Introduction

Scott Fulnecky: We are excited to be here today with you all and talk about the connected factory proof of value that Trek10 has put together with AWS.

I'm Scott Fulnecky. I'll be your host today, and my role at Trek10 is on the sales side of things. I'm joined by a couple of my colleagues and a special guest from AWS, Thomas Cummins. Thomas, could you go ahead and introduce yourself?

Thomas Cummins: Yes. Hi everybody, my name is Thomas Cummins. I'm a Partner Solutions Architect at AWS. Glad to be here today. So I specialize in IoT and helping our partners build industrial IoT Solutions. So I'm working with solution integrators, ISVs, OEMs, and in particular, been focused on this connected factory initiative over the past several months. And so glad to be here today.

Scott Fulnecky: Awesome, thanks Thomas. I'll turn it over now to Carlos Lemus, our principal architect for the IoT business unit, to introduce himself.

Carlos Lemus: Thanks Scott. My name is Carlos Lemus and I am the practice lead for IoT here at Trek10. I've been in the IoT field for six years, and before that, I worked in process control. I've also worked across several different verticals, both in the consumer and industrial spaces for IoT, and it's a pleasure to be with you all today.

Scott Fulnecky: Awesome, thanks, Carlos. And we're also joined today by another Trek10 teammate, Brenden Judson. Could you say a couple of words, Brenden?

Brenden Judson: Yeah. Thank you, Scott; I'm both a data scientist and cloud Engineer at Trek10. My involvement with this POV is that I will focus on the machine learning aspect of the POV. And I look forward to showing you some of the possibilities that are associated with that today.

Scott Fulnecky: Awesome. Thanks so much, guys. With that taken care of, let's dive into the agenda today. We've set aside about an hour to talk about the state of manufacturing today, where AWS is putting time and energy to connect factories to the cloud, and the offer Trek10 is taking to market with the AWS team. So we'll have a demo that will show some of the connectivity and function of this engagement and what it'll bring via SiteWise. Brenden will share a demo showing you a bit about the way that machine learning will be involved and what that might entail.

Alright, so let's dive in. I'll set the tone by giving a quick rundown of who Trek10 is and why we're all here today.

We are a Premier Consulting Partner with AWS, you know, we're born in the cloud, 100% focused on AWS, and we really specialize in leveraging the best tools and AWS managed services to design, build, and support cutting edge solutions for our clients. We've got hundreds of production deployments under our belt, and our team is one of the foremost experts in building out serverless applications and enabling enterprises to migrate to the cloud native architectures. Massive scalability, heavy automation, and remarkably low operating costs are our hallmarks.

As launch partners for the IoT competency with AWS, about five years ago. Now we've been building out solutions on AWS since then, with a focus on event-driven architecture built in a cloud native fashion with decoupled microservices to achieve this.

Industrial & Digital Revolution - Digital Transformation - Industry 4.0

3:20

I'll kick things off by, again, just kind of setting the tone for why we're here. It all starts off with the Industrial Revolution, and it'll move towards the present state of affairs. So, taking it all the way back to where it began with the Industrial Revolution, the Mid 17-1800s saw a transition from hand production methods to machines, new chemical manufacturing, and iron production process, increased use of steam power and water power. It also led to an unprecedented rise in the rate of population growth and social unrest, and as we all know, it was a major turning point in history. We saw the standard of living increase and catapulted society into the modern age, influenced almost every aspect of modern life, and it's still going on today. It's iterated to take on a few different forms during the Second Industrial Revolution, which took place between the 1800s and early 1900s. We saw the birth of globalization as the Industrial Revolution spread beyond Great Britain and the US and out into the rest of the world's developing countries. It served to increase the production of iron and steel on a large scale and connected cities with railroads, building on the foundational components that the Industrial Revolution had kicked off a century before.

Some of the greatest improvements to the quality of life during this era came from the proliferation of basic modern infrastructure like sewage, water, gas, and electrification for the masses and new technologies like the internal combustion engine.

Radio, the telegraph, the telephone, new materials, chemicals, and alloys that were invented again catapulted the quality of life and standards, far exceeding anything expected or experienced to date. As we transition through the Industrial Revolution, we hit the third Industrial Revolution, or the Digital Revolution, where we saw a switch from analog electronics to digital and brought the dawn of the information age. Like the first Industrial Revolution, we saw the rapid expansion of technological innovations. This time, instead of the shift in manufacturing to machines from hand production methods, it transformed both traditional production and business operations, utilizing new inventions like the transistor to make new products like computers, cell phones, and later, the internet and was mass adopted by consumers before where we land today, or the fourth Industrial Revolution.

Fast forward to December 12, 2015, an article by Klaus Schwab, where he states that the fourth Industrial Revolution and what it means and how to respond. He states that the current shift from simple digitization, or the third Industrial Revolution, to innovation based on combinations of technologies, basically the fourth Industrial Revolution, is forcing industry to understand their environment, challenge operations teams, and innovate like never before. So unlike the first industrial revolution where technology really took hold and expanded in the manufacturing sector, digital transformation happened first at scale with consumers with the mass adoption of new digital technology like CDs, cell phones, PCs, the internet.

So it's time for the corporate manufacturing world to catch up. Think about your smart smartphone and all the data at your fingertips, how it connects to your car easily, your speakers, your lights, your HVAC at home.

With that in mind, we want to present out here, some of the terms that get used commonly in the industry and speak to the diverse crowd that we've got gathered here with us today from digital transformation teams, IT directors, operations teams dealing with the manufacturing execution, product lifecycle teams, we've got some plant managers, I think we've got some guys from Skunk Works and Innovation Teams and a couple of C-level IT leaders, and maybe some potential partners on the webinar as well. So explained in a way that addresses some of the business problems that these disparate teams can have:

Digital Transformation is the process by which we implement these solutions. Industrial IoT is the result of that, and industry 4.0 is really this point in time and when it's all happening. So early adopters today have paid a premium, and everyone in this room knows the fate of the late adopters who haven't or aren't willing to ask these tough questions.
Like steam power, electricity, computers, and automation before, now industrial IoT, distributed computing, data ecosystems, and machine learning are becoming cheaper and more ubiquitous.

We're here to talk about Industry 4.0 adoption and moving beyond the discrete connections that exist within the 3.0 Automation Stack and towards the democratization of access to data for all these disparate teams. So, a Data Platform Ecosystem in the cloud that breaks down these data silos is the ultimate goal.

Today, we're going to talk about the engagement that Trek10 has created using the components of the industrial machine connectivity Kit to connect the data from the factory floor into AWS, gaining visibility and line of sight to manufacturing performance for root cause analysis and how we use that data, via a machine learning POC to inform decisions made for data where we create new information for that data. With all that in mind, I'll hand it off now to Thomas to speak a bit about the AWS connected factory vision.

AWS Connected Factory Vision
9:16
Thomas Cummins: Thank you. So good to talk to everybody today. Again what we've been working on over the past few months at AWS in our IoT partner team is how to put together a vision for what we're calling the connected factory, and what that means is, we will need to

make it easier, simpler, faster, and cheaper for our partners that we work with. Again, I come from the partner team, so I'm very partner focused here.

How do we make it faster, cheaper, easier for us to deliver: number one, proof of value solutions to customers to get them to really see into the business value of an industrial IoT solution at their manufacturing plant. Number two, how to achieve success criteria for those proof of values and then move on to scale to multiple facilities. Where we want to focus today, in particular, is going to be this piece right here.

So in particular, we want to focus on the machine to cloud connectivity piece. You're probably familiar with AWS services that are in the cloud related to data lakes, advanced analytics, machine learning, data visualization, business intelligence tools like QuickSight.

Today we want to focus on the machine to cloud piece, which we've seen be a struggle for our partners and customers because every single factory floor is different. There's a variety of different machines we're connecting to, PLC from different vendors, different industrial protocols we have to work with. And so this is the key technical problem we want to try to solve and that's what we're going to focus on today.

That's where this industrial machine connectivity or IMC architecture came from. What we're trying to do is simplify bringing data in from assets on the plant floor, whether those are PLCs, historians, or other scattered systems, and then bring that into AWS in a structured manner.

We're generally talking about bringing in near real time data, so streaming data, process variables for example: tags, bringing up to the cloud. What we want to do is, instead of building many different industrial protocol adapters and maybe running those inside of Lambda functions and Greengrass, what we want to do is rely on our partners' edge applications that are already used and trusted in by customers, such as PTC KEPServerEX, Inductive Automation Ignition Server, and we have several more partners that we're adding to this list. So we want to rely on those partners to handle the PLC, the historian, and connectivity, handle all the industrial protocol translation and then what we're going to do is use our service AWS IoT Greengrass to basically interface with those applications and then bring data into the cloud.

And the primary service we're looking at in the cloud to bring the data in and the asset model information is AWS IoT SiteWise and I'll get into a little more detail about how that works. The reason this is important is, currently bringing data in from the edge of the cloud is one thing, but you really need the context of that data, you need the asset hierarchy information, you need to know where that data came from and the description of that asset of where it came from, to be able to build really intelligent applications in the cloud that deliver business value to the customer. So if you're training machine learning models, you're going to need to know exactly what asset it came from. You're going to need to know other information about that asset in order to make intelligent decisions about it and show that to the customer.

So, AWS IoT SiteWise is a service that does three things for us. It allows us to model assets in the cloud. So it allows us to create, basically, digital representations of assets in the cloud. It allows us to structure those assets in a hierarchy that represents how those assets are actually arranged in the real world. It allows us to bring data into those assets and actually store them in a managed time series database. So, now we have the contextual data of the assets and then we actually have a database that's fully managed that stores the time series data that you can access via API, or move it into AWS IoT Core via an MQTT, and I'll show you that in just a bit.

A key part of this is this block called the asset model converter on the right hand side. That is a serverless workflow that we've built that basically maps the tags from the partner’s edge applications. So, KEPServer, Ignition are the two examples we started with. It maps those tags that are already defined at the edge by, say, a plant manager or an OT [Operational Technology] personnel, and maps that into SiteWise automatically. And so we don't have to go through the process of manually coding, or writing code to basically do all this translation. This architecture will do that for us.

Now, once we have all the context of the data and the data itself into SiteWise, we can basically move it into IoT Core on an event basis, so whenever a certain value exceeds a threshold, for example, we can trigger an event, send that IoT Core and then through its rules engine route it to a number of services. Trek10 here is basically building on this architecture to be able to deliver a proof of value offering to customers.

So I'll get into a little bit of detail of what's under the hood of SiteWise and actually, how are we modeling assets. Here's an example of how we're linking measurements to assets. So on the left hand side, say we have a physical Press Shop and it's in Portland, and we have a certain line called line one in that press shop, and there’s two presses, Press A and Press B. So, those are actual physical machines. Those tags are being generated on the PLCs attached to those presses.

We have an OPC-UA server that's pulling, actually getting those tags and that could be the OPC-UA server on KEPServerEX or Ignition, for example. We have a gateway that is basically consuming the data off of the OPC-UA server that runs inside of SiteWise, and then you see on the right hand side. This is how we structure assets within SiteWise. So, basically, you define a model to represent and it’s an abstract representation of a physical asset or organization, for example. So the first model we're using as a site, and there's a certain description of that site. It can be the location. It could be the size. It could be the owner. Then we instantiate that model to create a specific asset. So the specific asset here from the site model would be the Portland Press Shop and it's up type site. So, this is similar to the object-oriented programming concept of instantiating classes to create objects.

Then the next level down we have some sort of a model called a metal press line and that's what this is. Then we instantiate it to create a specific line, in this case, line one, and then same thing with presses. We have a model called press, and this is where SiteWise is very useful as in that model called press, this is where we define all the attributes of a press. So, this could include manufacturing name, model name, serial number. Things that are static like that, things that are changing all the time like the process variables, the vibration, the temperature.

We can also do what are called transforms. So, these are mathematical and logical operations we perform on that raw data to produce, say, an averaged value over the past minute of our windowed period, or a threshold detection determining if the raw value exceeds a certain value.

Those are called transforms and then there's another type of property that is defined within the model called a metric. And these are where you can perform more advanced calculations, such as building up OEE metrics for individual assets lines and facilities. So, all that's defined in the models in this case of press and then you instantiate that, in this case, two times to create Press A and Press B, and then what happens is SiteWise automatically links the tags that are being generated on that OPC-UA server. You can see the path example there for the tags and it automatically puts them into the appropriate place in the time series database within SiteWise to be able to store all that information.

Here's just a closer, another way of looking at the same thing. So, how we're modeling assets. So, in the icons or the diagrams on the right hand side you see, there's a stamping press, which is the model and we define different properties. We have attributes, measurements, formulas. Those are the transforms I discussed. Then you instantiate that model called the stamping press to create a specific stamping press and in this case, you identify uniquely by that tool ID property. So, this is kind of a different way of representing the same thing.

So, now let's talk about SiteWise. We're now understanding as a single service where you model assets. So, you have the contextual data, you have the metadata of those assets and then you can actually store the actual data itself.

So how do we get data in? There's three different ways: individual devices, you know IoT devices out in the wild, can publish over and MQTT to IoT Core, and then we have a native rule in IoT Core that will route data into the time series database in SiteWise. So, that's one of the most flexible ways for individual devices to send data.

The middle path, however, is sort of the preferred path, and that's when we were just looking at. Which is, data that's exposed on an OPC-UA server, we can pull data in via a connector that runs on Greengrass Core, which is an edge application that can run on any sort of standard industrial PC, industrial Edge device. That will pipe data directly into SiteWise's time series database. And that's the most performant and cost-effective way.

And then lastly is the most flexible, which is just using the AWS SDK and the HTTPS post request, which is an API that directly lets you put data into the time series database.

And then how do we get data out. So once state is in SiteWise, there's two different ways to extract data. You can use what's called the property update notification feature which is, I mentioned it before. But any individual time series value is mapped to a property. Whenever that value changes, you can instruct SiteWise to publish an MQTT message on a unique MQTT topic to IoT Core with that payload. So, it's basically a way to build a very zero code event detection system. So, any particular property, you can build, define a transform, or a metric, or just take the raw values if you would like. And anytime that value changes an MQTT message will get published, and then on IoT Core, you can use the native rules engine to route it to whatever service you'd like. You could route it into a data lake. You could route it into a service like IoT Events to do an advanced event detection model, things like that.

The second path is just using the API. So, there's a GET request that you can use to extract data out of the database. You basically specify the specific property you want to pull out, (property or properties plural) and then a time window that you want to pull the data out. So this is where you could do, you know, more batch analytics of historical data.

So, to wrap up, what we've developed here is something very specific. We're trying to simplify the edge to cloud connectivity of building industrial IoT solutions and we focused on beat of mapping tag data from these edge applications, like KEPServer and Ignition, into SiteWise and solving that problem to allow partners like Trek10 to be able to build solutions more quickly on top of this architecture, on top of these services.

So with that, I'm going to hand it back to the Trek10 team and Scott and Carlos to take us through the next few slides, including the demo.

Scott Fulnecky: Thanks, Thomas. Carlos, do you want me to stop sharing now, or are you going to dive into this slide, and then I'll stop?

Demo 1 - SiteWise

22:21

Carlos Lemus: Yeah, I think that that's a good overview of the IMC kit.I think we can move on to a demo. And what I'm going to do in this demo is, I'm going to show you all of this in action and we're going to be using Ignition as our deployment solution.

So, I'm gonna share my screen here. First, we'll take a look at some of the simulated assets which are part of the IMC kit.

Over here on the left hand side, you can see that we have some lines. We have a conveyor tag that has some properties in it like motor amps, temperature, vibration, and typical things that you would see out on a manufacturing floor. These are part of the IMC kit. They're simulated data. And so if we take these and then we go over to SiteWise, what we would hope to see is the same data that we see in Ignition, which is local on the manufacturing floor, would also be replicated up in SiteWise. So, if we expand this as a tree here, we can go to line one and again, we can go to conveyor. And then if we go to measurements, we can see that all of the data that we were seeing in Ignition is also available in AWS IoT SiteWise.

Alright, so next we will explore some dashboards that showcase this same simulated data from the IMC kit. So, to do that, we go over here to the left, expand this hamburger menu, and then we go under monitor, we can go to portals, and you can see that I have a lot of portals here that I've created, but let's go to the latest one, which is the one I created specifically for this demo.

We can get out here of getting started, we go to the IMC dashboard. And then you're going to be able to see all of this data. Within the last five minutes, this is all of the data that's coming in from all of our different assets. You can see booth pressure, motor amps, and so on.

So far, we've looked at simulated data. Part of the IMC kit, you can do this yourself, but now I want to show you an example of what real-life data looks like.

Carlos Lemus: Now to do this, we have built an application that gets weather data for different cities and it exposes that data in OPC-UA format as though it were any other piece of industrial equipment. And so then, what we can do is we can read that data through Ignition and then replicate it up into SiteWise. Let's take a look.

If we go back to Ignition, we can go to the city's folder which you might have seen earlier. And then you see that there is a tab here for the asset of Indianapolis, which is where I'm giving this presentation from. You can see that there are different variables typical for weather conditions. So, you have conditions, humidity, pressure, temperature. One thing to note here is that we have the temperature be 74.34 and we can put this as an LED display. So, you know, pretend like this is an HMI on your manufacturing floor. We can turn this into a label, and so all of this data is available locally for operators to use.

Now let's go back to SiteWise and make sure that this data is being replicated. So, we go back to assets and again, we can expand the asset tree here to list all of our cities. Go to Indianapolis, and then we can go to measurements to see all of our variables, and then we can see the value here 74.34. And again, that was the same value that we saw 74.34.

Alright, so now let's take a look at a different city. Let's try San Francisco. So again, we go to measurements and we see that the latest value is 54.07. If we go back, we can expand the San Francisco asset and we can see that it's 54.07. Great!

I want to show you now how we created this tag. So, we created a data type representing an abstract city and here we defined things like conditions, humidity, pressure, temperature, again, everything that you would see in your weather forecast.

Now, once we created this data type, that immediately creates what's called an asset model in SiteWise, which then is used to create new assets in SiteWise.

Let's try creating a new tab for a new city. We can right-click on cities to put it in that folder. We go to new tag, then data type instance, city, and we're going to name our tag Chicago. Then we can go down to parameters and we also got a pass in the city name of Chicago. Now, this is for the purpose of our application. You have the tag name, that's Chicago, but we also have to pass in the city name as a parameter for it to create the city successfully. We enter Chicago, commit, apply, and then we hit OK to exit. Now let's take a look and see that Chicago successfully showing some weather data, and it is right. We have again, clouds, temperature of 73.72, humidity 56 and that is our data on the Ignition side.

Let's go back to SiteWise and just make sure that it was successfully created We can refresh, to make sure that it loads everything properly. Expand cities, and then we go into Chicago measurements and sure enough, we have the data here, 73.72 and all the other data.

Now let's go to dashboards. I created a dashboard for the City of Indianapolis that shows some of these variables, and if we actually go in and change this to, let's say, the last hour, you can see that the temperature has dropped by about a degree for the last hour. This is one of the nice things about the SiteWise services that the portals allow you to very quickly spin up dashboards with all of the data that is coming in from your manufacturing floor.

You can see temperature is obviously the most fluctuating variable. Pressure and humidity don't change that much, and so you can see that the data changed over time there. Now let's go back and let's create a new dashboard for the new tag, we just created. We're going to call that dashboard Chicago.

Then if we go over here to the right, it's a little bit hard to see, but I believe that first one is the city of Chicago. Great. So, what we can do is we can take temperature and we can drag and drop that and you see that there's a single data point there. Let's go ahead and save this dashboard.

So a single data point, right, because we literally just created that tag. Let's see if zooming in makes it better, it's like five minutes. So, there's a single data point, right, because we just created that tag, but you can see that it's being visualized in the SiteWise portal. This has shown that we can create new tags in Ignition reading from, say, newly deployed equipment and then have that data replicated into SiteWise.

In summary, we have showcased that the simulated assets that are part of the IMC kit, as they would look like on the manufacturing floor and in the AWS cloud through SiteWise, we can have those replicated from one to the other.

We also looked at a live asset that pulls weather data for different cities and presents that data through an OPC-UA in the same manner as industrial equipment. Finally, we showed how those newly created tags are immediately put into the AWS cloud.

But, you might be wondering, what can I *actually* do with this data once it's in the cloud?

To give you an example of that, you will hear from my colleague, Brenden Judson, who will tell you about our Machine Learning Ops Framework, which is perhaps one of the most valuable portions of our connected factory offering. So with that, I'll hand it over, Brenden.

Demo 2 - Machine Learning and SageMaker
32:38
Brenden Judson: Thank you, Carlos, and thank you for the introduction.

I will go ahead and share my screen. Starting to look at the machine learning offering of our POV. To begin with, what have we already accomplished with the IoT POV? Well, we've unlocked the data from the factory floor and ingested it into our cloud AWS environment.

We have also standardized data across different data sources and different data formats, into a single common data format. Finally, we have stored this data into AWS SiteWise and ultimately, an S3 bucket.

There's a lot of value associated with the POV. However, I think there is an additional step to kind of close the loop on this engagement, and that is to utilize the data that we have gathered and captured to drive different business outcomes.

There's many different examples of how you can use data to drive different business outcomes. Some of them are to drive deeper insights, reduce uncertainty around your data and business in general, improve the efficiency of your business and processes, and identify different bottlenecks in those processes. You can conduct condition monitoring and anomaly detection in your processes and there's much more that you can do to drive business outcomes.

I think a key theme throughout this demo that I want to get across is, I will be giving you a specific example and a specific model to kind of get the point across. But, I want you to take a step back and think in terms of your own IoT data and your own problem statement, to see how you could leverage our tools to make your own model to solve your own problems.

So with that said, the Machine Learning POC. We want to showcase the realm of what is possible, and to do that, we're including a machine learning model POC with this POV.

Upcoming, I'm going to be showing you a specific demo of what this POC could look like.
To achieve this, and to implement and manage our machine learning model, we use an internal tool called our Machine Learning Operations Framework.

What is the Machine Learning Operations Framework [MLOps Framework]? Well, like I said, we built an internal tool that integrates with the IIoT POV. This tool provides you with a production environment that you can use to leverage and continue to drive business outcomes over time. It provides you with an environment for testing, training, and hosting your machine learning models, and it comes with additional features such as logging, quality assurance, drift detection, and automatic model retraining.

Now, the entire framework was implemented using the AWS Well-Architected Review, Machine Learning Lens best practices, and finally, and possibly most important, the framework is model agnostic. That is, any machine learning model can be used with this framework.

So, I think that speaks to the theme again of trying to take away the high-level concepts, so that we can apply machine learning to our own use case.

Now getting into the specific demo, I'm going to be showing an anomaly detection demo.

As we know, all machines eventually reach a point of poor health. This leads to malfunctions, which can have serious business consequences. We want to use our IoT sensor data to identify these anomalies and preemptively avoid these malfunctions.

I think a very common and fair question would be, so why do we need a machine learning model to do this? Why can't we just apply thresholds to our IoT sensors and also set up alerts when they pass that threshold? I think this slide kind of speaks to why a model is needed in certain situations.

A threshold could work in a very simple use case. However, in the real world, problems are much more complex and I think this will speak to it. So on the left, we have a bunch of data points in a two dimensional plane. And as you can see, it is very clear what record is the anomalous record. It is this one up here in the far left that is very far away from any other data points. However, if we take the same data on the left and reduce it to two separate one dimensional spaces, it becomes very unclear where the anomalous record is. This phenomena is drastically increased as you go to N-Dimensional space and work N-Dimensional features. And so there's these kinds of real world problems where simple thresholds are insufficient and you need a machine learning model to approach a problem.

For our specific use demo, we're going to be using K nearest neighbors anomaly detection algorithm. This is a very simple but powerful machine learning algorithm. I have a demo here to kind of show you how that works.

So to begin with, when training the model, it takes all your data in and plots in N-Dimensional space. For this example here, we're working with two dimensions, seen by the Y and X-axis, and all the blue points are the training data plotted in that space.

When it gets an incoming new record it wants to apply inference against, it plots that record in the same space, seen by the X here, finds the K nearest neighbors, in this case, the number K is three, and these nearest neighbors are circled in red. It calculates the distance between the incoming point and each of the neighbors. It takes the average of those three distances and compares that to a threshold, and if it is above the threshold, it marks the incoming record as anomalous.

So, you can see, instead of just comparing it to a single dimension, this model is able to include all the different dimensions and features to detect anomalies.

Okay, so what we're looking at here is the end to end, machine learning process workflow. This is a standard workflow that we use when approaching any machine learning problem and I can run through this really quickly to give me an idea of what it entails.

So to begin with, we always start with the business problem. We want to identify what really would provide value to the business and that is the ultimate goal. Once we have identified the business problem, we frame this into a machine learning problem and answer questions like what kind of model we use to solve this business problem. After that has been framed, we collect and integrate the data into our environment. Once the data has been collected and integrated, we need to clean and prepare that data so it can be used by the machine learning model.

Once the data has been cleaned and prepared, we want to do some (EDA) exploratory data analysis, data visualization, and analytics to get a deeper understanding of what data we actually have.

Once you have an understanding of your data, you want to create different features that can be used by the machine learning model. That is called feature engineering.

Finally, we get to the juicy stuff, which is actually training and evaluating our model. Once we have a model live, we ask the question, whether or not we have answered our original business problem. If not, we go back to the circular workflow and restart with EDA. If we have though, we have a model that we can use to leverage predictions.

So, going back into our specific demo and use case.

We are going to approach this demo from the same end-to-end machine learning process workflow. So to begin with, what is the business problem? Company X wants to detect anomalies in their data to avoid malfunctions, and ultimately avoid downtime on their equipment. Frame this in a machine learning problem: We're going to use K nearest neighbors to detect anomalies in their IoT data. Thirdly, data collection integration. This step has already been covered by the IIoT POV that Carlos shared with you. And so as I mentioned, I'm going to share a demo now and the rest of the steps of the machine learning process will be covered in that demo.

Starting at the AWS Console. As I mentioned, we already identified the business problem and we framed it in a machine learning problem, and we have integrated our data into IoT SiteWise.

We're working with the same IoT SiteWise data that Carlos has already shown and so I will go ahead and navigate there.

I will navigate to the exact asset that we are working with. It is a simulation data of an assembly line. Now I will navigate to the two measurements that we are actually interested in, and that is temperature and receiver hopper. These are two real numbers, and it is the values that we will be using to train the model, and also we will be looking at to detect anomalies.

With IoT SiteWise, in combination with our data platform framework, we have completed the next step which is cleaning and data preparation. Following data preparation, is data visualization and analytics. To do this, we use a tool such as Tableau to get a deeper sight into our data.

And with that said, I'm going to hop over to Tableau. You can see we've connected Tableau to our data to get a deeper sight. You can see here, it's receiving the temperature and both the hopper data. I'm going to go over to this dashboard I created to get a deeper look at the temperature data. As you can see, it's pretty regular between 150-130 degrees. However, the exact value in that range fluctuates pretty widely.

If we take a look at hopper data. Again, it's very regular in range and it's also regular in the pattern within that range.

So, as I already mentioned about the phenomena of increase in dimensionality, we can plot this in two-dimensional space and see that as very regular within this rectangular area. With all that taken into account, we have completed the next step of data visualization and analytics.

Following that step in the machine learning process, is to actually train and evaluate your machine learning model. As I mentioned, we use the MLOps framework to do this, and specifically, we use AWS SageMaker. So, I'm going to go navigate to SageMaker right now and I will navigate to models.

What you're seeing here is the model that was output by the ML Ops framework. So, this model has been trained on our temperature and hopper data and is the K and N Algorithm that we were looking at before. Awesome, we have a model, but how do we actually use it?

Well, including the ML Ops framework, we have spun up a SageMaker hosting endpoint, which is a 24/7 REST API endpoint that allows you to do real-time inference against your machine learning model. Excellent. Now we have a machine learning model and a way to interact with it.

The next step is to actually get the new incoming IoT SiteWise data and to get it to leverage against the hosted endpoint. To do this, we have created a simple pipeline to absorb SiteWise data in real time and leverage it against our endpoint. So, I'm going to the logs now to give you some insight into that pipeline, and as you can see, this is the incoming records being leveraged against our model in real time.

What you're looking at the first value is the timestamp, followed by the temperature value, the hopper value, and a true or false value as to whether or not the incoming record was detected as anomalous.

With that being said, what is the next step in the machine learning process? Well, we want to

evaluate whether or not our business question has been answered. So to do this, we need to actually inject an anomalous record and see if the model picks it up. We've written a simple script to inject a single record that is anomalous into our IoT SiteWise data. I'm going to give it a temperature of 100 degrees and a hopper value of 120.

Now as that data is processed, we should get a notification that an anomalous record has been identified. And sure enough, I've just received an email that there has been an anomaly detection.

I can refresh the logs and we should also see in the pipeline logs that an anomalous record has come in and been detected as anonymous. And sure enough, in the logs, is just that. An anomalous record that was detected.

I'm going to go back to the slides, really quick. I threw a lot at everyone right there. So I kind of want to just sum it up really quick. What did we just see?

At a very high level, it is this diagram and workflow seen on the left. Step one: we absorbed IoT data to train a machine learning model in AWS SageMaker. Step two: new incoming records were processed by a Lambda function in a pipeline in real time. Step three: That Lambda function took those incoming records and leveraged them against our trained machine learning model. Step four: If an anomaly was detected, it sent an email to the user notifying him.

At a high level, we integrated the ML Ops framework with our IoT data to test, train, and host a machine learning model. We leveraged incoming data against the model to detect anomalies in real time.

My final point that I'll leave you with is, what I have been saying the theme throughout the entirety of this demo. And that is, yes, this is a specific use case, but I think if you take a step back, you get the higher levels of how you can leverage your own IoT data against your own machine learning model to provide yourself business value.

Thank you for your time, and with that, I will hand it back to Scott.

Outro

45:28

Scott Fulnecky: Awesome, thanks for that demo Brenden, and thanks Carlos, and Thomas as well. I'll wrap things up here.

I'll just share my screen and wind down the slide deck that we started back landing on the POV itself and what challenges we're trying to solve against competitors in the manufacturing space.

When they're looking at their competitors, they're saying that they're always poised to do things better, faster, cheaper. Manufacturers are really always challenged to continuously innovate and remain competitive and with data being locked on the factory floor, that can be tough. So, operational teams can lack the visibility and additional insight into what's happening, both for the people on the factory floor, and in management. IT teams lack the data coming from that factory floor to make the predictions on where those various failures can happen and help the OT teams with that predictive maintenance and being more proactive with scheduling downtime, rather than reactive to breaks.

Also, when you think about employees and as the old guard is retiring from these factories, their knowledge goes with them. But as new groups of young people that are interested in the new technology and new ways of manufacturing are coming in, there's a need to capture that talent and their interest. This is definitely one of the ways to do that. And so the solution is, implementing the POV to connect to that operational data, unlock it, be able to utilize that data, and reduce the uncertainty by seeing more of what's happening on the plant floor and coming up with new information using that data, and the machine learning provides that prediction. That's that key ingredient for decision making under that uncertainty.

With this new visibility into that IoT data, what comes up is new opportunities even beyond that like inventory management, logistics planning, and others. Some of the benefits that you're going to get would be gaining visibility into that operational data for root cause analysis (RCA) and being able to streamline and enhance processes within the IT systems by exposing that data to additional systems like the ERP system, or manufacturing execution, or warehouse management systems for other business units and for the IT team to use. Through access to this data, and implementing predictive maintenance, and optimizing preventative maintenance to avoid lower quality, product contamination, and broken machinery, we're looking at lower downtime, decreased costs, and decreased unplanned maintenance events, being able to automate and accurately predict failure
When one breakdown can affect the whole supply chain, it stops production, packaging, shipping. Now you've got that line of sight and the 360-degree picture of what's happening that starts at the factory floor.

And so with the Connected Factory POV, the deliverables that we're looking at are, the physical deployment on-site, that edge device. The industrial POC will be deployed with Ignition and Greengrass and the SiteWise connector to pull that data from the PLCs, push it to AWS. Once it's in AWS, you're connected in with SiteWise, where you get your single pane of glass overview of the operational data on the dashboards and those monitors on the dashboards that Thomas and Carlos spoke to.

You also get that asset creation and asset model setup that shows the hierarchy and gives the visual representation of the machines on the plant floor. Then it's also going to come with sample alerting for three of the assets on the factory floor.

For the machine learning component, that POC inside the connected factory engagement. Essentially after collecting a few months of data, we would then have the historical data that will be fed into that machine learning process and the POC model is really designed to showcase the possibilities of how your IoT data can deliver the business value.

Obviously, due to the disparate nature of that data that would need to be collected and used to address the wide range of problems and KPIs that would come from that business problem, and the disparate number of sensors and outputs, the time required to collect that data is subject to change, and scope is limited to a POC model for the machine learning aspect of this.

If we're not able to address all of the questions today I would like to give you the ability to reach out to iot@trek10.com to get a conversation started about what's next, or just to address any questions that we don't get to today.

Again, I'd like to thank Thomas for joining us and Brenden and Carlos, and for everybody that attended today as well.

Thomas Cummins: Thanks Scott. Thanks, Carlos. Thanks everybody.

Carlos Lemus: Thanks, everyone.

Scott Fulnecky: Thanks, Thomas. Take care everyone