Data Mesh Radio Episode #33: 10 Reasons Why You Aren't Ready for Data Mesh Article

Interview with Thinh Ha

Listen (link)

Scott Hirleman

Welcome to Data Mesh Radio, produced and hosted by Scott Hirleman, the founder of the Data Mesh learning community. Data Mesh Radio is a vendor independent resource for learning more about Data Mesh. Let's jump in.

In this episode, I interviewed Thinh Ha, strategic Cloud Engineer at GCP. To be clear, this is only his own opinions and does not necessarily represent those of GCP, he was not on representing Google, but it should be the options of GCP because they're pretty much all rational and reassemble in my view. This is a bit of a spicy intro because this was just a fun conversation, and I think you'll all really enjoy this episode. I asked Thinh to be on after he published a post on Medium originally called 10 reasons why you should not adopt Data Mesh, which was later changed to 10 reasons why you are not ready to adopt Data Mesh. I had a few nit-picks with the article, but agreed with most of the points that he had made. You can find the link in the show notes. I do recommend reading it before jumping into the full episode.

Thinh originally came across Data Mesh, when a few customers requested he help them implement Data Mesh. He really liked some aspects but thought this wasn't for everyone, and as he's worked more on implementing Data Mesh, he decided there are some key concerns or hurdles, maybe a checklist that if you aren't at a certain stage, you're really gonna struggle trying to implement a Data Mesh. We went down the 10 reasons he gave in the article one by one to discuss, there were some useful tidbits that came out in addition to the article, in the conversation, most of which were from Thinh and a bit of my own color and flavor added in.

To Thinh, there's a point when decentralizing makes sense, but monotheists always have a place. Decentralizing too early is just extra work, make sure you aren't decentralizing just to decentralize. I think Andrew Harmel-Law on his episode with Danilo Sato also talked about splitting too early or splitting in the wrong way and having to glue things back together, I think this is where it comes as well with decentralization. To get Data Mesh right, take a lot of learnings from implementing DevOps, part of that is not having a gateway at the end for... And now you do these

five things, you need to be doing documentation, testing, things like that as part of the data product development process. And you also need to automate as much of it as you can to help your developers not just toil. You need to find good ways to bring people on board and show them the value of Data Mesh. The organization needs to make serving data an important thing to everyone, trying to only use the stake to get people to participate in Data Mesh will not go well. You have to find that incentivizing structure that is more of the carrot. You need to encourage the constructive conversations and keep your eyes open. Data Mesh isn't a silver bullet, it's not that there is a simple pathway and boom, everything's solved.

A huge part of Data Mesh is just encouraging the people process to have much more conversations and work together, you need to know what you were trying to solve, and then show people that you were trying to convince and you can probably get more people to come around. This is important to Data Mesh or for really any IT/ Data implementation project. Take the good parts of Agile and leave the bad. One year out planning is very bad. You can have a general pathway and north star, but you need to be able to measure and move the target based on those measurements. It's about agility and adapting to changes, shifting quickly, then measuring, then adjusting your plans accordingly, that fast feedback cycle is really important. You need to empower your people to make decisions and changes quickly, they need to feel intellectually safe and that there is a high tolerance for failure, failure is gonna happen. To do that, you also need to make a platform that makes failure much less harmful. You also need to let people feel useful and understand what is and what isn't part of their role, are they doing the right things, are they getting those things right, what is their impact?

Making it so you're not just doing for the sake of doing that people understand their part in the greater four. You need to build up your internal data talent, whether they are officially on a data team or not, and the only real way to learn that, according to Thinh, is by doing. So make sure you don't just outsource all the doing to contractors and consultants, those people are going to leave your company at some point, you want to build that up internally. Thinh referenced the DORA research that when you shift security, privacy and compliance left, teams deliver faster, fail less often, recover quicker, get more done and deliver more value.

So this is part of Data Mesh as well. Thinh summed it up by saying. "Make it easy to get it right and make it hard to get wrong for your teams." Part of that is making smaller changes that are more observable, build systems that make it relatively harmless to fail as failure is going to happen. I know it's all easier said than done, I think you'll get a lot out of this and that it will really help you to start to frame your conversations as well with your own maybe skeptics inside your company or outside, and that you can take a lot of this and kind of directly apply it onetoone and figure

out if you really are ready to be going down the full Data Mesh path or you're just looking to find a way to better collaborate across your organization with data.

With that bottom line up front done, let's go ahead and jump into this interview.

Super excited for this episode. Today, I've got Thinh Ha here who had written a really great article about the 10 reasons why you are not ready for Data Mesh, and we're gonna go through that and talk about some other challenges that we are seeing with large scale companies, whether they're doing Data Mesh or not, and I'm very excited. So Thin Ha is the strategic Cloud Engineer. He works for GCP on the professional services side, but to be very clear, he's not here representing Google, this is entirely his own views, as we have with all of our guests, these are only people sharing their own context, they're not here as a representative of their company. So with that kind of said, Thinh, if you wouldn't mind giving people a little bit of background on yourself, and then we can kind of... Maybe even a little background on why you decided to write the article as well. And I fully agree that I keep getting pushed back of people saying stop gate keeping, but I do think that people need to think before they decide to go down the Data Mesh path, and that there are many companies that aren't ready for it. And so I really like that you started this conversation. If you could give that background on yourself and what led you to write this.

Thinh Ha
Yeah, thanks Scott. Hi my name is Thinh. I am a consultant, pretty much for Google cloud services. I've worked with several customers on Data Mesh, and it's hard. I actually encounter by surprise. I started working with a customer who told me that they've tried building a centralized data team twice and in their mind, it felt twice, and then they sent Zhamak 's article on Data Mesh. And they said this sounds really good. Can you please do this for me? And I had no idea what I was doing at the time. I ended up leading that data platform team to build the Data Mesh, and they wanted to prove that the architecture works, and I learned a lot from it. I'm still learning a lot today, and that's the reason why I wanted to write this article. Personally, I believe in Data Mesh, I know it works, but I also believe in walking into things with your eyes open and I've messed up a few times, and I've observed people struggle.

Conversations around Data Mesh tend to oscillate between, it sounds really great, know that you have to do it or it's pure hype, I'm not worried about that. So I wanted to bridge some of this conversation and just give people the information, so that they can have a more constructive conversation around Data Mesh and Data Mesh adoption. It's not to gate keep, I promise. Just like the title, I am a believer.

Scott Hirleman

But I've talked about this as well, and I'm a believer. But I think having a practical approach to it makes sense, this is hard, this is not something that a 150 person company should be taking on when you think about full Data Mesh. When I did a live reading thing that I put on LinkedIn about it, and we had a little bit of conversation, it was like, this could be interpreted this way. And you're like. Well, I didn't mean it that way. And so even with that, it's one of those where I think you kinda talked about the one or the zero aspect, either this is Data Mesh and we should be going full in on this and we're just kind of confused as the how or it's all hype and it's like no, it's one or zero. It's not one or zero, it's halfway in between. We have these sliding scales and who does it make sense for? So I'm very excited to talk about this 'cause I do think all the points have... If we frame them correctly, I think all of your points are very valid and very true. [chuckle] So I was very excited to read one where I was like. Oh, this is not just... This is hype and BS, it was like. No, these are well thought out points. So if you're ready for the grilling to begin, we can jump into.

Thinh Ha
Yeah, go for it.

Scott Hirelman
So your first point was you are not operating at a scale where decentralization makes sense. So could you give a little background on that and we can kinda talk on that?

Thinh Ha
Yeah, so when I started with the Data Mesh journey, I had to figure out where are the areas where I can learn from, what are some of the lessons that we can learn from in other areas. And part of it, there's this whole comparison between Data Mesh and micro services with the decentralized data owners who lose coupling between domains and API-driven interaction, there's a lot of parallels, and I started looking at, okay, well, when does a microservices architecture make sense, when does a monolith architecture makes sense? And you know monoliths are actually easier to work with in some cases, you can run a stack trace and see all of your function course.

It's only when you have say 10 different teams trying to coordinate changes to the same spaghetti cobwebs where you don't know where things start and end that you have problems with it, and it makes sense to start defining domains and break out your application and have the teams autonomy to iterate within the domain and establish a contract so that they can talk between them. So there is a critical point where at the certain kind of scale decentralization makes sense, but before that, it is probably easier when you just keep it in one place, and let's say if all of your... If you own a small company and you're not really producing that much data and maybe you have one person doing reporting, there's not enough context to really

decentralize and make everybody responsible for all of the data times and all of the data engineering stuff. And the question I would ask is, are you meeting the analytics demand of your organization? And if the answer is yes, then maybe you don't have a problem, and maybe you don't really need to adopt the Data Mesh, that's the point I'm trying to make.

Scott Hirleman
Or from my standpoint as well, is decentralization your cause for not meeting your analytics demand? Where so much of this is that the domains historically haven't had to serve data, and so if they're just kind of serving data chaos and they're constantly changing their application model, and the data model is built directly off the application model, and you don't have that kind of abstraction between the application model and the data model that you're serving, yeah, your application team, your domain teams are gonna constantly be not meeting the demands of the data consumers because all you're doing is breaking the things out from underneath them, but is that the fact of the centralized data team being the bottleneck? Probably not for a lot of these companies, you just have to get better at a motion of... And Data Mesh does push you in that way, but it's also 10 steps further... If you're thinking about, is the data engineering team the cause of those challenges? No, it's that if we just get the domain teams to have some idea of ownership and you give them some autonomy if they wanna serve this data in their own way, great. But they can also just serve it via the centralized team and that you don't need to... If you don't have 30, 40 domains, which is kind of where I think a full Data Mesh starts to make sense, your microservices might be 10 domains, but that centralization isn't the bottleneck that it is on even the microservices side because your data shouldn't be evolving that quickly, so I fully agree on this one, I think you made a lot of really good points there. Do you have anything else you wanted to add on that specific one or should we jump to number two?

Thinh Ha
Yeah, no, look, I think with respect to centralization, it's easier for me as a data engineer to just do things myself. If someone say, Hey, I need this report tomorrow, it's easier for me to just go into the source system, then grab the data, wrangle and prepare the report. What is the point of suddenly trying to get all these different people to, okay, now you created the data, now you have to push it into the system and building all of these API interfaces, and then trust that they have to do it right. It's a really different way of doing things. And if you're going to have to, if you really want to do, you need to know what problem it is you're solving, and I think... Yeah, some people think that Data Mesh solves their problem, but maybe it is one of the solutions, but there is another solution somewhere else. And that's something that I think there was a conversation in the comment section in my blog, but I just said you know your problem, it can be solved Data Mesh maybe, but here's a few other ways

that you can solve it, so think about that first, and then think about Data Mesh.

Scott Hirelman
Is it a sledge hammer? Or do you just need a hammer? It's like, yes, you could hit that nail in, but it's a lot more effort to hit it in and do that... So number two, you do not have a strong business case for how adopting Data Mesh will deliver business value for individual business units. What were you talking about here? And what was the thought process behind this one?

Thinh Ha
Yeah, so it takes effort to build a successful data team. Now, suddenly in the Data Mesh, we're expecting every domain team, whether they are resource, system data producing team or the consumer team to take on more responsibilities than what they have done before, and they're gonna have to ask you what's in it for them. Why should I do this? Why should I invest all this effort? How would this produce faster time to value for me to get the data that I need, how would it produce AIML driven and application that I need to differentiate myself in the market? And you have to have a good answer, unfortunately, there's always going to be skeptics, maybe part of your organization is fully bought into the vision of the Data Mesh, they see it, it makes sense to them, but there will be a part of the organization that doesn't believe in that, they don't want this change, they're completely fine the way it is, and you have to convince them and to convince them in the business world, it just means building a strong business use. How is it going to have a measurable impact for the team, for the bottom lines, so that they can invest and spend time and effort into adopting this change. Without that, it's going to be really difficult to maintain the momentum.

Scott Hirleman
Yeah, I agree and disagree on this point because at some point, if your entire organization is going Data Mesh, those domains kinda have to get on board, whether they want to or not. But I do agree that the carrot is much more valuable and valid than the steak. I talk to somebody who is gonna be on an episode coming up here, and literally one of the things they do is when a team publishes a data product to the data mesh, they send them an actual cake, so it's cake driven development, they send them an actual literal cake. But I agree with you that especially at the start, to get moving, to get momentum, you can't go to the domain and say, your data is the most valuable, and therefore, you're the ones who are gonna be producing the data if they're not bought in. You have to be showing that business value throughout as you're bringing on incremental data products, because if you're not adding value to those domains and you're just adding additional work to them, then it's just not... They're not gonna be bought in. And there's challenges there. But at some point, you kind of also have to say, if this person isn't bought in, if the leader of this domain isn't

bought in, then at some point we have to change that leadership.

You don't have to say that everybody has to be full in from the beginning, 'cause then you don't have the ability to evolve and get everybody moving together, but yeah, I agree with you for the first 30, 40% of your journey, everybody that you're bringing on has to be bought in.

Thinh Ha
Yeah. And you don't need everybody to be brought in at the beginning, you can start with a few, a small group of people who really believe and approve the value, and get people to be confident in it. It's about building confidence at the end of the day and yeah, once you have a track record of success and success means a demonstrable impact. Not just, "Oh, I've built a Data Mesh." So what? And then other people will want to get involved. Regardless of whether they believed in it at the beginning or not just because they see that it's making value and they wanna get on board.

Scott Hirelman
Yeah, well and I think also playing to people's career aspects of Data Mesh is something that a lot of people are very interested in, so if you're somebody who's like, Well, I'm not that bought in, it's like, Well, but it can be very good for your career if you can talk about how you've implemented this in your team.

Did you have anything else you wanted to add there? Should we jump to number three? Okay, so number three is you treat Data Mesh as a technical solution with a fixed target rather than an operating model that continuously evolves over time. And before you start, I'm gonna say 150%. So what were you thinking about here when you said this?

Thinh Ha
So yeah, look, I've been in programs where before we even get started building a Data Mesh, people will want to plan everything upfront, let's build the entire data domain model upfront, we'll figure out who the leaders are in each of the domain, and then they're gonna have to give us a backlog for the upcoming year, and then we're raise towards that backlog. And it's a bit of a remnant of bad Agile practices, and I'm not convinced that that's the point of Data Mesh. Data Mesh is about agility, it's about giving people autonomy to adapt to change, it is... You can't plan out everything, and especially most of us are still trying to figure out how to do Data Mesh. So I think it's about shipping quickly, observing the impact that it has, and gaining confidence that this is working or not. And then if you've done it wrong then change course quickly. That's the most important thing. And on this aspect, adopting Data Mesh is no different to adopting DevOps, you just have to... It's about agility, it's about ability to change quickly. So once again, I'm learning a lot from what

can I learn from, in other words, this point came completely from the DevOps world.

Scott Hirleman
Yeah, and I think a lot of people haven't... If you watch the early Zhamak presentations, she talked about, Oh yes, and we're taking this from DevOps and we're taking this from microservices, and we're taking this from... It is about just literally saying what's worked in these other things, let's apply it to data. And I'm frustrated that the word agile and agility are linked because agility isn't necessarily about Agile as in capital A Agile, but the whole point of this, if you listen to, again, Jamak's early presentations, it's agility and scalability of data. So if you do have that fixed target, you're not measuring and moving, your target is changing. Your business is ever changing, if you're not thinking about how are we evolving and yes, you wanna have a target for how the organization works, but not for... This is the exact set up, and this is exactly what we're gonna do because you're gonna measure and find out all of this data product that we thought was gonna be super, super valuable, and it's very, very expensive to create, only 10% of it is used. This other 90%, we never use. So let's pull off the complications of creating that and just go with that 10% and that measurement and movement and having that conversation back and forth between the teams is super important. And the whole technology, especially people wanna know what tech do I use or even what architecture? It's like people process is far more important because you have to be able to do this...

Thinh Ha
It's always about the people.

Scott Hirleman
And working together. Well, yeah. The number of times I talk to people and they go, Yeah, we had this revelation when we put the data consumers and the data producers in the same room, it's like they've never talked, what? [laughter] I think this is such a crazy thing. So did you have anything else you wanted to add there or should we jump to number four?

Thin
No, I'm good.

Scott Hirleman
Okay. So number four, your organizational culture does not empower bottom up decision making. So what were you thinking about here?

Thin
Yeah, so this is another side of the same coin as the previous point, people in the box, people empowered to get things done quickly, and that means they have to learn

quickly and of course correct quickly and potentially make mistakes. It's okay to go the wrong way, as long as you realize it quickly, and you change, you course correct. And people need to feel safe to try out different things and to fail, it doesn't mean you can try everything indefinitely. And you can fail indefinitely, that's where I think SRE practices like innovation tokens and service level objectives come in. But the point is, people have to feel safe to do the best work, and without psychological safety, you might have people building data sets, but they don't share it because they're afraid that people may blame them if they find a problem in that data set or people noticing that something's wrong with the data set or in the pipeline, but not saying anything.

And it's not a good place to be in when you're trying to decentralize everything, when you can't have a tight grip over everything that's going on, you have to trust people to get what they need done. So yeah, I think that's the point here, everyone wants to do their best work and it's hard to adopt such a big change if people are not excited about Data Mesh, if they're not excited about trying out these things, and they're just fearful about it. So I think the point here is about making people excited and building a culture that empowers people to feel safe, to learn and to try new things and to do their best work.

Scott Hirelman
Yeah, I think if you give somebody... If you say, I'm giving you the autonomy to go do this, you'll be surprised at how many people their eyes light up. And if they're not super excited about having that autonomy, that might be something where they're not really prepared to be in that Agile type culture, but then you also have that centralized safety net, that you have that psychological safety, but you also have, okay, this isn't... I'm telling you you have to go do all of these things on your own, I'm empowering you to make the decisions, but I'm giving you the ability to work with a more centralized team when you have questions. And we've got that centralized sharing and that centralized togetherness.

But yeah, so I really like this one, 'cause I think if every decision has to be made bottom up, that's obviously a challenge, but you give people the freedom and the trust. People, when they feel trusted and empowered and given autonomy, they're much happier and they can go and do amazing things versus everything has to be approved. It slows down your agility and it means that you have brain drain of a lot of your best people. So that's kind of been proven repeatedly by lots of different studies. Are you good on that? Did you wanna add anything more there?

Thinh Ha
I really like the point you said about having a kind of centralized safety net. So data Mesh has a decentralized architecture kind of emphasizes the fact that you want

everything to be decentralized but overall, the central team is so important. They don't just build the data platform. They're the people who've had mistakes before, they're the experts in infrastructure, they're the experts in, let's say, data modeling or something like that. They know how to do these things well, and they're just coping with the amount of work by not doing everything themselves and empowering everyone else to do it. It should be more like an internal consultant. They're there to help everybody else and to be that safety net when people need it, so that they can go and learn and do all these things that they wanted to do before, but they never had the chance to. So yeah, I really, really like that point you said about the role of the centralized team and the safety net.

Scott Hirleman
And I think that's on the governance side too of empowering instead of gatekeeping of no, you can't have the access. How do we think about the access by default and that when we're putting the governance decisions on the domain teams, when they feel comfortable to make it, when they don't, there is a centralized body that we can go and ask and go, I really don't understand this relative to GDPR. Okay, great, we're gonna jump in there and we're gonna help you instead of you must comply with these 73 things, and they're not really in a language that you understand, and that the platform team is putting things in to making governance decisions easy instead of you, Domain team have to now own and operate your own governance tooling system. No, that needs to be part of the platform to give them the ability to make those decisions.

Thinh Ha
How many times have you read some guidelines and it says something like, you must have appropriate encryption in place, and the key word is appropriate. What does appropriate means? Tell me what appropriate means, and no one can tell you what it means. Exactly that, you have to think about it from the perspective that you're there to help them and get them and get to what they need to do quicker, it's not about you. So yeah, great point.

Scott Hirelman
And I think that's kind of where the role of a CDO goes. If there's two different roles of the enablement and empowering and that central backdrop, but it's also then to start pairing more with other C-level execs to start to drive their strategy decisions via data that you start to infuse that. But I think a big part of that at the start is just, how do you create that culture of safety net and empowerment. I like a lot of what you're saying there too.

Thinh Ha
100%.

Scott Hirleman
So for number five, you do not have clearly established roles and responsibilities and incentive structure for distributed data teams. So what were you thinking about here?

Thinh Ha
Yeah, so this is a bit of a difficult one. We had a chat about this offline. To me, it's not about putting people into boxes, it's about making sure people know what they are supposed to do, that they feel empowered to do it, and they know that they're going to be rewarded for it at the end of the day. I think the incentive structure is such an important part of Data Mesh, how do I know I'm doing a good job, how do I know I'm building a useful product? And my title today is business analyst, and someday I'm doing all these air flow and DBT and Grafana and what not, am I doing the right role for the team, am I being rewarded accordingly for that? It's a tough one, and I'm not a HI expert, I don't know how to solve it, but I know that people can't thrive unless they're getting positive reinforcement on whether they're doing the right thing or not, and they feel rewarded for it. And that's this point about roles and responsibility. I hope that makes sense.

Scott Hirleman
Yeah. Well, I think my nitpick on it, which I think when we started talking, I don't think was a fair nitpick was the word clearly, or the phrase clearly established roles, because I think when you're first starting, you have to be agile about your organization, you have to take that team topologies type approach of you have responsibilities and you have people that those responsibilities are on and that you try and paint the whole picture as to or you have to have full coverage as to, here's what we're trying to accomplish for the next three months. And so we need to assign out these responsibilities. But do we have to have fully mapped out those roles from the beginning?

And I think what you talked about in a lot of the previous ones of being agile, especially number three, when you talked about the target, instead of being agile, I think we need to have that from a role, literal role standpoint, an HR role standpoint, but I agree with you as to, we need to find those proper bucketing of responsibilities. So it's not just like, "You team, have these 15 responsibilities, here's your checklist, that somebody got these." Versus like, "Hey, it really makes sense for the owner to own these three things, to own kind of the final governance decisions." And when they need to talk to a centralized team or not, but the data product developer should be doing the data modeling, figuring out the tooling to measure when the application model change might break the data model change. And then who is gonna be kind of the main point of contact to your consumers, that type of thing, I think it makes

perfect sense. And the incentive structure as well, I haven't found a really good... Outside of giving people cake.

Thinh Ha
That is a really good one. That is a really, really good one.

Scott Hirleman
It's one of those things of just like, it feels ridiculous, but at the same point, then that person can take a picture, and then... But that we start to talk about, but this is good for your career, a lot of people don't wanna pay down tech debt, because in a lot of organizations, it doesn't make your career. It's not the thing, it's the new features, it's the new product, it's the new... So we have to talk about that aspect as well. And that it's not just, you're doing the right thing, so yay. It's that we align career progression and we'll align monetization and things like that as well. But that we kind of do all of it at once and then it's not just we're gonna pay people more if they're doing the data products right.

And it's like, well, then does that lead to gamification measuring and all that stuff, versus, we need to create a structure that makes sense, that gets people bought in and they feel like they're valued as well. So I think it's a very deep topic that I get weaker and weaker on the more that I say more sentences.

Thinh Ha
It's about the people once again. There's this analogy I've heard which I really like, think about a football match, you call it soccer, in a primary school. All the kids are just chasing the ball, they're just all chasing the ball, all 10 of them or whatever. And, is that the best way to play? No, people need to know where they should be on the field, so that they can work together and achieve a common goal together. And I think that's what this point is about, roles and responsibilities, it's about figuring out what works and then giving people the structure so that they can follow to be successful in their job.

And it's about understanding, "Okay, well, I am a data engineer, my job is to enable these 10 analytics engineers who are building DBT models on top of my database, that are building or these analysts are facing, they know who they're supposed to talk to. They're facing into the business stakeholders and they know what's important for them, they have the context, and they can build the things that are useful for those people. And instead of having the data engineer directly talking to the business stakeholders or the analysts, talking to the data engineer, it's about optimizing it and giving people the means to be successful and not making them figure it all out on their own.

And I think this is where... You asked me at the beginning, what are the things that we need to talk more about and figure out how to set up so that people can be successful? I think this is one of those areas, what is the successful team model of building a Data Mesh? Okay, so we know there's a domain and there's a data product owner, but then how is that team structure? How did they be successful? And I hope we can have more conversations around that, figuring out what's successful and what works and what doesn't work.

Scott Hirleman
And it's organizational dependent, there isn't a cookie cutter model for it. But there are ones that work and ones that don't. And so the more you can share about what's... But yeah, and I think the football/soccer analogy works much better than American football, where you have very, very strict roles, and people can't do certain things, based on what role they are. The keeper can do specific things within soccer/football, but everybody else... If the striker may be pulling back to do some defending and things like that, there's overlapping runs, there's all these things where you kind of support each other as a team, as a domain, but there are clear kind of general responsibilities that people have as well. And so that it is and you have your formation to say, are we more attacking or more defending? Are we 541 or something like that, and it's just all defense versus 433 is much more of an attacking style.

Thinh Ha
And that's a clear common objective. We all know what we're here to do. And this is how we're going to achieve it together. I think that's the...

Scott Hirleman
Exactly. Yeah, I like that. So number six, you do not have a critical mass of data talent. So what were you thinking about here?

Thinh Ha
Yeah, so you know what makes people afraid of Data Mesh? It's the question about how this is going to affect my role? Some people are gaining responsibilities they didn't expect to gain, and then some people are losing responsibilities. And I think the point I really wanted to make here more than anything, is you got to train your people. You need to give them the tools to be successful in this new world and empower them to learn and do what they need to do to be successful in this new world. So maybe I'm naive. I think that's how you gain a critical mass of data talent, and by data talent, I really mean everyone. It's not just engineers.

Thinh
To some extent, you can hire or contract engineers, but when they leave you lose all

the knowledge that they've learned and so much of Data Mesh is about learning how to do things, and it's the people that make a difference. So it's the people that try it first. The people who have taken the plunge and gone into it with putting their career on the line to try out this new thing that we're about going to have to become the leaders to train everybody else, and I think that's how you build a successful culture, and you spread the Data Mesh to build organization and knowledge about how to do it well, how to be successful in this new world. And that's the point I'm trying to make. It's not about, "Oh, you gotta go and hire a bunch of people right now." It's about how we make our existing people successful and give them a roadmap to being successful in this new world.

Scott Hirelman
Yeah. And I don't remember if I said that I fully agree with this, but I do fully agree with this based on what we're talking about here. Because, one, I think you need to have the critical mass to be able to understand what your data challenges are to be able to create a successful platform. You have to have enough people to create a platform, 'cause if all you're doing is saying domain teams, you now have to share your data in this way and you're not giving them the tools to do so, and the expertise and the backdrop help of, "Oh, nobody on the domain team knows how to data model." So do we put a data engineer into that team or do we give them the tools and we start slow and say, "Okay, this is just kind of your early days around creating your data model and things like that." Whatever, there's lots of different ways to do that. But I think then you also have to think about the critical mass of understanding consumptive patterns and being able to train additional people to level up your overall data literacy and things like that. So I think you're taking on way too much if you don't have enough people and enough overhead to try to get this moving forward.

And so I don't know what that critical mass level is. It might be different for different organizations, but I agree that when I talk to people that are like, " Yeah, our data engineering team is five or eight people and we wanna do Data Mesh." It's like there needs to be a far better understanding of tooling and integration of tooling and a slow run process to build up your data sharing capabilities. That there's more people that are talking about that very specific before you get there. I just... You're gonna get yourself in a lot of trouble.

Thinh Ha
Yeah. No, I think it's okay to start with a small data engineering team, just as long as you figure out how to get them to learn and how do you get them to start enabling other people. So everytime they do, as a reward you're 50% of your time. The only one issue is 50% of your time trying to automate your job away. Everything you're doing this year and the next year, you should no longer be doing. And that's the approach

that the data engineers need to take. Like, what are the things that people are doing that's producing toil and causing delays in getting value? Let's automate all that stuff, so that the next time they can get it done quicker, and everybody needs to start adopting that kind of mentality. And it has to be a journey, it cannot be, let's hire a hundred engineers today and then hope that tomorrow, we have a Data Mesh. So I don't think that will work.

Scott Hirleman
That's the old, if it takes one woman nine months to have a baby, what if you split that up and it was nine women for one month each? It's not the way it works, but kinda what you're talking about I think is also the slow build towards Data Mesh, where I don't think that you even have to have that end target of Data Mesh. When you've got a team of five, ten people, that centralization, you don't need to have that decentralization. They can still be a centralized team and they don't have to move to only being a data platform too. There is still is that central ownership of things, because there are certain things where you can't build the tooling for the very complex use cases to make it so that the domain teams can share all the data that they need, but that you're building that team muscle as to just sharing data. And it doesn't have to be in a Data Mesh way. It's sharing data.

Thinh Ha
Yeah, it's about making data more accessible so that people can get what they need done. And the best way to start with the Data Mesh, and I've noticed this a few times, I find the people who are the most hungry, who really want that data, but are struggling to get it, they can be data consumers. They can be the data scientist that was struggling to get the data that they need. Or the data producers who are really struggling to get insight out of the data, because everything is stuck in the system they can't access or run your analytics group. And just make them successful. Just like, "Okay, I'm not going to give you the data that you're asking for, but I'm gonna give you all these tools so that you can go and do it yourself." And you start with those people, and then slowly, slowly you can get to a point where you have the Data Mesh. You don't have to start all out at once.

Scott Hirelman
Yeah, and I just thought of an interesting through line from that, back to number two of, you need the strong business case for the business units. What I'm hearing from a lot of people is that the first consumers from Data Mesh are typically the other application developer teams that were trying to stitch together all this information that they didn't have access to from other systems, and they were trying to do this Frankenstein monster type query. And so it's kind of interesting that I fully agree with the intentionality around your data, what do you share and things like that, if you don't have that maturity level of being able to talk about why this matters it's gonna

be difficult. Did you have anything else you wanted to add on the low... Okay, so for number eight, you expect to find off the shelf software to help you adopt Data Mesh 100%, yes. So what are you thinking about here?

Thinh Ha
So it's like the start of the DevOps journey, or suddenly there was the DevOps software everywhere everyone's buying DevOps and then being really confused why they don't have it yet. So I think we're kind of in a similar space with Data Mesh and the tools, it doesn't fix everything. I've said it before, it's about the people and yeah, if you go into Data Mesh thinking you're gonna buy something and it's gonna fix everything, and so what it won't be a problem for you. You wouldn't for that time, you're not gonna get what you want.

Scott Hirleman
It's the equivalent of saying, “I'm gonna get in shape,” and then just buying some gym equipment and going, "Why I'm I not in shape?"

Thinh Ha
Exactly.

Scott Hirleman
Yeah. Now, and...

Thinh Ha
It's a tough one, but yeah.

Scott Hirleman
And I think we'll get better on the tooling, but the tooling right now, especially there's so much white space in between the tooling, and there are so many things where it goes, well, this tool, it helps me in these three ways, but I need it to help me to cover these five, and the tooling isn't extensible in the right ways where it's like. Okay, I can do this with some aspects of my role on my own, so I either have to say. Oh, this is good enough. Or I have to roll my own for all five of those pieces, and so I'm hoping... I've been talking a little bit about some of the stuff around metadata sharing and that I wanna see a lot more systems, a data product should broadcast out all of it's metadata.

And people are like, "Well, shouldn't you just push it into this other system?" It's like. No, you should have it broadcasted out, so that way any system that's listening can bring that metadata in and then all the other systems, if you can get them to broadcast out, great, but then you wanna start to think about tagging where this metadata came from, so that way you're not repeating the same metadata. You don't

have the same thing that's come from nine different systems, it's all the same. But I think that there's a lot of these expectations of and vendors are trying to sell this as, we just give you this capability as well. And so I don't think we'll ever have this as well as kind of what you talked about with the number three of a technical solution, you're never gonna be able to buy a Data Mesh even if the tooling is the best thing ever. Yes, you say to help you adopt Data Mesh, but we're way far away from even tooling that's really designed around Data Mesh, and then we're never gonna get to a place where you couldn't buy the tooling, that's just gonna be like. Okay, this is a drop in tooling, and our platform is done, it's like no so much of this is talking with your people and figuring out their specific needs.

Thinh Ha
But I do want to give credit where it's due, there are really innovative startups in software out there that are really trying to solve metadata, I think metadata is the next big data. It's now, it's not big data, it's big metadata. How do you... You have lineage, you have all these pipelines generating all kinds of different data, you have people copying data all over the place and building reports and using data in ways that you've never used it before. So how do you track all of this and organize it and make it coherent and accessible and discoverable and all that good stuff? It's a challenge and I credit where it's due. There are some really, really innovative subjects out there that are doing this. So don't dismiss the tool, just don't expect them to do everything for you.

Scott Hirleman
They're not right now, and they never really will be everything for you, but yeah, I agree with you that there are a lot of people out there that are approaching things in a good way. You're not here on behalf of Google, but I would say BigQuery also is something that I hear a lot because there are people that literally go. Well, we're just gonna do some stuff on GCP because BigQuery, I can just... I can do ELT, I can just dump stuff into here and then I can actually structure it and then it's not a pain to do this.

Thinh Ha
I said I wasn't gonna talk about Google, but it blew my mind when I joined Google and I can just write SQL and everything works, and it's fast and I don't have to babysit these Spark pipelines that run for 12 hours and then fail with unknown error exceptions or something like that. So I was just really, really happy. There are tools that will help you, for sure.

Scott Hirleman
Yeah, I was amazed at how many people were just literally adapting GCP for this. So number nine, you do not have buyin to shift left security, privacy and compliance. So

what were you thinking about here?

Thinh Ha
Yeah, I think this is part of the DORA research. The numbers speak for itself, the team that shift lab, security, privacy and compliance into the development process, deliver faster, they get more done. They deliver more value, they fail less often. And they recover quicker. So in a world where everyone is moving quickly and shipping changes in smaller batches, you can't have a single review gateway at the end of everything that set everybody back once if they fail it. It doesn't work, you have to make the changes smaller, smaller, safer, more observable, so when it goes wrong, you know immediately that it went wrong, and actionable at the end of the day, and all of this is done... Security privacy has to be embedded into standard processes, so that people understand why it matters and how to do what needs to be done. And yeah DevSecOps is kind of a new thing and I think... I don't know. It's kind of controversial. It's like. Oh, obviously. But yeah, I think it's the only way to go.

Scott Hirleman
Yeah, I think DevSecOps, which for people that aren't familiar, it's Developer, Security, Operations. And so I had a conversation with Azmath Pasha who was talking about X Ops, and it's just everything, something Ops, but I do agree with you that it's something where it's kind of unfortunately named, but it makes sense from the naming perspective, and it's something that I agree with, that it needs to happen. And I think this is where we talk about that federating governance. You still have the centralized team as that centralized backdrop, it's not the centralized bottleneck, it's the centralized safety net. So if you have questions about this, especially for governance, you need to have kind of a CYA or cover your behind type of approach to governance where if people have questions, they have somewhere that they can go to to enable them.

Thinh Ha
Alright.

Scott Hirleman
But outside of that, you need to be trusting your teams because otherwise, why did you hire these people? If you don't trust them, why did you hire them? Why are they on staff, why are they working for you? And as you said teams just perform better and you're gonna have better retention and all that stuff as well.

Thinh Ha
Yeah.

Scott Hirleman

Autonomy as an incentive is kind of an interesting approach.

Thinh Ha
And with security and privacy, we think about failures as the end of the world. The way that we have to think about failure and taking inspiration from SRE now is, it's a systems problem, it's not a personal problem, it's not... So when I'm asking teams to shift left to go to privacy and compliance, I'm not saying you are now responsible for security and I am not. I'm saying. How do we build an ecosystem so that we can prevent people from making these kinds of mistakes? How do we make sure that the guardrails are put in place so that when people veer out of the guardrails, they are notified immediately and no change is done and things has been reverted back immediately. So yeah, I think it... And once again, you have to trust people, failing is expected, and it's a way to learn, and how do you make failing as harmless as possible? That's probably the goal of all of this otherwise, you can't stop failing from happening.

Scott Hirleman
Yeah. I think the high tolerance for failure is very important and it's psychological safety, but it's also... I think people are really, really afraid of GDPR, and if you think about what the EU has done thus far with GDPR and things like that, it's teams that had an issue. I haven't seen anybody get fined for having a singular issue where they go, or it's a very, very small fine or something like that, where they go. "Hey, we were really trying, we can show you we were really trying and we just didn't code this in the right way, and we had the intentionality and we were doing that." I don't think regulators are out there to be malicious, people are afraid of them being malicious for the sake of being malicious, and it's like, they're not. They're there to make sure that you're doing the right things, but it's not okay, this thing was exposed for five minutes because somebody made a bad bucket permission or whatever. Okay, but yeah, how do we think about making it so that those failures are much easier to detect and that they're much harder to make and that...

Thinh Ha
Exactly.

Scott Hirleman
Yeah, so I fully agree with that. Did you have anything else you wanted to add on that or...

Thinh Ha
I'm gonna refrain talking about GDPR.

Scott Hirleman

Okay.

Thinh Ha
'Cause I could go on for ages, but yeah.

Scott Hirleman
Yeah.

Thinh Ha
I won't stop.

Scott Hirleman
The ThoughtWorks folks had an interesting webinar a while back where they talked about what they actually had in some of their Data Mesh implementations was a GDPR request data product, and so everybody else... And so they actually have a legal reason to be able to keep that data infinitely even, but they weren't, but that then all the other systems would check against this and go. Do I have any of this PII in any of my systems? And so it could run against that, and it was such an amazing like, Whoa, that's a weird approach, but you know what, it makes perfect sense.

Thinh Ha
And I have seen this before. Compliance is a good reason to push people to change and to do things better. With GDPR, that's the right to forget. So how I go to you as a corporation and say, "Mr. Whatever corporation, you have my data, I don't want you to have my data anymore, can you please delete it?" And it's such a difficult problem, it's such a difficult problem to solve, but to solve it you have to do all the things that's good for Data Governance. You have to identify where the PII are, you have to separate it, you have to be able to identify the customer record, and you need to have a system that would be able to go and delete the record, and managing a retention policy on demand. And I think once you have that, it's actually really, really good for everyone, and you've managed to get the organization to do something that they would have never gotten together to do in the first place. So it's not such a bad thing. I think it's quite an interesting space.

Scott Hirleman
It has value at the end even if it's a pain to do.

Thinh Ha
Yeah.

Scott Hirleman
And number 10: You do not consider Data Governance to be a core activity to be

prioritized against other activities in every data team's backlog. So what were you thinking about here?

Thinh Ha
Yeah. Look, the data world tends to be... I might get hated for this, but five to eight years behind the infrastructure world. So DevOps is over, well no, we figured out how to do DevOps, pretty much, we know how to do it well and Data Mesh is just starting picking up all these new things, and Data Governance, we need to learn what worked and what didn't work in the security and privacy world. For example, you can't have a checkpoint right at the end of the process to approve a launch, you can't have a checkpoint right at the end of a data journey to say "Now you need to produce a data catalog. Where are the metadata? Where are the tables? Where are the columns?" You can't wait until the very end to do that. It needs to be embedded in the process of creating the data.

We need to automate it. It can't be a spreadsheet. It needs to be an activity that's done by the engine as they are creating the data itself, and it has to be part of everybody's work. I feel very, very strongly about this, and Data Mesh gives us a partial solution, and there's an owner for every piece of data asset. Suddenly you have defined a model where there's responsibility, federated responsibility to get things right. And this will produce data, you are the owner of that data set, and here's the responsibility of owning a data set. That's the carrot. You're gonna have agility and ability to do what you need, but there's a responsibility that comes with that as well. It's like a CEO, you're managing... You want to generate value, but you also don't want to go to jail.

So it makes you think twice about, for example, collecting data you don't need or leaving data lying around. I think it's better and safer for everybody, and this is an area that's evolving very quickly, and I'm really excited to see how things progress.

Scott Hirleman
Yeah. And the Data Governance as the enabler, so exactly what you talked about. You're gonna create better data products if somebody isn't going "Oh, you didn't do my documentation right. Okay, I gotta do this at the end." It's not just the checklist, but it's like "Hey, here's the thing," and you can maybe even create a potential list of consumers. Who would say "What are you sharing information about? Oh, okay." I'm going to then push you with the potential consumers, and see if you can get in a room and talk about what they might actually want, and that it might shape your data product, and that the governance is even measuring how other people are using data products, and that's part of that kind of overall Mesh level. So I think it's not just... And it's showing people how. I think blueprints are really crucial.

Because if people have to generate this stuff from their own heads, is that on the governance team or the platform team? I think it's more on the governance team to say, "Hey, here is a core data model." We don't wanna have a core data model that you have to adhere to. We want an extensible core data model, but for a lot of domains, especially for the first data products, it's fine to just be like "Here's the data model that makes the most sense for a lot of things in our organization." So if you can keep your context while using this, great. You don't have to do that and schema checks, and all that stuff for... You don't make people do manual things, but then you also give teams the ability to do the right things at the right time. I fully agree with you that it's not just at the end. It's the "Okay, here's my hurdle to jump over." It's like "No, this stuff's important and here's why."

Thinh Ha
And have you noticed how documentation is always outdated? Like that spreadsheet that you got about the data, that doesn't actually exist anymore. So you have to keep updating it. It has to be part of the regular process. And as a Data Governance team, it's about building processes so that it's easy to get it right, rather than saying "You have to do this" and actually giving people no support at all. So that's another thing that Data Governance has to learn from security and privacy, that has to become an engineering discipline. I don't think it can exist as an ivory tower function. Every CDO is probably gonna have to think about "How do I turn my organization into an engineering function so that I can empower everyone to do what they need to do, and make sure that they're doing the right thing?" And the system, and the processes, and the total need to setup, to ensure that the right thing is getting done. That's the way I think we have to work.

Scott Hirleman
I think a good tagline for thinking about the data work within Data Mesh is exactly what you said, "Make it easy to get it right." As simple a summation as that is, I think that's...

Thinh Ha
Make it really hard to get it wrong.

Scott Hirleman
Okay, I'm gonna write that down as well, and make it hard to get wrong. Okay. So this has been phenomenal. We're coming up on an hour here. So this has just been fantastic. But do you have anything that you wanted to wrap with, or any kind of sum-up thoughts for folks, as well as where can people find you?

Thinh Ha
Now, this has been super fun, I don't get to talk about this a lot. And I don't know, I'm

still learning. So if something resonates or doesn't resonate, or you wanna talk more about it, you can reach out to me on Twitter, I'm Thinh Ha. So yeah.

Scott Hirleman
I'll drop that in the show notes, and I'll drop your LinkedIn as well as a link to the article. Yeah, exactly what you said, the whole point of this podcast is learning out loud. It's not to be "I am the expert." It's to be "We're sharing our context." So this has just been great, I really enjoyed this conversation, it's a really fun thing to have on a Friday. So thank you so much Thinh, for your time, and thank you everyone for listening.

I'd again like to thank my guest today, Thinh Ha, who is a strategic cloud engineer and the author of a very good post called "10 Reasons Why You're Not Ready To Adopt Data Mesh." You can find his contact info and a link to that article in the show notes.

Starburst Sponsor Spot:
With a Massive Move to distributed data architecture, it's essential to have access to all of your data wherever it is. A data mesh emphasizes domain driven data ownership, data as a product, self-service infrastructure, and federated computational governance, giving you faster time to value without needing to transport your data. Starburst allows you to achieve this distributed architecture by allowing you to run SQL queries across distributed data that connects sources, regions, and clouds. For more information on how your team can benefit from a data mesh strategy, check out our data mesh resource center on our website.

Scott Hirleman
Thank you so much for listening to this episode of Data Mesh Radio. Hopefully, it was useful to you. If you'd like to connect with the show, you can find us on LinkedIn or Twitter. If you'd like to connect with me, you can do the same. If you have feedback or especially if you'd like to be a guest, we've got some links in the show notes to tell you how to do that, would love to hear what questions people have and how I can be useful.