



REALITIES REMIXED

People · Culture · Industry · Tech



RR012

How Open Source is powering
AI with Richard Harmond, Red
Hat



RR012

How Open Source is powering AI with Richard Harmond, Red Hat

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

(00:00.59)

Yeah, and I think you're entirely correct, right? First time for everything eh Dave? I mean, it's been four weeks, I've been building up to this.

(00:23.318)

I'm Dave Chapman. I'm Esmee van der Giessen. I'm Rob Kernahan. And this is Realities Remixed, an original podcast from Capgemini. And this week, we are going to check in on open source and see how that fits within the world of AI. Joining us to discuss that this week will be VP and Global Head of Financial Services at Red Hat, Richard Harmon. Now, if you want to jump straight to Richard's conversation with us, you will find the timecode for that in the show notes.

But before we get to that, Esmee, Rob, you're here. How are you doing? All right. Yes. All good. Good. Happiness. you been up to? What you been up to, I've been doing a lot of experiencing deep democracy training. So I'm now up to level three out of five. Not level four yet. And it's really getting, I'm really getting the hang of it. It goes up to five, this one, Rob. In levels, yeah. You can't do that. You can't do that. It's not allowed. Do you know what level five is about? Go on then. Like really you yourself as an instrument level four is already the same, five is even deeper. If Rob was an instrument, what kind of instrument would Rob be? Don't answer that question. That's, that's just going. trombone. I'm thinking trombone. Triangle. it? There's something very cool about a triangle. It's very simple. You could definitely have it. We've been talking about triangles before when we did the photo shoot in Barcelona. Oh yeah. Yeah, there you go. Somehow you're connected to triangles. I love the idea. It's either trombone, very dramatic one end of the spectrum, loud, noisy, complicated instrument. Honking instrument. then the other end is ding. Esme, if you were an instrument, what kind of instrument would you be? I'd like to be a piano. I just love the sound of a piano. Yes. The piano is, I think, one of the greatest instruments. I think it might be the big dog, don't you? Yeah, it is the best. love like wherever you put piano, it always sounds like so you've got the classical piano, you've got it in house music, you've got it all sort of wherever it plays in rock when the piano pops up, always does a good thing. I think it's the it is the peak instrument, isn't it? And you Dave? So I think I would if I could choose my own. (02:41.998) If somebody else chose it, God knows, but if I could choose my own, I would be the Moog Analog Synthesizer. Oh no, you have to have a normal instrument. That's very specific. Yeah. Yes. Yes, that is a normal instrument, Robert. Well, is that the one where you put your hand near it and it makes a funny... No, that's the Pheromone. That's the Pheromone. Pheromone, that's it. I you'd be one of interesting choice. As I come near you, you always start to wobble. (03:09.036) I think we've just hit something there actually. I have an electromagnetic effect on your personality. And it's generally for the negative. A human theremin. So today, amongst other things, we are going to talk about the mix of open source culture and AI and whether they complement each other or not.

So can they come together and act as some sort of force multiplier? Does things like transparency and ways of developing in these various different tools, is there any read across between the two? Gut feel for this. So gut feel going in Rob, where's your head? So it's complicated, right? But I would say where it's AI and we're using models, understanding how the model was trained, whether there are biases in there.

quite important. Where you have closed ecosystems and you're using something like that, maybe you don't fully understand how it was trained and what results it was biased towards.

So openness is a good thing when it comes to things like models. The problem is though, for a frontier model, it costs you 150 to 200 million dollars to train it and you need a huge amount of infrastructure. That's something the open source community can't cope with, right? Because it doesn't have the access to the resources. So there's this bad idea, if you want the edge stuff, you're going to have to go private and close. But if you want the open and understanding the abstraction and maybe appreciating how it's going to operate, then open source does have a good point to it as well. Then there's the other part, is who's controlling the open source, who are the few key individuals that keep the project alive and moving forward and all that sort of stuff. Where's the development coming from? I think there's no doubt that the sort of the happy Nirvana of open source in its, its, in its track to Let's just call it enterprise, resilient, stable, trustable, has inevitably become more corporate at the point somebody productizes that, hasn't it? Think there's an inevitability to that, perhaps. And a lot of open source projects like Kubernetes, it's like the foundation of cloud native, that's generally controlled by a couple of big corporates. (05:13.226) And it's the same with OpenStack as well. So big corporates, I mean they are open in the sense that I can get hold of the code base and deal with it, etc. But let's not be under any illusion that some big corporations who control the direction of some of these very considerable projects, but then they're the ones that created them in the first place. So fair's fair, I suppose. And it's a balance you have to strike. If you didn't have the heavyweight capability of those orgs behind it, they probably wouldn't be anywhere near as progressive as they are from a capability perspective. From a culture perspective, there clearly is a lot of cultural belief in the world of open source and in AI native, guess I would say we're on the beginning of understanding what that means, though there might be some read across from cloud native thinking. Do you see any overlap between the two cultures that could be complementary or do you see potential friction? No, I think in essence, I see a lot of similarities. Talk about non-linear, open, interconnectedness without even knowing how it's connected. I think there's a lot in there, but as Rob mentioned, there might be some big corpse behind it. I think it could also be like a safeguard if there's ethics behind it so that you're sure that there are some corporations that do feel that they need to check out whether it's good, whether it's solid, whether it's safe, and hopefully not doing the wrong thing. And again, it's also about the balance. But if you look at the community behind open source, that's fascinating. The idea of you want to hand something over to the community or ask them something and then they start building it and then you can make it richer and better. Think that that's, but maybe that also sounds a bit romantic. And that could also be the case with AI. If it's for the good, it's for the good. But if it's not, then, or it could be for the bad as well. So I'm really interested to see how this is going in terms of the engineering community and how they feel, I think, obligated to do it in the right way in terms of ethics. Don't know. I'm just curious. It's that balance though, isn't it? So we love the idea of open source, the freedom, the availability, you know, but without considerable resources as the world gets more complex and we expect ever more functionality out of these things. (07:33.08) Then there has to be some form of big resource behind it to push it forward. I mean, I think that's part and parcel of the, I mean, going back to the amount it costs to train AI without private corps helping, you're not going to get very far, you? Mean, yes, you've got the Linux operating system that everybody points to, but that was something that was built over decades and maybe we need results a bit faster. So I think there's a balance to be struck, isn't there? Well, I think perhaps the other dimension to look at this is like, how can AI help with the development of open source code and develop a productivity in the world of more corporate development has obvious read across into the world of open source. But I think there in may lie a challenge. And one of the challenges might be then AI code that maybe isn't as robust or resilient as it needs to be leading to an influx and an increase in low quality contributions in the world of open source. You get an overburden then on the maintainers in the open source world, what they're doing, how they're spending their time, et cetera. And then finally, there might be something in all of that around ethical or

security risk where, you know, kind of nefarious code or core code for whatever reason from varying different backgrounds gets used in a way that it perhaps wouldn't in a fully human system. But that's happened, isn't it? So there was that open source project that like tens of thousands of organisations used. Its code base was largely dormant and security hackers came along and inserted malicious code. Everybody pulled it in, compiled it and it nicked a load of security tokens. So that has actually happened. when you use open source, because of its very nature, you have to pay attention to what's going on in the projects you're relying on. You know, the average open source project has like 3000 dependencies. So there's a huge human effort or a tooling effort that's required to track all this stuff as well. So we're building a world that is very complex and it's very easy maybe to corrupt that subtly, which we've discussed in the past, haven't we, about data sets getting slowly corrupted to influence training and such like. Well, look, let's go to our conversation now with Richard Harman. Richard is the VP and Global Head of Financial Services at Red Hat. (09:52.404) and dig in it a little further you (10:02.766) So we are finally here with Richard Harmon, VP and Global Head of Financial Services at Red Hat. Now I say finally, because it's taken a while to line this one up, hasn't it? I think we first met Richard back in August of 2025. We are now recording this in April of 2026. Dr. Mike, what got in the way of this? What happened? First, we had a sort of technical issue and the second time you had a big storm in the UK and there was a power outage. So you were out of electricity. That's right. So Richard, thank you so much for bearing with us and being so patient. No, it's pleasure. I'm looking forward to it. Delighted to talk to you today. Now, why don't you just tell everybody a little bit about yourself and what you do on a day to day basis.

Yeah. So Red Hat is the largest open source company in the world and I run our financial services vertical. So it's a global role and I talk, you we have thousands of financial service customers, but I spend a lot of time also with regulators and particular central banks. So most of the central banks globally are also our customers. So it's a great opportunity to spend time with regulators and understand what their priorities are. And then in some sense, share that information with our customers. as well as our teams where we're building platforms and capabilities. Did you always like financial services? how did you end up in that industry? Yeah, great question. Yes, I have a PhD in economics, specialty in sort of econometrics, mathematics. And I started my career at the Federal Reserve in New York and then a short stint at the Federal Reserve Board before I moved to the commercial side. with Citibank and others. yeah, it's, it's, started, you know, early on in, you know, undergraduate was focused in that area as well. how did you meander your way from, from that beginning into the world of open source and the tech side of financial services? Yeah, I, I, I, the first tech company I worked for was, was with SAP and I was there six years, started (12:17.026) I guess what you call now an AI consulting practice on the financial services. I was part of a bigger group. And then I moved to their capital markets business and it was managing part of that out of London. Then I moved to Cloudera, which was the Hadoop data platform provider. And that was my first real experience with open source. it, I was fascinated with it because it's not only the technology that's very innovative and, but it's the way. software gets built. It's a global community, it's not just one firm or one institution. And that whole model leads to, as we'll talk later probably, cultural aspects as well in terms of how companies manage and run themselves. bit I always find fascinating with open source is it builds some of most complicated software on the planet, operating systems and such like, and it self-organises, but it kind of works. And it shows that the human system with you know, everyone's a peer almost in the community, but they all come together and there's these great outcomes and you think maybe you don't need all the shenanigans of corporate world to do great things. It always surprises me. I love the model because you think it's democratised and it requires motivation to have to be involved in it. When the motivation is strong and these people are highly motivated, it creates great outcomes. It's content centric. as well in its thinking people come together because they are passionate, have ideas and drive out from that perspective rather than, you know, there is

some sort of hierarchical win out of the situation (13:58.968) good idea, meritocracy or the I've created a better bit of code. Everybody tends to accept it and it drags it in. it's actually quite a nice, nice system really for the human. technically it's free software. Yeah. Which we don't like anything free. Happy days. Well, obviously we provide for large enterprises, know, enterprise grade software. So we harden it, secure it, do a lot of things like that. Integrating all that in a holistic incredibly stable and resilient platform that's performant, in other words, very, very low latency. There's a tremendous amount of computer science skills that go in making that really, really successful. And the most important part, from the Linux side, is it has to be resilient. And it is the safest operating system in the world. And it has been for 30 years, 30 plus years. The other side of it, when I was in the financial service side, open source was something that was outside for decades. And in the last 15 years, the adoption has massively accelerated. Now people do not question the use of open source because again, it is very, very safe. It has very rigid requirements in terms of committers providing new content, constantly being checked. And you have a global community. you know, basically looking, working and using it. So you have more eyes and hands sort of involved in testing it as well. That's a very good point. 20 years ago, open source wasn't trusted. It was like, don't use it, you need proprietary closed software. And now it's basically yes, be open, see the code, a big community scanning it. Do you think that it felt like that was just a gradual change where the utility of the code was so useful that people went, well... The alternative is very expensive and this bit over here is free and it seems to work. Although there has been some very notable issues with like the classic being the Log4j one, which caused a bit of havoc for a while, if you remember. there have been some, but everybody has the notable security issues. yeah, it's a trust that's built in the system over time. It's quite impressive that it's done it. There wasn't a marketing campaign behind it. It was just the, it came out of the usefulness of everything that everybody went, well, it probably is all right, isn't it? Yeah, and the proof. (16:22.574) Probably, right? It's proven over and over again that builds trust in the system. Yeah, you're spot on. And I think the other side of it that's important to remember is that it's also very cost effective, but you're never locked in. actually have the access. can look at the code. take it and run it yourself if you want to support it yourself, right? And push all the pieces together if it's an infrastructure layer. So there are firms, there's some large banks, for example, globally that try to do some of those things. Some of do it very, well, but it's duplicating effort. I think the notion of open source, we have this in Europe, we have this thing around sovereignty, actually it's more global now, concerns. And one of the secrets of open source is the fact that you control the software, the user. They, they, they always have it upstream. They see everything fully transparent. They can contribute to it besides just being a user. And so that provides a unique capability from a sovereign perspective, because you're not dependent on one particular company that's in a particular country that you're worried about, you know? um, so Richard, maybe, um, for those of us that haven't been tracking the open source story as we've journeyed through the cloud era, think it's position and Linux's position pre cloud was very clear and everybody kind of knew where it stood. just paint a bit of a picture for us at how the community and how the software development has changed like during the cloud era and maybe sort of broadly up to date. Yeah. So with the Advent and adoption of public clouds in particular. We have three very large, we call them hyperscalers, but there's large cloud providers out of China and other countries. Europe's building its own sort of federated cloud environment. But out of those sort cloud technology comes sort of a modern architecture called Kubernetes. (18:44.918)

And that architecture is about everything is self-contained in a container. So you can replicate something very easily by just creating a new container. And if a container goes down, you can replace it very quickly. So it's this sort of micro foundation type architecture, which provides a lot of agility, plug and play capabilities. It also can be highly resilient Most large institutions, a lot of their software development now is in this sort cloud native architecture, which is the Kubernetes layer and having things sort of, know, largely built in a containerized environment.

Yeah. I was going say, when you go from a world of what was the mainframe area client server out to cloud, I think it's, and a lot still haven't got there with their systems, or they're still, it's only partially there. The freedom you get and the ability to check the change agility that comes out of it is amazing. It's like this incredible value release that most people know exists, but they're so, it's so, for some it's so difficult to get there to execute on it because of the organisational structure that exists around them. I'd be interested in your view on that and where you think the maturity of the market is getting to, because we've had cloud for a good old while now. been very mature for 10 years. Things like Kubernetes, containerisation, cloud-native ways of working. They've been well understood for a long time. Where do you think... If you said a sort of from legacy to cloud-native, where would you put the percentage of us on the journey? Are we 10%, 20%, 30 %? You know, it's that view, because the value is huge. Yeah, it varies among industries. So I mean, I manage the financial service business. let me That one I know much better than say Toco. I'd say it varies between 30 and 50 percent. You hit it right on the nose. The challenge is the legacy workloads, mainframe. We now have our application development platform called OpenShift. It's Kubernetes based, it's hybrid cloud, meaning you can run in any cloud environment. You can also seamlessly run it in on-prem. (21:01.474)

So you have the ability of having the same application run across every environment that you're using, which is sort of the beauty of having clouds, right? You have the agility, not just bursting capabilities that firms like to have and not having to be massive data centers, it's available in the cloud. I'd say the migration, the challenge, particularly for financial services, the legacy systems, still probably 75, 60 to 75 % of a bank's budget is maintaining the legacy systems. So there's a big effort, and it's been going on for more than a decade, of trying to modernize these systems. But when you get to something like core banking, that's open heart operations for the bank, right? Touching that and some of that, many of it's mainframe based, some of that code is all cobalt, it's 30 years old. Many of the programmers are not around anymore. They're all retired. So it becomes a real big challenge for some of these missions critical legacy systems. had a guy on the pod, Ron Toledo, he tells a story from the Netherlands where the dad started the code base and then the son ended up supporting his dad's code base on mainframe. And we weren't sure if it was poetic or if it was sad. We weren't, because it was like the code had survived. But you're right, because the banks were the first to automate with big mainframes and they got huge advantage from it. But now they're having to tackle the beast that served them so well for 30 years. Yeah. And we have this Kubernetes layer in the mainframes now. So firms can modernize their software. don't have to... mean, mainframes have incredible resilience. They have, you know... They're basically, they can be almost never down. So there's certain elements of mainframes that really fit a very critical function and a role. They always sound like they're old architecture, but everything under the hood, outside the box, is all completely new. Chips are new, networks are new, everything gets modernized in that. But the problem is the software, right? A lot of it, now it's changing, but a lot of it is all COBOL-based. (23:19.412) And so, you know, getting systems from COBOL into some other language, Java, C, whatever, you know, is a non-trivial task, especially the people who wrote the code, just Rob, like you highlighted, they're children of the ones that know the code now. You know, that actually is a bit sad, I think. Yeah, I think that's where we ended up. Sad, because I came, my first role was working on a banking mainframe. And I found COBOL and I mean, I was things like, what do you mean I can't just declare a variable in scope halfway through the code? What are you talking about? I have to know what I need before I start writing the code. What do you mean I have to write batch code? was like, I was confused for ages. I don't understand how this works. You paint a beautiful picture, I just, I did literally, they sat me down and go, you're going to be doing COBOL. And I had to look it up. I had to really look it up. And I'm quite old, Dave. So there you go. that's the... Before we delve even further into...challenges in financial services. I just want to finish the little arc around particularly the open source community. So as we went into the cloud period, you said, Kubernetes and that

space born from the same sort of cultural roots, how does the community develop over that period of time? Did you see it start to diminish as people got more interested in kind of the new wave of cloud native technologies or did you see the opposite happening? In terms of the open source community, it's exploded. There's over a million projects in the open source world. A million? Over a million projects. So you multiply that how many developers on a project. Some are small, like five or 10. Some can have thousands, many, many thousands, right? So that shows you the breadth at which open source becomes sort of the platform for innovation in many cases. I would say everything done in academia 99 % of it is done in the open source world, you know, because it's transparent, it's shareable, there's a central repository. You can pull down the data and the models from an academic paper and run it yourself and test it yourself. I would think that that is also the reason why in financial services, because it's so highly regulated and especially towards due diligence in the past decade. The focus was more and more on control and making sure that everything is visible. (25:40.59)

that would be an opener or an enabler, I think, for open source or not, even though it feels like kind of a contradiction? Yeah, no, it is. It absolutely is. And I think the learning curve and the experience, which took decades, was about that that's actually very, very innovative and reliable software, right? And secure. Because one of the things people were always worried about early on was, you know, is this secure? There's so many people working and developing in this. But the model that was developed through committers has worked extremely well. And it's very controlled, they're carefully checked in 99.9 % of the cases. So I think it's the fact that you have so many different people with different backgrounds involved in these kinds of projects, I think is what makes it actually not just community-based, but also makes it much more resilient. Right? Because you have people with different backgrounds to do different kinds of testing. It's the very heart of why diversity is powerful, isn't it? Isn't it an exact example of that sort of situation? Yeah. Yeah. And you know, let's say an institution doesn't have to be a bank that is using an open source capability or platform and sees something missing. They develop something to fit that gap. If they push it in the open source, it becomes part of that project, right? After it's been reviewed, checked and tested and validated. And so that's how these systems get very, very resilient, but they also address all the unique needs of very diverse institutions, from public sector, to financial service, to satellite companies, to anything. Now, as the culture moves forward again, so we come out of the cloud native period already, And to the point earlier on, we're somewhere probably between on a bad day in a slow industry, 20 % adopted maybe at the top end, 50 or 60 % adopted in certain sectors and certain industries. So it's almost like we haven't completed that curve. And then this behemoth of AI has now come over the horizon. (27:59.586) we're going to see some cultural ways of working changes that are going to be very dramatic, I think, all the way from the tech layer right through to the boardroom. What's your mental model going into that of how open source culturally fits into that movement? I think it plays a strategic role. Obviously, OpenAI and some of the other more explosive firms in the AI space, they're not open source, right? So they've built very proprietary

closed systems. it's almost the exact opposite because they're so highly protective of what's going on with their models and such like. Yeah. And look at the valuation they have, right? are hundreds of billions in some cases. that's a financial system model because investors can reap the benefits. The revenue stream can be very significant. Investors who will see that that could scale so profitability revenue even grows more. So that's a model that that's existed for centuries almost right going way way back with equity markets. But I think what's what's slowly happening in the AI space and it takes time just like anything adoption with open source. More and more of some of that innovation is coming out of the open source space and

many more of these tech firms. know, Google is one of many examples, but Google puts a lot of its innovation in the open source world. Kubernetes originally was one of the big projects that Google was one of the earlier developers on it, and Red Hat adopted what Google had done. And we made it basically completely open source, but global and very, very resilient. So that model, I think, works well. And I think you'll see a balance where the open source innovations on the AI side start to match some of these others. The problem is, you know, if you're building a massive large language model on hundreds of billions of parameters that cost 50 to a hundred million dollars. You don't have that, that, that capital in the open source community. Yeah. (30:17.112) Well, there was a big debate about where frontier models come from and who owns them and what bias has been programmed into them. Because when it's a closed system and you rely on these frontier models for the capability, you don't know potentially what's been used at the core and how it might respond. Because the provenance of how a model behaves is a very difficult thing to understand. So yeah, there's this bias that says a closed system may not always create the thing that we really actually wanted or is the right for the future. As soon as we get to a place where we can build these models ourselves, maybe for individual nation states or individual organisations like an Obersource one can afford it, then I think it will get a lot better. But you're just right, it's cost prohibitive to get the big models out the door, isn't it? I think there were a couple of threads just to build on. One was this notion of nation state and you touched it already, Richard, with sovereignty and the solution to that fundamental problem in countries and states where they're maybe so far behind the technological state of the art that the catch-up action is too expensive and almost impossible to try and orchestrate. That was one area where I think this is open source is interesting. the second one is it doesn't always have to be large language models, of course. The small language model and varying smaller instantiations feel to me like to have a lower barrier to entry, though maybe more specialized. think you're really spot on. I think the benefit, well, there's a cost benefit of obviously typically using open source, right, versus closed systems. But I think that, you know, particularly in the AI space, because things are changing so fast. mean, the tools that people are using today, a year from now, be using something completely different. I'll give you some examples in financial service. that I'm just starting to learn about what's coming out of the academic side, for instance. But I think around innovation, if someone doesn't have \$100 billion of capital to invest in data centers and other things, the open source becomes a very, very attractive way to stay very close to the edge in a very cost-effective way. But still, some of the more interesting innovations, it's not dependent upon (32:40.92) having massive, massive compute capabilities, you know, will be coming more out of the academic side as well as the open source side. So I think there's a nice balance. People really need to be adopting both approaches at this stage, taking advantage of what the private companies are building and, you know, hopefully negotiating a good deal on price. at the same time, there's a lot of innovation on how do we make these large

very expensive models, how do you make them more cost effective to use? For example, Red Hat we now have through an acquisition of Neural Magic, we've open source, and it was an open source company, we have what we call VLLMs. So, you when the large model, the large language model is built or any foundation model, you know, it's sort of fixed and static. And then we ask it questions, right? And get responses. That's sort of generative AI. And those questions require a lot of inference analysis. What is the answer to this question? And that consumes a lot of compute power, computes, you know, huge amount of access to data and information. And that's what we call inference. And that's turning out to be the most expensive part of AI once, because once the model's built, it's, it's built, you know, you could, you need to update it and you can create small models off of them and all the other things like Dave, Dave hit it right on the nail. But I think it's the inference cost now that's a big challenge

for most people. And you don't quite realize it until you start using it and you start getting the cost of the inference as well. it's that thing about agentic becoming the new business model. How do you redefine your business using agentic processes? You're essentially potentially trading salary for tokens. Now, salary is a thing that is...much easier to control from a financial perspective, but tokens aren't always. so you're inviting very busy. And one of the early things that happened with the cloud era was the variable cost that can be accrued around that created some pretty big surprises for some organizations that weren't on top of it. This feels like another way of potentially losing a little bit of control if you're a finance officer to say, are variable. (34:58.934) salary doesn't have to be so variable, or at least I understand what's going to happen with it. It be interesting your view on, are organizations prepared for that sort of, when they start to bring this in, they're going to have to think very differently about how they manage their cost basis. Yes, and I think what I see with most institutions, and again my view is a little more heavily weighted obviously in financial services, is they see these tools that enhance the productivity of their workforce. Okay. Obviously, think some of those roles eventually as these capabilities get really, really predictable and accurate, some of those roles will change. They won't be doing the mundane coding. They may be not pulling all the different data from 50 different data sources and manually trying to look at, you know, what's good data and bad data. You're going to have sort of agents in the agentic world sort of doing some of the really mundane, boring stuff. My background is deeply in modeling most of my career. And you you spend 80 % of modeler's time is getting the information in a place that you could actually start to model on. Now there's hope in a gentic world. You know, have agents that do a lot of this really manual tasks in some sense, very hard work actually. And it's also important that it does it quicker too. So it doesn't take a team of humans to go through a massive database and set it up to build models on. That could take months. Agents potentially could do that in days and maybe even hours. Let's maybe use that as a lever into the other side of your job, is the financial services specialism that you have. Financial services is an interesting world for a number of reasons. We've touched on its legacy. And we touched very briefly on the regulatory issues. I wonder going into this new era, what your take on that is in the sense of the financial services were relatively late to the cloud because of regulation and then they accelerated like very quickly into the cloud. I thought there was an interesting little movement that happened there. Are you seeing the same pattern emerge or are you seeing regulators get in on the action earlier this time around? first, definitely regulators are getting much more (37:22.328) uicker and agile in terms of how they manage risk. But I would say the one, one of the biggest challenge that came with the, with the big public cloud providers was this notion of lock-in, right? And we, and you know, I've written on this even going back 10 years, cloud concentration risk is sort of the famous term. And, you know, and, and when you have lock-in, you're, you're, you're sort of, your applications are using very specific, unique capabilities in one cloud. that if you need to move that application, you can't. It's very difficult. It takes a long time. have to redevelop certain capabilities. Or you have to create the same application but build it for two different environments, right? Because your uniqueness is in each. that's sort of what regulators call that vendor lock-in. In Europe, we have a Digital Operational Resiliency Act, which is ensuring that firms have the ability of running mission critical workloads, not just all workloads, on different platforms. It could be on premise, it could be in different clouds, could be eventually on a sovereign cloud. But the point is they have to have portability. Because if a system goes down, and we see this now in the Middle East, data centers were bombed in several countries. If those goes down, you need another platform to run on. And just being in different regions in one cloud is not necessarily foolproof, and as some of the examples we just saw the last two weeks, three weeks. So I think that that's one of the challenges. And when we move into the AI side, there is that same concern about being critically locked into a single technology that's closed and that you have less agility in choices. It might become commercially cheaper to do it that way. That's very true. But when I see, you

know, group CIOs at large global institutions, they're now seriously, carefully looking at the AI enterprise platform, the infrastructure to make sure that they can use hundreds of different tools, but they can manage that in a consistent way in terms of regulatory requirements, in terms of safety, monitoring systems, monitoring employees, how they're using these systems, you know, all of that stuff. So we saying this. (39:48.078) so the new enterprise stack as it's emerging and it's becoming reasonably clear now what the layers are going to be in the stack. There are a number of different variants on it. I prefer one that actually we use, is a relatively disaggregated version. So it shows the context and semantic different from agents, infrastructure, and then different from sort of the tooling layers. And then obviously sitting underneath that, you've got base layers, systems of record, things like that. So is the conversation going on in the industry that you're seeing at the moment that's picking any pick a layer, any layer. So I'll pick, let's pick the semantic layer just for random one that the lock in issue might exist at the semantic layer. And then also there is a potential locking issue at the model level, presumably. So therefore you might have agents that are active in your organization that are using context. and data and then a model, but may have to swap onto using alternate versions of that infrastructure for resilience. I must admit that is a level of complexity. hadn't actually got to, I don't think, but that introduces quite a lot of difficult thinking, doesn't it? Yeah. you know, I'd say financial service firms in general, historically have been, because they're in so many different lines of business, they have so many products. and different kinds of customers. They have built lots of silos, right? And one of the areas of great concern is when people were modernizing technology, the goal was don't just lift and shift into the cloud because all you're doing is creating the same siloed sort of complex structure that's very, very expensive and difficult to manage. You're replicating that in the cloud. You know, so you haven't actually really modernized what you're doing and how you're managing all these disparate systems. And we're talking about tens of thousands of systems, not just hundreds. So, and I think when, you your description of layers is exactly what everyone is, you know, you're spot on in terms of exactly, you know, what are the different layers and the different types of tools and capabilities and how even locked in a middle layer (42:10.776) can have an impact on what you can do in the layers above, in terms of what applications. It also could have an implication of what kind of chips you could be using to do your AI workloads. So all of it has to be holistically viewed. And this is what CIOs and CTOs and boards look at is, are we building something that's properly resilient in terms of security, but one that is optimal in terms of taking on new innovations, which is happening so rapidly in the AI space. I can't keep up, literally. You know, can you take those innovations and fit it onto your existing environment? Or do you have to go and build a whole new environment to support it? And that's the biggest challenge with such rapid change, right? Well, one of the things that we've observed on the show, probably over the course of the last 12 to 18 months actually, is the rate of innovation in the tech world.

is far outstripping the ability to adopt that in terms of sort of speed of change. And, you know, one of the aspects of adoption lag, I think that get massively underestimated is some of the complexity you're talking about. It's not necessarily as simple of taking one tool and applying that one tool and then, you know, great, great things arrive. There's a set of complex things that need to go on within, particularly within a regulated industry. that need to happen to allow the adoption of those tools. Yeah, and I'll give you one specific example. In the UK, the regulator here, have GDPR. It's a privacy law that's across Europe. Many countries have similar laws, rules. So it's about protecting the customer, protecting their identity, protecting their data is confidential, but also requiring effectively permission to use that data. that the customers give an approval. So when you go on the web, you say, do you accept the cookies? No, I only accept what's required. That's an example of what GDPR sort of provides. You dictate how your data can be used and what data can be collected. About, maybe it's two,

three months ago, the UK regulator extended GDPR into the agentic world, into the autonomous systems world. Now, and when I saw this, I was sort of shocked. (44:34.858)

Okay. If you have some sort of a Gentic system workflow that's automated in GDB, the customer has to be told about it. He has to give permission and he has to have a right of refusal. Just think about the web, right? You get onto a page. And so that could be incredibly complex. If you have many, many autonomous systems doing different bits of a whole process happening at machine speed, not human speed. Yeah. Now, maybe it's a legal thing and they check in the box and they say approve it. And to be honest, human beings won't know what those behind because you don't know what's behind there, right? But it does give you a sense that the regulators are now much further along than they were, say, with the cloud side, right? They caught up, took them a while. And I think that's exactly, Dave, what you're highlighting is that there's these complexities that The regulated institutions, not just financial service, but all regulated institutions have an extra degree of complexity that they have to manage. think the regulator should force explicit approval on everyone and they can send it to your phone. You just be walking down the high street and you just get constant approval messages, request, request, request, request. I want to your data, you don't know why. It's like, well, you're not going to get the special thing at the end of it if you don't click approve. the one is always that... Yeah, you'll miss out FOMO. But on the cookie one, they made it when you hit reject. Some organisations just say, all right, fine, you've rejected. But others go, okay, well here's 20,000 cookie providers. You now have to individually select each one that you don't want to pass the data. And there's no deny all, is it? It is literally releasing the Epstein files in three trucks worth of paper printouts, isn't it? But it is that sort of like, I'm going to increase the friction so much that you lose the will to live to go through 2,000 check boxes to say, I'm going to deny the access. I think when the regulations come in, the serious point is, and they're starting to do this in the UK with subscription services, where they must provide a one click cancel button. (46:48.554) an any subscription service. It should be the same with you're not allowed to present 2,000 tick boxes that you have to individually go through. Once you deny, you deny, that's it. I think sometimes the regulator misses out on the usability and the practicality of how the system can be done because there's always going to be somebody who tries to get around the rules, isn't it? Someone very smart in that kind of world is going to create an agent. Yeah, that just goes down. agent that automatically does all of that for you because you've given him what the permissions should be. So, so there, you know, mean, AI can actually help us as well, protect us, not just inundate us with all kinds of new stuff that we don't know how to, how to, how to judge. Let's hope so. Hey, let's try and maybe bring today's conversation to a bit of a close. Let's, let's do a little bit of future gazing in your world. Love a tech prediction. Come on, Dave. know we love them. Oh, it's always, always a difficult one to judge. Although Rob maintains that he's the best. at Tech Predictions. And we've done some trends for this year, which we're going to check in on towards Christmas. we look forward to that. I'm tracking very well against those trends, David. I've been looking at them and I thought, actually, crystal ball worked this year. It's failed many times, but this year it seems to be on Well, let's time be the judge of that. It's like the classic phrase, 40 % of the time it works every time. Yeah. On that note, Richard, let's have a look forward. What are you seeing happens in the next 18 months to 24 months in the world of financial services as we get into this scaling world. I think we've talked about on the show that we've moved from POCs into at least the challenge of trying to scale this stuff. Do you see us at scale in major organizations in the next 80 to 24 months in your sector? I think for certain applications that aren't mission critical, because that's where the regulator has a lot of restrictions on. I think we're going to see huge changes. I think there's going to be huge changes in how software is built within financial service institution. I could just share the example of Red Hat, right? We have our entire engineering organization now adopting a full agentic workflow sort of structure in terms of writing code with agents. It's human computer scientists having a set of agents available to them. They decide what LLMs or

whatever algos they want to use to drive the agents. (49:16.416) And they do part of the coding work, the human checks on it. And what we've seen, for example, in Red Hat, think we have 12,000, 14,000 of these engineers. On the AI engineering side, and I just learned this two weeks ago from our head of engineering. I was at a large bank with a group CIO, and we had a workshop with them. And he sort of shared that. You know, the AI engineering team has been doing this for almost two years. I was sort of shocked, right? Because I'm thinking, I thought agentic was sort of more recent. And he said, no, no, we have guardrails on certain agents. They do particular coding tasks or they do some data extraction, whatever it is. They do very specific tasks. And he said, some work, some projects which typically took six months of, and open source company, have top-notch engineers globally, right? It would take them six months, a team. Now they can do that in three months, in some cases even a month. So that's the productivity increase. And I think that's the benefit I see, the true benefit early on for firms is about workers becoming more productive, the work becomes a little more fulfilling intellectually. And there will be probably some jobs that will be eliminated because of that. But I think overall for institutions, that improvement accelerates the ability to adapt, to changes, to shocks, to all kinds of things. And that's the positive aspect I see out of this. As long as we have proper guardrails, safety, transparency, all those things that regulators are insisting upon. (51:13.454) Now we end every episode of this podcast by asking our guests what they're excited about doing next. And that could be something in their personal lives, like they've got a great restaurant booked at the weekend, or it could be something in their professional lives, or maybe a little bit of both. So Richard, what are you excited about doing next? Well, given we've had such a wonderful discussion about AI, I'd say the piece that I'm sort of really interested in is in sitting in a chair, tying myself down and trying to keep up with all the things that's going on in the AI space. Maybe you could use open claw to write an agent to keep you up to date. Yeah. Yeah. Probably with the technology, you can do all of that in a weekend, you know, Rob gave a nice example of, of, know, of when you're asleep, it's doing work for you. And that's, to me is great. That would be fantastic. Rob has got a team of architect agents deployed at all times. Haven't you Rob? I don't do any work anymore. It's just how long it takes everyone to work it out. maybe they've already always known. Are they in a virtual ivory tower? Yeah, no, that's what I call the sandbox I put them in. That's what I named it. (52:26.478) Sandbox tower. Sandbox tower. I'm not letting them out yet, but when I do, it's going to be impressive or disastrous. Most likely disastrous. God help us. Richard, thank you so much for spending some time with us this morning. again, apologies for taking so long to actually go from our usual meeting you and then actually getting the recording done. hopefully it was worth the wait. We certainly enjoyed it. you guys are so good. I had so much fun and I actually learned some stuff from you guys as well. So, appreciate it very, very much. Thank you very much. Thanks very much.

If you would like to discuss any of the issues on this week's show and how they might impact you and your business, please get in touch with us at realitiesremix.capgemini.com. We're all on LinkedIn. We'd love to hear from you. So, feel free to connect in the end if you have any questions for the show to tackle. And of course, please rate and subscribe to our podcast. It really helps us improve the show. A huge thanks to Richard, our sound and editing wizards Ben and Louis, our producer Marcel and of course to all our listeners. See you in another reality next week.

About Capgemini

Capgemini is a global business and technology transformation partner, helping organizations to accelerate their dual transition to a digital and sustainable world, while creating tangible impact for enterprises and society. It is a responsible and diverse group of 340,000 team members in more than 50 countries. With its strong over 55-year heritage, Capgemini is trusted by its clients to unlock the value of technology to address the entire breadth of their business needs. It delivers end-to-end services and solutions leveraging strengths from strategy and design to engineering, all fueled by its market leading capabilities in AI, generative AI, cloud and data, combined with its deep industry expertise and partner ecosystem. The Group reported 2024 global revenues of €22.1 billion.

Make it real.

www.capgemini.com



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

Copyright © 2025 Capgemini. All rights reserved.

