28002 219th Pl SE 3

(0:00 - 3:41)

As a quick intro, as part of my daily job, I spend a lot of time in ProjectsTarget, if you've heard of that in the news, where we are building big data centers for AI, basically these are OCI's, AI investments, and my primary focus is mostly on the software side plus the infrastructure, so there's a lot of work that we need to deliver capacity to the customers, that includes building, how do we model the data centers, how do we connect the dots that leads that model into a real life infrastructure, and how do we operate that once it's all brought up, so there's a whole bunch of things that needs to happen, and there's a software side of it, and the networking side of it, and the computer side of it, so I kind of focus on some of those aspects, so that's kind of what I do. I've been here for a year, prior to that I was at AWS for about seven years, so, and then, you know, I've been in the industry for a while doing networking, okay, and Rick is here, so Rick can give a quick intro, I'll be leading the interview, and Rick will be, you know, fly on the wall after some time. Hey Alex, so first thing first, thank you on behalf of OCI to spend time with us, so we take it as a great privilege that people like yourself are deciding to interview with OCI.

I'm Rick, I'm one of the system engineers in the group, and actually Surendra is my senior, so I reported directly to him, he's the architect, I'm the guy who does the work, he's the one who talks me through and trusts me, so I'm the soldier here, but anyway, so I've been in OCI for about a decade now, I've been there from the day one, I had the, I was grateful and privileged to see it grow from one rack to the biggest GPU provider in the world. I was in the data center, so that's the stuff you can apply for, and yeah, whatever happens, you get a job, you're going to be working with the best in the world, and yeah, we would love to have you sitting with us in the data centers doing hardware, software stuff, and I wish you best of luck. I'm going to go mute on myself because I'm learning on Surendra how to do interviews, so best of luck, Alex, and thank you for having us.

Yeah, no, thank you. All right, hey, Alex, so let's maybe, you know, take a minute for a quick intro, and then we, you know, have some conversations about your experiences from the past. Yeah, sure, yeah, so yeah, I got a lot of, like, electrical engineering, computer architecture degrees.

I went into a company that did wireless mesh networking for integrated circuits, SOCs, so, you know, like, building up applications to do throughput and latency measurements and hardware-constrained, memory-constrained environments. Did that for a long time, 10 years, built up three teams, a big emphasis on automation, and then, so I left that company after about 12 years and went to a walk-through metal detector company, and I was the first hire there in their QA group, and yeah, I was responsible for the end-to-end product, the whole ecosystem, and it was driven by AWS, did, like, the fleet management and the configuration management, error reporting. It started in the cloud, then it got pushed down into these, like, embedded systems, walk-through metal detectors, and Linux operating systems, and a lot of

peripherals.

(3:41 - 5:42)

It was a pretty big physical footprint, and there was cellular modems and internet modems, internet routers and switches and PoE, and there was a tablet, Android tablet for GUI, for customer-facing instances, cameras, and many generations, many hardware configurations, local area networks for integration into companies, intranets, and just responsible for delivering product to market, working in the co-manufacturer and making sure that the hardware coming off the line was valid and conformed to our requirements, and deployments into the field as well. Okay, awesome. So, it sounds very interesting.

So, in terms of the structure, Alex, I'll try to save at least five minutes for you to ask us any questions you may have towards the end, and in between, we basically go through your experiences from any part of your professional life, and as you probably noticed with all the other interviewers, I'll be taking notes, and even if I'm not looking at you, I'm paying attention. So, in terms of other disclaimer, there's something that I need to state as a policy. OCIS is pretty strict lately with people using chat GPD during interviews, so that will ban the people from interviewing with OCI ever again, right? So, this is a disclaimer that I have to offer before I start the interview.

Sure. Cool. Other than that, do you have any questions before we get started? No, I think that's just kind of like get through the required elements, and yeah, we can expound.

(5:43 - 8:07)

Yeah, one question before I do start. I was just looking at your resume. I've seen that you've worn different hats.

You've been an IC, and then you've been a manager. Yeah, absolutely. And this is an IC-specific role? Yeah.

Okay. Is there any kind of reason why you are switching from an IC role to a manager role? Is there a specific reason? Um, I mean, I guess if there is a reason, you know, I was, you know, I was at Evolve, the last company for four years, and yeah, I did like a lot of the management. And, you know, I've always kind of come in and kind of like an IC function, and they just give me the manager title, but because there's, because someone needs to manage.

But as I come in, I kind of like learn the technology and put it all together. So yeah, the title's always been manager, but I always do really consider myself hands-on, and being able to learn something new, and I've always worked on hardware, and OCI, you know, it's like a great opportunity to learn something new. Yeah, and be a part of, have an even greater impact.

Okay, awesome. You know, I've never been a manager myself, although I've entertained that idea now and then. I've always been an IC, so just curious.

Well, I'm sure you've done some management-esque activities, I'm sure. Yeah, of course, all the time. Yeah, exactly.

More TPN than a real manager, you know. Anyway, cool. So let's do this.

You know, like I said, you know, we go through a few questions, and then we can go through some of your experiences. To start off, let's talk about your experience, specifically with respect to a couple of situations that I'll offer you. One, since you've been in the industry for a while, I'm sure you've had to deal with customers.

(8:09 - 24:49)

Can you talk to me about any difficult decision that you've had to make that impacted the customer? If not, that's fine. I can, you know, I can ask you something different, but if that is an experience you've had, maybe we can just talk through for a bit. Yeah, I mean, you know, I haven't, yeah, I've been in the QA function for a while.

I guess, I guess, yeah, if you could, yeah, if there's an option to ask another question, I think I might be able to be give you something more substantive. Okay, sure. Similar, I mean, again, with respect to the customer, have you ever had to prioritize your customers' needs ahead of the needs of the, you know, the team, the group, the business that you're in? Ever? Yeah, I mean, I think, I think that the customer needs was, so we were working on, you know, on these walk-through metal detectors, and there was some internal architecture change, and the architecture change, yeah, I mean, we had logged it as an issue, so the problem is that we were retaining images for more than 30 days, which, you know, kind of like ran afoul of these, like, identity protection, like, you know, we're not supposed to retain identifying information, and so, like, when you watch the metal detectors, it does take pictures and videos, so we had, we're storing locally, you know, it's only supposed to be for 30 days, and, you know, we were using it, the raw data in order to just evaluation, and at times when there are security incidents, then they want to go back and find the scan data, the instance of when somebody brought some object through, and either, in most cases, it was, like, somebody brought through a weapon, and the weapon wasn't intercepted for, like, a lot of reasons, and so, like, we're doing our, like, forensic analysis, we, you know, as a company, we, as an owner of the walk-through metal detector, we need to have the forensics in order to make an action report, so we, and our data retention policy was only for 30 days, but we did a back-end change to where we moved it internally, and then disk cleaners and their retentions weren't updated, and so then, you know, we were saving damages for too long, and yeah, you know, I guess, I guess just, yeah, I'm always, like, how did you guys detect that? I mean, was that a violation that you guys figured out internally, or did the customer complain? Yeah, we found it, we found it, we found it internally, and I guess just kind of both of these questions that you've asked me, just, you know, my first inclination, which isn't the best, is that, like, we weren't, we aren't really the, like, the client-facing entities, like, we do, we do, we assess everything, and then we just kind of generate the reports and log the tickets, and traditionally, it's kind of been, been, like, other

entities that kind of, like, made determinations, so in this regard, we had logged the ticket, we had the automation, we knew it was a problem, and it was only until very late that somebody at the company realized that this was going to be a problem.

I mean, like, yeah, we could always, like, escalate it more, but things always move, like, very quickly, so, yeah, and so, and so once they decided that there needed to be a fix, yeah, absolutely, we honed in on it, we made sure that these disk cleaners and the data retention implementation was implemented and worked in the different ecosystems and generations and hardware configurations and across different, like, like, settings of the hardware, you know, I'm just sure we can all be familiar with just how many different configurations and permutations are deployed in fields, and so, you know, I really put automation first, so it didn't, it wasn't really that, like, difficult to, to test, but it certainly was something that was, like, an interrupt to just kind of, like, our normal process of kind of, like, the quality of, like, new feature testing, regression, performance, auditing, and then, like, releasing, we kind of accepted this change and ran these, these tests and still did, like, performance tests, because it is, it is kind of, like, something that would affect, like, the whole ecosystem, like, like, disk cleaning and checking the, the age, the, the date creation of different objects in the operating system, you know, that could, that could become onerous, you know, as, as, as the scan data gets larger and larger, so it's, you know, it's a very nuanced thing to characterize, but, yeah, yeah, we had to, like, adjust our testing schedule. Got it, okay, so, how, I mean, if, if I may ask, you know, you mentioned 30 days, what's the retention policy internally? How, like, how, how long did you guys go before you guys detected it? Just, I'm not looking for exact numbers, I mean, just the ballpark. Well, yeah, so, okay, so, with this change, that change where we were storing the, the data, it never got released, so it was never an impact, it never impacted the field.

Is that what you're asking, like, how, how long did this... Oh, I see, so, so this, but you said the disks weren't getting, weren't filling up, so this is not, and, so this is something that you guys detected before this went to the market, is that? That's right. Yes, okay, so there wasn't, there was never a customer impact with this one, or was there? I, I mean, no, there wasn't, there wasn't, because I guess we, we, we, we were the, you know, like, we're, we're always kind of, kind of, kind of thinking about, thinking about the, the, the customer. I think, I'm trying, I'm trying to, yeah, I'm trying to think of some kind of, like, customer, customer-facing issue.

I think, I think it would be, like, maybe with regards to, like, the cameras. So, like, maybe, like, the, the, the, you know, we produced some new camera, and this, this was in, this was in field studies, where we, where we worked with customers to bring them, kind of, new development hardware, and we saw that the customer reported that, that, that just the cameras weren't really able to, able to, you know, do the, do the, do the, the backlighting resolution and low light resolution. I wasn't able to really put together a good, like, grayscale white color disambiguation, so the, so the images weren't really showing up properly in, in, like, sunlight or low, low light situation.

So, that, that is something that, like, affected, affected the customer. Got it. Okay.

So, in, in going back to the first example, Alex, just to wrap that up, so we can switch to another question. In, in, in this case, was it a matter of just addressing the detection, like, was, was, because you said you were able to detect that, that this was, you know, missing, was it an improvement in the automation? Like, what, what was the, the learning from that experience? Well, yeah, I guess what I was mentioning is, like, we found the, we found the issue in QA, and, you know, we logged it, that these files weren't getting deleted, and I guess just how I always felt is that just, like, product wasn't really involved in, in, in our triage, and our bug scrubs, and, and, you know, it's funny, I'm just always talking about this, it's like, who owns quality? And, yeah, so I just kind of always felt that, that there, my own company had evolved, there just wasn't enough ownership, and, like, what, what bugs or features were in each release? So, the learn, the learning is, like, if, if we do in QA see something that we do think is going to, like, really affect the customer, we just need to go higher, higher up in the organization to get, you know, to get ownership on it, and, yeah, yeah, oftentimes there's just not a clear owner, and that's, that's when, you know, we kind of lack a decision, and then, so I think maybe, maybe kind of creating a filter of, like, every bug that we log should have a fixed version assigned to it, and then that might be, like, kind of an indirect way of, of seeing that it has been, you know, accepted by product, and, and assigned to a release, and, and so, yeah, we could, we could certainly kind of increase, state, like, more fields, and have kind of, like, more, more, more visibility into kind of ambiguous situations. Okay, and this was considered a serious violation, if, you know, oh my goodness, yeah, if it had gone out, and, like, and, like, a customer had retained, you know, identifying information for more than 30 days, yeah, that's, it's just, it's very serious.

Okay, fair enough, all right, and maybe another last one before we switch gears on this. Was this something that, I mean, you said that this was detected during the QA phase of the, the lifecycle. What was your role in this, you know, in this whole project at that time? Were you the manager? Were you the... Yeah, yeah, I'm the, at this phase, I was the manager, and it was, it was kind of, it was, it was like maintenance, sustaining, you know, it's like, it's like, it wasn't a new feature, I mean, this was, this is very, something very rudimentary, very simple, you know, these just regulatory requirements, so it's something that was easy to automate, so, you know, we, we generate scans, we create these artifacts on the operating system, and then we, we accelerate the clock, or we just change the system clock to a month in the future, maybe, like, reset the computer, and wait, wait to, for the disk cleanup, and all this stuff to run, and just see is all the, is the raw scan data there or not, so, you know, it's very low level of effort to automate it, so, yeah, it was something, it was something that was very easy to automate, and I work, and yeah, I kind of work on, I worked on, like, the new feature is what I personally was kind of had my hands on, so with regards to my role in the incident, is that, yeah, someone, well, you know, one of the manual engineers, or somebody who was just reviewing the results, they logged the bug, and yeah, I saw the words, and I saw that it had all the artifacts, and as the manager, it was, like, pretty clear to me what the problem was, and, and I suppose the, again, like, kind of, like, the learning is that, like, maybe it just, it just didn't register with the other teams, and there, there, there just wasn't this kind of, like, escalation procedure in place to, like, make sure

that the right decisions were being made about these issues, so, yeah, it's like learning.

Yeah, and did you, I mean, you talked about learning, did you do something about this, you know, for the future, or was it basically your team, or you, you dealing with this, and making some notes for yourself? So, what, what did we, what did we do differently? What we did differently was, you know, try to break down these silos, and, and invite the, the, the pro, the, the the product owner to, to be involved in the, in the, in, in the bug scrubs, or at least give them the, the availability, or, or make it clear to the, to the product owner what the, what the issues were, and, you know, kind of maintain a, a dialogue with the product owners. Okay, got it, so, so from that, from that time, time onwards, a product owner became part of this, this, bug scrub, I would say, yeah, the bug scrubs, yeah, got it, okay, cool. Maybe another quick one this time, and then we switch to some of your, let's see, so, again, any part of your experience, was there, I mean, can you talk to me about, like, you know, a, a time when you had to make, or act with limited set of information, or limited set of data, and basically, you know, like, you are faced with situations where there is a problem, and you don't have either the time, or all the data in front of you, have you ever had to deal with a situation like that? Yeah, I mean, yeah, there's a problem in front of you, yeah, yeah, there's, there's like, yeah, I mean, there's, there's always, yeah, in engineering, in this engineering space, yeah, we're trying to, yeah, the, all the information was, yeah, we were, we were making, so, we had this way to inject alerts into the software of the walk-through metal detectors, but it was all just the software chain, and so, I was just calling, like, an API call to inject into the software that, that there, an alert had occurred, and so, that would be able to test the software path, and test how the operating system managed processing the data, and passing it through the different interfaces, but, you know, that didn't really test the hardware, so, what we needed to do is, we needed to make something that was, that, that moved the metal within the electromagnetic sensing of the walk-through metal detectors, and so, yeah, it was, it was, it was a collaborative process, but we, we settled on making, making electrical fans that were connected with adreno boards, and then the, and then the adrenal boards would, would, you know, you could set bits in the GPIO registers of the adrenal board in order for it to, to, to drive, to turn the, the, the, the motors of the fans, turn them off and on, and then, and then those, the spinning of the fan is, is what, is what induced the, the walk-through metal detector to, to tell us that there was metal in, in the space, so, yeah, just quickly, what was the problem that you guys were trying to solve, and like, what was the information that was lacking? The, the problem we were trying to solve was, was passing metal, a deterministic way to pass metal through the, through the walk-through metal detector.

Okay, without, without raising an alarm? No, we, well, so, we, we just wanted to, to characterize how the walk-through metal detector would respond. Oh, I see, okay, okay, okay. Yeah, so, ideally, we could get the, the walk-through metal detector to alert.

I mean, that, that's like, you're kind of, like, testing the whole functionality. I got it. And, but there, but there are cases where metal can go through the detector where we don't want it to alert.

(24:50 - 25:47)

I, that's kind of, like, the whole, kind of, like, selling point of the system, that it's able to, to differentiate between, it was able to differentiate between metal that was a, was, that is a threat, and metal that's not a threat, to reduce the amount of, like, alerts. So, yeah, so, so, so, what was unknown was, just how, how was the architecture going to work? How would we integrate it within our automation? If, if the RF, if the, if the active RF signaling was going to interfere with the, with the, with the machine's ability to detect the metal, because it's just, it's just, like, a very sensitive piece of equipment, and it goes, does go through calibration during the power on self-test, but then when we have, like, electrical components powering the adrenal board, and then the adrenal board powering motors on the fan, and then the, and then all these, like, electric, active electrical components around the metal detector, we just, you know, like, weren't sure how the system was going to behave. Okay.

(25:48 - 27:54)

If we were going to be able to, like, deterministically assert that, that, that a metallic threat, you know, either is detected or not detected, depending on, on the, on the sensitivity settings of the scanner. Okay. So, how, how did you guys approach this? I mean, you talked about not knowing the architecture, not knowing how to integrate this, not knowing how the machines behave, or the electrical integration that you talked about.

Like, you guys have to sit down and, you know, figure out, spec the whole thing out, and, you know, deal with it, or, like, how, how do you, give me a sense of how you guys approach this? Yeah, yeah. So, I guess, just in the interest of time, no details, just, just to, you know, how did you guys go about solving it? Yeah. So, it's just, you just, it's just, you kind of do, like, incremental, you do, like, incremental approaches.

So, then we bring, like, an Arduino board to it, and then we, like, power it on. And then we do have all these spectrum analyzers, so that we can see what kind of noise it's inducing. And then we connect the fan to it, and then, you know, power on the fan, and then see if there, what kind of electromagnetic noise is being induced.

So, we just kind of, like, piecemeal, iteratively expand the footprint of the implementation, and just to see how the systems responded. Okay. And, in the process, I mean, this, even though you were doing this incrementally, you could not deliver this to the customer, or you couldn't, you could not ship it, because it's not complete, or, like, were you able to ship some version of this before completing this? Yeah, yeah, we did, we did, yeah, we did ship, yeah, like, some version of it, because they wanted to use it at, like, test houses, and they wanted it to use it in different places.

So, then, yeah, then there was, like, yeah, we didn't really have a good, like, deployment of, again, like, the integration into the infrastructure. And so, like, locally, in our labs, we could use it, but then when we took it to external sites, we just didn't have, like, the interfaces exposed.

So, people had to, kind of, like, connect a laptop to the Ethernet port and issue some commands, like, serially.

(27:55 - 28:31)

So, yeah, there was these, like, iterative, like, deployments into the field. Yeah. Okay.

All right. Let's switch gears, and I wanted to kind of broaden this discussion a little bit, and then also make this open-ended, so you can tell me more. Since you've been, you've been working for about, yeah, like, 15 years, I guess, so you've had a lot of experiences.

(28:33 - 28:59)

Talk to me about, maybe two projects. I'll, you know, we'll talk about one that you're really proud of, one project that you're really proud of in your career, and if you can talk to me why, and, you know, what are the reasons for that, and, you know, we can have some discussions. I will have some follow-up questions on that.

(29:03 - 34:09)

Something that I'm really, really proud of. Mm-hmm. I think, I think the, you know, what I'm, what I'm, what I'm proud of is writing the application layer that characterized the performance of these mesh networks that we were, that we were building at Silicon Labs.

So, you know, we were selling this product, and there wasn't really, like, a market for it, because we were, like, you know, the first ones coming out with this, with this ad hoc wireless networking stack, networking stack, and so we needed to kind of build out an existing, build out, like, real networks with real products in real environments. So we did, it's not the simulation, and so we built out the nodes, and the peripherals, and the power, and the ethernet, and the antennas throughout our lab in order to show that, like, a network up to, like, 255 nodes could exist, and it does work, and the stack won't fall over, and that it is able to support different levels of throughput based on your, based on how, how complex the routing algorithm is, if it's, like, proactive or reactive, and the size of the window, and the size of the, of the packets, and the size of the address tables. So it's just, just how, how the, how the performance of the, of the wireless network behaved under, under external interference, but then also the, just the different configuration parameters that are, that are supported by the, by the, by the protocol, and to be able to bring customers to the office, and show them, and then show, show them our diagnostic tools, and our, like, packet captures that we had, to be able to, like, show them that the tools work with real hardware, just really gave, gave customers the confidence to kind of, like, take on, take on our product.

So it's like, you know, we were using orderable part numbers, like, you know, parts that we could buy from the company, populating all these boards, building up the application, and gathering performance data. Okay, and was this a, like, a start, you know, startup, or a new company? Yeah. Got it, got it, okay.

What was your role, Alex, in this? Like, were you, like, a founding member, trying to define the overall architecture, implementing it, or, like, this is, again, the QA part, because you seem to be focused more on the QA. Yeah, it's, it was definitely QA. You know, like, like, it wasn't in, it wasn't in the development side, because the development side, they're the ones who gave us the, like, the, the static binaries, like, the, the networking stack implementations, and yeah, they're the ones who, who developed the internal tools.

So it was in, it was a QE, QA function, again, of, like, buying all the parts, you know, procuring all the parts, getting all the wiring, the power, populating all the boards, but then the big part with the audit was the automation of deploying the virtual machines, connecting to all the adapters, getting all the serial, all the, all the packet capture data off of the adapters, and then, like, collating all the information, so that you can kind of, like, see everything happening, and, you know, getting the wireless packet capture of a lot of nodes, you can kind of see individual packets going around the network. So I, I did, I did a lot of it. I mean, it was a team, so I certainly didn't all do it by myself, but I, I, I was part of the team.

I built boards, I, I wrote the scripts that connected to the, to the adapters, and did all the joining, and the provisioning, and I guess what I really did was I wrote that, that application layer that, that detected routes, sent packets, waited for packets to come back, and then kept counters of the size of the packets, and the windows, and, and how long it took for packets to come back, and to be able to get to those, to be able to get those throughput, throughput and latency numbers. Okay, okay. Right, so, sounds awesome, actually.

It sounds like a very, especially, I mean, I've been in those situations where, you know, you kind of get to do a lot of stuff without having people telling you what to do. Oh, yeah, we, we, yeah, we, you know, I guess we had, like, well, I had, yeah, like, networking degrees, so I kind of, like, knew what, what people were looking for with reserve, with, with, with respect to, like, throughput and latency across hops. It's a pretty simple graph, and, you know, I kind of, like, knew how to implement it.

(34:11 - 40:10)

Cool, nice, nice. So, what, what part of this project was, like, challenging or complex, and that kind of gave you this kick that you're proud of? Well, I guess, I guess, I guess the complexity was, was, like, bringing up the virtual machines and making sure that they would be able to, to connect to all these adapters that were, like, spread out through the network. So, I guess it was kind of, like, simplistic to think about in the past, but the benefit of hindsight is, I guess, it seems simplistic, I guess it always is, but I think, I think, like, monitoring the power, because, yeah, there were just certain times when when these modules would, would, would not be getting enough, enough power, and, you know, because we were doing, like, probably, like, too much fan out of, of what was being driven by, by our circuitry.

So, like, being able to detect round out situations, I think, was, was, like, critical. You know, we had these, like, voltage monitors on the integrated voltage monitors, and then, and then

aggregating all of the wireless packet captures back to, to a central repository so that we could then collate all, all of the packets being, traversing through the nodes. I think just, it was, you know, 15, 10 years, maybe 10 years, 10 years ago, 10 years ago, you know, back of, like, big data, we were using, like, vSphere to, in virtualization, so it was just a lot of combinations of these new technologies that I was working on, and just seeing them all, like, work in, work in concert together.

And, and, you know, we were working with IEEE to get these, like, certifications for these protocols. So, to be able to release performance data that is, like, purely a function of, of implementations of IEEE, I mean, you know, I just come out of grad school, kind of, you know, a few years out of grad school. So it was just, you know, very, very, kind of still had that academic flavor, and using, like, engineering in order to deliver, like, academic results is just all, like, kind of this beautiful harmony of industry and academia.

Okay. What was, like, you know, in doing this, or in going through this experience, were there specific, you know, problems that you had to uncover, and then go back to the, you know, the devs and say, look, this is not working, or, like, because you're trying to make sure that this is, this works as per the spec, and if you find, like, anything big, or, you know, the designers do, like, a fantastic job that it went through, like, and you did talk about writing the application layer. Was it more about, you know, everything is working, you're just making sure that, you know, we collect and, you know, you kind of said that characterizing the performance.

So is it mostly saying that, you know, here's the performance, and was the performance not where it was supposed to be, and did you have to go back to the devs and iterate on this, and then, you know, like, was your application really helpful in doing that? Yeah, yeah, yeah. So I think just, like, one of the issues that we saw was that just there was, like, a certain type of device that you could configure, and it was, like, a parent node, and the parent node, it's just a node that can accept connections to other devices, and so it's supposed to have a more, just have more tables in memory about, like, what nodes it's connected to, and you can have, like, set bits in the protocol for it to, for it to, like, proactively discover through, like, broadcast messages, and yeah, we just saw that, you know, in one of these, like, bits in one of the fields about, like, proactively broadcasting to discover in devices within its range, it wasn't inducing that behavior in the parent node, so we could see it in the Wireshark packet capture, we could see the bits of, you know, what does the field represent, and it just wasn't causing the parent nodes to go out and update its, like, routing table based on the nodes that responded, and it really came down to, like, the power management on the nodes, like, these, we had, like, industrial applications, so the, so these devices were meant to exist in the field for, like, seven years, so we were just putting a lot of effort into monitoring power consumption, and so, yeah, it was just the complexity, you know, on the dev side, they just weren't managing the static machine and the power levels, and so some of these more proactive routing features just weren't implemented properly, and then, you know, that's probably because it just wasn't as, it wasn't as much used part of the spec. Okay, yeah, I'll ask you kind of differently.

What differences did your application layer make for this product in, you know, getting that out to the market? What differences did it make? Yeah, because you wrote this, you know, you built this to characterize the performance. How did it make a difference to the business? Yeah, yeah, so the difference was, the difference was, again, like, people could come in and customers could come in and see it working, and then also, again, like, the test reports, the performance reports. That, you know, was part of the, you know, the website, and the SiliconLabs website, and part of the materials that the business could send out to customers to show what the throughput was, and so then it would be able to answer customers' questions of, you know, what was the maximum throughput that it could, different network topologies could support, and so that they could decide if our technology and microprocessors would be suitable for their applications.

(40:10 - 40:23)

So it just made it easier to get design wins. Okay. So, let's, we'll talk about another one.

(40:24 - 41:38)

So, you can use this, you use, you know, a different example, I'll say it's the same one, it doesn't matter, like, can you talk to me about a project that you worked on, which was large in scale, or very ambiguous, very complex, and that you had to work with, I don't know, different people, teams, orgs, what would be the largest project that you worked on? The largest project. Um, yeah, so, I guess, just kind of like what comes to mind recently is on the walkthrough, walkthrough the metal detectors, we did, we did, the developers did expose this interface of being able to inject the signature of metallic objects. So, earlier we talked about an electric fan that could like rotate in the field, and so that would be, that's like a object that's rotating in a fixed position in the magnetic field.

(41:39 - 42:13)

So, but we wanted to simulate, now we're moving into kind of like a simulation land of where we had a library of metallic objects, signatures of metallic objects. Well, what the development did was they exposed an interface where we could inject that electric, those electromagnetic signal on top of the RF stream that was coming from the sensors. And so we were able to assess if the system was able to detect these metallic signatures on top of the RF data that was being consumed by the walkthrough metal detectors.

(42:14 - 43:06)

And this was important because the walkthrough metal detectors were very, very sensitive, and if there was some transient, and if there's some transient interference, where there's some degradation of hardware or power while the, while the scanners were running, then that would change the baseline and it might, it might impact the scanner's ability to detect weapons as they're coming through. And so that was something which has never really had a, like a good

handle on is how did, how did the behavior of the scanners change over time, you know, while they were on, you know, certainly as a function of temperature and humidity and just, yeah, just the degradation of the hardware. So I've had to work with the team, we call them the, like the ATD team, the, the, the threat detection team to get the, to get the library of signatures.

(43:07 - 43:34)

And then I had to work with the host team to be able to inject those, those, inject those signatures into the, into the raw data stream. And then to be able to aggregate what decisions the operating system was making, the host application was making, if these were, if these were threats or not. And so, you know, I had the dictionary, the source of truth of, you know, these signatures should alert, these signatures shouldn't alert.

(43:35 - 44:04)

And so I was able to change all the configurations of the scanners, inject it. You could, we could change in the X, Y, Z orientation, how the objects were injected. So just from a simulation point of view, we, I was able to inject, you know, pass these objects through the plane at all these different slopes, inclinations, spins, orientations in different lanes and different scanner configurations.

(44:04 - 44:34)

And I was able to detect orientations of objects that didn't alert, that the scanner didn't, didn't, just didn't think were, were, were, were threats. So I was able to create, you know, nice X and Y heat maps of where objects were detected or not detected, and then collate those to the actual scan, raw scan data. So that, that there would be, you know, very impactful, simple visualizations of where the scanner was missing detections.

(44:35 - 44:40)

Okay, cool. Can you give me, this is awesome. I mean, sounds interesting.

(44:40 - 45:26)

Can you give me some data, anything that you can think of that tells me that, or that supports that this is a large project? I mean, in your view, I mean, right. In terms of people, organizations, or, you know, whatever you can think of, dimensions you can think of, like to say that this is why this was a large project. Oh, it was a large project because there was, this was, this was the only ability to, to, to evaluate how the scanner was, was acting in the field at runtime.

(45:27 - 45:40)

It was, it was a big project because it worked. I got it to work on all the different hardware configurations. So it was able to, to, to evaluate all the different versions that we had in

production.

(45:40 - 45:55)

So that, that was like a big part of, of, as you know, I was, I was both acting as a manager and, and as a IC in this, in this capacity. And I, you know, I made sure that we had all the, all the configurations in the field. And, you know, that was like difficult to do.

(45:55 - 46:22)

And so I had to work, be working with production to be able to know what, what was out there so that I could recreate those, those configurations in the field. So it was, it was big with regards to, you know, it was touching 100% of the customer base. It was, it was big because it gave the, you know, the research scientists, the ATD team, the algorithm threat detections, giving them real time behavior on, on real hardware.

(46:22 - 46:29)

Because they don't, they didn't, they didn't have real hardware. They just only had, they only had like simulations. They only really worked in simulations.

(46:30 - 46:53)

And so it was a big project because it, it, it opened up an ability for the cust, for the company to observe the, the, the real time behavior of, of all the scanners. And we were able to inject, scan, inject objects like in the field. And to be able to see if, if, if, if I, if, if the scanners in the field were able to detect objects.

(46:54 - 47:37)

Because, you know, up until that point, it was all very reactive. You know, I was like, what, what were the chances that the customer even observed that an object was missed? So it was, it was big on the, on the scope of hardware that it was running on and, and, and capacities of the company to evaluate the behavior of that. Got it.

Okay. And was it, what is the end result of this project? I mean, you know, you built this, you worked on this, you know, simulation and, you know, did all kinds of testing. What was the outcome of this? Yeah, the output is that we have, is that we have galleries.

(47:38 - 48:12)

We have like canonical sets of, of the, of the, of the scanner's ability to detect objects. And so we're able to see as we, as we incorporate new pieces of hardware, as we, as we make software changes, as, as yeah, we're evaluating, integrating some kind of new cellular modem, how all these changes are affecting the ability of the scanner to detect threats, which is like the core, the core, the core function of the, of the detector. And so like, I was able to like run all these

injections over all these sensitivity levels.

(48:13 - 48:42)

It's a big project just with regards to the, to the, to the amount of hardware and the number of configurations and the number of sensitivities. And I was creating galleries that showed where there were misses, where the scanner did not detect an object. And that would inform, that did inform what updates and what additions to the, to the, the, the machine learning, you know, they said it was like an AI.

(48:42 - 48:55)

So there was like a model, updates to the model so that, so that it could, you know, reduce the, the, the size of these holes to, to, to minimize the chance of like missed object, missed detections in the field. Okay. Yeah.

(48:55 - 49:27)

Quickly, in the interest of time, I don't want to spend too much time on this, but you know, maybe like a couple of sentences. Was there any feedback from the, from the field or the customers on how well this performed compared to, you know, what you guys benchmarked internally? Um, so it was like, it was just something that we used internally. So we never really gave it to the customers to be able to do injections.

(49:28 - 49:34)

So it was, it was, I don't, I don't think the customers knew anything about it. Okay. That's fine.

(49:34 - 49:56)

So last one, because I still want to save five minutes for you. Um, can you talk to me about any kind of lessons you learned from some, you know, we can, we can stick to major failures in the 15 years that you've been in the industry, like any projects, any, anything. I mean, you've been, again, a manager at NIC.

(49:58 - 50:25)

Yeah. Have you had any failures? And what are the lessons from those failures? Yeah, I think, I think the failure, um, I think the failure is to like invest, invest a lot of time and effort onto something and then, and then like not get adopted. So, and that's like as a manager, I would say, because again, that is kind of like where you have the biggest impact.

(50:28 - 50:46)

Yeah, we kind of, we had this idea and I guess there was somebody on product who wanted it. And we, maybe we, we spent, well, what we did is that we took something, we took a backend.

So we took like the backend from another team.

(50:47 - 51:06)

And, um, you know, we just kind of like shoehorned, uh, an implementation from, from another team is to monitor code size. So we had this, this situation with the product team wanted to see how code size was changing. And, um, so we, we, we took the backend from another team to store the data because it was there and they had it and they used it.

(51:06 - 51:13)

And it also came with like a front end and it was just like a lot of dropdowns. It was just very complex to manage. And maybe it was good for like a dev team.

(51:14 - 51:47)

But, um, you know, we, we were trying to expose this, this, this, this visual, this visualizer to like, I guess, less technical people, you know, like, like, uh, product owners and, and like, and, and, and managers and like business managers. So, you know, we, we put the effort to, to, to push all the code size into, into the backend and we set up all the relationships to show how it changed over time and how you could, it was just infinitely customizable, you know, like get, and then we kind of gave the front end to these kinds of like non-technical people. And they, they just didn't really use it.

(51:47 - 52:09)

And we just never really kind of closed that gap of like, oh, putting an overlay that's just like really simple and increasing, like, you know, driving engagement to be able to give them what they wanted. And then like that, the first product guy who wanted it, he kind of left and then it just kind of died and it just didn't really like go anywhere. So, um, yeah, this was like pretty early on, you know, as a manager.

(52:09 - 52:48)

And so I've always, from that point, you know, like the lesson learned is to, is to, is to get adoption from larger groups of people have, have prototypes and demos and kind of like buy-in from, from like multiple teams and multiple people and have, and have the check-ins to make sure that it's solving a problem. And then, and then, you know, and I, I think like really taking the time to pick the tools that work for us and instead of just like taking something that some other team happens to have and just kind of going through like all the adoption. I mean, that might be the good, that might be the answer if it, if it works for your team.

(52:48 - 53:00)

But, but I think in this case, it was just, it was just too complex. And we just like took it without doing our due diligence. And then we were just already kind of like sunk into tech dad of just,

oh, just like one more thing to get this thing to work.

(53:00 - 53:14)

And so we had kind of already put, put in too much emotional resources to just kind of like let it go. And we should have just like started fresh and just built, stood up something that, that did what we wanted it to do. Okay.

(53:16 - 53:20)

All right. So I think we're right on time. Five minutes.

(53:21 - 53:29)

And I'm going to shut up and you can ask questions. And Rick, you can come up there. Yeah.

(53:29 - 53:34)

I mean, yeah, I was looking at, I like, I like looking at people at LinkedIn. Well, first, thanks. Thanks for your time.

(53:34 - 53:38)

I know. Yeah. I'm sure you've got a lot better things to do than interview people.

(53:39 - 53:42)

But yeah, I guess it just kind of comes with the territory. So thank you. Thank you for your time.

(53:43 - 53:49)

Yeah. I kind of look, I like, enjoy looking at everyone's LinkedIn. And yeah, you know, like you said, you've done just like a lot of things at a lot of different companies.

(53:49 - 54:29)

And I guess I'm just, I've always kind of like struggled with this position of like not really, not really knowing, you know, exactly what the day-to-day is. And it just sounds like it's going to be kind of like building up and maybe running like maintenance and new features and accepting new hardware and this OCI lab that we're building, the hardware backend that we're building. So I guess just how, like, how does your role relate to, you know, how, you know, my role is going to be kind of like building and maintaining these like hardware labs.

(54:29 - 54:42)

Let me give you a stab at it, Alex. So the way the setup is for this interview is very similar to, it's called bartending or baris and glue. Me and Surendra, we don't report into your hiring manager.

(54:42 - 54:55)

We are kept impartial because we don't know what is the team requiring. That was done by like, it's like a separation of concerns. But what I can tell you, I'll go first.

(54:55 - 55:10)

So the way OCI was constructed, I think Surendra alluded to it at the beginning, it's very similar to Amazon. So we get shit done without no drama. Whatever the leaders decide from the business need, it comes down and we get it done pretty fast.

(55:10 - 55:19)

That's why OCI has been so successful in the last 10 years. And there's no drama. So when it comes, like the CEO right now is Clay Magook, some young guy from Amazon.

(55:19 - 55:28)

He's driving the business. And as of today, we are the biggest AI provider of the world. And how this translates to is whenever there's a need, we get it down.

(55:28 - 55:37)

Like I've been flying a lot in data centers around the world. Surendra has been architecturing stuff that's impossible to architecture. Like they're talking about nuclear power, gas turbines.

(55:37 - 55:48)

So for us, our daily... For me, I'm an engineer and Surendra is an architect. So from a daily... I think you and I, we're going to be peers if you get a job. So the daily stuff will be very hands-on.

(55:48 - 56:02)

There's going to be an architecture diagram from Surendra and the CEOs. We're going to come at and execute it within a few months time. Like the GB200 stuff that came about, right? It was architectured like six months back.

(56:02 - 56:13)

We were traveling around the world. I can tell you for sure, this organization is a very engineering-centric place. Even the CEO looks at code, look at design.

(56:14 - 56:33)

Surendra, do you want to say something? No, I think it's all good. I mean, like Rick said, we are not specifically tied to this team. And if you speak to the hiring manager, some of the role-specific questions may be appropriate to the hiring manager.

(56:33 - 56:51)

And then you may be able to get better answers. But generally speaking, there is no limitation to what you can do. And there is no prescription of what you should be doing.

(56:52 - 57:13)

We tend to adopt any ideas that come from anywhere as long as they're better than what we already have. So in that sense, the culture is somewhat similar to Amazon. But then we have people from Microsoft, from Meta, from Google.

(57:13 - 57:31)

It's kind of a combination of those cultures. And it lets you essentially do what you want to do without being constrained. And I can tell you from my own experience, I'm a software and a networking guy for a long time.

(57:31 - 57:48)

And if you probably looked at LinkedIn, you already know. And what I'm doing right now is so different from what I did before. So because people at the senior leadership, they don't give a damn about what your background is as long as you're able to solve problems.

(57:50 - 58:08)

And right now, I'm focusing on how fast we can deliver infrastructure. How fast can we build regions? When I say regions, I mean literally building, standing up a data center. That involves somebody makes a business contract in some region.

(58:09 - 58:36)

And then you need power and cooling. Then you need GPUs, networking racks, cables. And how do you bring all of them together and make that happen fast and give that to customers based on the contracts that we've drawn up with them? And how do you repeat this thousands of times? Because unlike the other companies, we are growing at the clip where we crossed 100 regions.

(58:37 - 59:07)

And we are aiming to cross thousands of regions, right? And that's where the business trajectory is. And if you have to solve that kind of scale, what does it take? What kind of systems do you need? What kind of thought processes do you need? You see how this gets super complex and at the same time exciting because you have the opportunities to go do things that people haven't done before. Yeah, and I think this role... I know we're out of time, but just to quickly say it.

(59:07 - 59:22)

And I think this role is validating the hardware and of course, the software that runs on the hardware. And so some components are going to be coming through Burlington that's sort of applying Burlington Mass and validating the new hardware that's going to be deployed in all these sites. Yes, yes.

(59:23 - 59:28)

Yeah, it changes sometimes. We build our own hardware. Things come from outside.

(59:28 - 59:36)

Oh, sure. Okay, well, I guess we're out of time. So yeah, thanks.

(59:36 - 59:38)

Thanks for your time. Yeah. All right.

(59:38 - 59:47)

Hey, very nice talking to you, Alex. And good luck to you. Yeah, and I guess I'll just reach out to HR about like, I don't know, timing next steps.

(59:47 - 59:51)

I'll just like wait a few days. I don't know. This is my last interview.

(59:51 - 1:00:01)

Yeah, if you have a recruiting coordinator working with you, they're probably the people that they should be able to respond to.