

05192020 Build Azure Scott Guthrie

**Build 2020**  
**Azure**  
**Scott Guthrie**  
**Monday, May 19, 2020**

**SCOTT GUTHRIE (EVP, Cloud & AI):** Good morning, everyone, and welcome to Build 2020. We have some great content and announcements to share today, and I'm going to be joined by some amazing speakers.

Our mission at Microsoft is to empower every person and organization on the planet to achieve more, and our goal is to empower developers with the platforms and tools that will make you incredibly successful.

In this presentation we're going to walk through and demo our latest development tools and cloud platform innovations.

Developers are the driver of innovation in the world we live in today, and this is especially true now as the world has had to go even more online and as every organization is having to rethink and rework their business.

Developers are more critical than ever before. With Azure, we've built an end-to-end cloud that enables developers to focus on building apps and not have to worry about the underlying infrastructure the apps run on.

Azure is open and it supports your favorite development language, open source framework, or tool. Azure also makes it easy for you to learn new skills and grow your career.

Every developer is welcome with Azure, and over the last few years, we've seen developers across the globe invent, build and deploy some truly amazing solutions using Azure.

Over 95% of the world's largest companies are running their businesses using Azure today, and throughout the Build event, many of our customers will share their stories about how Azure is helping them achieve business success, and how we are partnering with them during these challenging times.

We've also been rapidly building out our Azure regions. We now have 61 Azure datacenter regions around the world. This allows you to run your apps close to your employees and customers wherever they are. It also helps you meet local data residency requirements.

We recently added new regions in Israel, Mexico, Spain, Italy, Poland and New Zealand, and we're building out all of this cloud infrastructure in a sustainable way.

Every organization is affected by COVID-19. Microsoft is working closely with thousands of organizations around the world to help them use the Microsoft cloud to address today's challenges.

Businesses and organizations of all sizes are adopting solutions to enable remote work, while staying secure, and creating brand new applications because of the pandemic.

Swedish Health Services, one of the largest nonprofit health providers in Washington state, built an emergency response solution using Azure, Power BI and Power Apps. It allows them to track and manage real-time insights of bed and staff capacity, COVID patients' statistics, discharge pipeline, and even see supplies on hand like masks and gloves and the burn rate on critical supplies. The source code for this solution has been published as open source on GitHub, and it's been adopted by healthcare providers around the world.

We're also helping organizations around the world to enable their employees to work remotely in a secure way with Microsoft 365 and Azure.

Microsoft 365, which runs on Azure, is playing a critical role in helping organizations stay connected through video meetings, calls and chats, and Teams has experienced a significant spike in usage and now has over 75 million daily active users and over 4 billion meeting minutes per day. It is a cloud-native application built on Azure, and the fact that it was built on Azure was critical in enabling them to handle the sudden hockey stick of growth.

Teams is built using a microservice-based architecture that takes advantage of technologies like Azure Kubernetes Service and Azure Service Fabric, and this enabled them to dramatically scale up to millions of compute cores as usage increased because of COVID.

Teams also uses platform services like Azure Cosmos DB and Azure Media Services and uses a fully automated DevOps approach using Azure DevOps. What's great is that all of these services are broadly available for you to use as well. Using Azure, you can build cloud-native solutions that can scale to literally any business need.

With Azure you have access to an end-to-end cloud platform, with over 100 highly engineered platform services that provide you with building blocks and a secure foundation for all your applications. And in the rest of this presentation, we're going to go deeper on a number of these services and how you can use them to create great apps of their own. I'll be joined by a few guest speakers who will cover developing for the cloud, cloud-native applications, analytics and machine learning.

And to start, I'd like to welcome Amanda Silver, who'll be joining me today to talk about her latest innovation for developing for the cloud. Here's Amanda.

**AMANDA SILVER (CVP, Product Developer Tools, Visual Studio):** Thank you, Scott. I'm Amanda Silver, the CVP of Product for the Visual Studio family and our developer experiences. I'm delighted to be able to join you today from my home office. I want to express gratitude to the developers that built the apps and infrastructure that connect us today all over the world.

Now let's talk about developing for the cloud. The developer experience for Azure brings best-in-class tools, SDKs in every programming language, and a platform to support continuous collaboration and delivery.

Developers building modern apps all face common challenges. Every team needs to ship faster and iterate with confidence. Agility is top of mind. We all focus on faster time to market and shorter lead times to bootstrap new projects. With a more rapid cadence, we still need to ship confidently and securely.

Let's face it, we work in one of the most dynamic industries. We're learning all the time. And that's why we love it. We're constantly expected to master new technologies. Whether it's new languages, new frameworks, or new cloud services architectures, we're dealing with a ton of complexity and novelty. Now, more than ever, we need to collaborate with colleagues down the hall, down the road, or on the other side of the planet.

Azure has an end-to-end developer experience that helps you create reliable, global and secure applications faster. You can use your favorite languages, open source frameworks, and tools that make you feel at home.

Azure provides services that enable developers to focus on building applications, without having to worry about infrastructure. You can build, test and deploy from your code to the cloud with Azure tools and GitHub actions. Our tools have integrated collaboration to help you and your team master new skills together.

The app I'm about to show you is based on a common app architecture for modern cloud apps, a Static Web App, aka a SWA, backed by some serverless API. I used pre-built services in Azure to complete authentication, video transcription, geospatial services and storage, and plugged them into my app as Azure functions. The result is a geo-distributed, serverless web application.

I can quickly scaffold this Azure Static Web App in the portal, initialized from a Git repo, and that then creates a series of GitHub actions to set up continuous integration and continuous deployment for me.

If a new developer wants to contribute to the app, with Azure they can get a dev box set up in seconds and access it from any device.

Let me show you how my team and I are collaborating remotely to build a cool new Azure Static Web App. Let's start with Donovan.

**DONOVAN BROWN (Principal Cloud Advocate):** Thanks, Amanda.

This is Sunrise Standup. It's the app that we use because our team is globally distributed and working remotely, so that we can see what everyone's working on. You can record a short video and tell everyone what you've been up to.

**THOMAS MAURER:** Hey, Thomas here. Everything is going well in Switzerland.

**DONOVAN BROWN:** All this code is sitting inside of a GitHub repository. My goal, get the code out of this repository and into an Azure Static Web App. To do that, I'm going to visit the Azure portal, click on Create Resource and Search for Static Web App. Now, I'll simply click on this tile, click on Create, and then fill in the form.

I want to create a new resource group that I'm going to call Sunrise. I'm going to call the app Sunrise as well. I pick my region, and then I sign in with my GitHub. Signing in with GitHub is going to give Azure access to all my organizations, allow me to select my repository, and then from my repository select my branch. Now I can go ahead and click on Create.

In the background, Azure and GitHub are working together to make sure that I have a GitHub Actions pipeline. GitHub Actions is a workflow automation engine that's built in to GitHub that allows me to create CI/CD pipelines.

CI, or continuous integration, is the practice upon which every commit to the repository, the code is downloaded, compiled, tested and packaged for deployment. Continuous delivery, or CD, is the practice of taking that package and safely deploying into the resources inside of Azure.

If I go over here and click on resources, we can see that there's a URL we can click, and also that GitHub Actions have already been configured and are running in the background.

Now what I'm going to do is simply click on this URL and navigate to our application that is now running in the cloud. The GitHub Actions have run and safely deployed our code into production.

One other thing I wanted to point out was we also have this map feature. Now, it's not complete yet, but what we're going to do is we're going to have Amanda show you just how easy it is to add new functionality to this map by using Azure Static Web Apps.

**AMANDA SILVER:** So Donovan just reminded me that I need to add pins to our Azure maps so that we can visualize our global team. Now, what that means is that I need to add another endpoint to this Azure Static Web Application.

Now, what you can see here is that I already have the Azure Static Web App in my Visual Studio Code instance, and everything's defined by convention. So I have all of the APIs under this API folder, GetGitHubAvatar, GetSASToken, GetVideos, and as you can see, I'm already running everything locally on my machine, so that I can iterate really quickly as a developer. So we have GetVideos, GetGitHubAvatar and GetSASToken, and you can see that the Azure Functions application is up and running, and I can actually run the Sunrise Standup app right here. I can go ahead and play this.

**BRIAN BENDS:** Greetings, everyone. Brian Bends here in Las Vegas.

**AMANDA SILVER:** All right, so he's in Las Vegas. Now, let's go see if we can see him on the map. Unfortunately, no dice. There are no pins on the map. So we need to fix this. Let's go back to our Visual Studio Code instance.

All right, so what you can see is that I have this application in Visual Studio Code, but what's really cool is that if I install the Azure Static Web Apps extension into Visual Studio Code, I get some really key functionality in a few important spots.

Now, one of the areas that I get additional functionality is if I go into the Azure viewlet, you can see that I have this Static Web Apps viewlet right here. Now I need to add the additional API endpoint. So all I need to do is to click on this lightning button. Now, it's a lightning button maybe because it happens really fast, so pay attention. I just need to create a new endpoint, GetLocations for the name, and you can see that it actually scaffolds out a new function for me right here. And it does everything on the back end as well. So all I need to do then is just replace this with the code that I want to write.

Now that we have that code written, we want to see if it actually works. Let's go back and look at another spot that the extension adds some pretty cool functionality.

In addition to being able to launch the node functions, I can also launch a full stack application with the Azure Static Web Apps application, which launches both the backend with the functions, as well as the frontend for the full stack app.

I'm just going to switch to that and we're going to start debugging. And as you can see, it's going to actually kick off a new debug process. It's going to run NPM install. It's basically going to get the Azure Functions runtime up and running super, super quick. Cool. So now we actually have the Azure Functions running right here.

So now let's go look at the app again and see if it's working. So now we can see that the pins are actually loaded and we could see Mr. Bends from Las Vegas. But what's really cool is we can visualize our global team.

But even better, this app is already globally distributed without me having to do anything. Just using Azure Static Web Apps, it's actually been globally distributed all over the world, so everybody on the team gets super-quick access to the application.

Now I just need to get this fix to Donovan. Let's go back, and then I can use the GitHub Pull Request extension to create a new pull request from right inside Visual Studio Code. I can just go ahead and create a new pull request. It will automatically be populated by my commit. I can choose master, add map endpoint, and that will create a new pull request.

Now, all I need to do is just add Donovan as one of the reviewers to this pull request, click OK, and he should get the message.

**DONOVAN BROWN:** Where does this part even go? I just hope I can find it on the internet.

Even better, a notification from work. This is a pull request notification coming from Microsoft Teams, thanks to the GitHub app. As we can see, Amanda completed her changes, and I can click on this link and jump right to them inside of GitHub. A PR allows me to review the code before I merge it into master.

Reading the code is one thing, but running the code, that's even better. But I don't have all the tools that I need on this machine.

Luckily, Azure Static Web Apps has me covered. I simply click on the link that's automatically added to every pull request that allows me to see the code actually running in a staging environment. And as we can see, Amanda did a fantastic job. We can see the pins. We hover over the pins. We can see the avatars of the users.

The one thing I wish I could change is the theme. Here in my garage, it's really difficult to look at this light theme, but a dark theme, that would be better. But again, I don't want to run upstairs. I don't want to have to get cleaned up. I just want to do it right here off of this machine.

And luckily, CodeSpaces turns any web browser into a development machine, even this old machine that I use in the garage. It has no development tools on it and I installed Linux on it just for fun. But I can actually use CodeSpaces to develop.

Clicking on this link is going to spin up an entire development environment for me in the cloud, far more powerful than the machine that I'm holding. It's going to have Visual Studio Code on it. I can install all my favorite extensions, and all the libraries and frameworks I need will automatically be installed.

Now that my environment is spun up, I can do what I would normally do. I can open the file that I want to modify. I can go ahead and import the dark theme. And just like I would if I were upstairs, I can simply click on Debug. Doing that is going to spin up all the functions, which is the backend that we talked about earlier, and the changes that Amanda just made. Then I can click on this link that's going to allow me to see my changes right here in real time. I could set breakpoints, I could step through the code, but as we can see, the dark theme was applied correctly. I can commit this code right here from the web browser and go about my business trying to fix my car.

When you combine the power of Azure Static Web Apps, GitHub Actions and CodeSpaces, you turn every device you're holding into a development machine.

Thank you so much, everyone.

**AMANDA SILVER:** Today, I'm excited to announce Azure Static Web Apps. Modern, single-page applications don't rely on traditional web server frameworks. They deliver Static Web Apps with less compute, using serverless APIs. These apps can be built using frameworks like Angular, React and Vue or static site generators like Gatsby.

All you need to do is check in your code, build and deploy, code to cloud. With minimal Azure configuration, your app is deployed globally in minutes, and you can optionally configure routing, authentication and custom domains.

Azure has an end-to-end developer experience that makes it effortless to go from your idea to code to the cloud.

Back to you, Scott.

**SCOTT GUTHRIE:** Thanks, Amanda.

As Amanda showed, one of the unique things we're trying to do is to really provide you with developer tools that make you incredibly productive when building Azure-based solutions. As part of that, we're also delivering great cloud-native services that enable you to focus on developing your apps versus spending your time on the infrastructure below them.

I'd like to invite Jeff Hollan now to talk about building cloud-native applications on Azure. Here's Jeff.

**JEFF HOLLAN (Principal Program Manager):** Thanks, Scott. I'm Jeff Hollan, principal program manager for Azure Serverless, and I am thrilled to be with each of you here today at Build from my home office.

Let's talk some more about Azure. Azure provides a number of services to enable developers to focus on their apps and not the infrastructure. We're excited to share that everything from Kubernetes to AI, we have over a thousand new features and improvements. We also have unique end-to-end solutions so that you can partner with nontraditional developers to harness talent across your organization in building these apps.

Let's start with Kubernetes. Azure Kubernetes service is enterprise grade by design. It's a fully managed Kubernetes offering to build, scale and run your deployments.

We've recently announced a number of improvements to AKS. Windows container support is now generally available so you can bring existing Windows applications, lift and shift them to run in a fully managed Kubernetes service in Azure, the best cloud for .NET, so these applications get the benefits of scalability, portability and self-healing that Kubernetes provides.

Private Link allows you to communicate with the Kubernetes APIs, all within your private virtual network, and Spot Pricing provides cost benefits for workloads like batch processing or machine learning that can tolerate some interruptions.

Now, just as important as the compute is the data for your apps, and Azure has a number of flexible options for you to choose the right database. There's fully managed versions of Microsoft SQL, MySQL, MariaDB, Redis, Cosmos DB, and more, each of these fully scalable and backed by an SLA. This gives you the productivity that you need, but the freedom to leverage the tools your team prefers.

Today, we're announcing more cost-effective options for running Cosmos DB. The serverless capabilities coming soon enable you to efficiently run your workloads, even with spiky workloads. You can use the auto-scale feature to have your database instantly scale, but still get that single digit millisecond latency and our 99.999% availability SLAs. Using auto-scale with the new Cosmos DB free tier, your first 400 RUs of throughput and 5 gigabytes of storage are completely free.

Now moving to AI, AI is opening up new scenarios for every type of application, and any developer with any skillset can start to use AI with Cognitive Services. Cognitive Services provides the most complete options for models ranging from vision, speech, language and more. In just a few lines of code, you can start to access these models, no machine learning experience required. Each of these models is customizable, so you can fine tune it based on your data and your application.

We continue to innovate with Cognitive Services. Personalizer is the only reinforcement learning-based AI rationalization service.

Today, we're announcing a new capability, Apprentice Mode, so that you can run it alongside your existing recommendation solution, providing high accuracy results from the moment you deploy the service.

Recently we announced new voice options for Cognitive Services, newscast, customer service and digital assistant, to enable developers to tailor the voice of their apps to meet their unique brand and their unique scenarios.

Finally, you can run Cognitive Services anywhere using containers, and today, we're expanding that support with general availability of text analytics and language understanding.

Our tools enable everyone to be a part of the developer community and accomplishing tasks like building microservices and APIs and using low-code platforms.

As you know, Azure has a number of options to write applications using any language. But often one of the best ways to surface that information to end users is by pairing it with the Power Platform.

The combination of Azure, the Power Platform and Dynamics 365 is incredibly unique. It lets developers focus on those high-value tasks and accelerates the delivery of these cloud-based applications.

Power Apps is our low-code platform for developers and citizen developers to build rich applications quickly and easily. Power Apps is a low-code environment to write apps for both web and mobile.

I want to show you how we can use this to build an application, and we're going to build an app inspired by how we've seen our users leveraging cloud-native tech as part of the COVID-19 response.

The mobile application was built using Power Apps, and it's calling data and services that's running in Azure. I'm using those tools that I love that Donovan and Amanda showed us earlier today to build this application.

I'm showing this application now on my mobile device. This is an app for screening at different points of entry around the world. I could come in and enter the details or the symptoms of a traveler at this point of entry, or if they filled out a form ahead of time, I'll actually select this scan button and I can go ahead and scan a form. After I select to use this photo, it's now scanning this image and automatically filling in each of the fields here. Now when I select submit, all of the text was filled in without me having to do any typing.

Let's show how we built this experience. We're going to start here in the Azure portal, and every application needs data. For us, we're going to be using Azure Cosmos DB. Cosmos DB is perfect because it auto-scales to trillions of events every single second. I can already see some of the data here that we've uploaded to our database. For instance, I can navigate to screenings and see those details here. I'm even using the new Cosmos DB free tier, so I'm storing and retrieving this data at no cost.

Now that we have our data here in the cloud, I want to be able to access it through an API. If I come back over here to Visual Studio, I can create a brand new ASP.NET app.

I'm going to select the Web API project. Now, I need to get this API connected to that Cosmos DB database, and the new connected services experiences is just what I need.

I'll go ahead and select to add a dependency here. I'll select to add Cosmos DB. Visual Studio is connected to my Azure account, so I'll select my database. It's pulling in the connection details for me. I have the simple configuration option to choose to either store these details locally in a secrets file or securely encrypt them within Azure Key Vault, so they don't even stay on my machine.

Now, once I'm connected to Cosmos DB, we can go ahead and start to write some code. You can see here this is an example of one of our API controllers. This is for screening data so I can get screenings either by ID or all of them. I'm actually using the .NET entity framework with the Cosmos DB SDK, so I have something here like a link statement.

Now, I want to be able to test these statements, and I want to try this API, but testing APIs can be cumbersome, because I don't always have a UI to drive the experience. I'm fiddling with CLIs or other tools to try to navigate.

But I want to showcase this pretty cool new tool right inside of Visual Studio and .NET called HttpRepl. What HttpRepl allows me to do is browse and debug my API the same way I would navigate a folder structure.

So I'll start a debug session. It's going to open up this console window. You'll notice instead of being connected to the C drive, I'm connected to local host. I can now navigate using Dir, or if I want, I can type LS. It's going to return back to me the different endpoints, including our screening data that we were just looking at. Let's navigate to it and even do a get. It's now executing my code and returning that data from Cosmos DB, now within my Visual Studio project.

We have some code working, and I want to now deploy this to run in the cloud. My first step is going to be checking this into GitHub. Let's add a quick commit message here. I'll push this up to GitHub and navigate over to our repo.

This is just a regular GitHub repo, but I have a GitHub Action that is kicked off and is running now. This is just like the action that Donovan showed you earlier today. For my project it's taking the .NET API, it's containerizing it inside of Docker. I'm going to publish that container to Azure's Container Registry, and finally deploy it to run in the Azure Kubernetes service. Now my code can run across incredibly elastic compute, so I'm confident that this API can scale to whatever demands my mobile application might have.

My app is now running inside of Kubernetes, but I want to build some experiences around it. I need to put it in the hands of my health screening professionals. Rather than having to code this application manually by hand to target all the various devices my team might be using, I'm going to leverage Power Apps to quickly surface this data in an experience.

You can see here in the Power App Studio I can visually create this mobile application in just a few hours. I have access to over 300 connectors out of the box or I can use our own API, the screening API now running in Kubernetes.

Let's go ahead and add some information to this screen. I'll come in and insert a vertical gallery. I can even go ahead and choose the data source. Let's choose that screening data that we've been looking at so far. I can move the table down, go ahead and resize it.

I can customize the look and feel by choosing the layout. I like this title layout right here, so we'll go ahead and select that. I even have the option to choose different fields to surface. Because I published this API through Azure, it knows, hey, I have things like contact number and visitor name from this API. I'll go ahead and select it to show the visitor name and contact number. We've customized our form to show exactly how we want.

Now, this is a full mobile app. It could run across Windows, Android, any device. You can see here. I have a number of screens here, including our main screen and that screening data form.

I want to show you how we hooked this form up to those AI capabilities that let me scan a picture. I did that by navigating here, still in the Power Apps experience, to enhance my app with AI.

I have a number of capabilities at my disposal like object detection. I want to go ahead and choose form processing. We'll give our model a name. It's going to let us upload a few sample images. After I upload a few samples, it's going to create and train a machine learning model for me. Without having to write any code, I can now come back over here to Power Apps and tell it, hey, if someone clicks the scan button, go ahead, call that forms processing model. I can now map this to those different fields and do all the typing for me.

So our app looks pretty good now, and I just need to publish it to get in the hands of our users, our team, just like the 75 million others out there who have been using Microsoft Teams heavily during the COVID-19 crisis.

With new capabilities, I can now go ahead and choose to take any Power App and instantly add it to my team. By clicking this button, now my application has been deployed to whatever teams, whatever channels or my entire organization. Now, everyone using this Teams app has access to it, instantly deployed.

Let's open up my phone again. I can select to see the apps I have here, and there's that COVID screening app, integrated in the Teams experience, in this case, on my mobile device. I can go ahead and open it up and I could start to conduct those screenings.

You can see by using cloud-native tech, I was able to build this application very efficiently, both in terms of cost and in effort.

Only Microsoft provides a solution that spans from the power of Kubernetes to the collaboration of Microsoft Teams, all in a single cloud-native platform.

Thank you so much. Scott, back to you.

**SCOTT GUTHRIE:** Thanks, Jeff.

Now, Starbucks is a great example of a customer who is using the combination of Azure, Microsoft 365, Dynamics 365, Power BI and Teams to drive solutions across their business.

To share more about some of the great work that Starbucks is doing, I like to invite Julia White from Microsoft and Gerri Martin-Flickinger, executive vice president and chief technology officer from Starbucks.

**JULIA WHITE (CVP, Microsoft Azure):** Thank you so much, Scott.

Now, Gerri, great to be here today. I of course have my Starbucks ready to go.

**GERRI MARTIN-FLICKINGER (EVP, CTO, Starbucks):** And do I.

**JULIA WHITE:** Perfect.

Now, Starbucks has been on a journey with technology around their transformation and, gosh, starting with Office 365 back in 2015, and then adding things like Power BI and GitHub, and then even more recently, Azure Sphere. I'd love to understand what that journey has looked like for Starbucks?

**GERRI MARTIN-FLICKINGER:** It really has been a transformation. The first is helping our partners grow, and partner is what we call all of our employees. We've done this by ensuring that we're actually promoting intellectual curiosity and embracing a growth mindset.

Every year, we have a number of learning journeys for all of our partners, including something we call the Starbucks Innovation Expo. This past year, both Satya, as well as Brad Smith, came and spoke to our partners, and it was so encouraging and enlightening and really brought a new perspective on the external tech market.

We have really evolved from an organization that focused heavily on buying and installing software to one that also builds and thinks about modern techniques like microservices and CI/CD pipelines. We leverage the cloud extensively for scalability and flexibility. We're using AI/ML for hyper-personalization in many of our customer digital properties.

We are really thoughtful about making sure that our architecture is going to carry us into the future and give us the kind of agility and ability to innovate quickly that we need. It's been a great transformation journey so far.

Thank you, Julia.

**JULIA WHITE:** Thank you, Gerri, for joining us, and thanks for sharing your perspective and your learnings.

With that, I'll hand back to Scott.

**SCOTT GUTHRIE:** Great. Thank you, Julia and Gerri.

It's great to see how Starbucks is rapidly evolving to the changing needs of their employees and customers using Microsoft technologies.

Now, as part of that change, every organization is looking to use data and AI to run their business more effectively, and with Microsoft Azure, we provide a rich set of services and capabilities that make it easy for you to do this.

To talk about our latest innovations here, I like to welcome Rohan Kumar.

**ROHAN KUMAR (CVP, Engineering, Azure Data):** Thank you, Scott.

Hello, everyone. I'm Rohan Kumar, corporate vice president, engineering of Azure Data. I'm really excited to be here joining the keynote from my home office.

It is becoming increasingly clear that predictive analytics using AI and machine learning is a critical element in every enterprise customer's digital transformation journey.

Let's start by looking at what's new in Azure Analytics and machine learning and how it can help you build your solutions much easier and faster than ever before.

Synapse significantly reduces the complexity of building and operating end-to-end analytical solutions by enabling a simple, intuitive visual, no-code experience. We've deeply integrated Synapse with Power BI and Azure Machine Learning, making them an industry-leading combination to build predictive analytics solutions. Customers can continue running their existing data warehouse workloads in production today with Azure Synapse and will automatically benefit from the new capabilities which are in preview.

While we've taken a huge leap forward with Synapse, the barrier between operational databases and analytical systems has been difficult to overcome. Current solutions involve creating ETL pipelines, which are extremely complex, very expensive and challenging to maintain. These pipelines often cost significant delay of hours, days, or even weeks, resulting in decisions being made on stale data.

Which is why we're very excited to announce Azure Synapse Link. Synapse Link is the industry's first cloud-native implementation of hybrid transaction analytical processing.

Synapse Link links your Azure operational databases to Synapse, providing the ability to get immediate insights for your business. In fact, as soon as a transaction is recorded in your operational database, it will be available in columnar format for analytics in mere seconds.

And the best part, all it takes is a click of a button and we'll handle all the work for you. No need to build and manage extremely complex and expensive ETL pipelines or worry about negative impact on the performance of your transactions.

Instead of understanding what happened yesterday or last week, Synapse Link enables you to understand what is happening now, providing clarity in an instant. And in these uncertain times, making decisions based on the now is very critical as businesses must react to sudden changes. Available in Cosmos DB today, we will get this to other Azure operational databases over time.

Let's see Synapse Link in action through a demo. For the purpose of this demo, please imagine that I'm a data engineer at a large automotive company. We have multiple factory locations worldwide sending data from their production lines. I'm accountable to author and maintain a KPI dashboard using Power BI for our business users.

In the past, we've been never up to date with what's happening on the factory floor, which resulted in several fire drills. We address that challenge using Synapse Link. Let us see how we made that happen.

We deal with a very high volume of data and have extreme low latency requirements. Since our manufacturing plants are in multiple regions worldwide, support for features like georeplication

is essential for us to build highly responsive applications and workloads. Azure Cosmos DB is the perfect operational database, given our requirements.

Now, to avoid any disruptions to our supply chain, we need the database to serve our parts catalog and ingest input feeds from our factory floor in real time, 24/7, globally with zero downtime. We've achieved five nines of availability with Cosmos DB using the multi-region, multi-master writes.

Now, as you can see on the screen, my Cosmos DB database is ingesting millions of transactions per second and is able to serve data with single-digit millisecond latency.

Now my Cosmos DB database also happens to be the source repository for our failure reports. Let me show you using the Data Explorer view what the documents look like inside the database. Let me go ahead and click the supply chain database here and pick the quality assurance folder and items.

Now, as you'll see on the screen in a short while, there is a JSON document for every quality desk report for all our manufactured items. It contains all attributes of the report, including the part ID, description and the test result. This is exactly the data that we need realtime access to, to enable immediate insights on failure rates.

Welcome Synapse Link. With a single click, I can enable Synapse Link on my Cosmos DB account and select the Synapse workspace, an existing one, where my analytical queries will be run.

Now, behind the scenes, Synapse Link created a live, optimized, columnar representation of the Cosmos DB's operational data and made it available to Synapse without impacting the performance of Cosmos DB's transactions. I did not have to spend any time or resources building highly complex, very expensive and hard to maintain ETL pipelines. It just works, which is awesome.

Now, let me go to my Synapse workspace and query the live Cosmos DB data using the Synapse Link that we just created.

Here in my Synapse workspace, I can bring together all of my enterprise data integration, data warehousing and big data analytics into a single, unified experience for my immediate BI and machine learning needs. Let me enter the Data Hub.

Using the Data Hub in Synapse Studio, I can access all my data that's being managed by Synapse, including files, tables, data sets, et cetera. I can also access the real-time data that's enabled through Synapse Link from Cosmos DB. With a simple right click. I can create a new SQL script or create a notebook on-the-spot job on my live Cosmos DB data.

Let me go ahead and bring up a SQL script that I wrote earlier that essentially gets real-time failure rates from the Cosmos DB deed.

You'll notice that when I rerun the script, the timestamps get updated. The view that you see in the result set is not from last week or yesterday or even the last hour. The view that you see is from what's happening right now, which is really cool.

We've deeply integrated Synapse with Power BI, which enables me to create Power BI reports and enterprise-grade semantic models right from within the Synapse workspace.

This is the dashboard that's being backed by the Synapse SQL query that you just saw to get the failure rates. Let me go ahead and refresh this Power BI report board to get real-time insights. Our business users can rely on this dashboard to deal with incidents as they happen in real time.

As you can see on the screen, the failure rate for some of the parts is higher than the expected threshold. Let's figure out how to go handle this problem. Using the data flows in Synapse, I've already ingested the stock levels and prices from all our suppliers in our SAP system, and supplier contracts and delivery terms from our Dynamics 365 system.

To mitigate the current situation, we need to find a supplier to source the parts whose failure rate is above the acceptable threshold. To predict the next best supplier, I need to join the Synapse Link-enabled live data from Cosmos DB, which has the list of the bad parts, with the supplier data from Dynamics 365, and the stock levels and the pricing data from SAP, and run a predictive analytics query across the combined dataset. The SQL query that you see on the screen captures this logic.

One of the things that I wanted to showcase is the use of the predict function in the SQL query, which allows us to apply a model that was created in Azure Machine Learning to score against the combined dataset. Let me go ahead and run this query. In the results section in a few seconds, you're going to see a ranked order list of suppliers that meet our requirements.

You know, I'm not a data scientist and Synapse made it really easy for me to do predictive analytics by leveraging the model that was created in Azure Machine Learning to get the results, the predictive analytic results that I want.

Let's go ahead and go back to our Power BI report and add these predictive insights to this report, so we can share them with our business users. Doing that is really simple, just a few little clicks to add the insights that you want. I go ahead and click here and get the high value, and that's it. When I go ahead and click save, these new predictive insights have been added to the Power BI report and shared with all the business users. Now they can go to Power BI Online and double click to understand more details of how the prediction was made, what elements of the machine learning model influenced the prediction decision. It gives them a little bit more insights, deeper insights into what happened behind the scenes.

Some of you may be wondering, how did I create the model that I used in the predict function? Here I am in the new Azure Machine Learning Studio. It is designed to be a self-contained environment for all your data science needs, from model creation, packaging and deployment, through the monitoring and management.

Azure Machine Learning has tools for all skill levels. It has notebooks for a code-first experience, a designer for a drag and drop experience, and my personal favorite, automated machine learning click-through wizard.

Let me show you how someone like me, who is not a data scientist, was able to create a state of art machine learning model to predict the next best supplier.

Let me start by creating a new automated ML run. The very first thing I have to do is pick a dataset to train my model against. Given the deep integration that we've done between Synapse and AML, the dataset over here that I've processed in Synapse is automatically available to me in the auto ML workflow. Let me go ahead and select that and click next.

There's a few pieces of information that I need to provide, starting with the experiment name. Let me pick an already existing experiment that I have ongoing, the target column to score on, which is the forecast delivery days here, and then finally a compute environment for auto ML to run the training on. That's it.

I go ahead and click next, and finally, I pick the task type as classification because I'm dealing with a limited set of suppliers and click finish.

Behind the scenes, auto ML is leveraging the compute environment that I provided to train the machine learning model that best suits the requirements that I have. This process is going to take a while.

Let me show you how easy it is to deploy a model once it gets built. Let me start by clicking the experiments tab here, and over here you'll see the active experiment that I have ongoing. Let me go ahead and select that.

Now, in this dashboard over here, you'll see a lot of information about the experiment, the number of models that have already been created, the active one, you see Run 221 is the one that I just started. There's information around things like the accuracy of the model, et cetera.

Now, let me go ahead and back Run 166, which is a model that has been recently completed, to show you how easy it is to complete the deployment. From here, I just select the algorithm name and that's it. Here on the screen where I can just click the deploy button and then this model which I have built automatically gets available for Synapse to run SQL queries, start queries on. That's it. It's that simple.

As organizations look to adopt AI, they face a major challenge in developing and using AI responsibly. Customers want to understand the inner workings of their AI models.

To address these concerns, we are announcing several new capabilities in Azure Machine Learning. These are designed to empower data scientists and developers to understand, protect and control data and models throughout the machine learning lifecycle.

With these new capabilities, they can: understand model behavior by getting explanations of how models work, add both training and inferencing, as well as assessing fairness and mitigating bias; protect through differential privacy compute capabilities to build models using sensitive data, while safeguarding the privacy of individuals; control by tracking the entire machine learning process and maintaining an audit trail of all the machine learning assets to meet regulatory and compliance requirements.

This journey is just the beginning, and I cannot wait to share more with you soon. We can't wait to see what you build with all this amazing innovation.

Back to you, Scott.

**SCOTT GUTHRIE:** Thanks, Rohan.

We've covered a lot of territory in this presentation, watched some amazing end-to-end demos, and announced a lot of new services. You're going to see even more throughout these next two days in some of the other great sessions that we have here.

Developing with Azure puts the latest cloud technology and best-in-class development tools into your hands. No other cloud offers the comprehensive end-to-end platform that the Microsoft cloud delivers, and we really look forward to the amazing solutions that you build on top of it.

Thank you and have a great event.

END