EPISODE 1279

[INTRODUCTION]

[00:00:00] JM: Amundsen was started at Lyft and is a leading open source data catalog with a fast growing community and a lot of integrations. Amundsen enables you to search your entire organization by text search, see automated and curated metadata, share context with coworkers, and learn from others by seeing the most common queries on a table or frequently used data. Powered by Amundsen, the company Stemma, is a fully managed data catalog that bridges the gap between data producers and data consumers. Stemma adds features to Amundsen like showing meaningful data to individual users, adding metadata to data automatically, and documenting data on the fly. Stemma integrates with all the major data sources like Snowflake, Redshift, BigQuery, and Airflow.

In this episode, we talked to Mark Grover, the founder at Stemma. Mark cocreated Amundsen and authored the book *Hadoop Application Architectures*. He was an engineer at Cloudera before joining Lyft as a product manager.

A few announcements before we get started. One, if you like Clubhouse, subscribe to the Club for Software Daily on Clubhouse. It's just Software Daily. And we'll be doing some interesting Clubhouse sessions within the next few weeks. And two, if you're looking for a job, we are hiring a variety of roles. We're looking for a social media manager. We're looking for a graphic designer. And we're looking for writers. If you are interested in contributing content to Software Engineering Daily, or even if you're a podcaster, and you're curious about how to get involved, we are looking for people with interesting backgrounds who can contribute to Software Engineering Daily. Again, mostly we're looking for social media help and design help. But if you're a writer or a podcaster, we'd also love to hear from you. You can send me an email with your resume, jeff@softwareengineeringdaily.com. That's jeff@softwareengineeringdaily.com.

[INTERVIEW]

[00:02:00] JM: Mark, welcome back to the show.

[00:02:01] MG: Thank you for having me. Good to be back.

[00:02:03] JM: Last time, we talked about Amundsen and some of the problems with data discovery and metadata at a large company like Lyft. And I'd like to go a little bit deeper and also talk about the company you’re starting around Amundsen. But first, let's just give a brief review of Amundsen and some of the data discovery problems. Can you review the problems that you built Amundsen to solve at Lyft?

[00:02:29] MG: Yeah, for sure. So what's happened over the last few years is that companies have invested a lot of time and energy innovation in capturing, processing and storing a lot of data. And as an industry, we've had Fivetran and Stitch to bring in more data. We have brought in Airflow, and Prefect, and ETL Tools, and DBT to process more data. And then there's a slew of innovation happening in the data warehousing space with snowflake and BigQuery making it really easy for companies small, to big, to be able to store all of this data. At the same time, what's happening is there's a lot of innovation happening in the consumption space. So you have these BI tools that have existed for a long time. Now SaaS products that enable users who want to use data operators to make data-driven decision day-to-day.

But what's missing is that while we're bringing in more and more data in the organization and we have provided tools for users, operators, data scientists, analysts to make data-driven decisions, no one really knows what data exists within the company and what can be trusted. And this is the problem. I saw at Lyft where everyone had access to data, but few knew what existed, what was trustworthy, and how to use it. And that problem was so severe, data scientists and data analysts spend over a third of their time finding and validating trustworthy data. And that was the problem that Amundsen was created to solve. And the more I work on – Amundsen today is the leading open source data catalog. The more I work in the open source community, it's clearer and clearer every day that this problem wasn't unique to just Lyft. It exists that all companies of all different shapes and sizes, and happy to dig into the problem and the solution as well as we talk further.

[00:04:18] JM: how widespread is this set of problems of data discovery and metadata? Does every company, once it gets to a certain scale, start to have this kind of problem?

[00:04:28] MG: Absolutely. So this problem doesn't exist in super small companies where it's all in one person's head, or it could be easily documented in one single place like a wiki page or a doc. You could document what all your data exists and what each of the fields mean. It also doesn't exist where the setup is very stable. So if you have a small company where everything is stable and your data model is not evolving, the organization's production systems aren't evolving that produce the data. But that is a very small subset of companies. Most of the companies are big enough. The place where I see is a couple 100 employees for the company, you are big enough, where there's enough data, enough people in the mix that you don't have a single person who knows everything. And the company is evolving in a place where you can't keep everything up to date with just a simple doc. So anywhere I would say from 300 employees to up, and the signals here are the number of employees and the growth, both addition and subtraction, or moving of people within the company, as well as the amount of data the organization has and the change at which this data is coming into the organization are changing and evolving.

[00:05:44] JM: So you worked on the open source project Amundsen, just to work on these data discovery and metadata issues. And you are building a company around it. Tell me about the productization of Amundsen. What's the diff between the open source project and the closed source product?

[00:06:04] MG: Yeah, for sure. So like I was saying, we started from the problem – We saw this problem at Lyft. And what was happening there is that there were these gossip protocols of slack being built and shoulder tapping where people were asking each other what data exists. I remember a data scientist trying to optimize ETAs, and they didn't know what the source of truth for ETAs was, which was one of the core metrics for Lyft. And worst of all, data got delayed, deprecated and shut off. And the analysts and data scientists were the last ones to find out.

So Amundsen, I co-created the project to help solve that problem, which is an open source data discovery and catalog solution. So the product is getting metadata from various different sources. So it hooks onto your data warehouse, your HR system, if you give it permissions, your BI tools, and it's able to bring all this information together to power a view of what's trustworthy based on when was it last updated? Who else is using it? What's built on top of it? And what conversations are happening about it? And so it's an automated way of augmenting documentation and powering that experience of what is trustworthy, then relying on a single person, a data steward, or a volunteer who's assigned to keep something up to date on an ongoing basis.

So Amundsen – I'll get to your question in a moment. Amundsen is pretty successful as it has 750 users every week at Lyft. Over 75% of data scientists, data analysts, data engineers at Lyft use it every week. Amundsen is used by more than 35 companies in the open source. So these are ING, Square, Brex, Asana, Snap, and many more. Convoy has 80% of the company using Amundsen every month.

What Stemma does is it provides a managed version of Amundsen. So think of it as a superset of what the Amundsen project provides with two specific editions. So the first one is enterprise management. It has super easy deployment with enterprise-grade security. And the second thing it provides is more intelligence. So it uses existing metadata to infer richer metadata and personalize the experience based on user's role and activity and what is happening in the organization. And those are the key ways we enable more value for organizations that choose Stemma.

[00:08:30] JM: If I want to deploy Amundsen within my company to have my data indexed, what do I need to do?

[00:08:39] MG: Yeah. So if you were to deploy the open source project, the first thing that usually you end up doing is reading the documentation we have online. You also end up joining the Slack channel. But in terms of tactical steps that you need to do, is that Amundsen has four parts to it. Three of them are services. One of them is a library. So the three services are a frontend service that is the UI that powers this product. The second service is a metadata service, which is powered by a graph database. And this metadata has all the core metadata that is used in Amundsen. So it has information about what tables exist, what columns, what are the various relationships between them? What information are people querying? Who owns a particular table or a dashboard? Contains information about dashboards and what tables are they built off of? It has nodes for people. And if I am in one team, that I'm linked to another person in my same team. And it also has information about what I as a person use every day.

So that's the graph backend. The third service is a search service, which uses Elasticsearch in the back to power a search experience. So when someone comes in and searches for an ETA, they know what's behind it, and it gets the metadata from the metadata service to surface a trustworthy notion of a PageRank style implementation of data. And last as a data builder library, which allows you the mechanism to pull this metadata. We also have beta functionality to push this metadata into the product. So this is a library that you integrate with to ingest your data from your data warehouse, your BI tool, your HR system, and so on and so forth. So these are the four pieces. And as you deploy it, the first thing you do is you'd set up a PoC, and we have very easy to set up installed scripts and Kubernetes containers that you can quickly deploy using Docker on your local machine and connect your existing data warehouses to it. Once you're done with it, you deploy it in a more scalable, distributed manner, each of these components, and stand it up.

[00:10:49] JM: How does it query against Amundsen work? Yeah,

[00:10:53] MG: Yeah. So the two main query serving aspects of Amundsen are the search engine and the graph. And to give you a little more flavor of the product, the product is more like a Google search for your data. It doesn't have any capability to query the data itself. For example, Amundsen doesn't provide you a JDBC connector that you use to query the data that's in the backend. Our sole goal is to take this haystack of data that has various degrees of trust and provide you a system of information as well as ranking on what could be trustworthy based on some automated signals. So the goal here is that you come to the product when you start. Say you're developing a new ETL job, or you're developing a new insight. You come to the product. You search for something like ETA, and this would hit the search service. You'll get information in a ranked order of what is trustworthy based on how it's being used in the organization by whom? How often is updated? Etc. You click on that information, and then you get to a table detail page.

Now this table detail page has all the information you would want to know to establish and understand. Could you trust it? How often is it getting used? And how do I use it? So it has information about a description, which can come from an existing source, like the data warehouses description, as well as from curated means. So you can edit this description. It has information that comes from Airflow integration. So it brings in metadata about how often does this table get updated? When was it last updated? We parse the query logs, so it has information about who are the people who frequently use this data. And then we obviously have information about what are the columns and the types, their descriptions. And we generate column stats so you could see what the standard deviation or the number of distinct values in a particular column are.

And lastly, we have some information about lineage, which is, again, parsed from the query logs. We can see a preview of the data if you have access to the data. And if all of this looks good, then you go to the explore phase, which actually takes you out of Amundsen. That's the end of the discovery journey and onto the exploration journey. So from that point, you leave Amundsen and you go to the next tool in the process, which is usually a BI tool. So that could be a Mode or a Looker, or a Tableau, or Apache Superset, things of that nature. And that's how that interaction ends up working.

[00:13:19] JM: Can you extend that to like if I wanted to get some data that's indexed in Amundsen and run like a spark job against it, what am I doing?

[00:13:31] MG: Yeah. We've had requests to export the metadata that's in Amundsen into a relational form. So maybe you take all this metadata about quarry usage, user’s history, and who's querying it. Export that to a data warehouse, and then are able to run other SQL queries on it. So maybe you build the graph off what are the most important data sets being queried in the organization? What data sets that I own have been delayed by more than an hour in the last month, things of that nature.

And for that, what Amundsen does today is it actually has the backend graph, which the default for that is Neo4j. We also support Apache Atlas, as well as Amazon Neptune. And there's work happening to support an RDS for it. So that metadata, instead of being stored in a graph, while the graph models will still apply, it would be stored in a relational database. So that will allow both Amundsen to query natively and provide all the experience that it provides to engineers as well as data scientists and data analysts, but also for further analysis to be done on the top of this metadata directly on the backend store. So that way, you don't have to export this out, and you're able to do that same analysis you would do on the data warehouse using an analytical tool on the backend database at the Amundsen product.

[00:14:52] JM: Can you give a little bit more context into why Amundson fits into the current data engineering world? So among the popular tools of – You touched on this a little bit. Among the popular tools of the data warehouse and the DBT, and Census, and Fivetran, all these different things that stitch data together, what is Amundsen's place? Where does it fit into the workflow?

[00:15:18] MG: Yeah, that's an excellent question. So what's happening with data engineers is that they're constantly bogged down with keeping everyone informed about upcoming changes and current status off a table. So at Lyft, and many companies in the open source, a data engineering will build a pipeline, say, using DBT, and orchestrate that through Airflow. And then as the company evolves, data evolves, sometimes this data is late. And they have to notify all their stakeholders, particularly data scientists, and analysts usually, that this thing is going to be like, right?

But the reason these things happen is because they, and people upstream of them, which may be a product engineer, or a software engineer, don't quite know how their data is being used. Who is using them? So the most common way of notifying of existing changes is to spray and pray, right? And data engineers end up sending out blanket emails that no one reads. And when you make that change to the data pipeline, it breaks, or a downstream pipeline was delayed, and it surprises people.

So where this fits is in two places. The first and the most impactful place is when you have an existing pipeline that you're evolving. You get to see who is using the data that you are producing, and in what ways. So this includes both ETL pipelines that have been built on the data that you're producing. But it also includes dashboards that exists on top of the data you're producing, as well as ad hoc queries that are being run on your data. So when you are changing something that you own, you have a way to know who is using it in what ways and a way to notify them so that there are no surprises. That's the first and most impactful way this fits in their world. The second is when you're developing a new pipeline, you may be working with a lot of event data, say, that's coming from upstream eventing systems like Segment or Heap or something like that. You may have to use data that's coming from third parties, so your CRM system or things of that nature.

And lastly, you may be dealing with a lot of CDC data that's coming from production data databases and getting replicated into the data warehouse. So the second place where this fits in as a data engineer themselves has a discovery problem of understanding what data is available in the data warehouse? And can I trust this data? And how do I use it? So it's helping them discover the data, understand it, and then start to build their pipeline in whatever ETL tool they're using.

[00:17:49] JM: While we're on the subject, how has data engineering changed since you left Lyft, or since you were heavily involved with the data engineering at Lyft? It feels like the data engineering world is accelerating.

[00:18:03] MG: Yeah, absolutely. I would say it's changed in two ways. One is that there are tools for data engineers that are top of mind that weren't in the past. And these are products to help build ETL pipelines very quickly and easily. And products like DBT go in this space. There's easier integration to take these existing, the pipelines that were built in prototype, and orchestrate them. So schedule them in a production-ready manner through things like Airflow. That wasn't as common of a pattern when we last spoke almost two years ago. But that integration has become a lot more seamless. So that's one area of tools for data engineers to become more productive.

Another area where I'm seeing a lot of investment, and which is where Stemma and the Amundsen project both fall in, is data operations. Once you have made a pipeline, what do you do when you have to evolve this pipeline? So this involves change management. This involves making sure that data is getting delivered on time reliably. This involves data quality. That's the second category. And the third one, which is slightly tangential, but still related, is a little bit more of the democratization of writing of the ETL. And DBT, in my opinion, is a pioneer of this, where analysts and data scientists are able to write self-service ETL's that you don't have to rely on data engineers all the time. So those are the three big areas of investment that I've seen over the last two years.

[00:19:43] JM: What do you see is the most outstanding problems in the world of data engineering?

[00:19:47] MG: In my mind, the biggest problems in data engineering that remain unsolved are still around maintenance and upkeep of existing data pipelines. And these are the symptoms. So the symptoms involved keeping stakeholders posted when a pipeline is running late. Telling them when it's supposed to land. And that's a very highly manual process. And you never keep anyone fully up to date. You always miss people and you spam people, things of that nature.

It also involves knowing what changes are coming that are going to impact me as a data engineer and the pipeline I operate, as well as what changes am I making and who do I need to convey them so it doesn't surprise them, and then making sure that my data that I do own is getting delivered reliably and in a timely manner. So those are the areas of places where gaps still remain. And there are a bunch of different products in the space that helped with them. Mine is one of them. And I'm happy to chat more about what's involved here for other products as well.

[00:20:46] JM: I would like to go inside the perspective of starting a company around Amundsen, and like I know there some other like data catalog, data discovery metadata systems. And I guess I'd love to hear about your competitive stance. How do you compare to the other products that are out there and what's your strategy for competing with them?

[00:21:15] MG: Absolutely. So the problem that I saw at Lyft that I created Amundsen for that exists in a bunch of different companies – The status quo of solving that problem when I evaluated existing tools at Lyft was curation. So you have either a person whose full time job it is to make sure this metadata is up to date. And this metadata here involves descriptions, the cadence of delivery. What data quality rules apply? Things of that nature. Or you assign a volunteer responsibility to someone who already has a full time job and off doing something else to keep this metadata up to date. And the biggest problem with that is that this information gets out of date really quickly. And people already have other jobs, if they're volunteers. And if they have this full time job, they almost always don't have the full context. So they then have to go to other domain experts to fill this information in. And so that was the key thing that was missing an existing tool, in my opinion, continues to miss today, is that there's a heavy curation angle, and full reliance and curation, which doesn't work out in organizations where change is happening really quickly and you are democratizing access to data so more people can derive data driven decisions themselves.

Coming through today, the key place where Stemma differentiates itself is that we uniquely augment your data with automated documentation. So you don't have to document every single bit of information and certify every single data set. We'll support curation. And this is very important when you're changing user behavior. So if you're migrating from one particular data warehouse to another data warehouse, curation is the way you move people off. But at the same time, for the large majority of your data warehouse, you don't have to go document every single field. So that's the one thing. The second thing I find is that the data ecosystem is always evolving. And so the integrations and the metadata you obtained, and years ago, or even five years ago, are the state of the art integrations today.

And so today, you want metadata from your Snowflake, your DBT, your Airflow, and bring that in to stitch a model of what is trustworthy in the organization. And so it's very important for any data catalog, Stemma included, is to have the most integrations and keep them up to date. And I think that's one place where having a vibrant community, seamless product as backed by Amundsen, and Amundsen has the largest open source data catalog community. It's over 1600 people. And we have the most integration. So keeping them up to date so they work for the organization's today and evolve as they invest in their future is very important. And those are the two ways that we differentiate.

[00:24:11] JM: Have you managed to get some early like beta customers?

[00:24:15] MG: Yeah, at this point, I'm not in a place to share the name of the customers. We do have early customers. And they come from three different categories. About a third of them are Amundsen aware. So they know the project. They want it, but may not have the investment they need in order to deploy this themselves. They use us to provide a managed offering off Amundsen with the enterprise management as well as richer metadata through intelligence.

The second category are people who aren't Amundsen aware, but, one, a data catalog. And Amundsen is the leading open source data catalog, and end up choosing us to deploy that in the organizations. And the third category is organizations that already use Amundsen, are successful with it, and they are moving to Stemma in order to save the enterprise management overhead. Also deliver some of the richer intelligence features that we are working on that will really bring value to them.

[00:25:12] JM: In your conversations with these early customers, was there any feedback they gave you that changed your mind about your perspective on the product and the product direction?

[00:25:21] MG: Yeah, absolutely. One thing that's clear to me when working on Amundsen and on Stemma is that users and consumers now demand a consumer-like experience for their enterprise products. And Slack really started this trend. But it's an area I and Stemma continues to invest both in the open source project as well as in the company to make that experience a really seamless experience.

Especially for a product like us, we have a lot of metadata we show. We show the descriptions, the owners, the frequent users. When was it last updated? How often does it get updated? The columns, the fields, the stats, the preview, lineage, all this stuff. And it's very important that we understand these use cases, the user journeys. Invest our time and making that experience a very clean, curated experience in the product so users are not overwhelmed with the information. And the second thing that I've learned is that it's very important to create a managed offering that really lowers the time for dopamine. And taking that data warehouse and connecting it to your Stemma install should be just really, really quick and easy. And that's an area we continue to invest our time in.

[00:26:34] JM: So what's the infrastructure that you've built to have a hosted solution? Like when I spin up an instance of Amundsen on like the hosted version, on your product Stemma, are you build completely on AWS? Like give me a description of the underlying infrastructure that you're using.

[00:26:55] MG: Yeah. So if you choose Stemma, our goal is that you do as little work as needed for us to provide that same experience that has worked for Amundsen to you. And so what we do is a managed offering in which we deploy all the parts of Amundsen that you would have deployed and managed them. And so there we provide Kubernetes-based deploy for you where we run each of the three services. And then we write the integrations. Most of the companies we're working with have integrations that work out of the box, even in Amundsen. But in some cases, we have to write specific integrations in an organization that has their own time series database that's specific to them, or custom metadata that exists in the GitHub repository or a JSON file. So we integrate them as well. But all of our infrastructure is based on AWS right now. And we support deploying through Kubernetes or cloud formation templates, depending on the customer.

[00:27:59] JM: Have there been any particularly difficult engineering problems that you've encountered while taking Amundsen to becoming a cloud product?

[00:28:10] MG: So Amundsen was built for the cloud at Lyft. That's how the deployment was. There were a lot of learnings there and making that a cloud-first cloud native product to begin with, and I'm happy to talk about them. The place where I see us as a community, the Amundsen community, and some committers who work at Sstemma spending our time on is currently we have metadata around application context. So this is like what data exists? What are the fields? And what do they mean? We have added information about behavior, which is who's using this data? Who created it? What jobs? Or what people query it? And the last one is change, which is how is data evolving over time? So maybe you can look back and time travel back a month to see what was the lineage at that time? And what's the lineage right now? And what is the diff between those two lineages.

And the change part actually remains uncaptured. And it's something that we don't do today. And it's something that we would like to do in the future. And that's the place where that is still very hard just because of the veracity and the detail that we have to capture and track over time. And that remains to be a constant endeavor to bringing that changed information in the open source project and Stemma.

In terms of what we learned at Lyft when we build Amundsen in the cloud, one thing that we did right in my opinion was to make the frontend, all the services, configurable. And so one problem that open source projects have very commonly is that you end up maintaining a fork in the organization. And before Lyft, I worked as an engineer, a Spark developer at Cloudera. And I saw this very commonly in customers' environments as well, including Cloudera's, where you would have Hadoop and then you end up forking your own thing. And the engineer team was amazing at Lyft. And they've spent a lot of time and energy and making sure that forking was kept to a minimum, because upgrading these forks becomes a huge hassle. And so the repo structure of Amundsen at Lyft that remains consistent for the open source companies is that you would get a frontend configuration, which is placed in a separate repo that overlays on top of the Amundsen frontend repo.

And so a lot of thought was placed in the mind so that users don't have to maintain their own forks and manage them. And it was simply a configuration that got overlaid. And we continue to make investments in making that deployment really easy. So two of the more recent changes that we've done to the open source project is one that we have moved all these separate repos. So Amundsen used to be four different repos, one for each of the services and one for data builder. And then we had one umbrella repo that sub-module to all this stuff. We have moved all that to one repo. So it makes it really easy. But we were developing or deploying this as a user to wrap your head around and maintain and manage this thing. So that's one recent change.

The second change we have done is we have published a deployment guide. And there an Amundsen custom repo which makes that act of overlaying also very easy. So all you have to do is we provide some templates for configuration files .You fill that information in. And if you use this custom repo along with the new mono repo of Amundsen, it makes it really easy for you to overlay these config changes and deploy it in your own local environment the way you want it to.

[00:31:36] JM: Are you already starting to think about adjacencies to expand Amundsen into? Or do you think just the data discovery and metadata challenges and kind of the productization of open source challenges that you have ahead of you are sort of enough to keep you busy for a long time?

[00:31:53] MG: Yeah. So the space I am in is that of helping organizations and users in the organization just trust their data. And there are three categories of problems that need to be solved here. The first one is data governance. Am I using the right data for the right purpose as an analyst or data scientist, and I'm onboarding? Am I getting ramped up quickly based on what my team uses and what my team owns? That's one category. The second category is data quality, which involves putting certain expectations on your data and making sure they're getting met on an ongoing basis. And sometimes maybe these expectations are being automatically suggested to you. And the last one is data operations, where you are investing time in products. Making sure that data is getting delivered reliably on time every day. And it's very easy to keep doing that an ongoing basis by staying on top of changes that are happening.

We are in the data governance space. So the first pillar. And we want to do a good job of that space before we venture on anywhere. There is enough here for us to spend and invest our time in making organizations successful. It is also the place where most organizations have the biggest gap. So my goal, and Amundsen project, as well Stemma’s goal, is to make this problem not a problem for organizations. And that means we are integrating with other products like data quality tools so that Amundsen and Stemma become the single pane of glass for you to see, “Okay, what does this table mean?” What were the data quality checks run on it? And seeing whether I can trust him based on all that information? And that's our focus.

[00:33:42] JM: In terms of servicing more legacy companies, do they have a heterogeneity of data storage mechanisms that makes it difficult to integrate with their infrastructure to the extent that you would like to?

[00:33:57] MG: That's a good question. Yes, the older the organization, the more disparate and diverse their data ecosystem. And that could come from storage systems, BI tools, ETL tools, and so on and so forth. I have not found that to be a problem. In fact, that's the place where the product really shines. Because where you have a more fragmented view of the world, the need for you to understand what exists and what is trustworthy outside of my blinders is even more important. So I think the larger the organization and the more fragmented it is, the more value there is to a product like a data catalog that can help your users automatically uncover what is out there and what could be trustworthy, and who to talk to and who are the main users that I should be asking certain questions to.

[00:34:45] JM: Do you have anything else you can share about the ways that a company changes after it adopts Amundsen or adopts Stemma, the company that you're building?

[00:34:55] MG: Yeah, absolutely. So at Lyft and in the open source companies, over a third of data, analysts, data scientists time, is spent on finding and validating trustworthy data. After they have deployed a product like Amundson, they see an improvement in productivity of data analysts and data scientists by 20% to 25%. And that's because you provided them the context in order to find trustworthy data and see where it's coming from, who it's used by. That number is very high. And it leads to a very strong adoption of a product like this. So at Lyft, 75% of data scientists, analysts and data engineers use it every week. And we are seeing over 700 users at ING. 80% of the entirety of Convoy uses this product. So it's very sticky adoption, because these users consider a catalog product like Amundsen to be core to their workflow. And they use it both when they're creating new work both ETL pipelines as well as analysis, but also when their existing work needs to be evolved, or made sure that it's being communicated as to what changes are happening. So it becomes a very, very core part of their workflow.

[00:36:05] JM: All right. Well, as we begin to wind down any predictions about the world of data engineering, data infrastructure that you've learned from your work?

[00:36:16] MG: I'd say a few things. First one is that every data user will become a data engineer. They take on responsibilities to understand and make decisions based on data. But often they have to modify the data for their own needs. And I predict that more democratized tools for creating pipelines for managing, maintaining pipelines, for change managing those pipelines, for a more diverse set of personas with diverse skills, is something that's going to happen.

The second is that there's going to be products, experiences, built that help these – call them whatever you may. These new category of data engineers who are broader than the current skill set of data engineers that provide them additional tools of maintaining onboarding users to their so called pipelines. Seeing how they're being used, evolving them that are going to make that process really smooth. So one is a philosophical change, and the other one would require a bunch of tooling and product and experiential changes in order for these broader, more diverse skill set people to be able to write data engineering style pipelines.

[00:37:46] JM: So just to wrap up, if we talk about a world of improved data infrastructure, or a world where companies are using Stemma, or they're using other modern tools, how does the life of the data analyst improve?

[00:38:00] MG: Absolutely. So the problem with data analysts is that they're under extreme pressure to deliver reports and models. And they inadvertently end up using the wrong source or the wrong logic to do this work, because they don't have the entire context about what's out there and what's trustworthy. But what's worse is that data keeps changing underneath them. So something that was trustworthy yesterday may not be today, and data gets delayed, deprecated, or completely shut off. And then analysts and data scientists are the last ones to find out. So the place where a product like Amundsen helps them is that they always know the up to date status of a data through the augmented automated documentation. So you know how it's being used. When was it last updated? And you don't have to rely and ping a data engineer to be like, “Hey, is this getting delivered on time? What does this column mean?” And that is the change that this product brings in their day to day workflow. This makes the analysts and data scientists over 20% more productive on an ongoing basis.

[00:39:02] JM: Okay. So zooming out, let's just summarize what we've been talking about. So can you just review the problems of data discovery and what you've been working on and the progress you've made so far?

[00:39:17] MG: Absolutely. Yeah. So the key problem is companies are collecting more data than ever before, processing more data than ever before. But users, data engineers, data analysts, data scientists, very few of them know what exists, what's trustworthy and how to use it. This problem is severe. It existed at Lyft, where I co-created the Amundsen project. And it exists in all companies of any size once you reach a couple 100 employees. Amundsen is the leading data catalog that was started at Lyft. It's used by more than 35 companies, Instacart, Square, ING, Brex and many more, and users of Amundsen see 20% to 25% improvement in analyst and data science productive after deploying it. And I started Stemma. Very recently we came out of stealth earlier this month. And Stemma provides managed Amundsen, which uniquely augments data with automated documentation and provides enterprise-grade security that we managed so you can get the benefits of Amundsen and more from a managed offering. So if you're interested, check out Amundsen, amundsen.io, as well as Stemma at stemma.ai.

[00:40:30] JM: Mark, thanks for coming back on the show.

[00:40:32] MG: Thank you for having me, Jeff.

[END]