM: No. I was kind of using this is as case study of how cloud impacted a certain vertical. But I think you answered it pretty well.

[00:41:01] DD: I can add to that.

[00:41:01] JM: Yeah, sure.

[00:41:02] DD: I think what you've seen in a lot of other adjacent areas to security is provisioning, infrastructure management, is it's all being shifted into the coding development environment. How do you provision things? How do you manage resources in a way that's as dynamic as you code, right?

Today – Or I guess let's take a step back. A few years ago, VM's would last months, years. Now you have microservices spinning up daily, weekly. So manually provisioning things doesn't scale anymore, and so now you see in this kind of wave of infrastructure as code, right? How can you put kind of rules and guardrails around how you're provisioning infrastructure right into your kind of development environment without actually kind of touching the environment itself?

I think we're starting to see some of that permeate on the security side as well. Whether it's, one, our zero infrastructure kind of provisioned, deployed the way it should be, and are there any ways that can improve or are there any like vectors that might be exposed that we should be aware of? You can kind of start to run rules through kind of as like a policy engine, global policy engine across your infrastructure. I think that's one thing we're seeing.

Then, obviously, this notion of shift left where security is becoming a bigger component of the application development lifecycle, testing and development through production, but it's happening earlier on versus kind of once product is kind of out of protection.

[00:42:51] JM: Yeah. You put security in like the continuous integration process. Do static analysis there. Vulnerability scanning, whatever, or scan your GitHub repo.

Now, on the buyer side, one notable trend for investors is that these large enterprises like banks and oil companies and insurance companies, they are doing “a digital transformation”, which I don't know if that is really anything new. I don't know if they've been digitally transforming like since I was born. But, certainly, when I talk to some of these companies, it sounds like the cloud has caused these companies to open up their wallets in a way that has not been opened up before, maybe because maybe the buyer profile is changing, like you have younger CTO's, or younger CSO's, or people are just getting used to the cloud, or perhaps AWS just offers more services than was ever offered in IT before and so there's more spending. But do you a sense for how IT spend at these digital transforming companies is changing?

[00:44:01] DD: Yeah. I think it varies obviously by company to company, but I think a lot of the spend today of this “digital transformation” is really about migration of certain applications or components of applications to the cloud, one, because it's cheaper and it's more resource-efficient. I think at the end of the day, that's kind of what digital transformation is, is it's about lowering costs of maintaining all these things, driving efficiency internally, whether it's the resources or engineers that are required, or how much they have to spend on cap-X to actually get things up and running. Then, two, time to value and/or consumer experience.

If you can develop things faster, if you can test things without having to spin up a new data center or like bring a new rack online, it's a cost-benefit equation at the end of the day. So that's I think one budget line item around digital transformation. Then I think some of the other budgets are really around security. There's all these data within an enterprise, and as they’re starting to move things outside of the siloed environments, how do we ensure that those are protected, that we have the guardrails and policies in place to catch anything that might be leaking or could be exploited.

[SPONSOR MESSAGE]

[INTERVIEW CONTINUED]

[00:47:10] JM: Are there some acute problems within these enterprises that are unsolved or inadequately solved? What are the – Because my understanding is that as an investor, you're often times surveying the enterprises and talking to them about what are the biggest unsolved problems you have right now. What are the things that you really wish there was a company who could come to you today and say, "Look, I can give you this solution." What are the things the enterprises want to just say, "Hey, take my money," and solve this problem for today?

[00:47:44] DD: Yeah. There is a laundry list of items that meet that criteria. I think the one example I'll focus on this kind of the RPA space, and that there is so many wasted resources just doing the same manual, mundane, repetitive tasks, day-in, day-out. There are is whole offshore teams in various countries doing one job and one job only. So enterprises see this as, again, like a cost-benefit thing and if they can automate some of these repetitive manual processes.

[00:48:19] JM: RPA. This is robotic process automation.

[00:48:21] DD: Yeah. I call it generally just process automation because I think there are many ways that you can automate a process. You can develop an internal application using low-code, no-code type solutions and have that as kind of the workflow for our user internally. You can actually deploy an RPA bot in your environment to execute a repeatable task. It could be downloading or transferring files between two systems, or you can kind of hire a consultant to do it for you every quarter, or over some sort of cadence, or higher, or like outsource this to an offshore development team.

Generally, I think of it as process automation because there are many different approaches to how you actually automate a process. Yeah, I think that's a good example of this is just a resource drain by having a hundred people doing the same thing over and over again, and we can automate this with software.

[00:49:21] JM: This is often times a process that is manually done on a computer. It's usually like you have some internal knowledge worker who, on a daily basis, has to login to a CRM that's backed by Oracle. Some old CRM thing. They do some process, like they take data from one place to another and they have to copy-paste five things over from one cell to another, and it's like a process that you would think you could describe to a computer, but it's kind of hard to describe to a computer and code. So you have these RPA tools the watch your screen and then they turn those commands into a script. That's how it works, right?

[00:50:06] DD: Right. Part of it is the systems that people are working across are these old legacy systems that don't necessarily have outward-facing APIs. I think it works really well for the legacy systems of the world, but I think as more enterprises adapt these 2.0 or newer SaaS solutions, you start to have better opportunities to automate workflows through API calls. I think part of it is it's just where we are in – Call it the digital transformation lifestyle. Yeah. But I think there's a lot of wasted resources there that people are trying to reallocate elsewhere that might be higher value.

[00:50:54] JM: That RPA segment – I know there're a bunch of companies that are doing really well around this space, but there are certain problems that those companies still don't solve or it's just the process of figuring out where are the places that we can apply this technology throughout the organization. What is it that makes you say this is a technology that stands out as still kind of an unsolved problem?

[00:51:16] DD: Yeah. I think it's general applicability across use cases. I think it's worked really well in finance orgs. It's worked really well at legal orgs.

[00:51:26] JM: Rainforest QA, the QA example. I don't know if you saw that company.

[00:51:30] DD: Yeah, Rainforest QA. But I think the interesting thing there is to understand how much of it is really software. How much of it is actually services at least from an investment perspective and be able to kind of identify that and be honest with what you're underwriting there. I think a lot of these RPA vendors today still have a very heavy services component to it because It's hard to set these bots up. It's hard to maintain these bots, right? Then how do you ensure that they're continually performing the job that you specified it or programmed it to do.

[00:52:04] JM: Because if the UI changes, then your bots breaks.

[00:52:06] DD: Yeah, the UI changes, your bot breaks. Yeah, I think one thing UI path has done interesting really well at is educating the market. They have taught a bunch of business users how to develop using the UI path framework. Over time, I think that will help them reduce the reliance on providing services.

[00:52:27] JM: Selling to the enterprises used to be more difficult. You often had to go through a CSO. There is this-top down process that you had to do. Has the sales and adaption process for technology become easier? Has the budget gotten allocated to lower-level people in the organization who can say make a purchase of a new technology, or do freemium technologies? Is this like – This has given rise to the problem with the top-down model. It's given rise to freemium technologies and that's what has given rise to bottoms-up. Tell me about bottoms up sales and how the sales and adaption process for technology has gotten easier.

[00:53:16] DD: Yeah. I think the thing to note there is the buyer might not necessarily be the user. So the user can acquire a technology or solution kind of through themselves through their own network and start using it, start implementing it, and that will work up to a certain extent and then it get surfaced up to the actual buyer, the person who has budget.

It's this delicate balance between understanding who the users are, who the community is? How do you engage that community? How do you get them to love the product that they will force someone in their company to deploy an enterprise-wide license for it? How do you actually also identify your buyer at the same time?

I think the best companies in this space have been able to balance this. Take Zoom, for example. A lot of the adaption early on was fairly organic. It was this bottoms-up motion where someone would share a Zoom link, and as a user, you would click it. It's a delightful experience so now in a conference call and you didn't have to memorize 10 digits and any stars or hashes, and you started using that yourself when you sent out conference call links.

But they realize that after a certain extent, I think it was like three or four people within a conference call, paywall come up and that was a way for them to figure out who that buyer was. At what scale within an organization it made sense for them to go after and deploy actual x-sales resources to that? MongoDB I think has a very similar motion there where they have their open source version and after a certain usage they're able to identify when it makes sense to actually go to the organization and try and get them to buy an enterprise license. I think it's a great way to build momentum within the business.

Now, I think if you're a company, you have to realize that freemium is great in building a high-velocity sales motion and high-velocity traction, but it introduces a lot of noise into the equation as well, right? People are moving in and out of your solution. They're canceling. They're swiping their credit cards. These are small dollar value, things that are happening very frequently, and I'm just talking about at the early stages. In order to build more predictability into your motion, you do have to kind of balance it with an enterprise sales motion. I don't think that's going to go away. It's just how do you balance it? Where do you focus first and how do you remain resource-focused and constrained going through that growth?

[00:56:06] JM: Pick an investment that you've done at Battery and walk me through the mechanics of if, the evaluation of the business. How you start to figure out this was something worth investing in and you know the dynamics of competing for a deal?

[00:56:24] DD: Yeah. I can talk about one that was an existing seed investment we did, which Woven, and we really – Like I said. There're a few things that we think about. One, there's strategic importance. There is product. There is team, and then there is kind of the financials and maybe the go-to-market aspect of things.

The way I think about the financials of a company, it's necessary but not sufficient. There has to be proof points that the company has achieved kind of product market fit that there's some repeatability in their sales motion, that there is a consistent buyer, that there is a consistent use case or a core set of use cases that the solution solves. That I think will manifest itself into the financials of a company and you could kind of see that through net dollar retention, through the magic number, which is the sales efficiency metric, through revenue growth and how much companies are expanding or how much they're adding net new revenue. How much return or a lack thereof there is within the business? I think the motion of how they're selling, who they're selling to, how much value are customers extracting from this or see from it will manifest itself in the financials. I think it's necessary, but not sufficient.

Woven, f example, what we really focused on was team and technological differentiation. The team came out of Facebook, and so they had a strong understanding of kind of what this graph database looked like and kind of the use cases that you could build upon it.

[00:58:06] JM: So company as a graph database.

[00:58:08] DD: On the back. It's a calendar application built with your calendar event as kind of the central node of that graph. So then you can enrich it with all these different pieces of data relative to kind of the authorization and access that you have as a user of that. Right now, your calendar event is a static thing. when you send something to me, what I see is the same as what you see, and their viewpoint is that it should actually be very different, because what we need in order to inform us of this event is very different from what you might need, and you can enrich that kind of using that graph database structure on the backend.

Now, we understand kind of – We understood the competitive dynamics here, Office 365, Google Calendar, these are massive companies with broad footprint. But there is I think inherently a lot of frustration around these solutions as well, and we like the fact that it was technologically different from a lot of the other players in the space that are providing productivity solutions around the calendaring sector, or space, or function, and that was kind of what we built our investment thesis on.

Now, I think if you look at more mature investments we've done, some of the thought process and framework that we go through is are they replacing budget of incumbents? I there a team here and a leadership around them that we think can build a big business and can get in front of enterprises? Sell themselves, sell their product and communicate that value? Do they have a culture of developing products quickly and how have they built those products? Is it in a way that we think aligns with where we see a lot of the market trends going, multi-tenancy, cloud-native, consumption pricing? Those sort of things. We want to see that they have a good pulse on what's happening in the market and they're positioning themselves to take advantage of a lot of those tailwinds. Like I said, they have the financials that support that they're doing these things in the way that we think is scalable and will benefit in the long term.

[01:00:21] JM: That's just a seed deal. Were there any competitive dynamics in that position or was that more just like they were just looking for a seed deal. You got it done?

[01:00:31] DD: I think a lot of investments come down to relationships. How you've built those overtime? How transparent you are through your process, and then the motivations behind why people want to partner up with others. I think one of the big reasons I made the switch between corp dev to venture is that I felt in venture there is a closer, more aligned relationship between the investor and the founder. Versus on the corp side, it's very transactional. I would be a barrier for a founder or a CEO to be joining Google, because the sponsor was someone else. It wasn't me.

I think ultimately in the venture world, it comes down to that alignment of investor and founder. Do they want to work together? There are factors that play into that around speed, around valuation, that make it very competitive, and I think at a certain scale or stage, some factors are more important than others. But ultimately, I think it comes down to relationship and is there a kind of connection between investor and founder.

[01:01:40] JM: The M&A process. If you're Google, you’re acquiring a company, how can that be an adversarial process?

[01:01:50] DD: From the corp dev perspective?

[01:01:52] JM: Yeah.

[01:01:54] DD: Yeah. I guess the corp dev’s job is to get the best deal for Google and to ensure that they are protecting Google. Not necessarily to make the process as easy as possible, right? So there were certain terms and things that were really important to Google corporate that deal sponsor weren’t thinking about because they wanted the team and the product that they could work with and plug into their existing solution or roadmap and they wanted it yesterday.

We had to work with all the different parties to figure out how do we structure a deal that would work for the company and work for Google.

[01:02:37] JM: Long-vesting schedules.

[01:02:39] DD: I can’t say.

[01:02:40] JM: You can’t say. There’s somebody with a ball and chain at Google right now eating from the salad bar, remorsefully thinking of his interaction with Danel Dayan.

[01:02:55] DD: I luckily kept close contact with many of the founders that I was able to interact with and work with, and I always tell them don't shoot the messenger.

[01:03:06] JM: Brutal. What do you disagree with most tech investors about?

[01:03:11] DD: Yeah. I mean, the point around financials being necessary but not sufficient I think is something that unique to my perspective where it's okay if they don't have the certain metrics that a series A company should be at or a series B company should be at. But they have to show that they have a framework, or thought process, or structure in place that allows them to achieve that over the long term. That's one big area that I try and take into conversations with me that I think many tech investors might get stuck on.

There are a lot of companies out there that build their business in a way that makes sense for the vertical that they're going after and they are great businesses. Now they might not be venture businesses, but they're great businesses, and I personally still like those types of companies. I still love to talk to those types of companies, but it's sometimes hard to underwrite that from a venture-capital perspective.

I think one other thing that I like to think of when evaluating open source companies is the best open source companies also own or contribute to the actual open source project themselves. By that, I mean, Spark, and Databricks, the engineers and employees of Databricks are the number one contributors to the Spark open source project, for example.

Confluent and Kafka, another example of that. Terraform Vault of Hashi, another example of that. MongoDB and that open source project and the commercial version, it's another example of that. I think there's been on a handful of companies out there that have proven that if you can own the community of your open source project, then you have the highest probability of actually commercializing that as well.

One other perspective that I have is I think the cloud market is a lot bigger than we all actually I think can comprehend.

[01:05:24] JM: That I agree with. Even hyped men like me. Even I think I probably underestimate it.

[01:05:31] DD: When we think about like sizing the market, there a few industries out there that we just say we inherently believe where they're playing is strategic and it's indexed to the cloud, and they're building it in very scalable way that we think it will be big.

there's a few interesting data points that our team recently dug into around how big this market has actually grown. Take for example Informatica. They went public in 1999. They went public with 350 customers. Then they got sold in 2014, they had 5,500 customers. That's like a 1000, 500% growth in enterprise customers that they were solving or going after and selling into.

TIBCO is a very similar situation, right? Varonis, which went public in December 2013, when they went public, they had 2,400, 2,500 customers. Just take a few years between when Informatica went public to when Varonis public, the scale of the businesses and the number of enterprises out there that were already starting to use software almost 2X'd, 3X'd.

[01:06:51] JM: Well, all these metrics are really crude, because like I operate a business basically out of my apartment. That would not be possible without cloud computing. There're a lot of people like me. Am I an enterprise? I don't know. Maybe in the limit. You look at developing markets. We have no idea how many enterprises will be started in developing markets. We don't know how big they will be. We don't know how many people they will be. We don't know about like the WhatsApp phenomenon where it's like you have 50 people and you get a $19 billion company that I'm sure has plenty of infrastructure expense. There're so many really, really big variables that, all due respect to your open cloud report, it basically makes all of these metrics extremely crude and barely even directional.

[01:07:45] DD: Yeah. I think that's a fair representation of it. Part of the report was really to just frame that, right? It's this big.

[01:07:52] JM: Hey, you got to do something.

[01:07:54] DD: Yeah, and we're only in the early innings of it. Now, what I think we tried to do is look at what are the operational learnings that we've taken from companies that are building in this era, and that's around understanding your community, understanding that your user might not be your buyer, that there is these different approaches to how you sell your product through a marketplace, through a self-serve motion, through an enterprise motion. How do you balance all of those things? How do you create a feedback loop that allows you to iterate and improve on them? If you do that properly and if you do that it in a thoughtful way, then these are kind of the metrics that the best-of-breed companies have been able to achieve.

That's kind of how we thought about the report and helping you operators and founders how to navigate it. I think one thing we want to spend more time on is double-clicking in each of those kind of operational learnings that we highlighted an understanding like what are the components that really move the needle there.

If you're structuring your sales team, how do you incentivize your sales reps? Is it on land or is it on span? Is it both? How do you think about selling through the marketplace? Are they getting full quota retirement through commits or actual usage? These are some of the nuances that we're starting to really dig into that we hope can help operators and founders have ammo as they go through these conversations or thought processes.

[01:09:28] JM: Last question. I've heard the term venture bubble, and that could mean many things, but some people think this is a bubble. Whether or not this cloud growth is unending and inevitable. You could say, "Well, look, maybe the cloud is inevitable and the cloud is growing really, really fast, but there's not enough investment opportunities to satisfy the amount of investment dollars that are pouring in. There is not enough smart people building companies that really have enough of a chance to deliver the returns that venture firms need. Are we in some kind of venture bubble?

[01:10:12] DD: I'd say, from an investment perspective, one saying that one of the partners at Battery always says is you can have a good company, but you also need to have a good deal. It's finding that balance of a company that fits an investment profile that we think can also you generate meaningful returns.

Ideas are a dime a dozen. I think what's critical is how you execute on those ideas and do you have the one learnings DNA and team around you to execute on it. That can manifest itself in many ways. That's understanding how to sell to the enterprise that's having great product vision, that's having strong engineering team, that's able to kind of identify a technological differentiation in the market and build that.

I think it's more like the operational factors that differentiate companies in this era, and I think you have to separate that from where valuations are today. Valuations have crept up over the last 10 years, and there are these like critical high-watermarks that define where the valuation floors are. In 2014, that was SAP's acquisition of Concur for the longest time. Revenue multiples were somewhere in like 6 to 8X last 12 months revenue multiples, or 6 to 8X last 12 months revenue. When they acquired Concur, that was at 10X or 11X. From then on, the majority of high-profile tech acquisitions were done 10X and above.

[01:11:52] JM: Wow! You're saying when an acquisition occurs, if it's one of these technology acquisitions where we have no idea how to size the market, basically what happens is they end up asymptoting towards a price that is just the high watermark of previous acquisitions of that flavor.

[01:12:12] DD: Yeah. I think that's a fair assumption. You saw it again with Salesforce's acquisition of MuleSoft, right? Multiples were 10 to 12X at that point. They acquired MuleSoft for I think like 20 times, and now 20 times revenue is normal. Even the public markets are trading in that range. I think there are these few data points on the valuation side that have influenced where things are trading.

Now, the key is that these markets and these businesses are growing fast enough to sustain that belief that they're actually that valuable. Now, whether overtime reverses to the min, I don't really know, but I think that's kind of the environment that we're playing in now. So for us it's understanding, "Can this team really execute on this idea and is it this balance between good company, good deal?"

[01:13:05] JM: Danel, thanks for coming on the show. It's been great talking to you.

[01:13:07] DD: Thanks for having me. It was a pleasure.

[END OF INTERVIEW]

[01:13:19] JM: When I’m building a new product, G2i is the company that I call on to help me find a developer who can build the first version of my product. G2i is a hiring platform run by engineers that matches you with React, React Native, GraphQL and mobile engineers who you can trust. Whether you are a new company building your first product, like me, or an established company that wants additional engineering help, G2i has the talent that you need to accomplish your goals.

Go to softwareengineeringdaily.com/g2i to learn more about what G2i has to offer. We’ve also done several shows with the people who run G2i, Gabe Greenberg, and the rest of his team. These are engineers who know about the React ecosystem, about the mobile ecosystem, about GraphQL, React Native. They know their stuff and they run a great organization.

In my personal experience, G2i has linked me up with experienced engineers that can fit my budget, and the G2i staff are friendly and easy to work with. They know how product development works. They can help you find the perfect engineer for your stack, and you can go to softwareengineeringdaily.com/g2i to learn more about G2i.

Thank you to G2i for being a great supporter of Software Engineering Daily both as listeners and also as people who have contributed code that have helped me out in my projects. So if you want to get some additional help for your engineering projects, go to softwareengineeringdaily.com/g2i.

[END]