Data Mesh Radio Episode #293: Adapting Product Management to Data - Finding the Customer Pain and the Value

Interview with Amritha Arun Babu Mysore

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
To get their Data Products for Dummies book (info gated), please see here: https://www.starburst.io/info/data-products-for-dummies/
To get their Data Mesh for Dummies book (info gated), please see here: https://starburst.io/info/data-mesh-for-dummies/

List of all previous episodes (including links to available transcripts) is available here.
The folder of all previous transcripts is available here.

0:00:07 Scott Hirleman
A written transcript of this episode is provided by Starburst. For more information, you can see the show notes.

0:00:18 Starburst
Welcome to Data Mesh Radio with your host, Scott Hirleman, sponsored by Starburst. Starburst is a single platform to help you activate all your data, no matter where it lives. Check out our new Data Products for Dummies eBook to learn more about how your organization can utilize data products. To download your free copy, head on over to starburst.io.

0:00:40 Scott Hirleman
Data Mesh Radio is provided as a free community resource by Data Mesh Understanding. It is produced and hosted by me, Scott Hirleman. I started this podcast as a place for practitioners to get useful information about data mesh. We're at over 200 episodes. I've now left DataStax, thanks for all their help in founding things, but I've left to start Data Mesh Understanding, which is also helping practitioners to get to the information needed to do data mesh well. We have free implementer introduction and roundtable programs, in addition to the more advanced yet affordable offerings, so please do get in touch if you're looking for more information on how to do, how to approach data mesh. Just check datameshunderstanding.com for more info. There's also a helpful organization of past Data Mesh Radio episodes there if you want to dig into specific topics rather than digging through 200 different episodes. So with that, let's hit the funky intro music and listen to what you'll hear about in this interview episode.

Adapting Product Management to Data, Finding the Customer Pain and the Value. Bottom line upfront, what are you gonna hear about and learn about in this episode? I interviewed Amritha Arun Babu Mysore, who's the Manager of Technical Product Management and Machine Learning at Amazon. To be very clear though, she was only representing her own views on the episode. So in this episode, we used the phrase, "data product management" to mean product management around data rather than the specific of product management for data products. So it can apply to data products, but also something like an ML model or a pipeline, which I'll call data elements when I'm talking about it in this summary. So here are some key takeaways or thoughts from Amritha's point of view.

Number one, "As a product manager, it's just part of the job that you have to work backwards from customer pain points. If you aren't building to a customer pain, if you don't have a customer, is what you're building even a product?" Number two, always focus on who you are building a product for, why, and what is the impact? We just keep coming back to this. This is the key of product management, and it doesn't change simply because we're doing data. Number three, data product management is different from software product management in a few key ways. In software, you are focused "on solving a particular user problem." But in data, you have the same goal, but there are often more complications like not owning the source of your data and potentially, more related problems to solve across multiple users. So you're not as necessarily focused on a very particular user problem.

Number four, in data product management, start from the user journey and the user problem, then work back to not only what a solution looks like, but also what data you need. What are the sources? And then, do they even exist yet? Number five, product management is about delivering business value. Data product management is no different. Always come back to the business value from addressing the user problem. If there isn't a business value from addressing that problem, why are you trying to address that problem? Number six, even your data cleaning methodology can impact your data. Make sure consumers that do care about that, usually data scientists, are aware of the decisions you've made, and maybe bring them in as early as possible to help you make the decisions that will work for them and work kind of for all, so they aren't stuck with a cleaning methodology that they're not okay with. Number seven, potentially controversial, try not to over-customize your solutions, but oftentimes, you will still need to really consider the very specific needs of your consumers or your main consumer. Build for reuse, but also build where your consumers are actually having their needs met. A mediocre solution for all is usually worse than a few specialized solutions.

Number eight, prioritization is crucial in product management. This just keeps coming up over and over in any conversation around product management. But that applies to features within the products, but also the products themselves. There are many potential use cases that won't be met because there isn't enough value. That's the name of the game, return on investment. It's not about capturing all value possible, but making strategic bets to capture as much value on what limited resources you have. We don't have unlimited resource. Number nine, communication and building relationships and trust are foundational in product management. It's an art as much as a science. If you can't have tough conversations and get alignment, it is far harder to build a product that actually will meet customers' needs. Number 10, relatedly, establish regular communication with your customers. You shouldn't only be talking to them when things go wrong. Stay on top of what is driving value for them and look to augment your product proactively, not only reactively.

Number 11, product management requires patience as much as diligence. Sometimes your data product or your data element violates its SLAs, but it was just an outlier, a one-off. Don't look to overreact and jump to changing things. But you obviously to need have serious conversations if your data elements aren't meeting expectations over a more extended period of time. Number 12, if you aren't sure what products you should create in a new area, talk to people and find the points of frictions. What are the pain points? And is there enough value in addressing them to justify doing the work? Again, this is just kind of a key of product management, but far too often people say, I have pain and we, data people, try and jump to solving that pain instead of saying, should we actually do it? Number 13, it's crucial to deeply converse with potential users of a data product or data element to assess if it's really going to be worth the effort. There is always a chance you build something that isn't used or isn't valuable, but through deep investigation and ideation with potential customers, you can avoid that far more often. Really ask the tough questions of, is this actually a value to do?

Number 14, when you are building something, even before it hits GA or general availability, get validation. You can save yourself a ton of effort and re-work as you find a better solution sooner. You don't go farther down the path and have to re-trace. Number 15, product management is about collaborating to drive towards value. You are there to prioritize and coordinate. You don't have to know everything, but your job is to uncover as much understanding as possible to maximize your value creation and minimize wasted work, right? The product manager doesn't have to know everything, but the job is to figure out, where are the points that you really need to focus? Number 16, always ask what value building something for your customer will drive. But also ask what happens if we don't build it. What is the cost of not acting?

Finally, number 17, the only constant is change, especially in data. Leverage a "loosely dependent architecture," that way you'll be able to adapt to change and be open and honest with customers that things will change. Emphasize you'll work with them to adapt to those changes, but they have to understand change is always a comin. Okay, enough of just me. Let's hear from our awesome guest in this interview episode.

Okay, very, very excited for today's episode. I've got Amritha here who... Amritha Arun Babu Mysore, who is a product leader. To be clear, she's only gonna be representing her own views here, but we're gonna be talking a lot about product management within the data space. So we're gonna be talking somewhat about product management, but mostly about product management around data, and thinking about how we can apply that, and then we'll kind of do some of the stuff as to how that can apply more into data mesh and data product management. But just in general, product management in data is even a relatively new kind of discipline. And so, what can we learn? What can we extend that into data mesh? But I think it's also just a very, very useful, interesting topic.

We're gonna talk a lot about, how do we even just have good communication around data and needs, right? When is it time to actually have a new conversation? When is it time to re-assess different things? Is there a rule of thumb as to, this has deteriorated to X percent to really spark that we need to have another conversation? How do you identify the problem spaces to build data products and/or models and how AI and ML are stuck between the upstream of you don't control the sources and you don't really own a lot of the downstream where people are consuming your ML models and things like that. And how to measure when a model isn't working anymore and how to think about retrain versus shutting down, and we'll extend that a little bit to thinking about data products and how, when we think about, this isn't doing what it should be anymore. I need to evolve.

So we're gonna be doing some concrete things of her background, but we're also gonna be doing a little bit of thinking in the open space as to how could we apply this to other areas as well? So I'm very, very excited about that episode or about this content and these questions. But before we jump into that, if you don't mind, could you give people a bit of a background on yourself, and then we can jump into the conversation at hand?

0:11:47 Amritha Arun Babu Mysore
Perfect. Thanks, Scott. Hi, I'm Amritha. I'm a Product Manager at Amazon. I have worked at various big tech companies as well as startups throughout my career. I have built products in supply chain, as well as other domains. Yeah, happy to talk today more about data product management.

0:12:13 Scott Hirleman
Yeah. Well, and let's talk about that product management and the differences from what you've seen historically versus when you started to really think about... 'Cause you're creating products, you're creating ML models as products, right? And so let's talk a little bit about, how do you see those as the same or different when we think about software product management? And where I'm really trying to get a little bit more color on is, I'm struggling when I have conversations with people about, what do we take from software product management? And what don't we take? What do we have to... What don't we have to re-invent because it's a wheel, and what do we have to re-invent because it's an engine and we can't use the engine from a wagon because it doesn't have one? How do we think about... I know that's a terrible analogy, but how do we think about what's useful and what's not from the product management space of software into data?

0:13:15 Amritha Arun Babu Mysore
Yeah. That's a good question 'cause I've been thinking about that myself as I transition from software product management to data, and then an ML product management. So what does all this mean? And what does this mean for any newcomers as well, right? So in the software product management, you are focusing on solving a particular user problem. Let's say, for example, if you are managing contracts manually, let's say vendor contracts, you build an application for automating that. That could be building a workflow to record the contracts, it could be building UI to capture the contracts and all the information about it.

Similarly on the data or machine learning product side, there are scenarios where you would work on identifying, what are the customer problems, because that's where the similarity in both the product management is, that as you're working backwards from customer, you're making sure you identify a problem that is unmet, and then you can solve that and tie that to the business objective that you are company solving for.

So that is the similarity. But where it differs is, as a data product manager, you have to think more about, what are the data aspects? Like, for example, what are the sources of this data? Well, how was this data generated? What are the mechanisms or the schema of this data that is being stored in? How will this data be interpreted and used downstream? So there are a lot of aspects around data itself that a product manager has to think about before designing a particular solution. So yeah, there is similarity, but at the same time, there is enough difference for it to exist on its own.

0:15:23 Scott Hirleman
And it's interesting 'cause almost the exact same phrasing came up in a recent episode with Ryan Collingwood, where he was talking about, we... I think it was Ryan

who was saying this, but that there is a specific problem in product management in software, whereas in data, sometimes you have that specific problems, but often you don't. You're trying to represent what is happening and sometimes that is targeted at a specific problem, but often it's not. It's trying to encapsulate this for users that might be trying to look at multiple different problems. And I think that's where it gets even harder and harder to represent. When it's software, it's, again, I'm going to try to tackle this exact flow or this exact problem versus data, there's like, there's so many different uses of this data. So I'd love to hear how you think about, how do we encapsulate that or how do we find the problems, and how do we find a way to...

I know I'm asking you this incredibly difficult question, but how... Solve data for us. But how do you think about solving a specific problem versus creating something that can address, if not solve, multiple problems or kind of creating it in such a way that it isn't so only specifically tied to that one problem because the value, the big, big value in data often comes from multiple use cases, reuse of that data.

0:17:02 Amritha Arun Babu Mysore
Yeah, that's a brilliant question. I'm thinking to see if there are a couple of ways I can give you an example for that. But to answer at a very high level, to build a particular data product, whether it's a data asset or a data pipeline in itself, there are, one, the obvious way that is where you are building something from ground up. Let's say you're building a particular model for fraud detection in those... Like a financial services company who is building a model for fraud detection. Then you would typically work backwards from saying that, "Okay, what are the kind of transaction data that I would need? What are the kinds of sources that I have here?" And then you work back. You also map around the user journey to determine, like, let's say... Again, you need to get much more specific here to say, let's say fraudulent with, let's say credit card transactions, then you start mapping out the user journey for credit card transactions.

And then you can start understanding, what are the data inputs at any given point? What are the data outputs at any given stage? And then based on that, you start building your vision and requirement. Because here, the requirements has to be very specific on as to what are the data type, data attributes, data schema that you're getting at any given stage. And then you as a product manager, you're also going to work with various stakeholders like data scientists and engineers and ML engineers to understand what are the kinds of models that you need to pick because there are like so many different kinds of model architecture in themselves that can be used in any given scenario. So that's the path that you would follow, which followed by you would build it, test it and deploy it. This is a scenario where you have a clear cut out problem or a clear defined problem.

There are scenarios where, let's say for example, for any reason your particular model started deteriorating or you're not getting the right kind of recommendations like for example you wanted recommendations on a particular product and you're not getting good recommendations. In those scenarios, the model itself is not performing well. Here, the problem is not clearly cut out for you as a product manager wherein you are not sure whether it's the model that is not performing or is it the data sources that have changed or the data that is getting processed has any issues.

So here it becomes very tricky where product manager would have to dive into the technicalities of IT work, collaborate heavily with the ML engineers and the software engineers to understand what are the inputs and outputs at each and every stage, compare it with what was the original thought when we built this particular pipeline and this particular product. So that's how I would look at this particular scenario.

0:20:31 Scott Hirleman
Are you typically building a model for a very specific use case like one user or are you typically building models that are reusable? Because that reusability is so challenging. The reason why I'm poking at this so much is, I was just having this conversation with Ole Olesen-Bagneux of, we're really, really struggling in data to build things that are reusable for people that were... It wasn't built specifically for that use case.

And so I am asking, do you have a solution for one of the hardest one of the hardest challenges in all of data which I'm not saying, hey, but I mean how do we think about creating something that isn't only targeted, so overly targeted but is still at the level where you don't have then the user having to do 90% of the work you know that total cost of ownership falling onto the user. That's not what a product is, that's what maybe a project is, that's when we're thinking old school just delivery dump the data in the bucket and walk away. How do you think about mapping that? I mean is it just so much it depends that it's very, very difficult to kind of abstract away the patterns to figure out how to make something that's reusable?

0:22:00 Amritha Arun Babu Mysore
That's a very good question. So there are... Let me put it this way, a model is trained on a particular data set based on the objective that you are trying to achieve for that particular model. But even if you take for example let's say ChatGPT for regular conversational that any typical person would, could use, those same same ChatGPT can be taken... A version of those the GPT LLM, can be taken and also used for let's say if you want to analyze financial information of 10-K and 10-Q financial reports.

You can even augment it to provide you insights on that. So it goes back to, what is the LLM trained on? Was it generic enough like as in data around the world that's available on the internet on those... Is it trained on that? Then you augment it from there where you can fine-tune it, you can use part of the model to make sure this is the questions that you need to be answered or this is the scenario where you're going to reuse this particular model, so that's how I would think of, yeah.

0:23:27 Scott Hirleman
And I think maybe we can also move this a little bit into, you're the consumer of data too, right? How do you talk with your upstream sources? Are they typically building something that is custom to you or are they Zhamak with data mesh is saying maybe we don't need feature stores anymore or maybe we do, but they're actually just data products themselves, but most of the time ML engineers they freak out when you start to say, we're going to take away your feature store and then we go, but we're just going to give you a clean source of data and they're like, no, no, no, I want raw data and it's like, no, you don't, you want clean untransformed data. You don't want raw, raw data, you want clean yet untransformed. You want the vegetable as it's been washed you don't want it with all the dirt on it anyway, you want it that it's been cleaned for you and that you know that it's also not got any like salmonella or anything like that, but how do you think about communicating that upstream as well because you're taking downstream needs and you're again, in that middle where you go, I need this and how do you start to have that conversation as well?

0:24:47 Amritha Arun Babu Mysore
There are a couple of scenarios that have come up here. One is, as a ML engineer, we need to... So oftentimes, as a product manager, when you encounter such situation, you also have to think about where the person is coming from, what is their activities, what do they do, why do they need this data? Because product management discipline as such is about working with various stakeholders and oftentimes, they are all in different domains and that means that different skill set, different mindset. So it is very critical to come to the table with that to understand where they're coming from.

So oftentimes it is ML engineers taught us that, I could do this cleaning by myself and might be these are the data sources that I need. Probably when I get a clean data, I might be missing out certain features whereas if the communication is more turned around in a way that we are going to reduce cost, we are going to reduce time to experiment for you, we are going to provide you clean data wherein you as our stakeholder has opportunity to prescribe and be part of the journey where we determine what are the cleaning methodologies that we are going to use? What are the pre-processes that we are going to use so that you understand what is being taken from the source and how is it transformed or to a clean data, but not to an extent where it's just feature engineered for you, but you have a clean data enough that you can go boom and start feature engineering and increase your velocity to experiment. So that is a conversation that is very critical and the more you take folks along the journey the more easily it is to get by it.

0:26:42 Scott Hirleman
And so since you're the recipient of all of that, is it that your job is communicating what your downstream users needs are or is it more communicating what your needs are. I kind of think, I don't know if you've seen “Office Space” where, but there's the guy where it's like, so what is it that you do here and he's like, "I take the the requirements from the customers and I bring them to the engineers." It's like, so you do that physically and it's like, well, my assistant does it, but like sometimes product managers feel like they're doing that, but obviously there there is far more value in the job. I don't agree with Airbnb's whole thing of cutting out product management, but like how do you think about inserting yourself into the process as to what that downstream person needs and then creating your own tempo, your own timelines and having that communication up and down.

How do you think about that and then once we've talked a little bit about that, I'd love to transition into as well, how do you know when it's right to have a conversation when things haven't deteriorated or aren't doing those things, but how do you think about setting that pace and the, what's right and how often are you putting the end consumers of your model in product… Into contact with the sources of your data versus how should people start to think about that because this is… In data mesh, the central team has to move further and further away from being in the day-to-day, you don't have that as the person that's actually creating the ML model, so you are creating the data product in that standpoint of what we actually think of as a data product in data mesh.

So long, long, long explanation, but I think people are really really struggling as to, who does what and when and how does that actually flow like, do you have any tips as to what works well or maybe some anti patterns as to what really doesn't?

0:28:54 Amritha Arun Babu Mysore
Yeah. See, as a product manager, it's just part of the job that you have to work backwards from a customer pain point and a customer. If you're building a product without a customer, it just… I'm not even sure what is the product going to solve at that point. Are we… It could even in that like… I'm unable to comprehend that because even in that case if I'm building a product, I don't have a customer might be I have an intention, the intention is stemming from my own thought and biases about a problem, then I'm building a product for my own self.

So at any point you are still building a product for a given customer. So having that clarity of as to who is your customer, if you are an infrastructure product manager, your customers are all the data scientists, all the software engineers, all the teams who are dependent on you to build their models on your infrastructure. So similarly, if you are a product manager who's working specifically on building a particular model, then you need to collaborate very closely with an infrastructure PM. You need to closely collaborate with an application PM if the model is eventually integrated into an application and you collaborate with all the other stakeholders.

So the point here is that, at any given point, you have to be cognizant of whom are you building this for? Why, and what... That is the primary customer. And the secondary is like who else, if I build this what are the impact it will have on my secondary customers or other downstream or interacting applications. So making sure and being cognizant of that will help you build a comprehensive product and make sure that you enrich the product and keep everyone in the loop.

0:30:57 Scott Hirleman
Yeah. I think that's... A couple of points in there that are really important are, at least this is what I'm seeing and that's driving success in data mesh. So there's this concept in data mesh of creating data for use cases that you don't know exist just yet. And so some of that is just like, I don't have anybody asking for this specific column, but I am going to put that in and that's not that complex, but trying to model for somebody in the future, but exactly as you said, I think what you were talking about of, you have an initial customer, but you can have multiple customers of the same model and maybe there might even be some small tweaks between the two or you have that type of thing but when you first are building something, you must have someone specific in mind because you have to know what are the challenges because we're seeing this problem where people are telling lines of business, the domains, share your data.

And then they share the data and it's not used nearly as much as they expect and it's that data field of dreams of, if you build it value will come. If you build it users will come and we're seeing that there's a much bigger gap between what users will come and use, versus just the data exists and I don't know if that's because we don't know how to model that data well, if it's because consumers don't know how to come and actually do that or if it's just no matter what, data just needs to be sculpted far more than people think to actually make it usable.

So if they're expecting to just be able to consume as is, you're never going to be able to produce that in such a way. I don't know the answer, but it's something I just keep seeing and I keep poking at people to try and find an answer and honestly, nobody's got a great answer because the answer that I keep getting is, in the real world, it's not happening so nobody's got a great answer because nobody's figured it out yet.

0:33:13 Amritha Arun Babu Mysore
I kept thinking the same. I don't have an exact answer for you because I'm myself figuring out, but I do have a couple of hypotheses that I've thought for myself. One is, in the business world, like you absolutely said, I couldn't agree more is, the data is not in a readily usable format if it's like let's say you have unstructured a lot of emails that you need to understand or translate or extract data from. Not having easily usable data is a very critical factor because then it slows down your experimentation timeline.

So that is where a lot of people get stuck into that experimentation cycle and they never move to saying that, oh, I finished experimenting or it's sort of like delayed cognitive experience, right? Like saying that I can just jump in and do something, that doesn't exist, so that's one. The second is, I know as a business we are always focused on high impact, high value initiators. So the tail end always gets the last priority. There could be scenarios or there are a lot of value even with the tail end of the products are the problems. So there are so many tail end of the problems that most of them are not getting solved or most of them are not getting enough value. So with that, you might see a lot of data unused for the tail end, that's what I think.

0:34:46 Scott Hirleman
It's that return on investment versus return and the investment isn't worth the return, the juice isn't worth the squeeze because it's just, yeah, now I'm seeing the same thing. So I wanted to transition this into what we had talked about in the pre-call of, when you, as a product manager, either upstream or downstream, something has deteriorated to a point that it's time to have a conversation, how do you think about that? When to think about... Because okay, our conversion rate from this model was 7.5% well it's now 7.4%. Okay, that's not that big of a deal. Next month okay, it's 7.3%. Okay, three months later, it's 6.5%.

When do you think about having that conversation because it's a slippery slope, it's just like a relationship. Things can deteriorate in a relationship, but that it's over time and it's not as if there's usually... I mean sometimes there is a sudden breaking point, but it's more as the straw that broke the camel's back rather than it's the first straw is a steel beam.

0:36:13 Amritha Arun Babu Mysore
So I remember at the starting of the podcast as well, you had another question of something similar to this where, how would we have this conversation whether in terms of like as of a metric, if it's giving you enough insights or not or you need to move from one metric to another. Or similarly now, to your current question of, when do you know that this particular model needs some attention either attention in the

form of retraining or attention in the form of this is not working for us, we need to shut this down.

So these conversations or any such conversations whether it's about alignment where alignment comes into picture with a broad set of stakeholders is quite tricky and as a product manager, you should have this as an art and as well as a science to do this. An art form comes in the form of building the trust with the stakeholders that you work with and the also be intentional about what you're doing and consciously trying to address your own biases. When you bring that sort of an intention to the table you build the trust and take people along the journey so that's the art form of it.

The science form of it is to think, what was the problem we started solving with this particular model or with this particular application that we built and the metrics we started analyzing and work backwards from there to say that, to make sure that this is the problem that we are solving. Again, going back to, let's say, a fraud detection model, to say that, okay, it needs to detect fraud. Let's say the precision of the model is expected and when you release the model, the precision and the recall metrics of the model was about 80% and above, which tells, okay, if you are decent enough, but it could be better. But then you said, as you train with more data, it's only just started deteriorating. Then you really need to go back and look into, because it beats your threshold, because 80% was your threshold and you released it. It's come down to, let's say, 70% or 60%. Then it's not performing as you expect. It's not detecting the number of frauds that you expected it to detect.

Then there are a couple of things that you can do at this point, is look into the data to say that, what is the incoming data that the model is getting exposed to? Is there a drift in the model behavior itself? Is the change in the model data that it is getting to, or is the model overfitted based on the data that it has learned over time? And what are we unable to scale any other infrastructure issues, or do we need to feature engineer? There are some features that we are missing. So there are a lot of ways that you can think of, retraining is just one of them to think, like, what are the things that we can do? Why is it failing?

So identifying that root cause is a very critical factor, because that identification of the root cause will help you determine how much cost you will incur to go fix this model. Is it worth fixing this model and perform those particular trade offs and determine, are you going to go through the retrain, go through the re-updating this particular model, or are you going to just shut this down and go back to building something? So those are some of the decisions that you would have to go through as a product manager, using data and using your relationship and collaboration with your other stakeholders.

0:40:08 Scott Hirleman
And how do you think about when you know that you have to go and have that conversation? Like again, you said 80% is your threshold, okay, this thing went from 95% to now it's at 82% and it's deteriorating 2%, 3% a week. There's something going wrong. Do you head that off at the past? You wait until it has violated, or how do you think about... And I know it's very difficult from a hypothetical standpoint, but one of the things that I see a lot, the reason why I'm poking at this so much is, we don't know when communication should happen.

Say you've got a data contractor, you've got monitoring, or you've got whatever. When it violates X, why alert should happen. A lot of people are struggling with alerting and observability and stuff around data, but when it's conceptually not doing what it should be or something is seeming strange. How do you think about when should I actually go and have a heart to heart versus as well, like, hey, I'm seeing something, I'm saying something. It hasn't broken yet. It isn't wrong. You're still delivering me what you promised you'd deliver me. But there's something weird going on.

How do you think about when to know that? Because otherwise you could be having these conversations on a daily basis. Hey, I know our SLA is 80% for the week, but today it was 73%. Therefore, we need to have a conversation. Okay, the next day it's now back up to 92%. The next day it's at 79%. Okay, it violated this day even though your SLA is on a weekly basis, how do you think about setting that? Is it gut? Is it that there isn't necessarily a thing? Is it when it starts to feel like it's headed in the wrong direction?

0:42:19 Amritha Arun Babu Mysore
I think it's more data oriented and some sort of gut. But the gut is a small percentage more around working backwards from the data itself. See, you gave a good example of SLA, even in terms of SLA, yes, one day your SLA came down to 73%. But there is another day where or consistently, your SLA has been above the threshold, then that the 73% is an outlier. So at some point, you are going to... If the outlier repeats itself in a pattern over, let's say, X months, and you are curious, you want to make sure you have a ticket open or in a backlog to say that, "Let's look into this. What's happening? Is there something that we need to uncover because this seems to be a repeating pattern."

Similarly, in the model too, if your threshold for fraud detection or your precision came down from 80% to, let's say, 70%, it is an alarm bell for immediate reaction. But at the same time, if it goes back to 80%, then I would be curious to know, what was the pattern? Why did it drive down? If there's an opportunity that could be a feature that we builded, it could be an opportunity to enhance the model itself. So I don't want to lose those opportunities.

So the way I do this is typically have like a weekly review of our metrics to understand what was our down point. Why did we have that? At least have a grasp around what could be the potential root causes. So we are not like blindsided by something that we are totally unaware for us.

0:44:21 Scott Hirleman
It sounds like if I'm summing up somewhat of what you're saying is, have regular communication established anyway, and then you have crisis communication when it's starting to feel like... But you should both... It's kind of like with a good relationship, if you're headed towards a breakup or anything like that, it shouldn't necessarily be a surprise to both parties in most cases. Sometimes people are oblivious, but it's that you're in the constant enough communication because your value is dependent on that other person. Your business delivery of value. And same thing as if you're the producer. You should be wanting to talk with your consumer and say, is this still meeting your needs? Is there something that we need to look into? Or is there additional value delivery opportunity where I could enhance what I'm delivering to you and we both benefit from that value. I know there's needs for incentivization, but is that a good way of summing up a lot of what you're saying there?

0:45:28 Amritha Arun Babu Mysore
Yeah, very nicely, you summed it up because having regular communication channels is very critical, then those regular communication channels can be used for initiating new products, initiating new changes or keeping everyone in the loop and driving the transparency. And that is very key to building that trust and building that transparency, especially in a data product management, because data changes has to be communicated for wide audience. And the more easily it's accessible through regular channels, the higher the trust and lesser the impact of any changes.

0:46:12 Scott Hirleman
Well, and I think from a producer standpoint it's also you're higher in their top... You're top of mind, right? Maybe not absolute top, but you're higher in their mind. I don't know that anybody ever says that. So I think this transitions well into as well, how do you think about identifying problem spaces to build data products or models? Like are you waiting for everybody to come to you or are you going out there and finding where could I be adding value?

We're going to have a panel coming up on data product discovery, although by the time this airs, I don't know if that panel will already be out or not, but I haven't recorded the panel yet. But the idea of going out there and finding users for your data hasn't really been a thing, especially when we've had kind of data teams as responding to tickets. They get requests and you fulfill them versus, you think about product organizations and you think about how that actually works. You should be going out there and finding how you can deliver value.

So how do you actually go out there and identify those? Is it just going in and talking to the people and they'll tell you, or is it you have to do some ideation and ask people, would this be of value? Or how have you found what works well, and maybe if you've got any examples of what maybe doesn't work so well. I think that ideation... Well, you talked about it earlier as well, just creating something and thinking that other people are going to use it when nobody's told you that they want that. Maybe that's one of the anti patterns. But I'd love to hear kind of what you've learned there.

0:48:02 Amritha Arun Babu Mysore
Over the years, a couple of things have worked for me. Whenever I'm into a new domain or a new ecosystem, I try to understand what are the fundamentals of that ecosystem or what are the components of that ecosystem. Based on that, I determine, okay, what could be the potential products that could come in here? Or what are the potential workflows or the user journeys? Who are the players? Who are the personas here? Based on that, I have interacted with the stakeholders who are currently doing that particular job. What I mean by that is, let's take an example of, again, going back to finance, like accounts payable or accounts receivable sort of workflows, right?

So there, when I was a product manager in that particular domain, I understood that what are the invoices, what are the parameters of the invoice that comes in here? And this has to be approved or reviewed and compared against our contracts. And then based on that, you make a payment. Pretty straightforward workflow. But what are the challenges? What are the issues that... Those are some of the things I can only speculate. And the speculation needs to be validated, needs to be supported with data. That comes through only when you talk with your stakeholders, when you talk with the folks who are doing the job today, to understand where do you face the challenge? What are the variations in this invoice that you see? What are the kinds of things? Because then what happens is you enrich your problem.

You also enrich your product because this product will then address a wide variety of challenges. So it only becomes a better product and your ideation... Throughout this ideation and discovery process, you learn more about the space organically, including the challenges that you yourself are not aware of this particular space.

0:50:13 Scott Hirleman

Yeah. One comment there is the product managers that don't talk to their users. I don't understand how they can call themselves product managers, but I liked what you were saying there as well, of you come up with some hypotheses, right? And you go out there and you just say, I'm going to talk to these people. So you're not only waiting for them to come in to you, you're going out there and saying, is this a problem? But my question there is, historically, when we've asked the question of, would this be useful for you when it comes to data? Everyone goes, yeah, because more data is always going to be slightly more useful. Versus is the juice worth the squeeze? Is this benefit worth the cost?

How do you think about having that conversation and having a real assessment of, should we actually do this? Yes, it might deliver value, but is the return on investment worth it? Not is there simply a return? I know it's a very difficult question.

0:51:17 Amritha Arun Babu Mysore
It's a fantastic question. You put me into thinking mode. So, yes, having the return is great because the satisfaction of building a product and being a good product manager comes in when folks use your product as an option for your particular product. But I think well intended teams like whether it's your stakeholder, everyone is well intended because everyone wants good data. They are trying to figure it out. They might think that, yes, this data will be useful, but it might not be useful. It is part of the journey, that will happen. But what are the potential ways that we can reduce the occurrences are, thinking out loud as to, or mapping out the scenarios as to where would you use this? What are the different flavors of this data that will be useful in a different user journey? What are the problems where this data will be handy for you? And having this data easily will help you navigate certain scenarios.

Those are some of the very critical understanding or scenario play that brainstorm sessions that you need to have with your data scientists or with your ML engineers so that you take them along the journey, you go through along the journey and you educate each other to reduce the number of occurrences of building something and that's not useful and that's during ideation.

Other part is, as you're building, it's very key to get validation because it could be that you thought of X data attributes in Y format, but might be the data scientist was thinking about it something different. So being able to validate every now and then to say that, hey, is this what you figured for this particular scenario, is very critical. This just so goes along to say, you have to collaborate at work as a team rather than just product manager running the show.

0:53:31 Scott Hirleman
Yeah, it's funny because I started to think a little bit about, I was listening to a

podcast and somebody was talking about when they were a kid, they asked their parents for a skateboard and then they just bugged them and bugged them until their parents got them a skateboard and they played with it for about 20 minutes and they were like, oh, I can't do the tricks from Tony Hawk on this. I can't just automatically do that. I wish we had more of that in our life of like, okay, you're telling me you're going to use it, but how do we actually find... It doesn't equate, but I think a lot of what you're saying there is going, okay, I know this could be of use. Show me how it's going to be of use. Tell me what this is going to be.

And, oh, Alla Hale, in her episode way back when, I think episode 122 kept saying, okay, and what would having this unlock for you, right? If it's like, this would be useful for us. Okay, how? What would this actually unlock? And they don't have a great answer. Then it's like, okay, I'm not seeing the return there. So I want to give you a space to react to that.

0:54:46 Amritha Arun Babu Mysore
Yeah, one thing I like to do, and this I've learned overtime, is always asking, so what? If I give you this, so what? What will you do with this? That's like you mentioned. The other piece the way I would like to do it is to say that, what if we don't do this? What is the impact of not doing this? Marrying both together actually helps you bring out a product that truly fits the needs or a data attribute that truly is needed. That's one of the keys or frameworks that I have used.

0:55:26 Scott Hirleman
Well, I think it creates the space where you're going to create something that's slimmed down, where you don't have a lot of cruft, you don't have a lot of stuff that's in there that is not necessary. Because you go, well, if we don't do this, well, then we can't do this one thing. Well, you only talked about that for these five aspects of it, and you requested these other 20 aspects. Do we really need these other 20 aspects? And then they start to slim it down more and more.

0:55:52 Amritha Arun Babu Mysore
That's sort of what I've seen.

0:55:55 Scott Hirleman
Yeah, I think that's great. So I wanted to kind of, as we're heading into time, I wanted to wrap up on kind of, again, this idea of being between the upstream and the downstream. This is very relevant for data mesh as well, of there's going to be data products that are consuming from upstream, and yet there's also downstream consumers, but there's also maybe even downstream data products that then have their own consumers. How do you build things so that they can evolve? How do you build things so that you're not breaking downstream? Sometimes you have to break

downstream, right? Oh, this source no longer exists. We were getting this from an external party. This source no longer exists.

But how do you think about creating things to be able to evolve? And I know it's very, very difficult to talk about that in the abstract, but also, how do you prepare your consumers? Most people are not used to things when there's a change in data that they're consuming. It's a breaking change no matter what. So how do you get them able to be prepared for evolution?

0:57:11 Amritha Arun Babu Mysore
Changes are inevitable. There are going to be changes. It's right business practice to anticipate these and make sure when you're architecting or when you're designing your services or your data architecture and your integrations to keep in mind that there will be a scenario where things will change. The potential to break. What this mindset enables you to do is, helps you build a modular architecture, a something loosely dependent architecture. In this fashion, you are always not putting your services into a situation where they will break because of upstream changes.

Another way to do this is to have the data contracts clearly specify what is the handshake requirement. So that based off that, you are very sure that if any changes that you're doing, there are the X services that are dependent on you. These are the handshake requirements clearly specified. And I need to make sure even if I change this data schema or if I change the way this data will be cleaned, I need to make sure on the output side, these are the variables that I'm still generating so that the downstream is not impacted.

So these are some of the architectural ways that you need to think and design your system. The other communication ways are, you need to make sure that you communicate saying that, hey, here are the changes that we are doing. Here is our design for the changes. Feedbacks are welcome or make it more transparent so you drive the changes. If there are additional inputs or additional requirements that other teams can share, that will also help change and enhance the systems, those are also things that you can include. So being able to collaborate and being able to take folks along the journey as you're changing and make it more transparent will help you minimize the impact of any changes.

0:59:29 Scott Hirleman
Yeah, it's amazing how much of this comes back to communication. And you're talking about having data contracts, and so many people are using them only defensively instead of collaboratively of... I know that my contract says that I'm going to be able to do this. Well, that has changed. So let's have a conversation. And this is where I also kind of somewhat bully people. But I tell everybody that they need to

have all of your consumers register as an actual consumer. So when something is changing, you're not a silent consumer. I know that I can go and have a conversation with you and that I can have an actual conversation with you and extract what your needs are because I might be only meeting 70% of the way and I've talked to somebody and to me.

It kind of goes back to one thing that I did with work, but forever ago when I was an intern, but the consumer was untransforming data back to the raw format, and the person that was putting it into the data product had been transforming it in a certain way. And all they had to do once they actually had the conversation, which they had been doing this work for two months, and it was difficult and complex, and it hurt their time to market with actually getting the data ready and all that stuff was, oh, yeah, we can just drop the raw column in another column and boom, okay, it's very simple.

And it kind of reminded me of randomly when I was an intern and I wrote up this memo, and the person above me rewrote it, and then the person above them rewrote it, and it looked a heck of a lot like mine did. And then the person above them rewrote it, and it looked a lot like the person that was directly above me who rewrote it, and then the person above them rewrote it. And it was two words different from my original. So if it had just been... But anyway, yeah, I love that idea of just talk to each other. Just please have a conversation and have empathy for each other.

1:01:44 Amritha Arun Babu Mysore
Yeah, I think that is very key. I kept saying, for a product manager, that is very key. I think in today's very interconnected workplace and interconnected world, it is just very key to make sure you have open communication, have existing lines of communication, understand what is the intent that anyone is coming from and work with empathy.

1:02:12 Scott Hirleman
Yeah, well, I can't remember exactly what phrase you used earlier. But you were talking about like, everybody kind of has and works with good intentions. And if you assume that, every once in a while you run across people that don't, but if you assume good intent from your colleagues and stuff, then you can have conversations that drive towards, like, what are you actually trying to achieve? And I think that's a huge part of product management.

1:02:38 Amritha Arun Babu Mysore
Yeah.

1:02:38 Scott Hirleman
So we covered a whole heck of a lot of things. Is there anything we didn't cover that you want to or any way that you'd like to wrap up the episode?

1:02:47 Amritha Arun Babu Mysore
No, I think you did a fantastic job of covering it from all the aspects. Thank you so much.

1:02:54 Scott Hirleman
Yeah. Well, it was a very fun chatting with you. And I'm sure there's going to be a lot of people that would love to follow up with you. Where's the best place to do that? Anything specific you'd like people following up about?

1:03:06 Amritha Arun Babu Mysore
Any feedback on what are the other topics that they would like to learn, that would be great to know. Yeah.

1:03:10 Scott Hirleman
And LinkedIn, probably the easiest way to find you?

1:03:10 Amritha Arun Babu Mysore
Yeah, that's right. Sure. Yeah.

1:03:14 Scott Hirleman
Awesome. Well, we'll drop a link to that in the show notes so people can easily find you. But again, Amritha, thank you so much for spending the time here today. And as well, thank you everyone out there for listening.

1:03:27 Amritha Arun Babu Mysore
Thank you, Scott.

1:03:28 Scott Hirleman
I'd again like to thank my guest today, Amritha Arun Babu Mysore, who's a Manager of Technical Product Management and Machine Learning at Amazon. You can find a link to her LinkedIn in the show notes as per usual. Thank you.

Hopefully that interview episode was really useful for you. Please do consider getting in touch with guests from the show from these episodes. Most have said they'd really love people to reach out to them. And please, as well, if you've got a minute, rate and review the podcast somewhere. It really is honestly super helpful for other people looking into kind of data podcast to kind of get this in front of them. Data Mesh Radio is again, provided as a free community resource by Data Mesh Understanding.

It's produced and hosted by me, Scott Hirleman. In April of 2023, I left DataStax who are wonderful in getting the data mesh community stuff started, so give them a shout for streaming and real time AI needs. But I left to start my own industry analyst kind of information as a service firm. Our offerings are affordable, and you can do them on a one-off or a month-to- month basis. Read, kind of throw it on the credit card. Don't worry about going through purchasing and things like that. The services include lots of practitioner roundtables, one-on-one data mesh kind of planning or feedback sessions, and tailored introductions to other data mesh practitioners that are focused around your topics of interest, what are you actually running into challenges with? We also have some free programs around introductions and roundtables that people can kind of check out as well.

Check the show notes or just go to datameshunderstanding.com for more info or helpful resources. As always, if you have suggestions for guests or topics, please do get in touch as well. Have a wonderful rest of your day. Now, let's hear that funky outro music.