Data Mesh Radio Episode #103: 4 Years of Learnings on Decentralized Data: ABN AMRO's Data Mesh Journey

Interview with Mahmoud Yassin

Listen (link)

Transcript provided as a free community resource by Starburst. To check out more Starburst-compiled resources about Data Mesh, please check here: https://www.starburst.io/info/data-mesh-resource-center?utm_source=DataMeshRadio

Scott Hirleman

The following is a message from George Trujillo, a data strategist at DataStax. As a reminder, DataStax is the only financial sponsor of Data Mesh Radio, in the Data Mesh Learning Community at this time. I work with George and I would highly recommend speaking with him, it's always a fun conversation.

George Trujillo

One of the key value propositions of data mesh is empowering lines of business to innovate with data. So it's been really exciting for me personally, to see data mesh in practice and how it's maturing. This is a significant organizational transformation, so it must be well understood. Empowering developers, analysts, and data scientists with downstream data has been part of my personal data journey that reemphasized the importance of reducing complexity in real-time data ecosystems, and the criticality of picking the right real time data technology stack. I'm always open and welcome the opportunity to share experiences and ideas around executing a data mesh strategy. Feel free to email or connect with me on LinkedIn if you'd like to talk about real time data ecosystems, data management strategies, or data mesh. My contact information can be found in the notes below. Thank you.
LinkedIn: https://www.linkedin.com/in/georgetrujillo/
Email: george.trujillo@datastax.com

Scott Hirleman

A written transcript of this episode is provided by Starburst. For more information, you can see the show notes.

Adrian Estala, Starburst

Welcome to Data Mesh Radio, with your host, Scott Hirleman, sponsored by Starburst. This is Adrian Estala, VP of Data Mesh Consulting Services at Starburst and host of Data Mesh TV. Starburst is the leading sponsor for Trino, the open source project, and Zhamak's Data Mesh book, Delivering Data Driven Value At Scale. To claim your free book, head over to starburst.io.

Scott Hirleman
Data Mesh Radio, a part of the Data as a Product Podcast Network, is a free community resource provided by DataStax. Data Mesh Radio is produced and hosted by Scott Hirleman, a co-founder of the Data Mesh Learning Community. This podcast is designed to help you get up to speed on a number of Data Mesh related topics, hopefully you find it useful.

Bottom line up front, what are you going to hear about and learn about in this episode? I interviewed Mahmoud Yassin, a Lead Data Architect at ABN AMRO. Some key takeaways, they're thoughts from Mahmoud's point of view. It's very difficult to do fully decentralized MDM or Master Data Management, which led to some duplication of effort. That can mean increased costs and people not using the best data. ABN tackled this through their data integration access layer or DIAL, which is similar to a service bus concept. Number two, they are using that centralized layer, again called DIAL, to help teams manage integrations that are both consistently running and on the fly. It helps monitor for duplication of work instead of reuse. Number three, if Mahmoud could do it again, he'd focus on enabling data integration earlier in their journey to encourage more data consumption. Cross domain and cross data product consumption is highly valuable, but right now, it's just not happening at the level that they would have hoped.

Number four, the industry really needs to develop more and better standards to enable easy data integration. It's just not out there right now. We need people to really focus on this if we're gonna get Data Mesh right. Number five, Data Mesh and similar decentralized data approaches cannot fully decentralize everything. Look for places to centralize offering in a platform or a platform like approach that can be leveraged by the decentralized teams. I think this is something that a lot of people mistake when thinking about Data Mesh; you don't decentralize everything. Number six, most current data technology licensing models aren't well designed for or suited to doing decentralized data. It's easy to pay a lot if you aren't careful or even if you are careful. So really keep an eye on that. Don't do things for the sake of doing them. Really think about when, especially when people think about, "Should I do this in streaming or can I do it in batch? Do I have to do this in a super, super performant manner, or can I kinda fudge a little bit on performance?"

Number seven, a tough but necessary mentality shift is not thinking about being "done" once data is delivered, that's data projects. That's not data as a product, that's not that data product thinking. Number eight, try to keep as much work as possible within the domain boundary when doing data work. Of course, cross domain communication is key, but try to limit the actual work dependency on other domains if possible. Jesse Anderson talked about there's a 12x time increase when you are

doing crossdomain work; if it's just within the team, it's 1/12 of the amount of time and effort to actually move forward for his research. Number nine, a data marketplace enables organizations to more easily create a standardized experience across data products and make data discovery much easier. You don't necessarily have to tie your cost allocation models to the marketplace concept; you don't have to think about it with this real monetization bent. You can use a marketplace to create an experience, and that's totally okay. You don't have to really, really tie in the economics.

Number ten, sharing what analytical queries or data integration recipes people are using has been important for ABN. It drives insights across boundaries and also creates a lower bar to interesting tangential insight creation and development. If you've got a recipe, somebody created this recipe because combining these data from these four different domains really comes up with something that's really interesting, somebody can easily take that and augment it for their own practices. Number eleven, you should consider not allowing integrations across multiple data products by default. Producers should be able to stop integrations for compliance purposes, but also because the integration might not actually provide good or valuable or correct insights. The producing domain, this is again per Mahmoud's view, should be able to say, no, you shouldn't do that integration, let's actually talk about it. Number twelve, and finally, traditional ETL development is about translating the business needs to code. But centralized IT usually can't deeply understand the business context and needs, so they deliver substandard solutions. If you consider that business needs evolve, it gets even worse. I think a lot of people that are looking at Data Mesh, understand this, this is one of the big pain points, but it's good to reiterate these things from time to time. With that bottom line up front done, let's jump into the interview.

Very, very excited for today's episode. I've got Mahmoud Yassin here, who's the Lead Data Architect at ABN AMRO. And we're gonna be talking about a lot of different things. If you're not familiar, ABN AMRO has been on a Data Mesh journey before the concept of Data Mesh was really even out there. Their journey has looked a lot like what Zhamak has talked about. Piethein Strengholt, who was formerly at ABN AMRO, did a book, kind of about what they were looking to do called, Data Management at Scale, I believe is what it's called. And so, it's just a lot of interesting things, a lot of learnings about what they've seen that works and what doesn't work over the last three to four years. And so, I think it's really gonna be helpful for people to see somebody who's kind of far down this journey, and what they've learned and what are some of the anti-patterns as well, so that we can help people avoid those. So Mahmoud, if you don't mind, if you could give people a bit of an introduction to yourself, and then we can jump into the topics at hand.

Mahmoud Yassin
Yeah. Thanks a lot, Scott, for having me on Data Mesh Radio. I'm really happy to be here with you. And my name is Mahmoud, as you said correctly. I work for ABN AMRO indeed, it's a bank in the Netherlands. And I started my career as a data ETL developer, let's say, dealing with millions of records. And so I've seen, let's say the bane of doing data transformations in a classical, let's say data warehouses. And then, yeah, also move between businesses and stuff. So I also had the chance to do data lake architecture myself, as well. So I've seen how this works and the benefits of it, and also the drawbacks. And then all the way till I decided to come to Europe. And then, yeah, really the concept of Data Mesh, or as you said, even before it's being called Data Mesh, at ABN AMRO, we've been thinking how can we upgrade our data architecture, in a way. So we didn't call it mesh at that end, but we call it the DIAL project, Digital Integration and Access Layer, which I'm sure that we will touch during the recording. And I've been with ABN AMRO for almost five years, trying to help the company build that new architecture. And as you said, we're quite, let's say, not there yet, but at least we are not starting from zero. So, the project has started already, I think, 2018, so almost four years now. So, I'm excited to share the groans and the pain points, as you said, because it's about sharing knowledge. I'm quite happy and excited to talk with you today.

Scott Hirleman
Awesome. And there's a lot of different places that we could jump into this, but I think what you talked about a little bit is a good transition. What were you seeing that had been the issues with the data warehouse and the data lake, and how the concept of Data Mesh is trying to address those? And then we can talk about kind of the realities of, it's not that, hey, these things were an issue. And so now, Data Mesh solves all of the challenges. And you don't have any more challenges, right? It's all just perfect, right? No. So would love to hear what you saw as kind of those issues, especially as we hit towards more and more scale. And how you are looking for Data Mesh to address those.

Mahmoud Yassin
Yeah. So, as you said, I always say there's no perfect data architecture. So it's not also like, take it or leave it kind of concept. So, if you are not a data warehouse, then you should be a Mesh or a Data Lake. And I always say, you should choose what fits your company. And also, it's okay to mix and match. So there are certain aspects, definitely, that the data or the classic data warehouse is good at. Same for lake, same for Mesh, same for whatever will come next. And this is just a matter of choosing the best architecture, even customer architecture, that fits your organization.

But then back to your question. So when I started as an ETL developer, I was in the IT department, of course. And then, that main struggle was how to translate the

business needs into code and then get feedback on it, and then get it to action or to make a decision. And of course, this is not super easy, if you have an IT person trying to understand the business of what's going on. So, so often we go to our meetings with the finance, with risk, with many departments. Okay? They try to give us, or let's say spoon feed us, let's say the business or try to put us in their shoes, and that's quite hard, to be honest, right? Because let's say your background, your experience and the stuff that they're looking at, it's not the same angle that you are looking from it from an IT perspective, and that always led to, let's say that you don't get the idea in the first time, second time maybe if you're into the topic and you've done some R&D and stuff, you started to, to make sense.

And I remember a very complex project about billing in telecommunication, and then I was trying to understand how the billing system works in order to create an ETL job that does, let's say, evaluation of the customer usage versus their plan, rate plan of, let's say, of their mobile subscription. And that was horrible. So I had to understand finance, how billing works, and a bunch of other things that were super complex. And for me, that's one of the, let's say, disadvantages of putting everything in that central IT team to manage business processes and business demand, and at a certain point of time, this is really not scalable because Data Mesh I think is all about scalability. If you have a small business, maybe the classic data warehouse is actually more than perfect for you, but if you really want to scale that into the speed of data that you are getting and if you are a big organization that gives those millions and billions of records and terabytes, etc., then that one, that team will become the bottleneck for sure, there's no doubt about that. Because they have a limited capacity. It's not also about putting people and scaling it up, it's a little bit on the mentality and the ownership.

And that also is the same actually in the data lake. Data lake just deals with the way we interact with the data differently, but the same problem exists, and that's why I really like that concept of Data Mesh, where the domain gets the ownership of what they own, both, let's say online transactional processing systems and also analytical processing, because it's sort of a game changer. You push the game towards the people who understand and it is their responsibility to make their own even IT teams, if they have such a setup, understand that they live together, they work together, they are dealing in the standards together, so it is much easier than dealing with a completely different team, a centralized team sitting in IT somewhere and trying to do let's say the same idea. So that part I really like about Data Mesh, and I really find this concept or principle very fascinating. And if you manage to do it, it's gonna help a lot of organizations.

Scott Hirleman
Yeah, and a couple of things. One, you and I both know this, right, it's coming

through, but I always wanna be somewhat clear with people about it. It's not that the data warehouse or the data lake technologies are the issue, it's that the way that we've used them in that centralized approach.

Mahmoud Yassin
Yeah. A lot of Data Mesh, by the way, is in my opinion and the way I perceive it is a mentality shift rather than anything else. Because concept like, okay, the data as a product, that's a mentality shift, domain ownership, that's a mentality shift that people need to kind of grasp and understand in order to really feel that this product that I have, this piece of data, is truly a product that I want to own and I want to guard and I want to be, let's say, sexy and people like to buy it, and then that's a game changer. In the classical way of thinking, that's indeed my system, I get data out of it for analytics, I throw it away to let's say the data warehouse, data lake, good luck, I've done my job. If there is a comment, a feedback, a review, it has to go to a system, to a team, to a team, then it comes to me, then okay, I've delivered the data, that's not my problem anymore, and that is a really a mentality shift, and that's what also I like about those two principles in the Data Mesh.

Scott Hirleman
And the other aspect that I thought was interesting, is something that's come up more and more, was when you said, the second or the third try and that you still didn't always get the context, you still didn't always get it right. This is something that I'm seeing, I think it was Björn Smedman in his episode talked about, that's a signal that you've got an issue. It's not that you have the centralized team meeting with another team once and they're able to have that exchange of context and it works relatively well, the signal that your centralization isn't working very well is those second plus meetings for the same project. It's that, okay, especially if it's post delivery of something. If it's post delivery and then it's that meeting of, okay, let's try this again, let's start from scratch versus let's iterate on it. That's positive, that's positive change versus, hey, we didn't get what we wanted and that's a good interesting signal.

And I liked what you said as well of that kind of handoff of, I did my job, I delivered the data, I'm moving on to the next thing. Oh, you wanna change to this? Put it back into my Jira queue or my whatever, my backlog and so it's gonna be another three months before I get to it, 'cause I've got all of my backlog filled up versus that like, hey, we're gonna do fast iteration together and handing off between teams and teams. Jesse Anderson in his episode mentioned that it takes 12x more time if you have to work outside of your team. If you have to go outside of your team, that one piece takes 12x more amount of time and work.

Mahmoud Yassin

I fully agree. In a big organization, you have even those teams are from different, let's say departments, and if you want to even touch the team or the product team, then you have to go through a PI planning and maybe a three months, let's say already planned things, we have to wait for the next queue, and it can be logistically a little bit too difficult even to change just one column, and that happens... I fully agree. That happens also in most of the companies as well. So, crossing the boundary is kind of scary sometimes. So if you manage to contain, let's say, the development, let's say to a certain aspect, I would say because you cannot get rid of the cross domain, that's also, we'll touch that part in a way because that's a way to scale up. But at least if you can contain the changes and the feedbacks and offering your data or your product in a different shape and form within your own domain that really can be managed from a central backlog, from even one team or two teams, whatever, but then it would be much easier, the cycle will be easier and priorities can be called very easily.

Scott Hirleman
Yeah, and I think Piethein in one of his articles, or maybe it was in a conversation I had with him, said, you try to localize the work, but you don't localize the communication, so you still have that cross boundary work or that cross boundary, cross domain communication, but it's not that there's much cross boundary work if you can avoid it, because it's like, okay, what do I have to do within my own domain to work with this? And sometimes there's collaboration needed, but a lot of times it's like if you're really clear with what you're doing, it can be combined into another data product or whatever relatively easily, because you've set what you are going to create and that you're not saying, okay, we're gonna do a cross domain collaboration on this data product, it's we're gonna serve what needs to get served and we're gonna create it as a product and move forward. So I think that mentality shift that you were talking about is so crucial to get away from that like, okay, we're gonna have this big big cross domain project versus like let's break it down into chunks where we don't have to be reliant, where we're loosely coupled in even the work that we're doing to create the loosely coupled data products.

Mahmoud Yassin
Yeah, I fully agree, and I also thought of maybe an easier example also that I will use through my session with you, my recording, is actually, if you look a little bit on Amazon and how they sell products or how they facilitate, let's say, selling product, this example really resonates with me when it comes to Data Mesh, because you have, let's say, a producer of a product and you have a consumer of a product, and the role of Amazon, I think we will touch upon it in the next questions, is really crucial, and can be very much mapped to Data Mesh.

So, for example, if you are a phone manufacturer, let's say Apple or Samsung, and you want to offer your product to be sold in Amazon, if there's a bad review, who

picks this up? It's not Amazon, right? So Amazon facilitates like the communication, but then the one who gets on the feed will be the producer because they want to have the best product ever, and then they want to know why this is angry and then they start reach out to the consumer and then have the next version or update it or do something, or even compensate the consumer at a certain point of time. And that kind of annuls you, really makes it easier for me when I want ever to explain Data Mesh because if you have such a thing instead of course, the product, the physical product, you're talking about data, similarly, you can see it or you can make it quite a little bit easier, in my opinion, and I'm gonna build up on this story, I think in our talk.

Scott Hirleman
Yeah, and I think that product management learnings from outside of software, because most people are like, oh, I should do data as a product, I'm going to just go to software product management, and I think that physical goods product management is really useful analogy to think about because there are so many additional aspects. Software product management, there are some really good things to take from that, but it's also like, what is my actual demand? How do I do internal data product marketing? How do I actually test the market for what do people actually want? I'm thinking about creating this. Is there demand, or do I only wait for people to come to me and say, "I want this." No, you wanna be out there and being innovative and having those conversations, so I like pretty much everything you're saying.

So you had talked a little bit about this, and I'd love to weave in the journey and what went well, what you do differently, what's still really hard and all of that. But let's start, we talked a little bit about that centralization, decentralization. Centralization is a benefit until it's not a benefit, right, if you're a small company, centralization, the cost of decentralization is far too high. But as you're going along, there's so many different things that you need to figure out how decentralized or how centralized it is, if you could give a couple of examples as to what you've looked at and maybe even if there's a couple of different domains where the centralization or decentralization is at a different level, that's totally fine. Right?

If somebody really doesn't know GDPR, if they don't have anybody who really understands it in their domain, that that can be a little bit higher leveraged on the centralized team, so would love just kind of how you've seen that evolve, and again, what's worked well, what you tried, where you tried to go fully decentralized and it's like, no, that was a bad idea. Like all that.

Mahmoud Yassin
Yeah, so of course centralization, decentralization is one of the main, always a nice, interesting point to discuss when it comes to Data Mesh, because yeah, I think the

theory, let's say as it stands, goes for full decentralization and of course not for everything, because the governance layer you need to have, let's say, solid and cross all the domains, so that one is clear, but also when we thought a little bit of a fully decentralization, we kind of thought of benefits also of centralization to a certain extent, so what's kind of neglected in Data Mesh is the amount of effort that you want to do if you go with a full decentralization when it comes, for example, to master data management.

So if you want to build this golden record of your customer, the Customer 360 record that always ensures that you know what the customer is about and where he lives, etc. If you think on how can I do that in a full decentralization mode, that will be so, so complex and hectic, and that will lead to a lot of data duplication and maybe even more challenging times to manage all this data. So clearly, let's say the balance between centralization/decentralization, you need to kind of find that balance. And for us at ABN AMRO, I think what we've decided for, which is maybe it's not really like a data warehouse or a data lake, but then we have invented a service bus, let's say a layer that we call the DIAL layer between the provider and a consumer. That is not a data warehouse, not a lake, so it's just a facilitator, let's say between the two. But then what you do, if you put your data in this layer is that you ensure certain aspects to your data by default. So by default, the data gets checked. By default, the data quality gets applied to it. By default, the business metadata is gathered, the technical metadata is gathered, lineage information is gathered. And then everyone can benefit from that transparency that you have created with that layer in the middle. And to make it also back to the example or analogy of Amazon, so if you want to sell the product in Amazon, you have also two options, one which can map to the full decentralization. So the order comes in, you go to a manufacturer and say, "Okay, dispatch that order to Scott," and then you deal with him, let's say with the manufacturer directly. And then each manufacturer, if you can imagine, has a different SLA, different logistics, challenges, etc., etc. And then the end- to end customer journey will not be as smooth as, for example, Amazon has a concept which is called Fulfillment by Amazon, FBA.

So in the FBA concept, they have big warehouses in every country, whatever, scattered, and then you as a provider, give your product or deliver it as a bulk to that fulfillment center. And the moment this product physically enters the fulfillment center, tons of insurity is being done to the product, it's getting checked. The quality aspect is being done, it gets verified, it's not a fraud, the product exists, because sometimes you just order something from some people and then you get empty stuff and so on and so forth. And then metadata gets collected by default, and a bunch of other standards will be applied, logistically will be stored in optimized way, packaged in a, let's say, eco-friendly, so many, many things for the physical product just by ensuring it goes to the fulfillment center.

And then at the end of that chain, when a consumer requests a product or actually multiple products, they are in the fulfillment center. So they can easily package this, let's say, and wrap it in a wrapper and deliver that to you through their known SLA, known logistics, their own standards, which then you get that insurance that the data will get delivered in time with quality, let's say in a standard format, in an interoperable format, etc., etc. And this is what we've tried to do with the DIAL architecture. It's like a distribution layer, where you give your data, we take your data, we apply a lot of things to it before we make it consumption ready or available for a consumer. But once it is consumption ready, it gets a stamp from us that it is fit for purpose, and can be used across, let's say the organization. So that is, let's say, one of the main learnings and that implementation is done centrally.

So let's say that ecosystem around let's say the DIAL project is actually managed centrally as I code, but then used decentrally. And that's the main beauty, or the main trick here, that you don't create in Data Mesh, let's say services that the IT people are using, it is being built by IT, but used by business/domain as much as possible. And for that, in order to manage that complexity, we came up with the idea of the data marketplace, which is exactly amazon.com. It's a website, you hop on to it, you can search, there's a catalog feature, search and find, the data set, then you triangulate it, okay, this is what I want. When you click on it, it opens a whole page of description of the data, description of metadata, ownership information in which domain, stewardship information, description of it, how many times this data has been used as a sort of, let's say, incentive for people to know how much they can trust this data, you see a data quality of the data set, you see the data issues, if it has data issue, if it has data gaps, all of this is in a very transparent way.

So our goal is to let the consumer figure out the best product ever that fits his needs or her needs. And then you can also see a preview feature of the product. So seeing is believing. So let me show you what I have also before you can buy it, if that's possible. That would be cool, right? If you go to our showroom and then you can see the product itself, but in this case it's data so you cannot touch it up, but then you can see it in a viewer, then we do that as well. And then you get a bunch of information, and also feedback and ratings, etc. So we've created all of that, but we don't use it. So everything is done in a self service way where the domain hop into the portal. They have different accesses, where they can create that, okay, I have the product and I want a landing zone where I need to put my data, okay, fine, bunch of domains and automation will happen in the background. But you will get a path where you can put your data in whatever batch you can do, let's say whatever format and also whatever standard you want to apply. And then okay, but this data is just data, you cannot move to a next step without, for example, providing the business metadata of what you're offering; okay, this is my product, this is description of it, it consolidates

from these kinds of elements, each element has a definition, and the definition is registered, etc.

And then you build up that product thinking, but the key point here is who is building that, not us, not the IT people, it's being done by the domain. And then a bunch of workflows, a bunch of approvals, etc., also to ensure certain quality and certain, let's say, transparency, but at the end, once the product is published in the data marketplace and ready to be consumed, you know that this product is ticking all the gates of being qualified to be used within the organization. And then another journey is going to start, which is the consumption journey. So that part is really going very well at our end.

Scott Hirleman
And I've got like 60 different ways that we could go off of that. But I think one thing that you mentioned was the centralized governance layer. And I think another thing that a lot of people have missed 'cause I don't think it was as much covered in the initial articles, but the centralized experience layer, which is that, if you were to have, let's even just say relatively early days, you're year and a half in, two years in, not four years in like you guys are, but then you've got 75, 100, maybe even 150 data products. There are some companies that are getting to, yeah, within 12 hours of a request, we have an initial V 0.0.1 of a data product out the door and that you can start to iterate with your potential consumer, but that centralized experience is so crucial because if every data product has its own user experience, it's like learning a new piece of software for every single one, and then you're trying to combine data from across them, and so you're having to do... It's that thing of, okay, I'm pulling data from this person's spreadsheet and even just doing, this person does columns and this person does rows for how they store. And this person does dates in UTC, and this person does it in their personal time zone, and just even little obnoxious things like that, that it can take hours to clean that data and it's in an Excel spreadsheet. Now, if you start to think about software and learning, okay, I have to learn Power BI and Tableau and Looker, and whatever else.

Mahmoud Yassin
MicroStrategy also.

Scott Hirleman
Yeah.

Mahmoud Yassin
Yeah, I really agree with you because also just imagine how Amazon would be if you need to deal with each product portal or experience separately, right, and instead of having a shopping cart where I want this phone, I want this cover, I want this screen

protector and then order. Then you get the package as one package. And of course with data, you can strive for more, let's say if you want this data to be integrated, how cool would be if certain integrations are being done as part of this centralized layer to avoid duplicating that in every single domain, because for example, this is a crown jewel integration, so everyone wants the customer 360 data set, which comes from 10 different sources, and if you want to do that your own, then you need to copy the data from the 10 different sources to your domain, and then try to do or apply the transformation and then get the data out, and then you need to replicate this in, let's say five domains, and you can imagine what will happen. So translation can be done wrongly, also amount of duplication, which means not just cost, but also governance, GDPR, and a bunch of other things.

But if you push that to that central layer, then you can ensure that the data, for certain aspects, we are not saying that we are going to do a classic data warehouse or data lake where all the transformation has to fit a model, and then we do it there. No, we have invented, let's say, a concept that we call integrated dataset. So as long as the integration doesn't change the value of the data, then that's fine, we can do that in this centralized layer. If you really want to do a custom integration that's really and purely related to your domain, that can be pushed there because maybe that's something that you want to do for a specific report, for a specific regulator, that's fine, but for the common, let's say integrations that let's say will cover maybe 60% or 70% of what people looking for, how cool would it be if this can be generalized and then done in a central place, and that centralization can also be virtual. So let's not underestimate the power of virtualization here, so it doesn't have to be physically kind of integrated, it can be also logically, or let's say on the fly integration for certain aspects, and of course for certain cases, if you want to cash this in, then we have came with also a concept where you can physically store that integration, we call it the data access store in our architecture, and then that can be for performance reasons, can be reused.

So centralization is not a bad thing, as you said, I really like your statement, unless it fails to, let's say give the purpose and the lesson that we've learned that if you centralize this layer, you are in much, much more control than replicating this centralized layer in every domain. And to be honest, also there you will pay a lot, and also the tools are not designed for that, and that's maybe a very important aspect that the technologies that we have now around data, the architecture of it has been built for quite some years now based on the concept of classic data warehouse like, okay, I have this big machine that's in a data center centrally, and then I put everything on it and then figure out what to do with it. Yeah, it will take, in my opinion, quite some years for the tools and technologies also to mature to, for example, being, let's say, centrally managed, but then decentrally used, right?

That is how maybe I've seen some tools now doing that, so for example, Microsoft Purview is heading into that direction, so it is a central tool that can be implemented on that tenant level, organization level, but you can use it decentrally by each product or domain separately, but still you have that control plane layer, let's say, to administrate, to ensure certain quality, certain standard, etc. So yeah, I think also technology, we need to give it some time also to digest the concept of Mesh and then maybe at a certain point of time we can be fully decentralized, I don't know. But for now, if you try to do that, first of all, you'll pay a lot of money, because licensing is not done to be decentralized, and also duplication of the data, the complexity of the data you need to deal with, so maybe till the technology catches up with the theory then our advice is to, let's say do the data exchange, maybe in a central layer, but then as I said, develop it centrally, but try to make it used decentrally and selfservice is a great, let's say keyword here, that if you manage to create that to let the domain teams manage their own product, let's say like what Amazon does with their product in the portal, that is pure beauty, in my opinion.

Scott Hirleman
You made a bunch of different really good points in there that I wanna talk to, but one was that centralized integration layer, I think is an important concept. One of the first interviews I did was with Wannes Rosiers and he was at DPG media, and what they did was instead of having that centralized integration layer, in most aspects, what they did was actually swim upstream, push upstream. And what that meant was that if you were a downstream consumer, I think it's a lot more difficult with a company at your size and your complexity level, but anybody that was downstream consuming information from a domain, that information was pushed up to that domain, and there was a, everyone should understand how your data is being consumed downstream. That doesn't mean that you have to make changes, but it may be that kind of that central integration layer that you're talking about, it's the same concept of, well, if we've got five different people that are using this information that's been transformed in this way, we should just push that into the initial data product.

Now, if you do have that customer 360 view, that's not really possible because there are, like you said, ten different domains that are publishing into it, so you can't have each one of those domains own the other nine pieces from other domains and things, but you should think about pushing that up, so that way at least they know that this transformation is happening and then they can make a conscious decision, oh, I've seen four different people that are transforming it in this way, and they're all doing it slightly differently, and I think the correct way to do it is this first way, so I'm gonna have that discussion with everybody and say, "Hey, here's why we think you should use this," and you have a higher quality source, because we're doing it live and that you're not having any delay in that you're not having that repeat of that master

data management is somewhat about we wanna make sure that we understand that this is the right information, but it's also that duplication of what you said of, not doing the same work or essentially the same work, where it's like, okay, somebody is transforming these four columns and somebody else is transforming column 1, 2, 3 and 5, and transforming a new column of 5, but those 1, 2, 3 are looking exactly the same, so let's just push that upstream if we can.

Mahmoud Yassin
I really like the example that you gave because we also, at the beginning, we thought of making this as a hard, let's say, thing, so the Customer 360 can mean something different between you and me, right? So this also evolved a little bit in our thinking and we came up with, okay, this cannot be like one size fits all, so what we're going to do in the next phase is that we are going to create our integration recipe of how this integrated data has been built, also transparently, right. So you will go to the data marketplace, you will see that piece of integration that has been done by Scott for this particular use case, and then this is the recipe of how this has been done.

Now, you want to look into this and you can see, okay, okay, he's done this, he joined, he used this key, and then he filtered on that and then he eliminated certain aspects, and then you have a couple of options, maybe it is exactly what I'm looking for, and don't underestimate that as well, then you can say, okay, I just want the same. But you have also the option to clone the recipe, let's say, and create your own version of this integrated data set, and then add your own, let's say logic to it, and then publish it to the data marketplace, then you have two different variations, but that's healthy and that's fine, because maybe a third consumer will come in, then they will see, okay, for this Customer 360, I have the version of Mahmoud and the version of Scott. And then the difference between them is Mahmoud looks at it from a finance perspective, and let's say Scott looks at it from a, let's say a risk perspective, then the third one can decide, okay, if they want to reuse one of them and opting in for that reusing of one of them is much cheaper as well because the data integration is already known, the recipe is known, and based on your implementation, whether it's a physical persistence of this data or let's say virtualization of whatever the technique that the company will apply, then it's just add another let's say consumer for this.

And even let's say the other choice of copying or let's say cloning this recipe and trying to integrate, let's say, or add your own logic to it, there are also ways to make it more optimized, and then when the company matures and that amount of integration grows, I think hardly you will... Let's say if you're a data scientist or an analyst, you will go and go and skim through, let's say, all the integrated data sets, and then you will find that maybe 70% of what I'm looking for has been done already, then I only need to focus on the remaining 30%, and then I can take that integrated data set, push it to my domain, that's fine, and then try to do whatever analysis or

analytics, advanced analytics, machine learning, whatever you want to do there, and that can really save a lot of time, and that's I think how we envision that integration, let's say to be in that layer as well, so hopefully that will work. We've started, let's say, doing that already, so the concept is implemented, but now implementation wise, virtualization versus physicalization of the data, how the recipe will work, that is something maybe that we can chat about in a year's time to see how it evolves, but this is at least the line of thinking that we are heading.

Scott Hirleman
And one question that I've gotten Chad Sanderson at Convoy especially was kind of pressing me on this was, would those kinds of integration recipe type things, are those considered a data product or not, and who manages that? It becomes another question, because if somebody should be managing their own consumption from these different data products, but if those things are still changing upstream, that integration capability, that standardized format, that interoperable format, they have to keep meeting what they've been meeting, otherwise it's the same issue that we've been having with the Data Lake where everything upstream keeps changing, and so I have to keep adapting what I'm doing. So it's an interesting question that I haven't heard. I think you're even saying, we'll figure it out in a year, 'cause we're gonna test like how does this...

Mahmoud Yassin
So what we've done here, so I think might apply to you is, let's say it's about data sharing, let's say between, let's say the different products, and for that, let's say we've also invented, let's say a product that we call data sharing agreement application, that one will tackle this problem because you cannot just opt in for a certain integration without getting the consent or the approval of the original data owners and data stewards from the particular data sets that you're going to use.

So let's say in that Amazon example, so I'm going to use that phone and I want to put a screen protector and I want to put a cover to it, so we will get, let's say, the approval of the three, let's say producers that we at Amazon, or let's say data marketplace will do that integration and then will club the screen on top of the phone and put a case for it, and then we are gonna sell that towards the consumer. Do you agree with that? And do you think this is correct or not? And that will be part of the approval side, so it's not like, okay, a wish or something, it has to be discussed. It has to be approved, and then that means that those three will sit together, decide together with the consumer of the integration of the builder of the one who requested the integration, and then have a conversation, and once it is agreed, then everyone knows. So I'm going, my data, my product will be used in that way by this integration recipe, and then it will be used for this purpose. Now someone opts in and wants to change it, he or she has to go through, again, the approval part because it may have

been you're looking at it from a different perspective, maybe I can correct you, okay, take care, this case that you're trying to fit is not for this version of my product, then do it differently, so this will be tackled by kind of this process. And everything, in my opinion, if you have the really high transparency level in, let's say, whatever product that will manage this, so in our case data marketplace, we don't hide anything. So, in one shot, you can get to know about this integration, it is being done by these three or four data sets that they have this kind of ownership and stewardship, and this is the recipe of integration and these are the steps, right?

And then if anyone tries to play and change something, it means something downstream, so that kind of impact needs to be assessed, and that's also something that we're really actively looking towards, but yeah, we haven't implemented that whole ecosystem, but at least hopefully, this will tackle the question that you mention, and of course avoid, let's say that scenario. Let's say, if you go back to, let's say, changing upstream without, let's say everyone knows what's happening, so I hope that that will solve this problem as well.

Scott Hirleman
Yeah, and I think it's interesting right now talking to a lot of people, they're saying that the way that we share information, because the way that we share data as in the ones and zeros, API is fine, in most cases. The way that we share information and analytical queries, like analytical APIs, are far different. We need them to be able to do big, complex range bound queries of like, okay, I want everything with a bunch of different filters and things like that, and that we're pulling a lot of data instead of small amounts of data, and yes, we can do that somewhat with SQL and things like that, but it's a complex topic the more that you start to dig into what should an analytical API look like. But that should be the interface, and so we can still have upstream changes within the application in the operational system as long as it doesn't break that interface, that analytical API that's being shared, that we still have that change and that we can quickly evolve our operational side, but that we don't constantly have this one to one tying between our operational and our analytical because those just are constantly evolving and breaking, and that's what's been the big issue.

Mahmoud Yassin
But that's one of the hard pillars of data as a product, right? So you don't also offer one product, your product can have versions, right, the phone can have get smarter, slimmer, brighter, the camera can enhance, but then you can ensure also backward compatibility, and that's something that also in the product thinking when you implement, that must be one of the main pillars that, let's say when you provide, when things get evolved and you provide, let's say, a next interface, be it an API, another batch or a stream or whatever, the data, your data, your product will be

consumed. You have to ensure backward compatibility and for whatever reason you want to decommission a certain interface, then through that transparency layer, again, you get to know via the lineage capability, who is going upstream is going to be affected. And then, of course, it is your own responsibility as a domain to communicate that and figure out a way with all your consumers to roll out or get rid of this interface and then let them switch to the newer model or the new version of it. So that also, I think, will be a great addition if we manage to create that mindset, and also if the tools can allow for such a thing, that will be really nice to have as well.

Scott Hirleman
Yeah, that's something I've been pushing for is an autoincrement from version to version. If you can test how somebody is using your data product, you can say, will this version change actually impact them? And if not, then they're automatically moved to the next version because there's no reason for them to not start using the new version, so we want that as well, but that's running. We're still at crawling stages in a lot of cases.

Mahmoud Yassin
Indeed, indeed. So it's just a journey right, so we have to embrace that this is a journey and the direction is clearer, but then you know how to go there. Yeah, over the upcoming let's say four or five years, I think there will be some struggles, and I really believe in what you're doing now, because sharing these experiences and failures and things and also successes is the key, let's say, to not just evolve ourselves, but also the technologies also evolve in the direction, otherwise then you want to go right, but then the car doesn't take you right, then you will be like going nowhere or going to a diverse direction.

Mahmoud Yassin
So that's really key, I think here when it comes to Data Mesh because let's put it fair and clear, it's new architecture, and then data warehouses have been let's say they're from the '70s. And now if you go to any company and even they have the ready made model for you based on your industry that you can just put in the data in, and then you will get a lot of integration already happening, that kind of maturity, because it has been there for 50 years, 70 years, but yeah, Data Mesh is really, really fairly new, so don't try to, let's say, push too much because it will take some time.

Scott Hirleman
That aspect that it's a journey is crucial and that there are pains and there are things that Data Mesh hasn't fully solved for, and it's not that, okay, you move to this and everything is all of a sudden golden, it's not that data becomes easy, it's that you were making these changes so that we can have that scalability and agility and stuff.

You mentioned that there were some failures along the way. If you're comfortable sharing, I would love to hear a couple of places where you tried something and it didn't work, and then we can talk a little bit about why that didn't work, because the point of this podcast isn't to sell Data Mesh, it's to talk about how people are actually attempting to do it, what's working well and what's not, so that we can create our list of antipatterns. We kinda have certain anti patterns of like, we want Polyglot, don't force everybody to use just SQL or anything like that, and don't just give everybody the full thing where they have to do all of their own governance and that you have no standards and no interoperability and all that. Those are kind of somewhat relatively obvious first level antipatterns, but would love to hear some stories of some things that you were trying that didn't necessarily work and how you've evolved those into a place that's working much better.

Mahmoud Yassin
Yeah, I think we already covered the solution of the challenges that we have faced already in the talk, but to be more precise, over the last, let's say, two years or even three years, we've added most of our capacity and effort into that layer, the central layer that we call the DIAL architecture into getting data in. So let's say rolling out, let's say the very enriched programs about data ownership, data stewardship created, let's say, functioning organization in order to realize what this means to you, etc. That went very well and now everyone knows about that, and then they started to onboard their data through DIAL and then get to know the benefits of doing that. But what we kind of underestimated during this period is that we always said, okay, let's leave the consumption for now and we will figure it out later.

But after, let's say, those scaling up a lot in the sourcing side of getting the data, and if you now look into the ratio of the consumption, it's not actually as what we have hoped, but that it makes sense to me because we didn't invest much there, and that's why I really advise so don't try to tackle it from one angle only. Try to, let's say, okay, work on the onboarding of your data and whatever solution that you will use, but also think of the consumption and how it will hunt you even if you don't, let's say, think of it in the same time, that's why, for example, it was obvious from the beginning that we need to have an integration kind of thing, but then, okay, how we are going to do it and what kind of a style virtualization versus, let's say physicalizing the data, all this stuff just came recently, and we feel like we've maybe if we could have done that earlier, then we could already have been, let's say, already producing or let's say consuming data in a much faster rate, and also automation is a key here because we facilitate a lot of consumption, but this is done manually.

So a team is trying to help everyone, let's say in the consumer domains to, let's say, realize how data should be integrated, etc. But if I have another, let's say, time, if I go back in time, I think we would have them look into how to automate that from the

beginning, which we are going to start anyway nowadays, but maybe my advice is, okay, don't underestimate the consumption side when it comes to Data Mesh and also the challenges when it comes to duplication of the data and how you're going to manage that and how, let's say, hard would it be to know from a lineage point of view and from a metadata management, where is what and what's happening to your data? And if you are really in a higher regulated business like ours, that can lead to big, big fines if you don't know what's happening, and then try to, let's say, avoid the chaos, let's say, of fully decentralization, and then because it comes with its own also, cost, in that sense. So I hope that's clear.

Scott Hirleman
Yeah, and I think, especially on that last point of, you don't jump from centralized to decentralized. It's not like, I talk about, with microservices, there's thin slicing where you take off a microservice at a time, you don't go in and smash your monolith with a hammer and it just goes into a bunch of shards and it works, right. You need to think about what are you actually trying to accomplish. And I think that there's been a lot of talk. It's very interesting 'cause I'm seeing a few different kinds of patterns emerge around what is the kinda genesis for a data product? What is the factor that causes a data product to get created? Is it that the domain thinks that they have information that they want to share? Is it that there's a specific use case that emerges? There are some people that are focusing on sharing more of their data, but not actually as a product initially, and then that they kinda have low quality data, but everybody understands that it's low quality data and then people can understand what data is available and on offer and that they then say, "Okay, this is the data product that we want to emerge. But then it's kinda weird because it ends up not being like to a specific domain," it'll be like, "Oh, we want this information from these like four different domains." And so then they kind of create these micro data products to almost just serve that specific use case.

And so it's very interesting, but I think what you're talking about is something that's... When I talk to people in financial services especially, is that a huge part of this is the regulatory, it is the governance, it is compliance. Do you need to have every piece of data integrating with each other up front? No. But you do need to create ways to allow for that. And we don't have standards. Nobody really has published... There's a couple in like BioLife Sciences and things like that, but we need more standards around how people are actually doing this integration. So it's not that every company has to reinvent the wheel themselves because right now it's, like, "Okay, do you just kinda do the old like linking key of, okay, we're gonna do all of our timestamps in this way, in whatever, and we're gonna have a couple of universal IDs if we can get them." But even in banking, it could be that one company has like seven different subsidiaries. So like how do you do all sorts of fun stuff?

Mahmoud Yassin
Even if you want to do it, yeah, if you are in a multinational, let's say business and then each country has its own regulations as well. Right? So don't underestimate that. And that also we yeah, we have to deal with that also in a way. So indeed, I think on the consumption side, we really need to find, let's say, or team up in a way to figure out how can we do, let's say, if possible, standard integrations and so on and so forth. What I've seen going on nowadays is there has been some discussions also on some actions even to standardize or standards for lineage sharing, for sharing the metadata. And that can be very powerful in that example because, you know, first of all, you need to know where is what? Right? And even if you build those kind of integrations, maybe in low quality data, if you don't know if that integration exists, then you cannot spot this and even challenge the consumptions and also ask them for a better version all the way through the producers.

And for that, there is a standard, I think, although data lineage is being built by, I think, OpenMetadata, that's the initiative. And then they really strive to create the standard of sharing lineage information towards a central capability in organization because transparency is the key here. So if you know what is happening where, that will help a lot. But nowadays, you know, each integration can happen in, I don't know, in a specific domain, but you will never know about it as a, let's say, central capability and that will complicate things moving forward. So I really hope to team up in that area also with many people to figure out how can we solve this.

Scott Hirleman
Yeah. And I think we're seeing a lot of tools emerge around data quality and things. But there's a trapped metadata problem, right? Where there like, I say, multiple panes of glass is a major pain in the ass because if you have to go and figure out, like, I want to understand this information and you've got your marketplace, right, you say, "Your marketplace has all of the information." If I have to go into all these different things or if I have to do these complicated translations just to create a simple standard set of how we do our metadata, it's such a big pain point and I'm just not seeing... OpenMetadata is trying to kinda tackle that. And OpenLineage is also one that's very specific to lineage, but it's kind of narrow minded around just being lineage instead of like lineage is part of metadata. And metadata is kind of the crucial aspect. But yeah, I think there's just a lot of these challenges.

Mahmoud Yassin
There's a new concept that we are working on actually already for a year now that we call... But then don't hate me, we call the metadata lake. So we figured out that, you know, to harvest all the metadata across the organization, first of all, there is no single tool in the market that can cope with, let's say, large data types and large technologies, etc. So we have decided to build, let's say, a tool agnostic lake just for

metadata, because it's so crucial in this.

So if you have all the metadata in your organization hopefully in one place, and then there are tons of opportunities that you can do apart from, let's say, reporting on it or sharing it with everyone for backward compatibility, for knowing what's happening for regulators, for whatever. There's many, many use cases for that. But then we found that we need to have a dedicated focus for metadata. And then, but maybe for another episode, we can talk about the challenges with the metadata harvesting and how can you build a tool agnostic solution. So in our case we have like three or four different technologies; they are trying to extract the information or the metadata from all kind of systems that we have in the bank and then put them in that central place, in that case, centrally because we don't want to do Mesh for metadata. Anyway and then we store that data locally and then we build on top of it the really, really beautiful use cases. And then it's really useful for lineage of course, but then for other aspects as well, for operational optimization, for cost optimization, for connecting the dots between the data, the processes, the technologies and the people as well. But I think that's a topic on its own and we can maybe chat about it later.

Scott Hirleman
Yeah. I was gonna say there's another couple of episodes that we could just dig into any of these specific issues. So I think we've covered a lot of really interesting topics and things. Is there anything we didn't cover that you wanted to cover, or is there any way that you'd like to kind of wrap up the episode as well?

Mahmoud Yassin
No, I think we covered a lot, but of course, the data sharing and the consumption of the data, that's something that I really call for help also for all your audience. If you really have some cool ideas that you want to share or even co-create something together, I'm very open to see how you've done it, and maybe we can team up and try, as you said, also figure out the standard or consumption, I don't know, but really, this part will be a little bit complex if you leave it at the end, so let's try to realize that now and figure out a way to solve it as well.

Scott Hirleman
Yeah. I tried to launch a thing that I called Data MOSS, which is Data Mesh OSS just 'cause it's terrible, but it's memorable. And it's something that you can actually search for relatively easily. And I think that's gonna be something that emerges in a couple of years; I think I was way too early with it. But I think exactly what you're talking about, we need standardization around just like, how do we store events, right? How do we store events and so that there's just a way that...You know, you make them extensible and things like that, but that every company doesn't have to reinvent everything each time, and that we can start to say, "Hey, we tried this and it

didn't work. Let's evolve away from this," and things like that. So you kinda gave a good summation of what you want people reaching out to you about, but is there anything else that you want people reaching out to you about? And where is the best place as well?

Mahmoud Yassin
Yeah, I think the best places is, we can definitely connect to each other through LinkedIn, but also I think I will stick around also you, and I think you will also do meetups as well to gather people around and tables, etc., so I would be also very keen to join and then listen and find out or team up with people from these kind of events. And otherwise, I'm really open to and reachable as well, I can share the details of my own with you, and then, yeah, really welcoming any feedback and also co-creation, if that's possible with you as well.

Scott Hirleman
Yeah. I'm hoping to launch in a lot of different countries, the physical meetups and local, that it's not managed by me, that it is managed by the community.

Mahmoud Yassin
I can host the first one in the Netherlands, for sure.
Scott Hirleman
Well, we actually had...

Mahmoud Yassin
Reserve that for me. Oh, you already have...

Scott Hirleman
We had one a week or so ago with Piethein and there was... I can't remember... Oh, Paolo Platter from Agile Lab and Ferd Scheepers from ING. Piethein was supposed to actually reach out to invite you, so I've gotta give him a little bit of grief, but... So Mahmoud, this has been really great, really enjoyed the conversation. I think it's gonna be really helpful for a lot of folks. So I really wanna thank you for taking the time today, and I'd like to thank everyone as well for listening.

Mahmoud Yassin
Thank you. Speak to you later.

Scott Hirleman
I'd again like to thank my guest today, Mahmoud Yassin, who's the lead data architect at ABN AMRO. You can find a link to his LinkedIn profile in the show notes as per usual. Thank you.

Thanks everyone for listening to another great guest on the Data Mesh Learning Podcast. Thanks again to our sponsors, especially DataStax, who actually pays for me full time to help out the Data Mesh community, if you're looking for a scalable, extremely cost efficient multi data center, multi cloud database offering and/or an easy to scale data streaming offering, check DataStax out, there's a link in the show notes. If you wanna get in touch with me, there's links in the show notes to go ahead and reach out. I would love to hear more about what you're doing with Data Mesh and how I can be helpful. So please do reach out and let me know as well as if you'd like to be a guest, check out the show notes for more information. Thanks so much.