Tegus Transcript

Dataminr - Head of Data Engineering at a Large, Publicly-Traded Financial Services Company

Interview conducted on April 19, 2022

VP/Head of Enterprise Analytics Lab, data science, and engineering at a Large, Publicly-Traded Financial Services Company. (Wells Fargo) Expert can speak to Spark.

VP at a Large, Publicly-Traded Financial Services Company, a customer of Ontic and Dataminr. The expert is responsible for leading corporate vision, strategy, and execution of Data Science, Big Data Engineering, AI, ML, NLP, and analytics capabilities in Chief Data Office and Chief Analytics Office, while supporting over 80 global lines of businesses. The expert also leads data and insight-driven cultural transformation by enabling AI, ML, and NLP innovations, operationalized 10s of AI models, and trained 100 data scientists. The expert architected and built on Hadoop Big Data Lake, as the AI, ML, and NLP innovation platform with data science and data engineering tools (Spark, Python, H2O, Jupyter, Hive). In the role, the expert has reduced data processing time from days to hours, enabling near-real-time decision making, and has adopted Analytic Target Operating Model - deploy models into prod in weeks vs. months. The expert works with tech teams in defining, architecting, and agile implementing the enterprise Big Data Lake with data governance, data security, and privacy – significantly speeding up innovation. The expert leads adoption of AWS and GCP cloud analytics, and architected self-service analytics using Business Led Analytics Development Environment – build models in weeks vs. months. The expert also uses AI to accurately detect Unfair, Deceptive or Abusive Acts or Practices, financial crimes, Suspicious Activity Report, and Unusual Activity Report.

Currently, the expert uses Ontic and Dataminr. The expert can speak in-depth about the protective intelligence platform space.

The expert transforms business using technology – from traditional analytics to artificial intelligence and builds new products for fraud, risk, compliance, customer intimacy, and operations – saving $10MMs with data governance, security, and data discovery. The expert specializes in driving business transformation across multiple domains including financial services, high tech, and fin-tech. The expert is an industry speaker and has 2 patent-pending inventions.

Q: Do you use Spark?
A: Yes, using for 7 years. I evaluate and make the purchase decision, while my groups use related products and services including Spark (SparkSQL, Spark MLlib, Spark GraphX, etc.) used in model development life cycle (Artificial Intelligence and Machine Learning data engineering and model development) using PySpark, and deployment in production (via CI/CD – continuous integration and continuous deployment).

Q: Is reducing cost a top priority for your company/team?
A: Yes.

Tegus Client

Hello. Thanks for taking the time. I'm researching how to deploy a big data or machine learning job, for example, like a large Apache Spark job to the cloud. I'm researching a company that helps people launch their jobs to the cloud and then profile your job, and then tell you exactly how you should provision Amazon or Google or Azure.

Head of Data Engineering at a Large, Publicly-Traded Financial Services Company

Yes. So how does that make it cheaper, by optimizing the resources allocated in the cloud?

Tegus Client

Yes. Yes. Exactly. They basically optimize the resource and make sure you're fully utilized. A lot of people misconfigure or don't really know how to configure the machines, or some machines are better for your jobs than others, right?

There's a compute-optimized, memory-optimized storage, network-optimized, et cetera. Which one is best for your specific workload? It depends and changes. So it’s basically like a matchmaking service, right? You have your workload, the cloud has all of these compute options and then it tells you which one is the best, basically.

Head of Data Engineering at a Large, Publicly-Traded Financial Services Company

So let's say, I'm in GCP, I am doing some deep learning model. So you may say, you know what, forget about the CPU, forget about the GPU, just straight go to TPUs. Something like that?

Tegus Client

Right now, let's say you're running like Google Dataproc on GCP. Which instance should you use? How should you configure Spark to use the network one, like compute optimize, et cetera? Should you use the GPU version of Spark, right? Or does NVIDIA has that kind of GPU version of Spark that you could use? Like does that make sense? Or should you use Photon, which Databricks released? But The Photon only really works on certain workloads. So does that make sense? The solution is a prediction algorithm.

Head of Data Engineering at a Large, Publicly-Traded Financial Services Company

So you can get resources on the spot for these kind of things? No, not really, right?

Tegus Client

Spot instances, like spot pricing?

Head of Data Engineering at a Large, Publicly-Traded Financial Services Company

Yes.

Tegus Client

Yes, so spot is like one element to optimize over. I was just curious about your use case for big data, ML. Is data your main workhorse? Do you have a lot of machine learning workloads? I'm kind of curious what's your primary workload on the cloud, what you will mostly do.

Head of Data Engineering at a Large, Publicly-Traded Financial Services Company

Sure. I'm into a lot of stuff data. So I have built decades ago data warehouses, operational data stores. And many years ago, on-premises data lake and have done also cloud data warehouse like Snowflake and have done kind of delta lake or lakehouse and used AWS and GCP mostly.

And at one time, we had Azure on-prem private cloud along with Cloud Foundry and yes. So that's just a broad set of data. And we are a very information-intensive company, a lot of data. We are very in the data manipulation business and all, very similar to amazon.com, except they have intellectual property in logistics.

We have some other intellectual property in banking and other stuff, but it's very data-driven. With that said, data is a lifeblood. And data governance, that is compliance, privacy and security, is very important to us. It's not uncommon that more than $1 trillion flows through our system in a single day.

And perhaps in a whole year, we could lose billions of dollars to cyber threat, fraud, so on and so forth. So given all that, the workloads are everything and anything, but I am right now kind of leading, where I have built about 160-plus AI/ML NLP, artificial intelligence, machine learning, natural language processing models.

Many of them have gone to production using MLRs, machine learning operations and some DevOps and some DataOps, all the different names. And so some in the cloud, but mostly on-premises.

And our workhorse is basically Spark and PySpark when it comes to EDA exploratory data analytics. And we use many of the ML libraries from within Spark and invoke via APIs within Python and the PySpark environment.

We do some GraphX and do some streaming, even though we use regular Kafka for lot of our streaming. So and there are some other open-source tools and some vendor tools like H2O, which is open source and a vendor product.

And also, we use DataRobot. And so our data lake has tens of petabytes of data and we realized the data lake will not answer all of our prayers. So now the whole idea is the data fabrics of interconnected islands of data. That's my terminology. But that's where we are with respect to data and the kind of work I do mostly among other things.

Tegus Client

I'm curious, you said you're split on on-prem and in the cloud. I'm curious roughly what percentage is on the cloud and what percentage is on-prem.

Head of Data Engineering at a Large, Publicly-Traded Financial Services Company

It's a little sad story. Given we're highly regulated, our compliance people still believe that cloud is not that safe. Even in the last 10 years, not a single breach has happened in the top five cloud providers.

But so let's say, about 20% in the cloud and 80% on-prem, but that's shifting. Maybe in two, three years, it will be more than 50% in the cloud. We will never have 100% because we have certain very restricted data like passwords and ATM PINs will never go to the cloud, plus some of our compliance stuff when we audited by our regulators or we have something called UAR and SAR, unusual activity report, suspicious activity report, that can be on customers, partners and all employees.

So even the CEO doesn't get to see some of those reports if we suspect something funny he's doing or a trader is doing in our wealth and investment management group, so on and so forth. So sensitive data will always be on-premises.

And of course, we are trying to do different things. We redact certain data. Like we keep the last four of the social number instead of the whole thing. We tokenize the data using data guides at Delphix. Plus we're trying format preserving encryption as well as synthetic data generation for the machine learning modeling. We also scrub the data, and we also mask the data. So that's important. And of course, while data is in transit or any storage, it's encrypted and generally has in AWS and GCP. What else?

Tegus Client

So the portion that's in the cloud, and that is growing over the next three years, is your high-level cost or performance a concern? And so I'm curious, is the cloud bill substantial enough for you all? Is it a concern for you all? Or is it more performance that you just want to get your stuff up and running fast?

Head of Data Engineering at a Large, Publicly-Traded Financial Services Company

Yes, the initial workloads were mostly how to get hyper-performance and more agility, like faster innovation. But going forward, when we migrate more, the cost will always be a concern. And we don't want users to have like an unnecessary instances hanging around. Like an AWS bill, there was a little joke or a little cartoon somewhere.

It said this poor guy in like rags sitting in a park and someone asked what happened to you, you were very good some time back. No, by mistake, I had five of my large AWS instances on, and that's it.

So that should become a big concern, yes. We will monitor and regulate all that. Now it's a little free when only specific workloads are going. And yes, there will be misconfigurations and unnecessary stuff lying around. And the storage devices like not do the very cheap storage for the archiving stuff rather than like for anything and everything. So the layers of the types of compute, the types of stories, the types of memory, everything else will be important, and the right configuration, optimization will definitely be very important.

See, the cost has always been effect and cloud is not cheap. So the cloud is only cheap in certain circumstances. If you burst into the cloud, that is you have a steady workload but if it's like data warehouse and others, like various more or less steady workload growing a little low over time, that percentage can be high, but it's still growing a little over time. On-prem, if it's properly managed, it's a lot cheaper than the cloud.

So and the bill can be humongous, but if I burst in the cloud, for example, in my machine learning workloads, I may be training intensively using 100 CPU instances or like VMs with CPU and 50 GPUs and 25 TPUs, but I mean we're doing that for 2 days a month.

I'm not doing it all like 30 days. That's where they get the biggest bang for the buck, right? Fast provisioning and like can I just pay you for what I need? And that is very inexpensive versus me doing it all myself and paying for the whole thing. So that is very critical. That is important to us. But a quick question, if you don't mind. And I know AWS says, but the documentation is horrible and we still make mistakes.

And if I set up like some minimum instance, it will be so difficult to fix it. But like many of these guys are claiming all they have. So they may have the best algorithm and everything, but how would they differentiate themselves?

Tegus Client

So, obviously, cloud and optimization is a huge space with a lot of companies. So one, a lot of companies do like local optimization. Like there's like spot stuff. You can optimize spot instances. Or even like Spark, you can optimize like Spark configurations.

Some companies claim to do that. What this company can do is they do Spark configurations, the hardware and the economics and then even the scheduling, all simultaneously, right?

Because everything impacts everything. It's not isolated, right? The memory you select in your Spark configuration has an impact on the hardware memory you're going to select. But then that impacts scheduling because well, maybe at a certain time, the spot availability is bad for that machine, right? So you have to kind of do everything at the same time. And two, they actually profile your Spark workloads, basically mathematically modeled Apache Spark, mathematically modeled Amazon hardware and Amazon economics.

Head of Data Engineering at a Large, Publicly-Traded Financial Services Company

Yes, yes. No, that's good. So kind of constrained global optimization kind of thing. And what resource is required or what it ran out of or what it was high on, things like that.

Tegus Client

Yes. It looks at the history, the logs of your job and look at all the guts, the internal stuff, the low-level stuff. And then it will hyper-optimize basically your cluster utilization, the network, CPU, network storage, et cetera, and make sure that the instance you're running on is the correct one. Maybe there's another instance type that you can be cheaper or faster on.

Head of Data Engineering at a Large, Publicly-Traded Financial Services Company

But it’s not looking at the workload itself, but the log from an earlier run of the workload. Is that right?

Tegus Client

Yes. So that is your property. It’s just the log. It's like the metadata of your job. So you might not look at a static Spark job, but the data coming in, depending on the size or the SKU or the schema, kind of have a huge impact on how the cluster is used. So it can expand. It's not just looking at Spark log, but maybe the rest of your infrastructure.

Head of Data Engineering at a Large, Publicly-Traded Financial Services Company

So basically, any workload on the cloud, be it the warehouse or lakehouse or MLOps or ML model, training doesn't matter, right?

Tegus Client

That's right. So for your cloud usage, are you looking for like managed services like Databricks? Or are you most like an open-source shop? Kind of curious what is your take there. Or are you even like serverless? That's kind of an option also.

Head of Data Engineering at a Large, Publicly-Traded Financial Services Company

We do some serverless, but the rest going forward, when it is large volume and everything, we'll probably manage it. And if you are in GCP, the two key cloud providers, public cloud providers going forward will be Azure for business apps and GCP, but mostly analytics workloads. And so that's what we are thinking of going forward.

Tegus Client

I was just curious, do you like open-source, Spark, Databricks or like serverless?

Head of Data Engineering at a Large, Publicly-Traded Financial Services Company

Yes. So right now, internally, it's all open source, okay? So cloud, I don't think we'll pay big bucks for Databricks and others to manage a whole lot. But each of the groups have someone will say, well, I will do it this way in GCP or that, another way in Azure, that's fine.

The investment banking group may use different products or different IT philosophy than the core banking. And we are technologically pretty savvy, tens of thousands of engineers. So we do a lot of open source. So my base platform is mostly open source.

But if it's efficient and effective and is cheaper, then we'll use it because we don't necessarily have to bulk the servers, so we know nothing, please do it for us. Like in some cases, we do serverless, but typically for some of the standard workloads, maybe not. Does it make sense?

And suppose I have a three-hour ML training job, the initial data ingestion and some exploratory data analytics might require different type of workload than later on if I'm just doing my model pipeline, not only the cloud should be elastic and kind of scalable, but also we should be able to use it through this mechanism of real-time kind of like allocation of resources.

Tegus Client

Exactly. I'm curious, I know you do a lot of machine learning. I know you use Spark as your main workhorse. Do you use like PyTorch or TensorFlow? I'm kind of curious what other large open-source platforms you use.

Head of Data Engineering at a Large, Publicly-Traded Financial Services Company

We use PyTorch and TensorFlow, yes.

Tegus Client

All right. Are those your main workhorses for like ML training?

Head of Data Engineering at a Large, Publicly-Traded Financial Services Company

Yes. So if you think of it today, despite everything else as others are telling and except from specific things like AutoML, driverless AI kind of thing, if you leave a few of that, like most of the work is very well done by open source, like Spark and Python combinations, right?

Tegus Client

So I want to ask, some large enterprises now that their spend is on that scale and they're drowning in cloud bills and they're desperate. They said even 10% would be meaningful. I mean some are in the hundreds of millions per year. And in some cases, it's doubling every year. And a few years ago, maybe they didn't care because it was too small. They said, whatever, cloud is still cheap.

But now a lot of these companies are forming kind of like cloud police within the company. They're saying, okay, the spend is now getting crazy and it's not shrinking. And they now actually have a team dedicated to just lowering cost and making sure everyone will base best practices, that they don't leave idle machines on, et cetera. It's becoming a very big concern.

Head of Data Engineering at a Large, Publicly-Traded Financial Services Company

Yes. So we are setting up a police also. That's what we are doing.

Tegus Client

Yes. You need a cloud police. Developers don't like to think about this stuff. And so you kind of need someone to enforce it.

Head of Data Engineering at a Large, Publicly-Traded Financial Services Company

So this company is going to be the weapons or the guns of the cloud police?

Tegus Client

That's one way to think about it. Developers also like using it because it helps them debug faster because it can kind of basically project how their code will do on the cloud in different instances. So they can kind of see, is my code scalable? Like am I building it right?

So it's not just the cost savings tool. It also has a lot of advantages for developers. It can actually project and let you know how best to get your code running. So it’s a performance tool, kind of a nice cloud feedback mechanism for developers also.

Head of Data Engineering at a Large, Publicly-Traded Financial Services Company

No. And that would be good. That is very pertinent, especially if some of the cloud users may not know all the nuances of like properly optimizing resources. And given some of the documentation is not good and the way they set it up, they do a little trick also. It's not very simple to really get away with like bring it to zero instance and then like fire out through Kubernetes and do that all the time. Because you will have some residual stuff even if you like want to get to zero. Like let's say at the end of two days, I want to get to zero, it's not so easy.

Tegus Client

Right. Exactly. And the cloud changes so much, right? Like there are new chips every month. Amazon has a new Graviton chip. Like who knows if you should be using it or not, right? Maybe the old Intel stuff is good enough and a lot cheaper. It changes so fast.

Head of Data Engineering at a Large, Publicly-Traded Financial Services Company

No, that's pretty good. And then you're saying on top of that, I can get a feel the extent of their scalability or what?

Tegus Client

Yes, they can because there’s a graph that can actually show you how this thing scales. So if you run it on 5 nodes versus 10 or 100 nodes, how does it scale? How much faster can you get? Because in distributed systems, typically, the scaling stops, right?

It's not always that you add more compute. It goes faster. Eventually, it does stop. So the question is, when does that stop occur? Is it 10 nodes? Or is it 50 or 10,000? Like how scalable is your code?

There's also like this annoying step for developers, which is configuring Spark, picking all the Amazon stuff. It's a very manual thing. And developers really hate that. It's really annoying. They're very unsure what to pick because it's a different skill set, right? It's not writing PySpark code. It's like interconnect storage. Like they don't know. And they also really have no interest in that stuff.

Head of Data Engineering at a Large, Publicly-Traded Financial Services Company

So typically, you would expect some linear scalability at least during the initial part, and then I’ll see where that is and where it starts to become nonlinear, right?

Tegus Client

Correct. Yes. That's where there's a lot of cost savings because some people think naively that it just scales forever, but even Spark, Spark actually does not scale very well.

And so especially if they're like a machine learning job, they're quite complicated. And so there's like basically a maximum performance you can get. And if you provision more, you're literally throwing money away. Like it does nothing, right?

Head of Data Engineering at a Large, Publicly-Traded Financial Services Company

So when they say,of course, that's a marketing jargon, that it's infinitely scalable, at the end, of course, But the question is, what is that infinite point, right?

Tegus Client

Yes, it depends. I mean it depends on the code. It is highly, highly dependent on your code. A lot of it is like, how much communication? Is it embarrassingly parallel? Is there a lot of communication? Is there a lot of data transfer? Are you like moving data in and out of storage? How are you partitioning the file? It gets very, very complicated.

Like this thing starts to die at 5 nodes. So if you're a developer, like, oh shoot, my code doesn't scale well. Let me figure out why and they go back and like make my code more efficient. That kind of feedback is nice. And you don't have to actually manually run it on Amazon.

Head of Data Engineering at a Large, Publicly-Traded Financial Services Company

So for example, if you are like trying to read a lot of data at a time, maybe well, they do partition it. And plus Spark itself has many of the built-in parallelism in memory, compute and all the partitioning. So Spark does a lot of that, doesn't it?

Tegus Client

It does. But the performance of Spark depends. Like if you get a lot of shuffling, they're using a lot of networks. Network is really how things start to die, right?

If it's "embarrassingly parallel," which is kind of a regime of computing, that's from the high-performance computing world, where like every node just independently calculates something and never talks to anything else, then that kind of code scales very well.

But if you have like a machine learning job where you're doing like MapReduce where you're doing computations and then each node has to transmit data and combine and sort or move around data, then you're bottlenecked by network.

And that usually is what kills your scaling. And then eventually you are network limited and adding more and more nodes does nothing because your network cannot keep up. So it's a function of your code and how complicated it is. So there's a lot of shuffling, there's a lot of communication, then your network is going to kill you and then your Spark job will not scale.

Head of Data Engineering at a Large, Publicly-Traded Financial Services Company

So basically, one has to optimize the memory. And then you minimize or optimize the shuffling and maybe have the other worker nodes colocated so that you're not doing any long-distance network traffic, right?

Tegus Client

That's correct. But that's something,the colocation thing you mentioned, where if you colocate the nodes that are talking to each other and exploit because especially on the cloud, when you get like 100 machines, the network is not uniform.

And so colocating jobs so that the high network nodes are next to each other is kind of how you want to basically minimize network traffic. And so there's a whole lot of optimization and distributed computing is really a resource allocation problem, right?

Head of Data Engineering at a Large, Publicly-Traded Financial Services Company

Very nice, and they can specify that like colocate this instance like VMs or the instances that I'm asking for. One can do that?

Tegus Client

You can do that for certain workloads. Yes, for like a lot of high-performance computing workloads, you can do that. With Spark, it's a little bit trickier. But basically, that is kind of a well-known thing from the high-performance computing world, is like colocating nodes to make sure that the network is really optimized.

Head of Data Engineering at a Large, Publicly-Traded Financial Services Company

So there are two parts. Spark, of course, has its own like abstraction layer on top of everything it does. But let's say in AWS or GCP, I can say, give me five large instances that are colocated to each other, next to each other. I can do that?

Tegus Client

You can request within like the same region or same zone, right? So yes, you can request that within Amazon. Yes.

Head of Data Engineering at a Large, Publicly-Traded Financial Services Company

Right. So even though the blade #1 and blade #100 maybe a little like diagonally opposite in that data center. It's possible, right? But if it's 10 gig or 100 gig, who cares, is it or did I say it wrong?

Tegus Client

Yes, it depends on your workload, right? If your workload like has very little network, then actually, you can save a lot of money because you're like, well, maybe I could just scatter my workload across a huge area because it doesn't matter that your nodes are not talking to each other. You can save a lot of money that way, right?

Or maybe your workload has a huge amount of network now that, okay, you really need to colocate this and put this all ideally as close together as possible. So how do your developers know that? Most of them are just worried about the logic of their code. They don't think about network.

Head of Data Engineering at a Large, Publicly-Traded Financial Services Company

Got it. And then there’s hints to the developers besides telling that they scaled out at 10 nodes or at 15 nodes. Are there hints like do this to increase my app’s scalability?

Tegus Client

Yes, they basically provide an analysis and say, what is your bottleneck, right? So if your garbage collection rate is really crazy, you should increase your memory per core. That's my bottleneck. Let me see if I can do something with the code.

Head of Data Engineering at a Large, Publicly-Traded Financial Services Company

So it's the workload optimized to minimize the network traffic in communication. And of course, the standard thing typically, the more the memory, it will handle many of the problems. Is that correct?

Tegus Client

More memory is always nice. The downside is cost, right? Especially for your driver node or worker node, the more memory typically leads to kind of more expensive instances.

So that's where a lot of cost savings is kind of like rightsize the amount of memory your job needs. Because a lot of people just like really overprovision memories. They usually give like way too much.

You're paying like 3x more than you should because you don't know the exact needs of your application. You have to profile it and see, oh, you're only using this many gigs, but you're allocating 3x as many. So it's unnecessary.

Head of Data Engineering at a Large, Publicly-Traded Financial Services Company

The AWS, what is it called? Is it CloudWatch? Or whatever it is that does the resource utilization, it doesn't give all the details, right?

Tegus Client

They don't go to the level tha’t’s needed. So this company is analyzing your Spark event log, going kind of a layer much deeper. The stuff that like high level, that like a lot of the monitoring companies do today look very high level, like CPU and memory usage, but they don't really know what the application is, right? It's just like just two numbers. Like my CPU usage is this, my memory usage is this, but they don't know why.

Head of Data Engineering at a Large, Publicly-Traded Financial Services Company

Got it. They don't tie to the application and what's happening. They just monitor their resource, the memory and CPU, how much is being used, that's it, kind of.

Tegus Client

Exactly. Yes. They are too high level. You can do a lot of good stuff. I'm not saying they are not good products, but that's kind of the old way of doing the cloud. And this is like a new frontier, like general cloud and compute optimization.

Head of Data Engineering at a Large, Publicly-Traded Financial Services Company

No, it's very nice, very nice. And so right now, MVP after the proof of concept and anyone can go and try it out for free, just cut and paste like in a cloud log, like in a log file, and tell them, set this many instances, this much memory, this much storage. Is that right?

Tegus Client

Yes. That's it. That's exactly right. I don't know if you know the world of optimization, but like you can make a lot of gains if you have a very specific problem. And so cloud and like Spark, for example, is a very specific problem. So you can build a custom solver just for that problem and you can go really fast. But you have to know the whole problem and you have to own the whole stack to be able to develop a solution.

Head of Data Engineering at a Large, Publicly-Traded Financial Services Company

Right. So there are some similarities. In my machine learning, I do hyperparameter tuning, right? If it's random for us, how many trees at a time? What's the best number of branches and number of leaves at each branch? So this is also dealing with some hyperparameters, right?

Tegus Client

That's right. That's right. Yes. Optimization and ML are actually very similar, right? They're actually both optimization problems, right? There is, of course, some hyperparameter tuning, but because the problem is so similar, right, it's always Spark, there's always compute, storage, network, et cetera. That never changes.

Head of Data Engineering at a Large, Publicly-Traded Financial Services Company

So they can take this log data, and in a matter of a few seconds, they spit out the answers?

Tegus Client

It does depend on the size of the logs. Some are absolutely humongous Spark jobs that might take like 10, 15 minutes. Some are really small. You'll get it under a minute. So it scales on the size of your job and how many tasks are in your Spark job.

Head of Data Engineering at a Large, Publicly-Traded Financial Services Company

And so what is the business model? So we will pay based on the number of search optimizations or what?

Tegus Client

Yes. Eventually, kind of like a Datadog model. I think they might scale by volume. Some companies have like 100 production lines going every day, some have 1,000. So if you use more, obviously, you pay more. So it's all that kind of usage.

Head of Data Engineering at a Large, Publicly-Traded Financial Services Company

And when you say to use more, the number of optimizations you do, that's the definition of more, right?

Tegus Client

Yes, typically, the number of times you use the service. Yes, exactly.

Head of Data Engineering at a Large, Publicly-Traded Financial Services Company

And what is the footprint? So because when you do the compute, you will be doing it within the environment. It's not like a service that you will use your own compute and do everything and give us the results? Or will they have the agent sitting or actually running in our environment infrastructure?

Tegus Client

So it doesn't sit at the client side. You transmit your metadata and you get the answer, right? So all of the footprint is not on your side.

Head of Data Engineering at a Large, Publicly-Traded Financial Services Company

Got it. And they keep the data secure and all that kind of stuff, right?

Tegus Client

That's correct. Long term, of course, having all the security compliance, SOC 2, GDPR, et cetera, all of that stuff.

Head of Data Engineering at a Large, Publicly-Traded Financial Services Company

Yes, that's where I've spent my 18 years here. Before that, I was an entrepreneur in Silicon Valley, all that stuff, but I have been here for quite some time. That if you want to really market it to large enterprises and be regulated in the space, be it financial, insurance or health care, there are a lot of things to watch out for.

Tegus Client

Yes, absolutely. Well, this was insightful. Thank you for the time.

Tegus is not a registered investment advisor or broker-dealer, and is not licensed nor qualified to provide investment advice. The information published in this transcript ("Content") is for information purposes only and should not be used as the sole basis for making any investment decision. Tegus, Inc. ("Tegus") makes no representations and accepts no liability for the Content or for any errors, omissions, or inaccuracies will in no way be held liable for any potential or actual violations of United States laws, including without limitation any securities laws, based on Information sent to you by Tegus. The views of the advisor expressed in the Content are those of the advisor and they are not endorsed by, nor do they represent the opinion of, Tegus. Tegus reserves all copyright, intellectual and other property rights in the Content. The Content is protected by the Copyright Laws of the United States and may not be copied, reproduced, sold, published, modified or exploited in any way without the express written consent of Tegus.