Deloitte.

The Deloitte On Cloud Podcast

David Linthicum, Managing Director, Chief Cloud Strategy Officer, Deloitte Consulting LLP

Title: **Google's head of developer media discusses the tech talent shortage and how AI can help**

Description: In this **episode**, Forrest Brazeal, Google Cloud's head of developer media, talks with David Linthicum about why there's an increasing talent shortage, how to address it, and how generative AI can be used as a tool to solve the problem, rather than a replacement for humans. In Brazeal's view, companies should use generative AI not to generate code that might not end up working, but to help new talent learn the process of software development on the job.

Duration: **00:26:22**

David Linthicum:
Welcome back to the On Cloud podcast. Today on the show I am joined by Forrest Brazeal, head of developer media at Google Cloud. How're you doing, Forrest?

Forrest Brazeal:
Hey, Dave. I'm doing well. Thanks for having me on the show.

David Linthicum:
Tell us the Forrest story. How did you get to Google? How has your career progressed to where you are right now? How did you come to becoming a well-known book author, and we're going to talk about that during the podcast. What were the building blocks to getting you here?

Forrest Brazeal:
Just like a lot of us, Dave, my journey in tech has been multipronged. It's taken a lot of detours and switchbacks. I've done a lot of the things that you can do in tech. I started my career many years ago working at an MSP. I've been a DBA. I've been a front- and back-end software developer. I spent a lot of time as a cloud architect and cloud engineer. This was back in the earlier days of AWS. So, I'm working for a large enterprise software company at the time, and we're taking some of their hosted SaaS applications, these large ERP systems they had out of their datacenter, and we're figuring out how to move them into AWS at a time when we didn't have some of the enterprise-ready features that the big cloud providers have today.

So, things like just shared storage were major points of contention for us and we're figuring out what can we bring in from the cloud provider. What are we going to have to go and find with these little kind of fly-by-night vendors that are co-located along the side? That was super-adventurous. We migrated some of the first large-scale ERP workloads from our legacy datacenters to the cloud there. Then somewhere along the way there I got very involved with the cloud community. I actually became one of the first of what are called AWS serverless heroes and became very involved with the serverless community. I helped to organize a conference called Serverlessconf. It was big in that space for a few years. Serverlessconf was run by a company called A Cloud Guru. At that time, I transitioned to doing what I do now, which is I've made this switch from cloud to cloud. I've become a true multi-cloud engineer.

I've gone from the AWS world to the Google Cloud world. I'm lead developer media at Google Cloud, which just means I get to come up with all kinds of creative and unusual ways to educate people about the cloud, help them to get the most out of it. That's something that I've been very passionate about for years. While I remain as technical as I possibly can, my real passion is helping other folks to understand how they can put these building blocks together in ways that are actually helpful for them, and I plan to continue doing that for the foreseeable future.

David Linthicum:
Yeah. You and I share that as a passion. I like teaching people about cloud. Doing podcasts is one thing. It's one media source, but I also like the fact you focus on nontraditional learners that are moving into cloud, because if you think about it, we kind of have a skill shortage in the cloud computing world. We're not able to attract enough people to get the skills that we need to be successful in many organizations, and certainly the enterprises are screaming about this right now. So, it really is the supply chain, your ability to get learning opportunities in the hands of people that we may not consider as potential cloud pros, but a lot of people out there are just looking at it as a way to change and progress their career. They're looking at it when they're starting their career, start moving in this direction versus maybe something they learned formally in college. There just seems to be a pent-up demand out there for training that isn't necessarily focused on computer science whizzes.

Forrest Brazeal:
You're absolutely right about that, Dave. A couple years ago I started a community initiative called The Cloud Resume Challenge, which you can check out at CloudResumeChallenge.dev, that is designed to hopefully help with this exact problem, which is we've got this terrible, terrible supply and demand mismatch going on right now in the industry. It's kind of as old as time. The tech industry in general has not been great about creating formal paths to raise up the next generation of talent.

Unfortunately, I think the problem is getting worse, not better, and there's a couple of big trends right now that are making it harder for let's say nontraditional career changer type of folks to get into tech, as opposed to people that are on like that four-year computer science college degree kind of a mindset.

Those people do tend to have some formalized ways that they can get hired, but for folks that don't fall in that bucket, there are two big things that are working against them right now. Number one is the shift to remote work. I actually like remote work. I work remotely most of the time. But the sad reality is it's tough when you're just starting your career to be able to get the mentorship you need, to be able to be successful when you're not able to just sit close and be next to people that are doing this type of work on a daily basis. The second challenge is the rise of generative AI. We're early on enough in this that it's – we're early days with knowing exactly how this trend is going to play out, but I am talking to hiring managers, to engineers that say, "Hey, I might be able to bring in an intern or an apprentice," that could sit alongside me and maybe take some tasks some off the side of my desk."

But now, I can feed that to a ChatGPT or Google Bard or whatever. I would argue that that would always have been a misunderstanding of how you should be working with a junior engineer. They're not there just to write pieces of code that you feel like are undifferentiated or that are beneath your level. They should be there in some type of a program that allows them to grow up and handle the type of tasks that you handle, but the reality is a lot of folks do think of a junior engineer as someone who takes tasks that are too easy for me, and we have these generative AI tools now that are coming in and seemingly can do some of that. So, it's cut out one more avenue where people have traditionally plugged in a junior engineer. The AI tools, Dave, are not a long-term substitute for having a sustainable talent pipeline. We all intuitively understand that. But at a tactical, practical level, how are we going to go in and make those opportunities? We have to be much more intentional about how we do that.

David Linthicum:
Before we get deeper in this topic, let's talk about your book authoring, the books that you just recently published. They're unique and really kind of made my day when I read through your bio and looked at your body of work. Tell us about that.

Forrest Brazeal:
Yeah. A couple years ago, I did a book with Wiley called *The Read Aloud Cloud*, which is a – it's kind of an illustrated introduction to cloud computing. It's designed for folks from age six to anywhere past that. It does take you through some of the big ideas of cloud computing, from elasticity and high availability, all the way through some security challenges. It even has a little data NAI section at the end. A year or so after that, in conjunction with the Cloud Resume Challenge Project, I did create some guidebooks to help these nontraditional career changers, again, try to figure out: how are you going to make your way and get that first opportunity in an industry where the cards are kind of stacked against you?

We have seen some really cool success stories come out of that. We've seen people who are plumbers, and miners, and loggers, the kind that cut down trees, and architects, the kind that build buildings, and teachers and dentists, just every possible background you can imagine, folks who are returning to the workforce after being a stay-at-home mom out for a period of time. All these folks, they bring an incredible self-starter attitude, a willingness to learn, a motivation. They've got the aptitude to do really, really well in this field. What they really need is someone to say, "Hey, here's the things that you need to focus on so that you don't get way off in the weeds. There's any number of things that you could be learning, but here's the things that are actually going to be interesting to a hiring manager. Then here are some projects that you can build that actually help you do that." The Cloud Resume Challenge Project, if you haven't seen it, it takes you through 16 steps.

It's kind of a spec. At the end of that time, you'll have built a personal website, but you'll have done it in some very specific ways. You'll have used source control. You'll have written code in at least two programming languages. You'll have built a small CI/CD pipeline. You'll have built a little serverless API. You'll have done some security, some networking, some authentication. If you can complete all those steps, and a lot of people who try the Cloud Resume Challenge can complete them because it doesn't tell you, "Click this, click that," in order to do it. You've got to open the Google tabs and go down the rabbit holes. Basically, you have to learn like an engineer.

That's how we actually learn on the job. But if you can complete all that, you will find that you've covered things not only that a lot of people coming out of four-year computer science programs haven't done, but also a lot of professional software engineers haven't put all those skills together in one place either. That does tend to be very interesting to hiring managers because it's real, and these folks that have been able to go in and get hired off of the strength of

this project have typically seen great success, and they've become beloved and relied on their teams pretty quickly. There's not some magic in the Cloud Resume Challenge.

Again, these people are brilliant. They are very, very self-motivated. They have a huge appetite for learning. They're going to succeed wherever they go. They just need someone to give them that first chance. So, Dave, when I created the Cloud Resume Challenge, I didn't do it really even to help these learners because they're great. There's nothing I can give them that they're not going to get somewhere else. It was really to kind of show hiring managers that there is an actual, incredible, untapped well of talent out here.

My friend Joe Emison, who runs an insurance startup called Branch, has done a great job of bringing junior engineers on. He refers to this type of talent as undervalued talent, meaning that it's not being given its due in the marketplace right now. I mean you have the opportunity as a hiring manager to bring on these folks who are going to be incredibly loyal to you, you have a chance to bring them up in your company culture, and to grow them in a way that you would want folks on your team to be. That just takes commitment. It takes vision. It takes buy-in from your senior engineers.

Somebody has got to sit alongside and mentor these people, so you've got to have incentive structures for your senior engineers to do that. It takes buy-in from your executives that you're going to carry some folks on your payroll for a bit who are learning. This is how you create sustainable talent pipelines. I go out and speak about this, and just beat the drum and hope that somebody is listening. We're going to need 40 million more cloud engineers in the industry by the end of this decade. It's incumbent on us to go out and make an industry where those folks can feel at home.

David Linthicum:
It's absolutely imperative. We sponsor a number of things here at Deloitte, including veterans who are reentering the workforce and getting in the training they need, and people who are out of the workforce say 20 years, went off and raised a family, men and women, and your ability to kind of get them reacclimated into having current and up-to-date skills and they can examine it. By the way, the ROI from those things is a thousand percent. It's just a win-win for everybody who sponsors that stuff. So, we have to manufacture an incentive and really kind of look at the supply chain of expertise if we're going to make cloud computing work, and we need to get more people skilled up in how to make this happen. So, speaking of that, the evolution that's going on around us right now, the generative AI stuff, how does that affect the learning process? What are some of the enterprise challenges that you see that are facing organizations that are trying to figure out how to make this work for them?

Forrest Brazeal:
That's such an interesting question, Dave. A couple months ago I was out speaking at the DevOps Enterprise Summit in Amsterdam, which is the wonderful, wonderful event put on by Gene Kim and his group at IT Revolution. If you've never been to that event, they run it in Europe, they run it in the US. I highly recommend anyone listening to check it out when you get a chance. But I was just speaking with a bunch of IT leaders there from around the world. Everyone was excited about these generative AI tools, to a degree that I feel like I don't see from IT decision makers a lot. They tend to be – not jaded, that's an unfair way to say it, but they've seen a lot of technologies come and go.

They're not quick to be excited about any brand-new flash in the pan thing. But every single person I talked to, there were two really key things that came out. Number one was, "Yeah, this is real. I plugged a task into this generative AI tool that would have taken me or one of my engineers several days or a week to rough out, and it just handed me what I needed. It's impossible to deny the potential of that for developer velocity." The second comment that they would always follow that up with was, "But we can't figure out how to use this at an enterprise context because it freaks our legal people out. We don't have a good sense of where this data is going when we put it into the chat window," and all this kind of thing.

So, I think from an enterprise standpoint, you're going to see these enterprise AI tools have to get a lot more specific and a lot more transparent about exactly what your contract is with them and what you're doing with their data. I've seen Google do some good things here. I'm sure there's going to be lots more good things to come around the industry, but I would be watching for when the banks and the heavily regulated industries feel like they're clear to start using this generative AI. That's when you're going to start seeing a lot of things happen. From a developer experience standpoint, one of the things we see is the better you already are as a developer, the more you get out of generative AI. Now here we're talking about coda systems.

These tools, they can't tell you what to prompt. So, the better your prompt is, the better the code that you're going to get out of it, and it seems to work better on a microlevel rather than a macrolevel. So, you've got to have some sense of what the application is you're trying to build, what the best way is to solve the problem, then this thing can fill in the blanks for you. So, I always tell the learners that I'm working with that right now, it does appear that the best way to get better at coding with generative AI is to get better at coding without generative AI. That sounds paradoxical, but that's just the reality of these tools.

We talk about context windows. These generative AI models have what's called a context window, which is some number of characters that you can put in to enrich the base model and help it know what kind of response you want. Then you as a human being, as a professional with some number of years of experience, you have a personal context window that's made up of just your general knowledge, as well as your understanding of what your company's needs are. Then it's up to you to continue to expand that on your own, because that combined with these LLMs, these generative AI models, can yield some things that really do help speed you up and provide you with some useful support. Don't neglect your own growth, but that's where the value of this stuff really kicks in.

David Linthicum:
Yeah. I think you kind of hit the nail on the head in terms of the value that this stuff has. Early on people were – you know, I'm also a cloud architect, and they were saying this stuff can maybe do cloud architecture. Well, I don't think it really can, but it can tell you if you've done a good job at creating an architecture. In other words, look at the optimization and the efficiency of that because we know how to ask the right questions. So, if becomes a better tool than we have now to get to a better state, but we still need the innovation. We still need the creativity to really move things forward. So, what are you guys doing at Google in terms of this AI revolution?

Forrest Brazeal:
You're seeing this model that Google has created called PaLM. It just shows up all over the Google Cloud console, providing helping hands for people and that type of thing. There's a security version of it called Sec-PaLM that's starting to pop up in a bunch of our security tools such as Chronicle in Mandiant,

the company that we acquired a year or so ago. I think you're starting to see this in other cloud providers as well. I've been working with cloud for a long time, Dave, and I've always had the belief that a great cloud console – so here I'm talking about the UI experience.

The value of that is in discovery and learning, because when you're a professional and you're shipping code every day, you don't want to be making a lot of what we call ClickOps type configurations in the console. You want to have things automated. You want to be using CloudFormation in the AWS world or what have you, to make sure you're automating all the changes that you can make. But the clicking around the console is great for understanding how a service works. It's great for identifying, hopefully, where something goes wrong, and some of the consoles are better than others at this. Where we're starting to see the generative AI tools really helping there is all of sudden now you can get potentially an AI summarization of where you have an issue.

I was looking the other day at some things that have rolled out in GKE, which is Google Kubernetes Engine. They have these tools now where you can pop in and say, "Okay. I have got something that's broken. There's an error in my logs. Can you tell me what the – explain this problem to me, and then give me some common steps to remediate that. Show me how to fix it." It just kind of walks you through what a typical fix for this would be. Then the best part of it is then there's a little button at the bottom that says, "Okay. Now generation the automation for me, so that I can fix this automatically next time. I can remediate this issue without having to break the glass and go in manually." That's what a great cloud console should be doing, Dave, is providing you with that type of discovery, awareness, and learning, and then transitioning you on to where you can actually have that automated in the future, and I think generative AI can be a big help there.

David Linthicum:
It's going to be huge. It's going to be a great tool. It's going to provide us with the assets and tools we need to become better at what we do as humans, and I think that's what technology needs to do. One of the things, I think, in trying to understand the roles of different technologies and different platforms, we're looking at all assets within an enterprise as being of value. So, we're not talking about just moving everything into a public cloud provider. We're talking about enabling the edge-based systems, all the IoT and devices and mobile systems, and certainly legacy systems, which in many instances may be servers you bought last week because that's becoming cheaper and better focus of what we do. So, what's your approach to using legacy systems or traditional systems, traditional platforms as an asset in the whole architecture realm?

Forrest Brazeal:
There's an amazing book by Marianne Bellotti that came out a year or so ago from No Starch Publishing called *Kill It with Fire*. Marianne used to work for the US Digital Service, which they take some of the gnarliest, most regulated legacy government type systems and figure out how to modernize them. She has a wonderful, wonderful perspective on this whole issue of: what is the role of legacy in a tech ecosystem? I highly recommend reading that book. She talks a lot about, and a lot of us talk about this idea, that legacy does not have to be a bad word. Legacy sometimes is just another word for, "Hey, this is a thing that worked. It didn't stick around forever because it didn't work. It stuck around because it fulfilled some useful function."

Another thing that I've talked to her about that comes back to me a lot is sometimes, if you show up at a new job and you look at this system that's 10, 15 or more years old, or, heck, maybe it's still running on a mainframe or let's say it's this database system that is, I don't know, it's got this custom thing they built on top of it to do replication, and you look at it and say, "Why would anyone build that? That's terrible. I could do this so much better today with some feature that's just come out in Postgres. This is just a terrible, terrible hacked together thing." But you have to remember that the tools that you want to use now did not exist at the time that this older system was built. These engineers were building it the best they could using the resources that they had, and if you had to go back in time and build it yourself the same way with their tools, you might have done the exact same thing.

So, I think it's valid to kind of celebrate and acknowledge that legacy and say, "Hey, this was state-of-the-art at the time. It fulfilled a real need. Potentially it's made a lot of money for us." But it is then still incumbent on us, once we take the time to understand that, then to figure out where we can improve it, where we can make meaningful improvements, not just bringing in the new and shiny stuff for the sake of it, but to understand where does the new technology actually add value. Heck, mainframes, they are not slow. A mainframe is just a computer the size of a refrigerator. It's just not going to be slow.

David Linthicum:
Legacy systems aren't always mainframes. They're traditional LAMP stacks, things like that that are not that old.

Forrest Brazeal:
That's exactly right. We talk about Chesterton's Fence. If you haven't heard of this concept, it goes back to the author G.K. Chesterton, who talked about seeing a fence stretched across a field that did not seem to have any purpose and wanted to tear it down, but then went and talked to the farmer and learned that there was a very specific reason that this fence was there and had something to do with the say the livestock was penned in. When we talk about Chesterton's Fence, we say don't get rid of a system until you understand why it was put there in the first place. That's where I would start with any of these legacy assets. Rather than just throwing them out the window, make sure you understand what they do, why they have value, and then you can build on them and add even more.

David Linthicum:
Yeah. Everything is in play, ubiquity of processing and the ability to have – you know, leave it where it lies. Also, the business case for actually changing this stuff, we're starting to question that stuff and that's for a good reason. We don't need to move everything to a particular cloud platform. This has to be looking at the optimization of the platform and everything should be in play, including legacy systems, systems that are 30 years old or 30 days old, it really doesn't matter. Which ones are more efficient and are going to bring more value back to the business? So, multi-cloud is a huge thing. We did a survey. Obviously, it's been growing over the last five years and last year. About 95 percent of the people out there are doing multi-cloud, depending on what survey you listen to. So, what are the challenges there in getting the skillsets aligned to doing that well?

Forrest Brazeal:
I tend to think that multi-cloud at the workload level is not really something we see. Here I'm talking about a single application that's deliberately built to stretch across multiple clouds. That's a dangerous game to play. I recognize there may be very specific use cases where it makes sense, but if you're not sure if you have one of those use cases you probably don't. Where we do see multi-cloud a lot is more at the organizational level. So, past a certain level of organizational complexity, at this point in the history of IT, you are likely going to have services either from multiple cloud providers floating around in your environment, or just let's even say multiple SaaS vendors.

At some point you're just going to have this mass of different tools. So, even if every individual application is just running on one cloud, at some point your engineers have to have some facility with multiple clouds. This gets more important I think as you get more advanced, more senior in your career. So, if you want to move into IT leadership – again, if every application you're building is just on one cloud, you are going to need to have a good enough situational awareness of the whole landscape that you can make appropriate build versus buy decisions, appropriate decisions about, like what I said with Chesterton's Fence, what to get rid of, what to consolidate.

If you really, really bet your career on just one cloud provider and that's all you learn and you don't expand beyond that, then I think you're going to limit your opportunities. So, when I work with these career changers who are trying to build a good resume, I do tell them focus on learning one cloud provider. Make that your native language, if you will, the thing that you feel most comfortable with, because let's be real, if you go into a job interview with no experience, having no experience on two clouds is not going to be more impressive than having no experience on one cloud. Once you get into that job, I would still recommend having one cloud that you feel like you speak natively. Then maybe you'll get certified on a second cloud, even if you're not building on it a lot day-to-day, just to expand your awareness of the landscape. You're never going to be sorry that you have that knowledge and understanding. So, I think multi-cloud engineer is a reasonable thing to aspire to.

David Linthicum:
Yeah. That's very valuable. They're tough to find. I made this case in my blog a number of times and on this podcast as well. We really need generalists, people who understand a little about a lot of stuff and understand how to configure this stuff. Again, you have to have some expertise and some deep knowledge in particular platforms, but man, we've just got to have everybody consider everything. We just talked about that, your ability to look at the entire landscape of technology and making sure we're assembling the right technology, including cloud brands, and what are the trade-offs in the different cloud brands. Who is doing storage well and not so well? What does it cost? All these sorts of things are very tough decisions that I think we need to make in light of getting a multi-cloud strategy running. So, where do you go to learn this stuff?

Forrest Brazeal:
I already mentioned Marianne Bellotti's book, *Kill It with Fire*, which is my – *[crosstalk]* – for anyone trying to think about how to manage legacy systems. If you're thinking about AI and how to learn there, I do always recommend Google Cloud Skills Boost. They just released a tremendous generative AI track. It's not necessarily just Google Cloud stuff, but they dig deep into input/outputs and what vector embeddings are, just bread and butter terms that you're going to want to know as you start to look at how these things fit in. Again, if you are a nontraditional learner and heading to cloud, I do recommend checking out CloudResumeChallenge.dev. The challenge is free for anyone to try. There's a bunch of other resources at that site as well. It is multi-cloud. You can choose AWS, Azure, or Google Cloud tracks to go through that, and I hope that you will.

David Linthicum:
I love that you guys are not just looking at your own stuff but promoting a wide variety of technologies that you need to understand to be successful for actually creating solutions. So, where can the listeners go to find out more about you on the Web and follow you?

Forrest Brazeal:
That's a great question. You can always find me on LinkedIn, or the platform formerly known as Twitter at @ForrestBrazeal. You can also find out more about me at ForrestBrazeal.com. I'm easy to contact there if you have any further questions.

David Linthicum:
An absolutely important conversation that we had. This is where the battlefield is right now. It's not necessarily trying to figure out how to make different technologies work. We're good at that. We know how to build solutions out of different technologies, but the ability to do it right and in an optimal state is kind of where we're missing out. We don't have enough skills around for people who are making this decision, and we've got to have those skills around, or else enterprises are going to find they're going to be absolutely behind the eight ball. So, if you enjoyed this podcast, make sure to like us, rate us, and subscribe. You can also check out our past episodes, including those hosted by my good friend, Mike Kavis. Find out more at DeloitteCloudPodcast.com, all one word. If you'd like to contact me directly, you can e-mail me at dlinthicum@deloitte.com. So, until next time, best of luck on your cloud journey. You guys stay safe. Cheers. Bye.