

Dan Ciruli: The more we realize the benefits, apps can do this, apps can do that, and the more use cases that over time we have more data, we're gathering more video, audio data, they just enable more things. And so, I have no doubt that as we, you know, data centers become more powerful, the things in our pockets become more powerful, the computers at the edge become more powerful, people have more and more ideas, we'll continue to write more applications than ever, you know, ten years from now we'll be blown away at where, you know, container-based applications are running and what those things are accomplishing, and all of that means, yes, more apps.

[Related: [AI and Cloud Native Spark Explosion of New Apps](#)]

I'm not the first person to make this analogy, but watching AI right now reminds me of late 90s, mid to late 90s internet, and just kind of realizing we don't know all the ways this technology is going to change everything, but it's clear it's going to change everything, right? And that's where AI is right now. It is, you know, let's say it's mid 90s, because it's in that time when you just knew it was going to be incredible, e-commerce hadn't been invented yet, I mean, companies really weren't even figuring out how to market on the internet yet, they were barely putting up websites and taking out, you know, radio ads to tell you to go visit their website, but you just knew it was going to transform, it's going to transform technology and in turn society. AI is at that, you know, it's very early on, but you can already tell there is no doubt, and that in itself will mean lots more apps, right, and AI will be, it'll be in the data center, it'll be in the cloud, it'll be at the edge, it'll be on your phone, it'll be on your watch, it's going to be everywhere, and all of that will be transformative.

[Related: [AI, Cloud Native and Hybrid Cloud Fuse to Run Apps and Data Anywhere](#)]

Virtually all apps that are being written today are running on cloud native, that's true of all the AI apps, right? All of the big companies, they're running all of that AI stuff on Kubernetes, for sure, you can't, I don't know of anybody who's really writing new AI based apps, running those models in VMs, that's all happening in Kubernetes. So in that sense, AI is built on cloud native and is an example of, it's a new application, by the way, and it's one that companies are starting to run in the cloud and just like their other workloads, they're going to want to run on prem, for sure. Your provisioning timelines are different, you've got to buy that hardware and it's expensive, however, that's where your data is, you don't want to send all of your most important data that you need to train models on up into the cloud, so it's another one of those. On the other hand, in a sense they're orthogonal too, because ultimately when you're consuming an application, you don't care if it's running on cloud native, you don't know if it's running on cloud native, when I open up my bank app and I check my balance, that might be running Kubernetes on the background, I don't know, right? Eventually it might be using AI in the background, I don't necessarily know. So in a sense they're orthogonal. One other thing is I think that when we talk about gen AI, it is one of the things that is going to solve the usability problems. We hear a lot about skills gaps and cloud native, and I can't hire people who understand Kubernetes well enough to run it

and then be asked to do that. To me that's not a skills gap, it's a technology gap. The problem there is that Kubernetes is too hard to use, and AI is going to be one of the things that solves that. And when we get to what we call intelligent infrastructure, or invisible infrastructure, infrastructure that is using ML to understand, oh, here's a problem that maybe hasn't happened yet, but is about to happen, and here's what we do to solve it. And you can imagine a day, and we're already seeing this, we've already got AI chatbots built into our software that allow you to say, in plain English, why am I seeing this crash loop error, and what can I do about it? And have AI help to solve that problem. So AI will help us bridge that skills gap by making the technology easier to use.

[Related: [Making IT Infrastructure AI Ready](#)]

Almost all application development now is happening in containers, in Kubernetes. What happened over the last 10 years, Kubernetes is having its 10 year anniversary, but the first three or four years was very experimental. So more over the last five years was, yes, organizations were adopting it, but they were adopting it mostly in the cloud. Because the cloud vendors, Google, where I used to work, and Amazon, and Azure, were all making it very easy to get a Kubernetes cluster. And so for many organizations, for the last five years, new development efforts were happening in the cloud, and they were happening in Kubernetes, everything on-prem was still running in VM. There's two things that are happening. One is that now even some of the stuff that's being written and run on-prem, they're saying, well, this is new development now, we want to do it in this modern way. And two, companies are rethinking the economics of cloud computing, and realizing that essentially renting is more expensive than buying, and doing all of your stuff in the cloud is very expensive. So they're saying, hey, we like these benefits of running in the cloud, we like this idea of velocity, we can get changes into production faster, but we need to be able to do that on-prem too. And so increasingly, IT departments are being asked, hey, how can you give us Kubernetes clusters? You're the one who can give us VMs on-prem, how can you give us Kubernetes on-prem?

[Related: [Building a GenAI App to Improve Customer Support](#)]

Cloud Native started, I'll say it started at Google, although that's not entirely fair. There was a bunch of people at a bunch of different companies that were trying to figure out how to ship software faster, how to make it from the time someone had an idea, put that in code, put that into production, could reduce the amount of time to do that. And at the same time, also make sure that when you did, it would be scalable enough to use at some of these big companies. And Cloud Native is what we use now to describe what they were developing back then, which was a, it's essentially, it's more layers of abstraction between the developer and the machine that the code is running on. Because traditionally, getting a code to a piece of machine involved server installs, and server installs are notoriously complicated and complex, and it would take a long time to get something just installed and running. And with Cloud Native-style architectures, you could put together a pipeline where a developer wrote some code, they checked it into source code, pushed it to the

source code repository, and then it kind of automatically and automagically got deployed to a server somewhere, no one cares where, no one's ever going to SSH into that server. So Cloud, and then over time, a bunch of this technology got open sourced or created again in open source, and now when we refer to Cloud Native, we refer to these projects like Kubernetes, that ultimately are aimed at this, and aimed at how can we make it faster for developers to get code into production.

[Related: [AI and Cloud Native Alchemize the Future of Enterprise IT](#)]

In the beginning of cloud computing, we would talk about hybrid, we would talk about hybrid as a temporary state, and we really thought that hybrid was going to be how people ran until all of their stuff was in the cloud. And I think now where we are as an industry has realized that hybrid is not a temporary state, it's not a transitional state, this is how most organizations will be running for the next decade, right, for the rest of my career probably, is that, oh no, we will be absolutely using the cloud appropriately, but we are still, we're investing in data centers, we're realizing for reasons of cost efficiency, for sometimes data sovereignty, sometimes just control, there's stuff we're going to want to run here. And so solving hybrid, not as how do we have this temporary bridge, but how do we put in place something that permanently lets us use the cloud and the data center appropriately, taking into account costs, taking into account where the data is, and taking into account sovereignty rules. So that's what's different about hybrid and multi-cloud today, is that companies are realizing this is a permanent state, and how do we make sure that we're putting in place the right structure so that we can take advantage of each of them for what they're good for.