Capgemini

CR007

The Cloud Value Flywheel Effect pt.1 with Anderson, McCann & O'Reilly

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

CR007

The Cloud Value Flywheel Effect pt.1 with Anderson, McCann & O'Reilly

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

[00:00:00] I'm not going to do that. In this first part, we're going to talk about the pyro and pyro, fuck it. Three for three, Dave. Well done. Yay. I'm going to say good work. I'm going to say good work.

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.

We're excited to be back for season two and for what's coming in 2023, lots of good things on the horizon. Sjoukje, Rob, did you have a good break? I spent my Christmas holiday cleaning up the attic because we are moving houses somewhere between April and June. So I had a very [00:01:00] stressful Christmas holiday this year.

And you Rob? I did virtually nothing other than go through the Christmas motions, which I was very excited about. We all know about your Christmas motions, Rob. However, I did have three weeks off, so I got back to work and had mild panic on arrival at my laptop, trying to work out what I did for a living.

But I got back into it. It's good to see you guys and excited about the year ahead. To kick the year off, we've got a special two part episode talking to the authors of The Value Flywheel Effect. Power the future and accelerate your organization to the modern cloud. So huge welcome to David Anderson Mark McCann and Mike O'Reilly. Guys, it is great to see you. Tell us a little bit about yourselves. My name is Dave Anderson. I'm a author and the value flywheel effect and contributor service agent based here in sunny Belfast. And I'm also a practicing architect. So I still like to do work as well as write books.

But I'm married with two small children. So very happy to be here. Welcome. I'm Mark McCann, architect with Globalization [00:02:00] Partners. Helped with the value flow effect book. I'm a

summery of the guys, practicing software architect. Currently employed with... globalisation partners, contributor with the Serverless Edge and co author with Dave Mark on Flywheel Effect. Good to see you guys. In this first part of a two part episode we're going to talk about the good work that you guys did at Liberty Mutual and how that led to the thinking in the book.

So let's start with when did clouds first start to come onto the horizon? We started massive virtualization push, which was brilliant. Give us a lot of flexibility within the data center. And then we started the burst and they do less, but around about, I think it was 2013, I took a CTO role in, in, in Ireland.

And then we started. became the architecture team and I was quite lucky because around 2013 [00:03:00] we were starting to have more serious conversations with AWS to talk about what could we do. So for me personally to be sitting at that table representing Liberty Mutual talking to AWS about we're now investigating public cloud that was a fantastic opportunity.

But then as my peers in the like the enterprise architecture team we were as we were looking around a lot of the brilliant technical leaders. Their first concern obviously was security, networking, data, on ramp, all the infrastructure type concerns, which absolutely are number one.

But I was sitting thinking our specialist is software. So how do we write software in this new environment? And that's the problem I took back the, our team in Belfast is like, this is going to happen. It'll, it's not going to happen overnight, but we need to think, a few years down the line, what does software look like?

And that was the question we started asking ourselves. That is a big adoption step, isn't it? When people, when a penny drop moment happens that actually we have to architect for [00:04:00] this, that there's something fundamentally different about it. It's not like we're going to get a thousand page contract that's going to, that's going to lock all of this in for us.

Yeah, and all the whole ideas of all the cloud principles around elasticity and all that stuff. How do we bake that in? We can't, it's got to be completely automated. And I used to always use the wave of disruption. This is common, the great wave. I used to always remind people this is common, so we need to start thinking differently.

And what was your wider conversation with the rest of the organization about it at this point? Did, were you declarative about it? Guys. Very quickly, we need a cloud strategy and from an LT perspective, our leadership team perspective, we're going to need to take a view on this. Or did it start more tacitly and more technically, if you like?

I think I think there was, I think we knew that we needed to change how we wrote software. And we were very open in those conversations. And there was we didn't know we had no idea around as architects what the answer was. And we were exploring different things as an organization.[00:05:00]

We had a kind of pretty mature like Java EE type practices around that traditional, enterprise design patterns. We had a kind of a lot of exploration around PaaS and Cloud Foundry, which was fascinating and that whole way of, a pairing and TDD based stuff. We had the big kind of Docker contingents looking at containerization and that kind of swish.

And then what really happened. For me, the penny that dropped for us was when Lambda came out in 2014. We were at reInvent when Lambda was announced. We thought, ooh, that looks interesting. But what specifically was it about that caused a bit of a light bulb moment? Something, Mark, when you're always quite keen on is that the, we used to have a great friend of ours, Ed Carmody, one of the, one of the lead architects, we say this phrase.

No undifferentiated heavy lifting. No, we're not going to do things that we don't need to do. And he's been saying that for 15 years. So that phrase always stuck with us. Yeah, I think a lot of the, we were very aware of the [00:06:00] operational burden of a lot of the systems we were building and some of the fragility of them, especially at the scale we were operating at.

So whenever some of that. Operational burden could shift to the cloud provider. That was a big, ah, okay. So we can focus more on the differentiating value. We can position ourselves hyper value chain, and we can offload a lot of this to the cloud provider, I think. That penny dropped, that, removing a lot of the things that we would have to have had to do in the past was now.

The value line moved. We were able to move ourselves up the value chain a little bit further. Yeah. And we actually used to we used to talk about the line, something we would talk about, like from way back then over 10 years ago we talked about where will we draw the line?

This idea of and really what that was, what that's become now is a shared responsibility line with a s But we were, we've been talking about that line before we moved a SS. Where's that kind of line of the platform and what we need to do? Yeah, and a lot of it was around time to value because we've been through this journey from 2000 [00:07:00] all the way through from mainframes to, big SOA systems to BPM systems to more microservices, Spring Boot, Docker, and then Lambda.

came along and it was all about how can we get value out more quickly. How can we reduce the lead time for, hey, we need to order up more servers or more AFLs or more racks for to now, okay, can we rapidly get the value into the hands of our customers? Can we more quickly get that value to who needs to have it?

In the long list of things that happen in that route to value and like route to utility and actual, customer centric service, what we're using to guide. you in that in terms of understanding this is a value component versus this is a undifferentiated piece of heavy lifting? There was, I think there's probably two important things there.

I think the business leadership of Liberty were always very clear on what value looked like. There was never any confusion of what was important and it was always absolutely the policyholder number one, never any mistakes. So there was very clear messaging from [00:08:00] leadership from out, so from a business perspective.

That was crystal clear. But around that time we discovered Simon Wardley, starting to talk about Wardley mapping and how he, and it was the Wardley mapping technique was fascinating, but we completely did not understand this. We were lost. Maybe just say a word, Dave, for those who haven't heard of it just say a word on what it is.

So Wardley Mapping is from the researcher, Simon Wardley. Simon had been a CEO of a company and he just realized that a lot of the strategic narrative wasn't good enough. So he came up with this concept called Wardley Mapping. We pretty much start with a, there's an x and y axis where there's your customer at the top and a value chain.

And the value chain starts from the most visible customer need. at the top, and then the things that are dependent on that need. If you need a cup of tea, there's a dependency on a kettle and some water and a cup. That really clear value line. And then across the x axis, you've got the evolution of technology or the evolution.

area access which really goes [00:09:00] from genesis which is something that's brand new and the fact that it's never been done it's like it's a wonder to custom where we understand what this is but it's quite difficult to build to product which is there's customer need and customer demand for this to a utility or commodity which is the price of doing business.

And I think that, that concept of wordly mapping, we just thought was fascinating. But more so the way Simon, his language and his mindset, how he talked about technology evolving. We could see that happening with what we were doing, but we didn't have the language to talk about that. We could see things going, that's become a commodity.

And it often frustrated us when we'd see software teams. spend a lot of time on a logging framework. We said go on a logging framework doesn't help our policyholder. We should just buy that I'm sure they'd be fascinated by it though. Yeah so and it was really that gave us a language and a way they share our thinking on that.

Yeah and then we really started to think about your who are [00:10:00] users? What are their needs and what are we doing to meet those needs? And then for the components of that value chain that Dave mentioned, how evolved are they? So we started to think in this way, and once the penny dropped, we started mapping everything, right?

We started to draw maps and talk about this in this way. And when you mapped out the value chains of components that we had, At our disposal at that time, a lot of your on premises stuff, a lot of, things that may be a long lead time or pretty well custom built.

And then Lambda Kims comes along and it's like a commodity that you can just leverage. It's ah, okay. We can use that, but that there is a lot of that undifferentiated heavy lifting that we talked about earlier and allows us to, position ourselves higher up the value chain. We can just leverage some of these things.

And you also need to be really clear on what's valuable, like what's the valuable piece of software. And often for insurance, it's like the rules engine. And rules engines had often started, we'll just need one or two rules for this and they grow, they've come this unwieldy custom complex thing, and we spent lots of time dealing with complicated [00:11:00] rules engines that didn't have to be, so we could just see this everywhere.

So it sounds like in, like within your team and within a group in the organization, you're developing these like very forward thinking pioneer and approaches, bringing in as much new thinking as you can, organizations. Yeah. generally have challenges then of communicating that out a bit more widely to wider, even wider IT stakeholders, nevermind wider business stakeholders.

So take us a little bit through that, how you communicated outside of the sort of best practice area that you were working in. I would say the only way to drive this type of change is via show, don't tell. There was no way we were going to say, this is the direction of travel. Here's a document.

Now go forth. We had to pick something. Build something and then actually show real kind of value, show low cost, show speed, the market, show rapid response time to change. For me, that's, that was the only way. And it was [00:12:00] only until we had maybe like a portfolio of maybe 10 or 12 really solid projects that you could start.

We didn't do it in secret, but no one's going to listen to you unless you've proved out the value of this idea. Yeah. There, there's a bit of a. top down approach and a bottom up approach, right? So on the ground, how do we enable these capabilities with the teams on the ground? How do we enable and empower the development teams to be able to embrace these techniques and practices and processes?

But also, what's the language for the C suite? What do the presentations look like? What are the decks? What are the real talking points that help them get get where we're trying to, we're trying to achieve? Yeah, I would just add to that, like the bias for action, so really what we were trying to promote within squads at the time, certainly what I remember is, a bias for action.

So try this thing, push that sort of agenda forward, but a lot of the squads were getting a lot of the confidence from the architecture team. So the architects would communicate with squads would, we would work with certain areas and, we had we would [00:13:00] encourage teams to, try something else, move something from Genesis through three custom and attack the end of the product space. So it's really the teams that were really good at this were had a real strong bias for action. Dave's point was, we did it and we, we learned a ton along the way.

It didn't always go according to plan, but again, we could then use that information that you know. And I think that's where like situational awareness really comes in and just Simon Bordley talks about this a lot, but we were able to. And we were absorbing all the stuff, we were reading all the blogs, we were going to all the conferences, we were just sponges for a lot of the stuff.

But we had better situational awareness and Mike says, with that batch fraction, we were, they were almost like, they were our sensing engine. They were probes to see where are the impediments, where are the things that are slowing us down? And us as the architecture team, what can we start to invest in to get ahead of that?

What are the blockers we can remove? New capabilities that will make that easier for the next team. We started to really build up, whether it was maybe [00:14:00] it's just internally we were talking about this a lot, but we had a really good situational awareness of where everything was.

And we would draw on the wall and we'd talk about it quite a lot. And we challenge, we challenged each other and that situational awareness, mark did that, was that industry innovation, situational awareness or was that applied? to, to Liberty at this point. So your situational awareness wasn't just like an academic one.

It was like as an organization, this is how we're mobilizing at the moment. I would say both. Yeah. I would say actually both, but I think the organization was probably more important because when we're our small architecture team was starting to grow by this stage, coming to 2016, 2017.

So we get done maybe nine, 10 people and. As the organization in Belfast was also growing, we were working across all areas of Liberty Mutual, across all lines of business, and also geographically, with people dipping into all sorts of projects. When we came back to our daily stand up or weekly more detailed meeting, we could sit and go, what's going on around the [00:15:00] entire enterprise?

Yeah. And see patterns, you say there's an experiment here, an experiment there. Which is good, right? It's like when you do look at success in. Multiple organizations, it generally comes from that type of approach like multiple lighthouse projects that actually when you join the dots of those lighthouse projects it becomes something more.

Yeah. Substantial. Exactly. And then I as well had the from knowing the C suite and being in those conversations, like I knew where, what the concerns were, what was happening. And then also as being a big company, you talk to a lot of vendors and people will share, come in and share ideas.

So you're getting a pretty good read of the industry as well as seeing what's happening internally and you can join the dots. It also sounds like you used an agile way of working. Was that a methodology that you were using? Yeah. Yeah. back then? Yeah, absolutely. The CEO at the time, James McLennan, famously said that around that time, we decided to change Liberty Mutual.

So we had a cloud transformation, an IGS transformation, a customer centric transformation, all at the same time, [00:16:00] which he says was a fair bit. It was good to do, but looking back, we may be shooting it on all three at the same time. Which was good, it was brilliant. It was a busy time.

We had a lot of figuring out to do, but I had been a proponent of agile since I remember even before the manifesto was signed. We had a lot of people who had deep inside knowledge and we were also trying to work that way of working as well, push that way of working. With the teams and everything going on, there must have been did you suffer a lot from the cognitive overload?

Obviously there are those who are better with it than others, but you've got people transformation, business transformation, technology transformation, you're fast followers on adopting very new technology on the edge of what's just being created in the, in that thought process. Must have been a bit of a strain on the organization.

Is there any points where you thought it was getting a bit too much? Or do you always find you obviously found a way through, but must have been some testy times with all of that. It's funny. I've been reading and experienced software for many years. And sometimes some of the older practices and techniques, you realize they're not so [00:17:00] bad.

So good old fast and change management. It was vastly underrated, so at least just talking to people, explaining what's happening and why it's happening and just giving people a sense of security. I think the human element never changes, does it? So how you work with the humans being empathetic, talking them through things, being patient sometimes.

Yeah. Yeah. Obviously, one of the things in the book we talk about is that's psychological. Safety and certainly I think one of the roles that the architecture team provided was, working with squads and providing a certain degree of our cover and, so that would be like setting realistic expectation, so if we had like a monolithic product that we were looking to migrate or we were looking to do something, merge a new area or something like that, we would always start relatively small and limit You know, what we would take on and again, it would just be celebrating the wins, again, the culture is very important we're, we pride ourselves on the engineering side of things, and there are certain kind of first order principles that we would always adhere to, we get, we architected, [00:18:00] but yeah we certainly, that's, that, that is something that we Yeah.

Yeah. We did consider quite a bit, we always talk about like meeting teams where they're at and we wouldn't try and put too much on a squad if we, that team didn't have that level of experience, we'd always set, we talk about next best action quite a bit and, and small incremental types of approaches.

And we also celebrated like engineering excellence as a thing and talking about things like software is a team sport. There is no hero programmers, all that really good mindset, where people not resources, a whole bunch of things that we took from the Agile community and reinforced those and helped teams and peers, not only in Belfast, but across the entire enterprise.

And, be very clear about being collaborative and working with people and helping people. So that's as a generous kind of mindset was absolutely important. There was no competition, it's about we, we all win together. So that was absolutely important. Yeah. So guys, you were very fast [00:19:00] adopters of new technology being deployed by the cloud providers.

Was there any point where you exposed some big issues that gave you? headaches with the technology, or did you find those versions going live with things like Lambda did work as they were advertised?

Pity you didn't make it in the main flow of the thing, eh? Yeah, I thought about it went, oh yeah, but the flow had moved on. So I thought you know what, I should ask it now, I think. Yeah, I'm glad we finally got to it, guys. I think we were, I don't know if it was lucky or by design, but we were always heavily investing in engineering excellence.

So we're always at the forefront of how can we have more confidence in our systems and how can we test things better, right? How can we make sure that we have good CICD pipelines and we were able to have stage gates and confidence at every level of the SDLC, right? So whenever you introduce a new technology, we were bringing that.

your mindset and that, engineering excellence approach to anything new. So [00:20:00] all the sort of the, especially because we're insurance and, policyholders number one, you can't really mess this stuff up. We always had that culture of doing the right thing and doing it well. So like something like Lambda.

It’s a different sort of paradigm, it’s a different model, but things like unit testing, integration testing, end to end testing, security, those things were always top of mind. So we were still always thinking about how do we apply those same, that same mindset to this new technology. So yeah, we were lucky in that regard.

So we were able to catch a lot of the early stuff. It’s good to think as well in kind of design patterns and what was the story that it would work for versus what it wouldn’t work for. And there’s also, again, to be honest, not just with the cloud providers, with many vendors, there’s many stories where we would come in with a fantastic idea of a product that wasn’t quite ready.

And we would knock the edges off it and try it and give feedback to the vendor. And lots of those vendors are very successful right across industry but there was [00:21:00] lots of. platforms that we incubated for many years and again, just good solid engineering practices to try and, use what was working and maybe compensate for what was not quite ready yet.

There’s often a thing that when you take a product from a vendor, there’s this expectation that it’s bulletproof and it’s going to work perfectly, but when you put it in the hands of engineers who are going to create things with it. They often work with it that in a way that wasn’t perceived by the original, first design of that product.

So that feedback loop with the vendors and coaching them on how you’re going to use it, what you’re going to use it for and how you’re deploying it, I think can be very valuable because then they, they’re very motivated to quickly fix those issues because they’re seeing real world. Use on something very new.

And there’s almost for me, there’s like a, there’s almost like a two step process with a vendor is the products interesting and are the only partner and not improvement journey. And I guess as a Fortune 100 insurance company, the bar was very [00:22:00] high for security, compliance, auditability, encryption, right?

You name it. So we were always having those conversations with any sort of new technology around, Hey, what’s the security story you like? What’s your encryption addressed capabilities? What’s your encryption in transit? How do we audit this? How do we do this as code? How can we do this in a pipeline?

So a lot of those early conversations were all about that type of thing, right? How do we make this fit for purpose for a big enterprise? And that’s a great measure, I think, with the point you’ve made is the product interesting? Is the vendor willing to work with us to improve it because we’re sure we’ll find some hiccups on the way and then have they got all the necessary, table stakes associated with what you’re going to need for the industry and environment.

Vendor engagement and flexibility when you’ve got a pioneering mindset, I think is nigh uncritical. Yeah. You can’t take rock, you can’t take boxed in service at that point because actually what you might want to do might be way more valuable than what they’re thinking of doing and what [00:23:00] they’ve got prioritized on their backlogs.

So it becomes ecosystem like I think when you’re particularly when you’re in the early days of a paradigm shift like cloud. And then like Dave mentioned working with those vendors and influencing their roadmaps. Yeah. And articulating your needs and making sure that they are incorporated into the features that they release as part of that release cycle.

I think one of the. big success factors for us is we didn’t just let the vendors come in and run their workshops and do their show and their song and dance shows. We would typically embrace it, try it ourselves, and sometimes we would even tailor their materials and workshops and make it work for our context.

And that then made it real for our teams. And it wasn't some vendor coming in doing it. It was one of us. Doing the workshops or doing the setup and then that gave us the feedback loop. So we were able to find out what the challenges were, where the impediments to flow were for teams.

And then we were getting ahead of us so that it increased the chances of that technology succeeding because we took a personal ownership of it. Being [00:24:00] applied, right? Yeah. And there's also a thing with vendors is that like you said, I don't want to talk to the salesperson. I'm not really interested in that conversation.

Yeah. But if a vendor has a good, strong account manager, that'll probably go well. And there's some vendors we spoke to that, that maybe the account manager was just okay and didn't go so well. And it's not all it's not all about, but software engineers, a really strong account manager can really make a massive difference.

Yeah, because it becomes a content conversation, right? Not a, I'm selling you on this conversation. What I'm really interested in getting a sense of though, at what point did, was the catalyzing or tipping point, do you think, where you look back and you go, what you guys were doing, what the cloud transformation was doing, what the wider business transformation was doing, when did the seesaw go over?

And you think actually we're now predominantly we might not be fully transformed when you say count workloads in the cloud or something. But actually we're really working in a different way now. What was that feeling like? And how do you know what were the key steps moving up to [00:25:00] that? Do you think? Oh, that's a good question.

There was probably, there's probably two aha moments for me personally. I'll let the guys share what they were like. There was probably locally. We as we were celebrating engineering a different way of working, we had a sort of a, I would say a transformational event. I think it was in 2017. We had an open space, which was, it was an on conference, but open space is different in the sense that.

It's a very specific format, open space technology. So we had an open space to and you ask a question and people participate. So we had a big, we, and the way I pitched it was that Would anyone like to attend this little company? So we got maybe 60 percent of the conference. So you have to attend.

Yeah, so everyone attended by choice. And then everyone landed in. And my boss the day before said, So where's the agenda for this two day event that we're spending a lot of money on? There's no agenda. We're gonna figure it out when we get there. And he was like, okay. I love that because that kind of different [00:26:00] approach to work from yeah complicated style of we have to plan everything and everything has to be monitored down to the last moment, which is like no, we're just going to get some smart people in a room and we'll work out what to do next, no?

Yeah, oh, completely. And now there was a very, Matt Winn helped us. He was coaching time, so he helped us with the format and just steering us in the right direction. But it was just, but the two day event just really catalyzed everyone and got everyone really excited that this is real.

We had a couple of visitors from the US over and people were just like, this is incredible. And then probably the second one was Which for me personally it was I think it was about 2020 the AWS had a serverless first function event that they only specifically focus on serverless first and they had contacted me about a speaker and Gillian McCann did a great talk on it.

Oh yeah. Werner Vogels was doing his introduction. I was sitting watching the introduction and he talked about this is in the cloud there's two types of way of working there's infrastructure and application. infrastructure is what we all know. And then there's applications. And this is serverless [00:27:00] first.

And this is a type of organizational nirvana. And then the next sentence out of his mouth was companies like Liberty Mutual. And I almost fell off my chair watching it. And I went, wow. Congratulations. This is real. Amazing. It's just, there's, for me, there's something about the engineers and the staff being excited, but this is different.

And then to the external recognition. as a company, you're doing something interesting. But there was many small wins led up to that. Yeah. And I got a lot of it around that 2014, 2015, 2016, all the way through the present day. It was all about creating that right environment that was set up for success.

And we were working really hard as a team to put the right things in place so that our teams could. go fast. And you mentioned cognitive burden, removing some of that cognitive burden. We did a lot of hands on workshops. We did a lot of, facilitated collaborative practices with teams to help them on the journey.

But the penny really dropped whenever it was like, Oh, we need to solve this problem. And like the next day, Oh yeah, we've got something working and it's [00:28:00] in production. That time the market was just incredible and we were starting to see more and more of those wins. And we were really good at starting to capture some of the success stories because it then helped other teams.

Remove any of the fear factor from adopting these ways of working under these technologies. We were able to, and Dave was great at collecting and presenting these in a way that the C suite got and everybody in the organization got that, Hey, teams have adopted this approach. They're able to deliver this really impactful business value in days and you can too.

And here are the processes and practices that you can apply to be the same. Yeah. That flywheel really started to turn. Completely. Toward the 2019, 2020, 2021. Yeah. People would tell you things because like we certainly weren't involved in every project. You couldn't be, it wasn't possible.

I just want to say, did you hear what happened yesterday? No. Remember the project that was nine months long that was supposed to be ready in three months time? They just put the production yesterday and you're like, really, how did they do that? They used serverless [00:29:00] first and well architected and blah, blah, blah.

And we're like, wow, I didn't even know they were doing that. So it was just some team thought we want a bit of that. So it was, it just, so as Mark says, it was a complete flywheel effect. It just went. Yeah. Yeah, we also started to roll out engineering excellence sort of reviews that were periodic and we start the review teams and and let them talk about their successes and the teams were then just There's a flow of fantastic progress, like here we've hit this customer KPI.

We've delivered this new feature with, we've improved our engineering excellence by a certain amount. We've improved our well architected status by a certain amount and we're going to do this next. And we were, we would do this across the entire org and it wasn't to compare one team against the other.

It was so they could all improve in their own context and you started to see that really starting to accelerate. towards 2019, 2020. It to see. So if you look back then, how long did it take to go through this full transformation? That's a [00:30:00] good question. It's probably still happening. If you ask the people downstairs, they're not finished.

Yeah. Yeah. I think, a friend of ours, Sheen Bracillus from Lego, he describes it like a rocket ship. Once you get on, you can't get off. You just keep going. There's always more things, not in a pressurized way, but as the fly will keep turning. I would say Liberty has probably been in constant transformation for decades.

Definitely seems to me that once you get into the new way of working, which includes all of the pioneering spirit, bringing new things in the business as usual then becomes almost transformational rights, sort of small T constantly working like that. Yeah, one of the things I noticed as well, it was a massive sort of signal of change was there was far more experimentation, for me, that was the main kind of trigger, so we talk about like tack and the engineering culture, but the business was far more well in the experiment.

And for me, that was something I was wasn't used to up until recent years, towards the end, my time in [00:31:00] Liberty, like we'd build out a product to try and see if a customer would use it. And then what we'd find is really what they're not looking for was that they're looking for something else.

And then I was actually removing functionality, which we weren't doing, and and for me that, that was a real sort of sign of momentum. And I think the Azure transformation really began to kick in once we had, we had transitioned a lot of our thinking to the cloud and we were beginning to move.

So that was, but I agree, I think it's, I think it's a continuous thing. But maybe to bring the first half of our conversation to a bit of a close, I wondered if you had any advice. So if organizations are maybe thinking about going on a journey like this, then maybe started doing some work within their teams, but actually maybe there isn't a big cloud transformation going on that they can partner with.

What advice would you give them? Because it can be quite disheartening and quite challenging at times, can't it? I think there's a few things. I wouldn't discount some of the models out there. A lot of models are useful, and find what works for yourself. One thing that [00:32:00] we did was the story we're telling, it's not all about architects.

There's many change agents that drove the story. But one of the things that we used was the elevator architect approach by Gregor Hulp. That's you've architects speaking to the C suite. Architects may be driving portfolio change and architects in the teams driving that change. So it was really important that people could operate at different levels at different times.

So I think actually being curious and look at the models out there and taking what you, what will work for you and certainly not taking a single framework and think we'll follow up by the latter and it'll work for us that really works. Yeah, I think we get a lot of value to the elevator architecture approach and it gives us a lot of situational awareness.

But I think that coupled with worldly mapping is a really great way to go. Like even you don't even need to call it worldly mapping. Just start asking the teams who are your users? What are their needs? What you doing to meet those needs and what can we do to improve that? Is a great starting point for any team to start to think about their [00:33:00] purpose, why they're, why they exist, what value they're delivering.

And a lot of that then you can start a tad to the cloud transformation. You can start to tad to, improve time to value better engineering excellence. But I think do, doing the elevator architect and applying the board lead mapping sort of techniques is a great place to start.

Yeah. It's all about keeping your world small. Isn't it? We talked about a couple of times purpose, big challenge with these sorts of things is removing the vanity metrics from conversations and really understand what's the purpose, what are we actually trying to achieve and in starting with that and then just encouraging and plus another really important technique that we haven't mentioned yet was the team.

Is the element of execution, not the individual, whatever else. And then when you're thinking about a team, it's team topologies comes into play. Because one of the most powerful questions that we use with the team topologies work was what's your team's purpose? What type of team are you? Teams often say we're a platform team and we're a value team and we're enabling [00:34:00] this other team.

Okay, so you're three team types. You should only be one. Have a single purpose and do one thing well. Don't do three or four things. That was a great sort of piece of coaching advice for

teams. So Sjoukje, what's trending? So each week I will do some research on what's trending in tech. And this week I'm going to talk about that. According to a study, almost half of the businesses struggle to control their cloud costs. And that is according to a report that was published by business monitoring company Anadot.

And they state that 49 percent of the businesses find it difficult to get cloud costs under control and 54 percent believes that their primary source of cloud waste is a lack of visibility into. their cloud usage. So a huge portion of these inflated cloud bills could have been avoided because most of [00:35:00] the respondents said that at least one third of their cloud spend is wasted each year.

And this sounds like a big issue at this moment, since most organizations are now cutting costs. And really see that most organizations are struggling to get the appropriate insights. And what do we think drives that? Is that the steps of successful cloud transformation? It seems to me to be a banana skin that most organizations, and I say this entirely constructively because I know the one that I was involved in, we did the same thing.

You get to the cloud and you suddenly realize after your migrations, Oh, I might not have all the capability that I need to have. And, cost is one of the first areas I think to demonstrate that. It's also a new thing because just because you're in the cloud, you may still be paying in a capex manner.

You may still be budgeting based on trends because you're just thinking, we'll just do what we did last year. And that's our flat budget, but really the, and the whole idea of FinOps, and we've [00:36:00] talked a lot about kind of GreenOps sustainability is when you. Start to think about go from trend based forecasting to value based, you can scale with the value of your business and your cloud bill fluctuates with the value.

But two things, one, you need to know what value is, and two, you need to have the right cloud strategy and account set up that you can actually see that you don't all have everything lumped in one big monolith. But one of the things that we've taught, we touch on a little bit in the group. In the book is the idea of sustainability and a well designed well architected system should be sustainable.

And you can see your, even your carbon usage, which should be lower. And that really is a nice kind of a proxy or metric for a good cost or a good well architected system. Yeah, I think a lot of organizations don't have good observability in place around their costs.

And we talk about this when we get the serverless and more modern applications, but your cloud bill is your architecture effectively, if you have a properly instrumented, and we do a lot of work [00:37:00] with, and we have it in the the long term value section of the book architected framework, there's a cost.

optimization pillar in that. And we would do this with the teams and say to the teams what's the run cost for your solution? Do you know what it costs to run? Do you have a dashboard that shows that? What are the trends look like? So again, throughout all levels of the organization, cost needs to be front of mind.

Everybody needs to be aware of the cost of their solutions. So they can they have mentioned that value based conversation. The team should be able to tell you, Hey, Our solution cost this much to run, cost this much per request or per event, and we brought in this much revenue. Those are the type of conversations the team should be starting to be part of in the cloud.

Yeah, transparency is extremely important. Yeah. I think there's a change, it's always the accountant's fault, that's what I'll say. But the it's the idea that, We used to buy something, capex, shove it in a data center, I'm going to depreciate that over five years. We don't have to worry about costs now.

So the old mindset of the organization is that's [00:38:00] done and dusted. We did our paper based exercise. We do it again in five years and cloud has fundamentally changed that, which is the optimization cycle has to be. continuous as you discussed. And I think it's just organizations catching up with the, you've got to think differently, but it goes all the way up through the finance lines, up to the CFO and they need to think differently about this stuff.

So I think, again, it's a natural maturity cycle that most organizations have to go through because it's just, it's a surprise unconsciously walking into something completely different almost. Yeah you're spot on. I think it's epic to use in setting. And we had some interesting challenges with some teams, especially at scale.

Hey, we're going to, we're going to do a performance testing run. And then suddenly that month's bill is 20 grand, whereas before it was 1, 000, right? So again, it's just being aware of the elastistic sort of nature of the cloud and your bills will go up and down based on usage and setting expectations with your key stakeholders that.

This will be operational expenditures, not just [00:39:00] CapEx that's been written off over seven years or five years or whatever. It's also a good test for how good a performance test is. That's going to cost you 10 grand and that's green dollars. So what's wrong with it? There was this conversation we had earlier about undeterministic processing.

So you get to AI algorithms, different data sets. How do you predict what this action is going to cost me? So you can do a performance test run. You get it from 20 to 10 to five. Improve the optimization or improve the efficiency of the system. But when you click the button, the fear of, I don't know how long this has to run for to get me my answer.

And there's that type of, how do you bound that context? I don't know if you have any viewpoints on that. We combat that through the engineering side, so her sprint, I'll work with each squad. We'll look at the trends. We'll look at the costs. We'll look at how effective we are. We'll look at safeguards.

We'll look at, it's almost like a form of analysis service. Sometimes, how do we protect from, unbounded events from occurring, those sorts of things, putting in good safeguards, I think

Mark, you talked about that [00:40:00] earlier in the sort of the conversation.

It's problem prevention and really get out in front of that. But our main weapon there is the engineering, rigor engineering. That's the plan, but also the observability of it and just making sure that we are being a wee bit, predictive when it comes to looking up costs and what we're expecting.

Yeah. Just to build on that, just tell the teams that they're responsible for the bill. Yeah. That's the best solution. Yeah. Yeah. That's the other thing. It's like the, what you find is particularly with the teams that are working in more. Sort of modern cloud type architectures is the biggest expenses, the team itself and you got to factor that into things as well.

Yeah, there's a whole total cost of ownership conversation here. There's also especially when you get the, high performing teams the teams almost want to over optimize or prematurely optimize. It's like your Lambda bill's 2 lads. You don't need to, you don't need to spend any more time on this.

Work on something else, right?[00:41:00]

To summarize the advice, really set up budgeting, alerting upfront in each and every layer of your architecture. I see lots of organizations still don't incorporate that in their architecture and also use the highly dynamically nature and architecture of the cloud itself. Scale up, scale down automatically when needed.

And there are some new. Innovations and technologies on this topic as well. There are lots of companies already using AI to automate these types of management where the model really learns and adapts to the demands on the cloud services to automatically reduce the runtime costs. And I think that's extremely powerful.

To invest in for the future, a huge issue, both for organizations as they're coming out the back of migration and just generally getting their heads around capability built. But as you said at the beginning, especially moving through 2023, we know the economic headwinds are going to be quite fierce, and it's important that.[00:42:00]

This kind of thing is dealt with almost as a job one as a capability build because it becomes so critical as an enabler to technology and business change going forward. Yeah, and I go all the hyperscalers have architecture frameworks that give good practices and best practices that, that'll guide doing this and like Google, Azure and AWS have their architected frameworks that have a cost optimization pillar in them.

There's great stuff in there, so if you haven't, have a look at that. It gives you great tips. Definitely recommend that. We like to end every episode of the show by asking our guests what they're excited about doing next, and that could be what you want to do over the weekend, or it could be a bigger next big project at work.

Mark, why don't you kick us off? Yeah, I think so I'm working with Globalization Partners, and we've alongside Michael and we are applying a lot of the same things that we have talked about in the book to a new context, to a new domain. So that excites me, right? How can we do [00:43:00] this again in a new area in nearly a year now?

I can feel volume two of the book coming on already. Yeah, so doing it again, doing it from scratch, re articulating those same things that we you know, stuff sometimes we took for granted or we had the arguments or had the fights to five, six, seven, eight years ago. Now you're having to have them again in a new audience.

And then how do you distill that down? So that's exciting. And it's great getting that flat wheel turning. Michael was in ahead of me, so he got the flat wheel turning already. So it's just a building on that. So that's what's exciting. And that's the next challenge. Mike, what about you?

What's exciting you? Man United versus Man City tomorrow at 12. Oh, there it is. Are you red or blue? I'm red. So looking forward to that. But yeah, just building on what Mark said. Certainly it's, we've been asking ourselves those questions in terms of the book, in terms of applying it to that new context.

[00:44:00] And I think what we're finding is, we've been able to actually do it. A lot faster because we are applying. We are able to move past certain things that maybe would have slowed us down in the past. And and certainly we're, I'm looking forward to a lot of the, outcomes I'd say over the next few months anyway, in terms of what we've been ramping up over the last six months.

And it is. That's a lot of what we talked about today and what we're finding is it's, we're validating a lot of our previous experiences and taking that forward, getting lots of good feedback as well. I think the other thing there as well is we're doing a lot more externally too, so we're looking at, so how do we do a bit more externally, working with people, do a bit of training, but kind of workshops there and even the local communities too, so get more involved there. We've been running meetups and speaking with engineers from different backgrounds, different organizations, different contexts, and feed that into the conversations as well and engage in that too, which is great.

So yeah, there's lots happening. I think this year's going to be a big year. We wish you both a huge amount of luck. [00:45:00] We're going to hear what Dave's excited about in part two of the conversation. So join us over there. So a huge thanks to our guests, Dave, Mark and Mike. So we're not done yet. Join us in part two to hear our three guests talking about how they encoded their experiences into the Flywheel model

Thanks to our 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 everyday by its purpose of unleashing human energy through technology for an inclusive and sustainable future. It is a responsible and diverse organization of over 360,000 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.