Capgemini

CR045

So, what is Cloud Native anyway? With Massimo Ferrari, RedHat

CLOUD REALITIES
LISTEN NOW
Capgemini's Cloud Realities podcast explores the exciting realities of today and tomorrow that can be unleashed by cloud.

CR045

So, what is Cloud Native anyway? With Massimo Ferrari, RedHat

Disclaimer: Please be aware that this transcript from the Cloud Realities podcast has been automatically generated, so errors may occur.

[00:00:00] I have to ask though, do you sit and observe the Roomba just cleaning as an event and find yourself mesmerized by what it's doing? I mean, who doesn't?

Welcome to Cloud Realities, a conversation show exploring the practical and exciting alternate realities that can be unleashed through cloud driven transformation. I'm David Chapman. I'm Sjoukje Zaal, and I'm Rob Kernahan.

And this week we're going to bring a couple of worlds together which we think have got a great deal to do with each other.

We're going to talk about open source and then we're going to talk about cloud native. And how those two things are very synergistic and how they drive value in an organization. But before we get to that Walked into the office the other day and our office where We're on the fifth floor [00:01:00] generally, so we're quite high up and walked into the lobby I could hear rob's voice and i'm like my god, he must be talking loudly So I get into the lift go up five floors And rob was standing on his wheelie suitcase, and he just finished Sort of talking to the office about something.

Now the interesting thing about this is like when MPs leave their office in UK government, they get their red briefcases, which they carry their ministerial briefing. And they put them on the floor and they talk and talk to the office and say, thank you for, you know, thank you for supporting me during my tenure and that kind of thing.

And Rob, who does a lot of work in public sector had obviously seen this wheelie bag on the floor. He was just stepping down. I was like, Rob, what's confusing you this week? A good creativity on that one, Dave. I'll give you top points. So it was it's a conversation that's heating up actually. It's about technology singularity, which I was getting confused about.

So, I don't know if you know the theory, but John von Neumann thought about this in 50s, [00:02:00] which is technology keeps ever increasing in cycles. And you know, the sort of rate of pace of change builds and builds and builds. And essentially the theory goes that at some point. Technology will become super intelligent, will not need the human, will take over itself.

And then you get this incredible pace of change, 10, 000 years of technology development in a day type thing. Yep. It's curve just keeps rising. That's the theory for the human. What does that mean? Either means one will live in bliss for the end of days or two, the technology will decide we're not very good and we might have to scarper fast.

So that's line one, and everybody focuses on that one with Gen AI. However, alternative theory, which is before the singularity occurs, technology and humans will fuse and create a sort of augmented version of ourselves. And then we will become super intelligent and the human element will sustain. Is that like a pilot, Rob?

Yeah, well, it's a bit beyond co pilot day, but let's go with that one. Yeah, but it's that fusion, right? And I can't work out if it's going to be the [00:03:00] purely logical technology that takes over and humans get left behind or whether there will be this fusion and actually something different will emerge in the future.

And the sort of timelines on this are that people saying it's not that far away. Maybe it's bad tech predictions. I don't know. But there's these two pathways tomorrow. And I don't know which one we're going to take. So I am generally confused. And it's reliant, isn't it? I think on the human getting out of the way or being pushed out of the way by a technology that is now sort of self sustaining and self developing in some way.

Is that right? Is that basically at the heart of both of those? I mean, we keep hearing stories about people who've lost their vision and we. put technology in their eyes and they can see again, not as well as the fidelity we know today. But there is this augmentation that's occurring where the human and technology is, you know, a very small level starting to fuse.

So that's the alternative pathway. So many overly focus on the tech will just run away with itself. You know, it's the terminator vision maybe who knows. But the other one is that we'll fuse differently and we'll become super [00:04:00] intelligent. It'll be the thing that allows us to get to a higher level.

Isn't the subtle difference in that though, that the. iN the example you just gave of like using technology to solve or augment our biological issues is humans making decisions to leverage technology in a way that suits them isn't some of what you're talking about around the singularity, like irreversible of that, which is like we're no longer in the driving seat.

I actually saw a funnily complete coincidence, but I was watching a little video today of Jack Ma talking about artificial intelligence sat on stage next to Elon Musk. And he was making an argument whether you agree with it or not, but he was making an argument. much like your second route where the masters of the domain, where, you know, where we're making the decisions that are leveraging technology and he felt that will always be the way, like he had an implicit assumption in his thesis that will always be the way.

And Musk sitting opposite him was, is effectively eye rolling his [00:05:00] way through the conversation and saying that actually. You know, you're constrained effectively. So I think for me, that's the dynamic in the middle of what you're saying is like, who's in the driving seat. Yeah, absolutely. It's the consciousness that a purely logical AI, I don't think would be much fun to live around.

the human effect and the consciousness and things. And how does that sustain? Can't predict what technology will become or will replicate the human effect. Who knows? But there is this confusion about the pathway and will we remain in control or will we have to be push aside and something else will take control.

Mason Well, that's a good one and a hard one to unpick. I think we are just going to have to keep watching the technology evolve. But I think I do subscribe to. We might know the answer to that, you know, within our, certainly within our lifetimes, you know, the next, you know, the next 20 years might be even, you know, we might still be making shows when that sort of stuff's happening, given the rate of innovation that's going on at the moment.

Let's see what 2024 [00:06:00] brings, but let's get onto today's subject. Joining us for today's conversation is Distinguished Product Manager from Red Hat, Massimo Ferrari. Massimo, good to see you. Thanks for taking the time to join us on the show today. Do you just want to say hello and introduce yourself?

Well, thank you for having me first and foremost. And yes, I'm Massimo Ferrari. I'm one of the product managers in the Ansible Automation Platform team here at Red Hat. So we effectively create our automation product, which Piggybacking on the conversation you were having before is one of those technologies that we are still trying to augment with artificial intelligence and everything, but to support humans to do more as opposed to replace humans.

Okay, well, let's dig into Red Hat then. I'm sure everybody's aware of it, but Massimo, maybe share us a bit of history and more interestingly, perhaps Red Hat's philosophy in the open source world. So spoiler alert, I don't come from the open source world. So I came in from business consulting, which made me, it kind of forced me to study a little bit what that special shine [00:07:00] was back in the days.

But basically that still is the largest open source or enterprise open source company out there, which means that basically, Roomba decided to start now. What a good timing. So going back, was that was that, what was that device that just A Roomba. A Roomba. The Bot Hoover. Is that what it was?

Yeah. Maybe you could bring him onto the show if you want. I think it could replace you, Dave. Just the whole editorial is just beeps and bops. One of the best. Purchases that I did in the past five years. That saves me so much time. It is a terrific thing. Well, maybe we'll come back to that a bit later and what you're excited about doing next.

Filling my house with Roombas. I have to ask though, do you sit and observe the Roomba just cleaning as an event and find yourself mesmerized by what it's doing? I mean, who doesn't? I think it's the whole thing. It takes four hours and it will stay adjustable for four hours. Well, let's see if he [00:08:00] cleans over there.

He cleaned over there! Well, let's get back to Red Hat. So you just joined Red Hat as a, I mean, obviously we can talk about Rumba for the next, it'll be a different episode than what we've got in mind. So I joined Red Hat nine years ago and I joined Red Hat from business consulting. So a very different world.

I had, you know, the classic outside view of open source. Plus 10 years ago, open source was already very widely accepted. But there has been A big transformation in the past 10 years that we can discuss later. But basically what ADAT is today is still the largest enterprise open source company in the world.

Which is fascinating, especially for me being a product person that there is this very strong connection with what we call the community. So basically the contributors out there, the people that look at the open source code and you know, provide any sort of contribution, which sometimes it gets conflated.

Into all day, write a piece of code and a piece of code will end up in the product is not always the case. People review the code from a security standpoint, from a viability standpoint, from a quality [00:09:00] standpoint. Sometimes they write documentation. Sometimes they just provide feedback and all of that is.

And I will give you a very unpopular view of the things, but it's basically having a huge and massive R& D department that is out there in the testing environment, which is a huge advantage for companies. And that's basically the reason why everyone jumped into open source in recent times. You think about that.

That's a lot of what a lot of companies have been doing including our business there, the IBM, big open source contributor but a lot of companies that historically never done that, they started doing that. That said the being an open source first company has requirements. Which, if you will, is the philosophy part.

So from a product perspective, what we do is that we take from the community, all these feedbacks, all these contributions that they do, and then we polish them, repackage them and make them enterprise ready, which is the whole thing. Because you know, if you're an enterprise, you have a whole lot of red tape to go through to adopt [00:10:00] software and you have requirements that individual users tend not to have.

So we package everything. We do bug fixes. We do all the standard software lifecycle things that you have to do. We ship that to customers. Great. And so far so good. And then what we do is that we take all the work that we've been doing and we put it back. into the community.

So basically we really release everything to the community.

So the community starts from this new fresh batch of code that we just polished and the cycle starts again. I think this still exists out there like a suspicion that if you use open source it is somehow less reliable or less stable code than say You're less transparent code or, you know, traditionally non open source, but that's a complete myth, right?

Because products that are based on open source code go through the same robust testing and release cycles as traditional code, right? And actually, if anything, you get more developer innovation as a result of the wider community. Well, innovation is part of it. The truth is that again, I like the [00:11:00] innovation part of the truth is that the more you go towards.

A lot of people think about open source projects and they think about operating system, they think about Realm, CentOS, et cetera, et cetera. So they think about something that has a very user centric component. I will be using that operating system. And as a user of that operating system, I have the full capacity of understanding what it does and how it does it.

And then there is an enterprise part of it, which is like, Oh, you will deploy 20, 000 of these operating systems. The truth is that open source now is so pervasive that you get into the enterprise software space. Now, if you like what we do automation and automation is, yes, there is a tactical, what we call, honestly, we call selfish automation.

There is a part of that, but the requirements that you have for the automation are probably 10 percent of what an enterprise would do. So what it, what that means is that the contribution of the innovation you can get from the community is much lower. Then what you can get on a [00:12:00] more consumer grade product.

So the truth is that there has been a transformation in adoption of open source. More and more software is based on open source, which means that now more enterprise software is based on open source. At that point, what you get from the community, honestly, is testing and review. Which means if you compare with closed source, you get a lot more testing.

A lot more, you know, breaking the code and checking if things are in line. And after that's where we jump in. We jump in and do the traditional testing and packaging, right? That's what we do. I'm a massive fan of open source. I think community eyes on gives a stronger, more robust thing. But then you do get Issues with security like log four J's very memorable from a few.

Was it 18 months ago? A massive vulnerability was spotted zero day type thing. And then because it was so pervasive and everybody uses it because it is so good because the whole community has developed over years. It's a fantastic software library. Suddenly everybody had a lot of work to recover against the patching and the upgrade cycle.

So I suppose that's the example [00:13:00] of the double edged sword, which is you. When it is so popular, because it is so good because the community developed it. If something does go wrong, it has a very wide impact. And I think that didn't damage open source for the enterprise mindset, but it did put a tiny little dent in the, should we be using this stuff?

But then you go back to what's the option, what's the alternative you go, maybe this is just the thing you have to deal with as part of the, you know, that. That big life cycle that you just talked about. Well, the truth is that from my point of view and I can say that because it wasn't my team doing that But it's a praise at most to the support team in Aradat is that if anything it kind of created a better Venue or a better perception of value of what a company like Aradat does because when that happened our sec ops and security teams jumped in and they patched the entire thing in, you know, zero time and then the deployed everything to customers first, obviously, and then it came back to the community, which means that having look at that like a sort of check [00:14:00] and balances.

So you have the community that goes and grows and creates popular technology and that's something that is started at scale. And then something very specific and very sensitive happens. Then you have a company you. with resources dedicated that jumps in and fix that out right now. The question that I would ask is, like, how long would have taken to understand that we have the vulnerability in closed source?

Yeah, I know. Indeed. Absolutely. Let's pivot the conversation slightly then and just bring in the other subject that we want to bring into the discussion, which is cloud native. Rob, maybe you could just give us a quick definition of cloud native first, in terms of how we see it, and then let's explore how the two things come together.

Sorry, a definition of cloud? Have you seen the debates going on in today? Cloud native, actually. I didn't say it was an easy question, Rob. Thanks, Dave. I'll help you out with it if you don't get it right. So don't worry. That's all right. So it is a test. It is a test. Yeah, of course it is. It's a level four test.

[00:15:00] Oh, yeah. Exactly, Shauk. Exactly. So, you go back to the epochs of computing. We've been, our operating model is constrained by the technology used. So mainframe client server, the sort of big core, little core type approaches that we've been through. Cloud allows us to decouple our resources, disaggregate.

So it's the loosely coupled, highly cohesive approach to the world. And so what that means is we can fundamentally change our operating model over the top. We can change the way we bring software to life. We can change in smaller cycles. We can do less risky releases. So you can change your ethos. And so for me, cloud native isn't so much about the technology.

The technology has allowed it to become what it is. It's about the way we operate across the whole software lifecycle. Yeah. It's that I can behave differently as a human being when I think about pushing value through software and it's that it's not moving to the new cloud platforms, but it is absolutely changing your whole ethos.

Above that. So it's a people thing [00:16:00] and it's a ways of working thing powered by the technology that allows it. So if I was to sort of put it into a bracket, that's a very complicated thing to do. And I'm sure there's many other viewpoints out there. That's how I tend to think about it. I would hold it exactly the same in the sense of.

The real big compute eras, so mainframe to client server to cloud continually removed boundaries from the organization and allowed the organization to respond and change and become something different as a result of the enabling technology. I think it remains to be seen about whether AI, for example, becomes a.

a fourth era of transformational change or whether it's contained within the cloud era. It's a big question for me at the moment. We probably will be subject for some of our shows next year. I think exploring what significant AI change actually looks like, but returning to the notion of open source and cloud native then.

So Massimo in your head, how do those two worlds come together for you? Like it feels like there's a decent amount of [00:17:00] synergy between cloud native technologies and open source thinking. Does that resonate to you? It does. And I agree with what Rolf said. There are different angles. Again, this has been a debate going on for, I don't know, 10 years, what the definition of cloud native is.

I think there is a nice, you took a nice, let's say, lateral point of view. Non controversial. Let's go with non controversial. It's the Switzerland, it's the Switzerland response to that. There are two points of views of how open source fits into this picture, which is, again, Part of that is as an enabling force.

So let's put this way. The reason why cloud is a consumption model is an enabler for people to change the mindset of the way they operate is because at the end of the day, what you're doing is commoditizing the infrastructure layer. There was a say like five, six years ago, it's like, Oh, people at some point, they will start with the very same infrastructure.

You will remove the competitive advantage [00:18:00] of being able to spend more money in your infrastructure stock. Now, we all know that it's not the reality as it is today, but let's imagine this ideal world where you're a new company and you just jump on the very, you know, you start from the very same place like anyone else.

Now, what happens is that you shift the focus of what you are doing on, now it's the software part as in, Developers that create applications that deliver services. So people think more as a service delivery model than they think about the Apple technology. But what happens is that all this stock that you don't see anymore, how is that managed?

How secure is that? How reliable is that? How has it been built? Say, what is the accountability mechanism behind that? And at this point, what in AWS of the world, or, you know, even Azure to a certain extent, they own.

So what happens is that you created a, if you want a slice of the market, and right now the open source market is in contribution is more and more [00:19:00] professionalized, which means you have fewer, you know, passionate people to just contribute for the sake of contributing, and you have more paid people coming from different companies and working in that ecosystem.

So what do you get is this check and balances mechanism, exactly the one that we described before. For the underlying infrastructure, so the idea is that there is a peer review and a common paradigm pattern, architectural pattern adoption in creation that is happening in the community, which gets translated then into the public cloud.

So at some point. You still won't care anymore about what is your enabling part underneath, but at the same time, there will be some sort of review and consensus and market motion happening somewhere else, right? So that's still active is just instead of being in your company or instead of being in that vendor, it happens in the community.

So I guess what I'm interested in now is we brought those two things together is what the benefits are. So what does it give? Two different communities in my head. What does it give to the tech community in an organization and allow [00:20:00] differently there? And then subsequently, what does that give to the business community and frankly, the wider organization and business that surrounds this enabling technology?

How do you hold the advantages and disadvantages for those two groups? Well, that goes to the second part of where open source influences cloud adoption, which is the fact that, as Rob said, the more people change the way they operate, So the more you become more flexible, the more you become more service oriented, the more you use software to express what you do, the more you have to adopt a different.

Let's say working model a lot of that, you know, there is this whole say that we keep telling infrastructure people, which is, Oh, you need to start thinking more like a developer. What that means is like, Oh, you have these part of that is using, you know, pipeline for checks and automate as much as you can.

Fair enough. Part of that is, Oh, you should start working in a community like kind of scenario. So at that point you, the consumer of the infrastructure stock that now is completely commoditized, you are working in a [00:21:00] community. Type of scenario. What happens are a couple of things. First, there is cross pollination of ideas, which is most of the advantage of working in open source world, which means whatever application you're working towards, you can start picking up.

Patterns, and you can start picking up, you know, the latest and greatest things that somebody else is doing for something completely different, which means that all of a sudden that accelerates the classic innovation paradigm of open source, you can accelerate the way you introduce new technologies, whether that is, is that AI or something else, you introduce that in what you're doing, right?

And the other side of that is that you have this ability to break down the silo of your own development group or your own company. You go out in the community, whomever you are Netflix or think about video games developers. So Rob is a gamer. You know that a lot of game developers like Gorilla Games, those kind of people, they do a lot of talks.

They open source a lot of their tools. And why is that? Because you have a way now to take [00:22:00] your team of 60 developers and and make the core of the work and then share that with. I know 6, 000 other developers and all of a sudden you have, as we said before, you have this giant R and D community out there that will just accelerate what you're doing may even fix things that you weren't able to fix before in this particular community, in the dev community, the synergy between open source and cloud native is right at the heart of what you were just talking about, which is like they're both Very dev centric, very creative organizations that the philosophies of the two things just come together wonderfully, I think, to sort of self support each other and actually in, you know, modern tech organizations in the cloud era, if you're not focused on enhancing developer productivity, driving developer creativity, you're really missing a trick, right?

You are, and let's be honest, it works wonderfully until you stay in a technology conversation. So going back to the video game example, [00:23:00] what you open source and start discussing about is the underlying algorithm you are using to generate clouds. How do we make more realistic clouds? How does it work? The shaders, this kind of stuff, which is not necessarily the IP part of what you're doing.

So that's the tricky part is when you start talking to enterprise, the tricky part is where you draw a line between, I want you on board as much innovation as I can. Yeah. I want people to discuss as much as they can. At the same time, what I'm really doing is that I'm making money based on my AP and what is my AP.

And how do you fence that? It's the toolbox. The important part is the IP comes with the assembly of the toolbox, how you configure the tools to be able to deliver the in game experience and the clouds look fluffier and all the water looks really realistic or whatever. That's an application of the tool that is your secret, isn't it?

That's the bit you keep protected, but actually you're building upon the community and the tool set that they have made. And that analogy, doesn't it, even beyond gaming organizations, the specificity. All the [00:24:00] expert application of technology to whatever the business problem is, whether it's making a cloud look fluffier or whether it's trying to take, you know, 30 percent of the emissions out of supply chain, the underlying developer community and the creativity that's that could be unleashed in that space can have upward business value impacts.

It does. Let's use a concrete example. So in tech, we used to do a whole lot of reinventing the wheel all the time. Why we did that? Because humans tend to do that because you want to create your own things especially tech people. Fair enough. But also because you don't have access to anybody else.

You're saying that's a bad thing? It is a bad thing. You're sure we can make a better wheel, can't we? I'm going to make it more rounds or somebody says, I'm going to make a better wheel. I think a square might be a better answer. Exactly. We'd make it, you know, hover. Part of that is because we all have opinions and we all like our opinions to be louder and more broadly adopted than other opinions.

[00:25:00] The other part of that is that you didn't have a chance to interact with anybody else that may have gone through the very same exercise you're trying to do. So, so, and that and to keep that, an example that, that drives something out is like, imagine that you are your insurance, like classic insurance, life insurance, and now you want to enter the travel insurance space.

You want to have these, you know, four days length of insurance that you have to deliver and via an app and do all the approval, the risk management, blah, blah, blah. That becomes, you know, a mobile first experience, which becomes a distributed system, and you never saw a distributed system in your entire life.

So you would have to reinvent that, or now you go in the community and in the community there are people like Netflix. So basically the people that pioneered large scale distributed system that are offering services in real time to customers. And now you have access to that, which is fantastic.

You don't have to reinvent the wheel. You can go there and learn and piggyback on what they did and adjust that. What is the tricky part there? The tricky part is that. You're in the community. People know that you work for a traditional [00:26:00] insurance. How do you go and poke the community without telling them about this very secret project that you are doing?

That's always the tricky part of open source. Like how do we draw a line, especially if you're in enterprise as an enterprise, I can draw a line and saying that's what can go out and that's what needs to stay. And then underneath this enterprise umbrella, you have 20, 000 people and the individual developer that is very excited about having a chat with the Netflix guy, you know, how do you control that?

That's the tricky part, right? Right, right. And well controlled then. So we've talked about the amazing impact these sorts of philosophies and ways of working have on developer communities. When you do extend that up in a way that's functional to organizational level, like What's the business upside of all of this, do you think like, so I'm an FMCG organization.

I've gone to the trouble of making the shift more fully into the world of open source. I've been transforming into a more cloud native environment for, you know, for the last N years and probably investing a fair amount in that transformation and change. What are [00:27:00] my upsides? Why have I done that? Oh, I've done that for many reasons.

So I can tell you the direct impact that has on us being an open organization and open companies that now we have a whole lot of customers that are very, that have, so our software runs a lot of mission critical business around the world, right? So organizations depend on what we deliver them. Now, they may have requirements that are unique to them.

They may have requirements that are not unique to them, and they are very happy to tell you what the requirements are. But putting those requirements on a roadmap and having something delivered in a product that you are using is not that straightforward. I'm a product person, I can tell you. If I triage something and approve something goes in the roadmap, you can get that in two releases.

Now you have. a venue where you can go and potentially even input your code directly into the application. So you can, let's put it in this way. You have a more direct way of influence the software that runs your core business, right? That's one [00:28:00] side. The other side is what we said before, is better acceleration, you know, let's say a viable way of introducing innovation in what you're doing, a wider, broader way of testing.

And trying the software that you are producing if faster, let's say adoption cycle when you go and you have to modernize whatever you're doing and again, you start a software project, whomever you are, whether you are an FSI or you are an oil and gas company, you are building your software to manage it.

The PLC that you have on your oil rig, etc. You start the project now, the project will end in two years. And what is the risk? One of the risks is that you start with a technology base that is already five years old, which, which could be a good thing in certain scenarios. It could be a bad thing in other scenarios, you know, having moved to an open source kind of philosophy gives you at least all the options available, which means you can introduce.

Latest technology as you go, you can spot problems [00:29:00] as you go, you have a larger crowd out there. that can vet the work that you are doing. That's one of the big business outtakes. And it seems to me that aspect and then combining that aspect with the similar philosophies that exist in cloud native and the similar sort of technology accelerating properties that exist in cloud native platforms is the thing that helps businesses go faster, right Robert?

It's the thing that kind of underpins a faster iteration of innovation, which therefore can lead to, you know, business value X, Y, Z. Yeah, it's in my head going back to the point there is the beginning. It sits in a number of domains. It's overall cheaper. So you know, for those who want the top line answer your T.

So your cost answer. It's T. C. O. Down. Yeah, you're faster to market. So you can go faster, but with more confidence because you know you're on a tested and checked and tried platform. So you're not inventing All of the critical [00:30:00] parts. People are happier. That's always a good one. It's happier to work in a cloud native environment and fundamentally there's less toil and you're spending more time on high value work so you can do more with the same.

You know, capability within your organization. I think the big thing for me is organizations pushing the change across the people and ways of working. There's this big culture shift you have to do to get to it, but when you get there, you sort of get this lovely benefits package that comes out the other side, which is some.

It's sort of a win, win type scenario. I feel like we are slowly moving towards the digital transformation conversation, which is the thing that I avoided, you know, on purpose, because that feels like, oh, we have this big digital transformation happening. And because of that, you need to accelerate everything.

There is more complexity and people need to change the mindset, which is all true. It's just that I've been such an abused. They Adam for so long, that is true. We always joke about the transformation failed, best change the technology and then it fails [00:31:00] again and they go change the technology again. So the technology always gets a bad rap, but it always comes back to the leadership and the people and the culture, isn't it?

But you're right, you get a lot of people have got transformation fatigue. I think that's, I think that is right. And you're right. It's also right that a digital transformation is a very much overused term, but I do think we are at a point where there is something tangibly different at the end of adopting some of the philosophies we've been talking about that does have business value attached to it.

And I think what people You know what? Somebody like Dave Snowden talks about, for example, we had him on the show and I think it's episode CR 20 if you want to refer to it. And the reason I bring that into it is because he doesn't see transformation as being something that should be, Hey, now let's stop everything we've been doing and, you know, reboot the whole thing and start completely differently.

He says it sees much more of an evolutionary curve and I think held in that. Way in front and framed in that way to me feels like a much healthier way of adopting [00:32:00] some of these new technologies. It doesn't say like everything we've been doing bad, all of this new ideas, stuff, good. It's like the new idea stuff is really good and absolutely has all of the benefits that we've been talking about.

But it doesn't mean you throw the baby out of the bath water and you can evolve forward. I think, and it's just a question in your organization of how fast you want to do that evolution. That's what sort of reflecting on some of what Dave says. And, you know, some of the conversations we've had that transformation should probably have a small T, you know what I mean?

I, and I don't disagree, but there is a point, let's put it this way. We've been talking about digital transformation for how long? The past 11 years, 12 years. Yeah. So Rob's been talking about it for maybe the last 18 months. It's just Is that why I woke from my stupor Dave and realize that things had changed.

Yeah. All right. Cheers

10 years in trial and error to understand what digital transformation or transformation in general means, right? And we finally landed [00:33:00] to a point where I was actually reading a paper last week that said, there is this very boasting title that said, Oh, 70 percent of digital transformation projects fail.

Okay. Fair enough. Why? And then the paper was going to, Oh, it's never a technology issue is always a mindset. Issue is always a people issue, which I agree with heartily, but the point is we are finally at the point where we have an understanding of what we got wrong for 10 years, and we are moving towards this idea of change and transformation should be incremental, should be constant, should be a part of your philosophy of how you operate, right?

Keep refining, keep changing things. I mean, my, in my opinion is still, there is still a tipping point. There is a tipping point where you need to get into the mindset that says, well, the way we've been working on software application projects, whatever. They had a development cycle and then they have a stable cycle when they go out in production, they don't touch them anymore or you don't touch them that often.

You remember the old [00:34:00] myth of like, you will all get into a CICD scenario where you will update your application 20 million times a day. And then most of the guys said, well, no, when he's in production and he does what he needs to do, it's fine. We'll do bug fixes, but it's fine. But there is this attachment between the people that develop and then the people that run the application until you move over the hump where those two groups are one group.

And you can actually start introducing the changes, whether those are technological changes or operational changes. Until you go down the tipping point, it's very hard to introduce changes incrementally. Because you still have a big wall you have to go over, right?

Sjoukje, what have you been looking at this week? So each week I do some research on related ideas and transformation in tech. And this week I thought. should take a look at how generative AI and cloud compliment each other. So gen [00:35:00] AI and cloud computing are complimentary capabilities that can be used together to drive adoption of both the technologies.

Gen AI can simplify the migration to public cloud and help enterprises decipher and translate legacy code into cloud native nine languages. And the use of GenAI tools could reduce cloud migration time by about 30 to 40 percent. GenAI could also help address skill shortages and other common use cases, including coding, content creation, and customer engagement.

So a question to you. Are these the main areas where infusing GenAI in cloud computing are beneficial? Or is there more? I mean, you got a lot of execs very excited with your 40 percent reduction there. Shout out to you, that's the stat that makes you sit up and you go, Ooh, blimey, let's have that. Can I have that please tomorrow?

A double? Oh yes, we'll have it. We've got the maths to back that up. I think the concept is though, doing AI, starting out, many organizations starting out on AI. Not [00:36:00] doing it in a cloud native way and using clouds to power it seems like an absolute, total, massive mistake for anyone. It's like, why would you start on a journey that's new, not using the new ways of working that we just discussed?

It feels like. You'd missed a massive trick and why bother and you're gonna have to convert anyway in the future. So it's a kind of like you'd have to have a really good reason not to combine the two together because it's the whole point about organizations and they are is they're all learning.

They're all testing. They're all trying to get their mastery and yeah, the best way to test and learn and fail and make sure you get those feedback loops going is in a cloud native way. Otherwise you might find you have some very interesting investment decisions that don't Come out to deliver the benefit that you probably thought they might at the beginning.

So, so yeah, for me, it's a no brainer. I'd maybe even go one step further than the no brainer, which is it's part of the cloud stack, isn't it? Yeah. So for me, it's the, it's part of [00:37:00] the era or epochs discussion we had earlier, which is the super set. It's still a cloud and it's dev stack and it's a ways of working is a very significant innovation.

That's that sits within the center of that. That's largely been enabled as a result of cloud ways of working and the cloud stack and, you know, frankly, in the background, the internet and the data set that's providing, right? So use your word, probably Robert is it is entire no brainer in my thought that If you're using AI, you're probably going to be using the cloud stack and therefore you're in that world, aren't you?

Am I missing something there? Totally true. And you should do that because also from sustainability perspectives, right? Not reinvent the wheel. Don't start training these models yourself. If I can offer, if I can offer a lateral point of view, so everything that you said is 100 percent right, especially generative AI because of the nature of how generative AI operates and get trained and how expensive it is, it's obviously a cloud first [00:38:00] delivery model.

If you use that, you will be using that on the cloud, but it's something that we said before, which is like, Oh, that could actually help accelerating. the adoption of the cloud. And that's true for accelerating the adoption of X, add whatever technology you want. Obviously, we went through the exercise of building our own generative AI because why not?

We started a while ago working with the Watson people and it was a fantastic learning opportunity because those are the scientists that create the algorithm, train the algorithm. It's a very interesting exercise by itself. But the interesting part was the You know, you set the charter to say why the hell do I want to invest in generative AI?

What is the end goal? And the end goal was, well, there is a massive skill gap and skill shortage in the market in tech, right? You can, and you can frame that as you want the right type of developers, the right type of operations. I was talking to a customer the other day that told me, oh, we started looking into automation massively because we have five senior folks that are retiring.

And we don't know where to [00:39:00] find that expertise anymore. And the approach that we took to generative AI, and a very good use of generative AI, is to help more and more people to do what you need them to, right? We use that on automation, obviously. Our idea was to democratize the way you access. Automation.

So how do you make people that are not automation experts or even, you know, people who make them writing more automations. But if you apply that to cloud transformation, the idea is now you have a technology that is able to take a Let's say slightly ambiguous idea or brief. I need to do this thing there and translate into something that is actionable, which is fantastic.

It's a fantastic acceleration technology. As long as you take care of the, you know, the, what it, what falls into the area of the ethical AI, which I'm sure you, I'm sure you will. No love. An interesting part from my point of view, again, I'm, I work in automation and I build automation tools in an interesting part, especially with generative [00:40:00] AI is that generative AI is non deterministic by design, right?

It's it leaves, so to speak, if we can say that generative AI lives, lives to generate things, you ask a question, we generate an answer, whether this answer is well known or not, we generate a new one, which has a fundamental conflict with it. Automation is an example, which is deterministic by design, right?

So it's interesting how we are trying to make that technology more viable in a world where we need consistent answers and governance and that kind of stuff. But for now, the best place or the best opportunity for generative AI tools, in my opinion, is to help people to get and move and be trained and be more effective.

Yeah. Yeah. There's that human issue where there just aren't enough people to get the productivity up. And everybody's suffering from productivity issues. So the angle is very well served, which is that this is going to help organizations do what they need to do and get people on high value stuff, as opposed to the boring stuff that is today necessary, [00:41:00] but tomorrow can be just taken away.

Well, to bring it full circle to one of Sjoukje's examples, the migration problem, which is. For anybody who's been involved in scale migration, it can be quite torturous on you end up, you know, that's an understatement. Yeah, and you generate the to generate the right speed you're having to make, you know, kind of, you know, you're sliding lower and lower down the value stack to try and get the thing to move actually at a pace that you want.

So, so something that can like bring in sort of highly capable compute and automation around migrations. I think the first company that cracks that problem. There's loads of automation attempts on the market at the moment, and some of them are better than others, let's put it like that. But there is still a sweet spot, I think, to be hit there with, like, fully automating migration.

That doesn't say it's going to necessarily automate modernization, but actually to have the conversation and the humans focused on the modernization, and to just get the rest of it swept across in a [00:42:00] way that is Effortless and releases things like facility and technology and some of the basics of the business case that's still to me feels like in 10 years.

We haven't quite cracked that problem yet. I don't think no, we're getting better at it, but we're still not there. No, that's true. It's still a roadblock and it's a good opportunity to take that out and have people focus on something else. Something that is more high value. Yeah, exactly. And I think that's a good place to leave today's conversation because I think all of what we talked about today is really about that, isn't it?

It's about just kind of trying to unleash value somewhere. And we also had a really useful contribution from the Roomba as well earlier in the conversation. I think that's the first robot we've had on the show, isn't it? Yeah, it is Dave, but that's, maybe that's where the threat comes from, right? It's the Roombas that are going to take over the world because everybody's buying them.

They're all sat dormant in your house and then they go out and everybody thinks it's benign, right? Maybe that's the As soon as they learn to do stairs.[00:43:00]

Great conversation today, Massimo. Thanks so much for joining us. It was a lot of fun and I, you know, really scratching on some important stuff. Thank you for having me. So we end every episode of this show by asking our guests what they're excited about doing next and that could be going for a great Christmas lunch or going for a good dinner.

We're in festive season at the moment. So it could be something like that, or it could be something in your professional life. So Massimo, what are you excited about doing next? Well, first it's Friday, and I'm very excited about the weekend ahead of me, which is the very first weekend that I spend home.

And my plan is all to go and see the festive Christmas spirit around London with my son, which is going to be a first. I'm also excited about next week. What I'm traveling again, I'm going to see a couple of customers is the end of the year, you know, as you know, I knew, and you would do the kind of the visit of the Christmas spirits at the end of the year to large customers.

And it's going to be fun. I'm going to, I'm going back to Italy because surprisingly enough, I'm one of the people that speaks Italian and [00:44:00] I'm going to Netherlands as well. So I will be in Amsterdam, I believe at the end of the week. And the exciting part is that there is a whole different conversation that we didn't touch today around how you enable in enterprise organizations that change the change of mindset, right?

Talk about or changing the operational model. How do you do that effectively? How do you have, I do start that into large environment. How do you help people to go through this transformation when you handle, you know, a fragmented organization, thousands of people, et cetera, et cetera. And there is this.

paradigm that is coming up more and more, which is the idea of having a platform team or a platform engineering model, people that are tasks to support that mindset transformation via technology. So two of the customer visits that I'm doing are entirely about having that discussion with people that went through.

A version of that. So it's more like a learning opportunity for me, which sounds a lot of fun. I mean, that does sound great. That sounds like two podcast episodes, right? That's right. I was just thinking, that's a tick. Yeah. [00:45:00] Enjoy that safe travels and particularly enjoy your Christmas tour around London with your son.

Thank you very much. So a huge thanks to our guest this week, Massimo. Thank you so much for being on the show. Thanks to our supportive producer Marcel, our sound and editing wizards, Ben and Louis, and of course, to all of our listeners.

We're on LinkedIn and X, Dave Chapman, Rob Kernahan, and Sjoukje Zaal. Feel free to follow or connect with us and please get in touch if you have any comments or ideas for the show. And of course, if you haven't already done that, rate and subscribe to our podcast.

See you in another reality next week

[00:46:00]

About Capgemini

Capgemini is a global leader in partnering with companies to transform and manage their business by harnessing the power of technology. The Group is guided every day by its purpose of unleashing human energy through technology for an inclusive and sustainable future. It is a responsible and diverse organization of nearly 350,000 team members in more than 50 countries. With its strong 55-year heritage and deep industry expertise, Capgemini is trusted by its clients to address the entire breadth of their business needs, from strategy and design to operations, fueled by the fast evolving and innovative world of cloud, data, AI, connectivity, software, digital engineering, and platforms. The Group reported in 2022 global revenues of €22 billion.

Get The Future You Want | www.capgemini.com

This presentation contains information that may be privileged or confidential and is the property of the Capgemini Group.