



January 30, 2019 — 8:30 AM - 5:00 PM | Detroit, Michigan

Azure Dev Day

Learn, architect, and develop solutions on Azure



#AzureDevDays
for developers, by developers

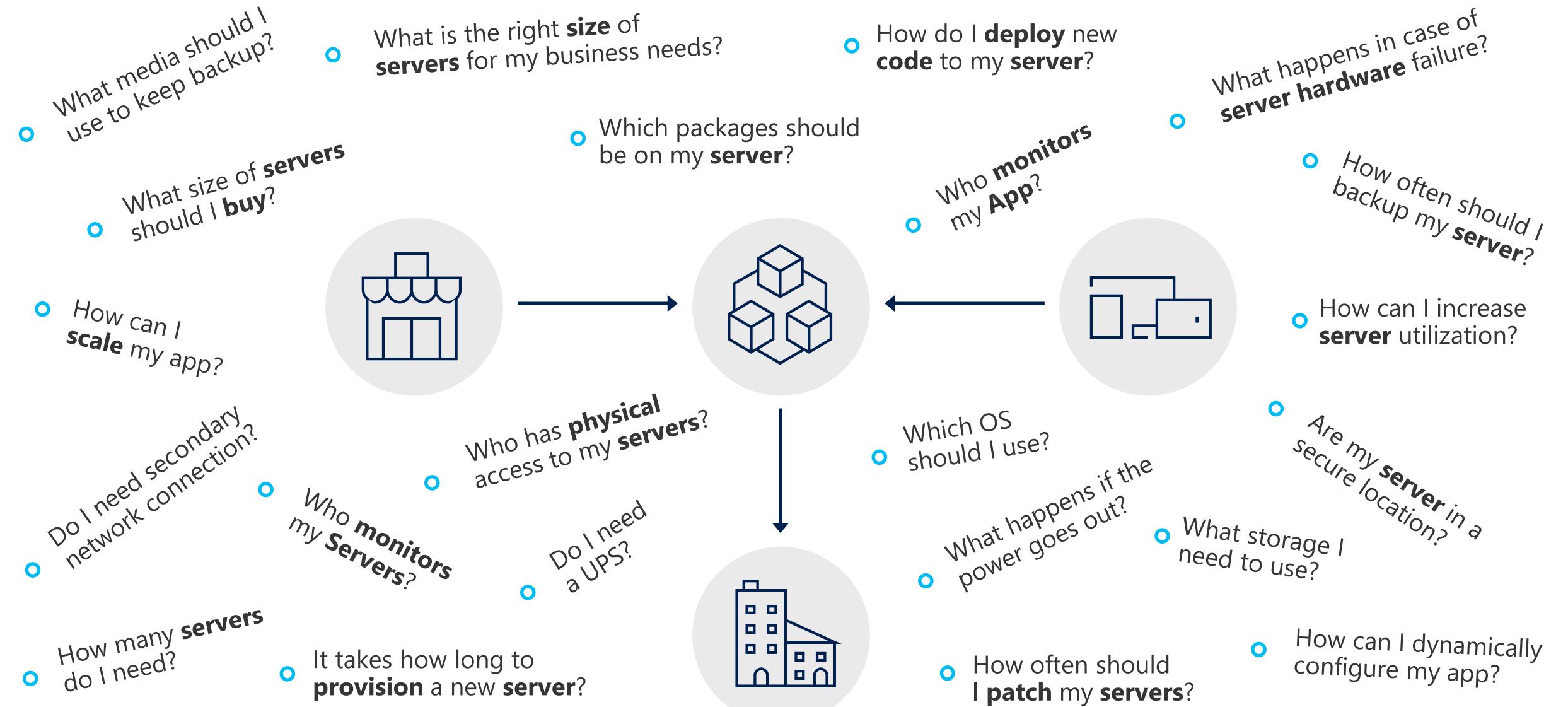
Learn.
Connect.
Explore.

Serverless Computing

Randy Pagels
Azure Technical Specialist – App Dev



Learn.
Connect.
Explore.



On-Premises

The “evolution” of application platforms

What is the right **size** of **servers** for my business needs?

How can I increase **server** utilization?

How many **servers** do I need?

How can I **scale** my app?



How often should I **patch** my **servers**?

How often should I backup my **server**?

Which packages should be on my **server**?

How do I **deploy** new **code** to my **server**?

Which OS should I use?

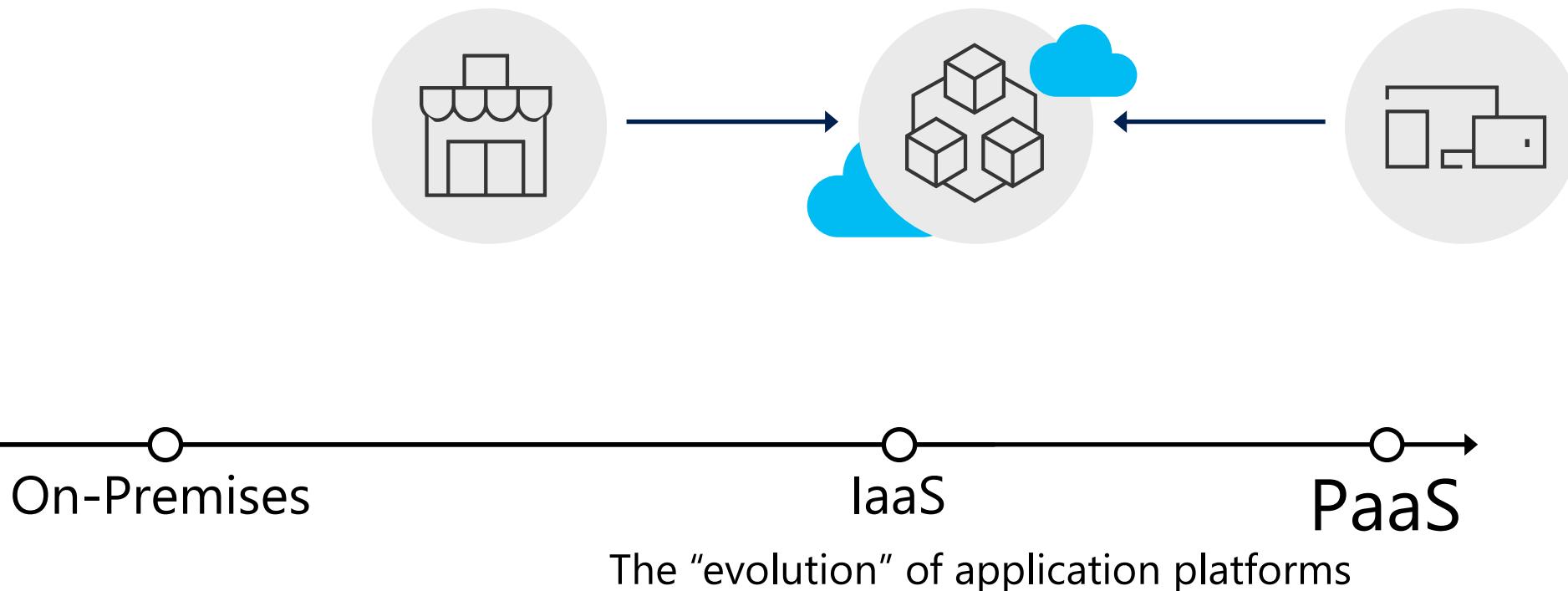
Who **monitors** my App?

On-Premises

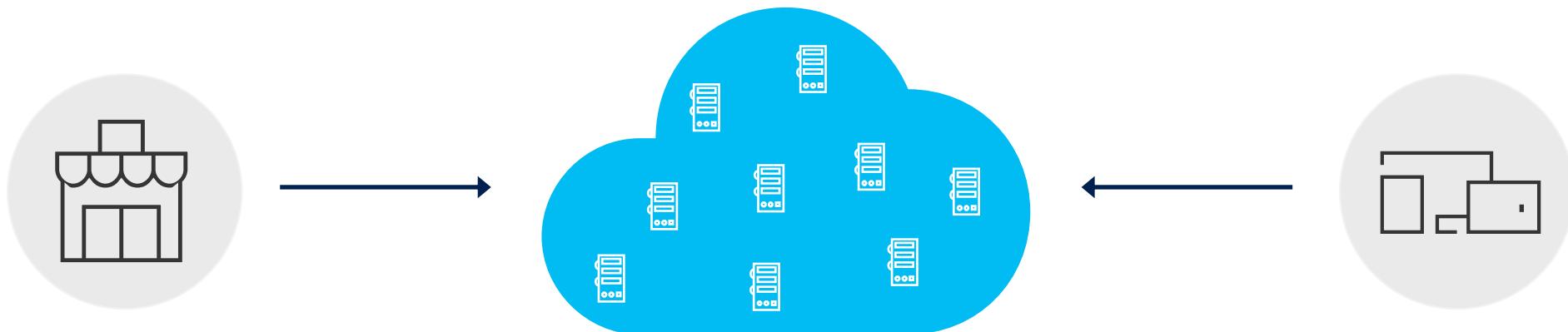
IaaS

The “evolution” of application platforms

What is the right **size** of “**servers**” for my business needs?
How can I increase “**server**” utilization?
How many “**servers**” do I need?
How can I **scale** my app?



How do I **architect** my app?



Serverless, the platform for next gen apps

On-Premises IaaS PaaS Serverless
The “evolution” of application platforms

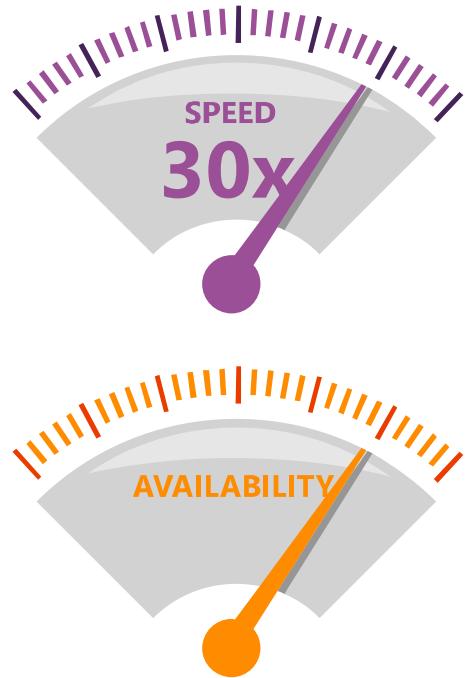
What is serverless?

I need to run this CODE when
this EVENT happens...

Serverless: Focus on code, not plumbing



Abstraction
of servers



Event-driven
scale

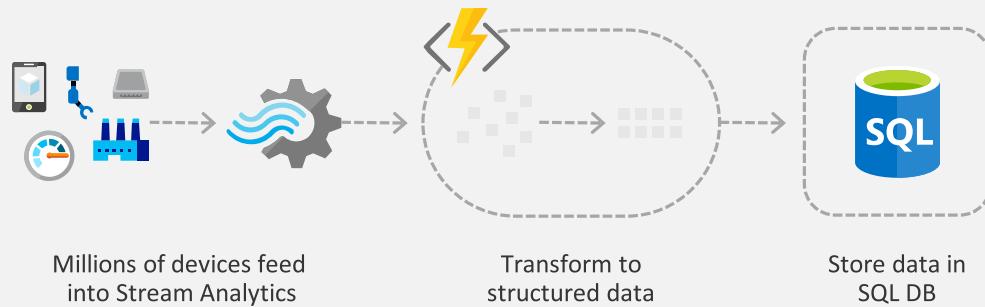


Sub-second
billing

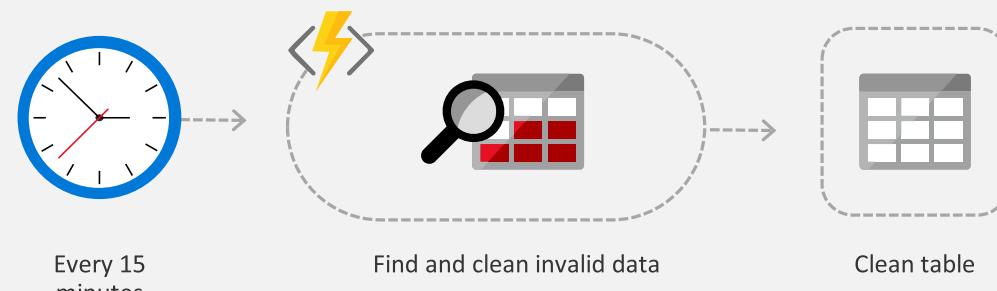
Scenarios for Serverless

Anything that needs to respond to events

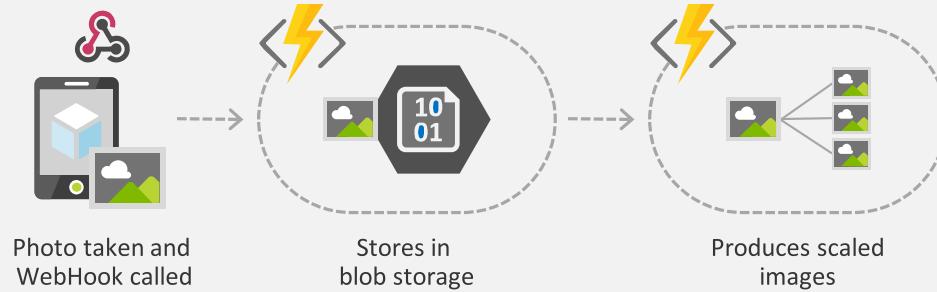
Real-time stream processing



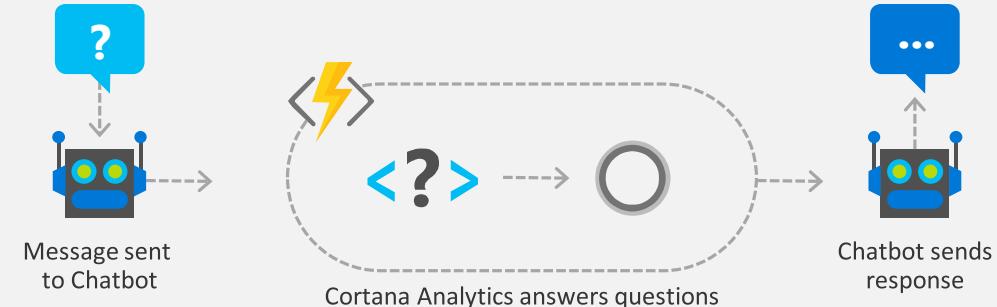
Timer-based processing



Backends (Mobile/IoT/Web)



Real-time bot messaging



What are the benefits?



Focus

Solve business problems—not technology problems related to undifferentiated heavy lifting



Efficiency

- Shorter time to market
- Fixed costs converted to variable costs
- Better service stability
- Better development and testing management
- Less waste



Flexibility

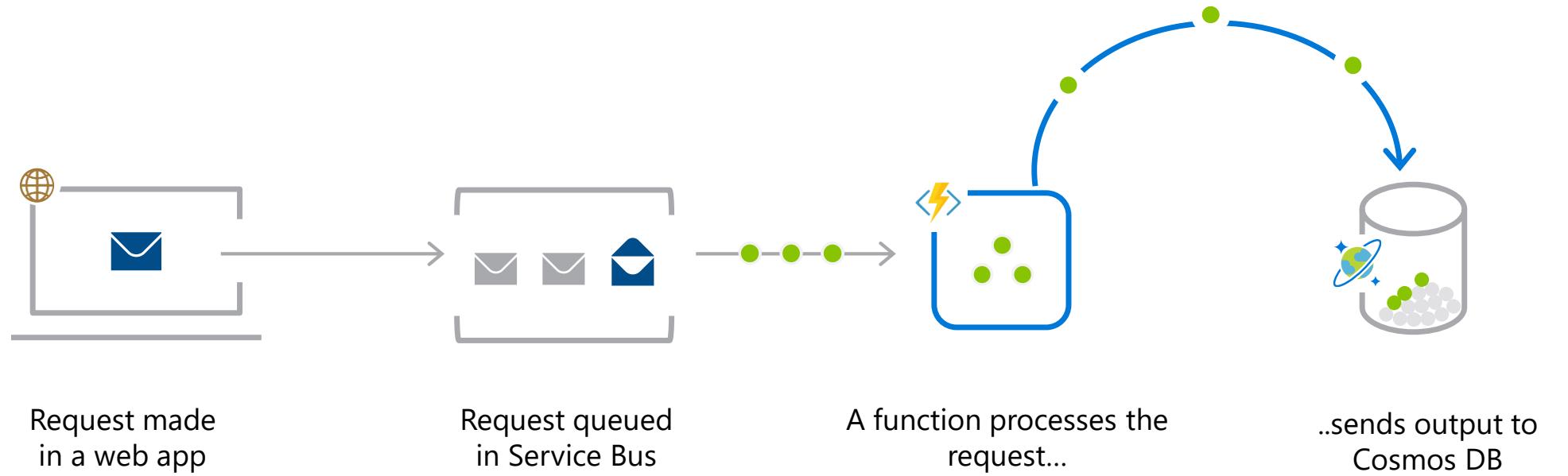
- Simplified starting experience
- Easier pivoting means more flexibility
- Easier experimentation
- Scale at your pace—don't bet the farm on Day 1
- Natural fit for microservices



Web application backends

Scenario Example: Retail

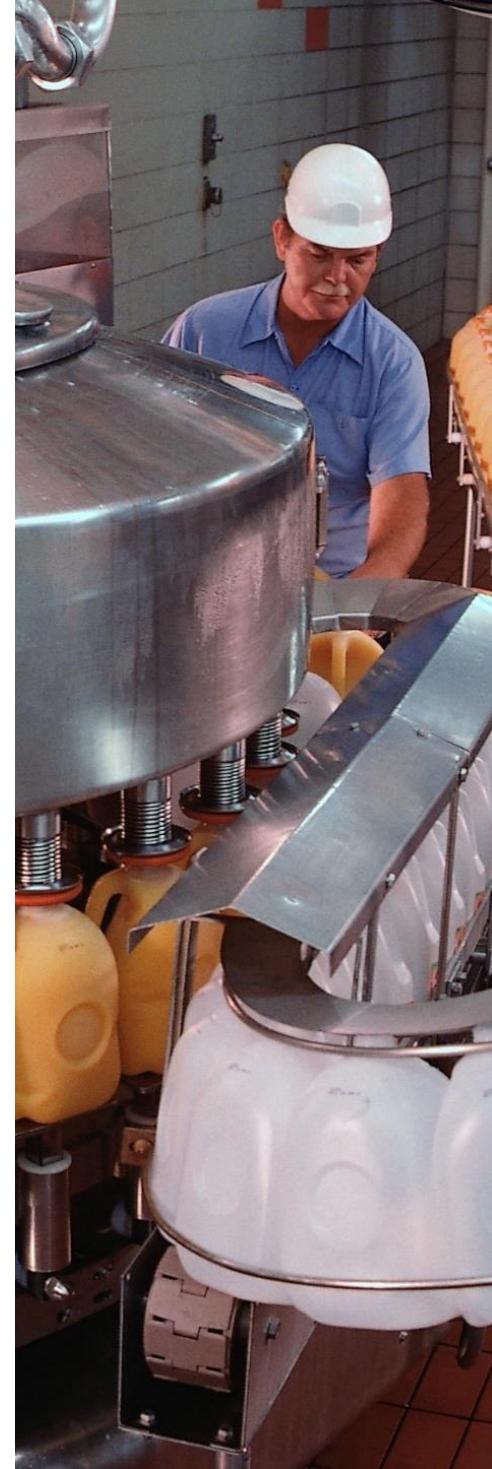
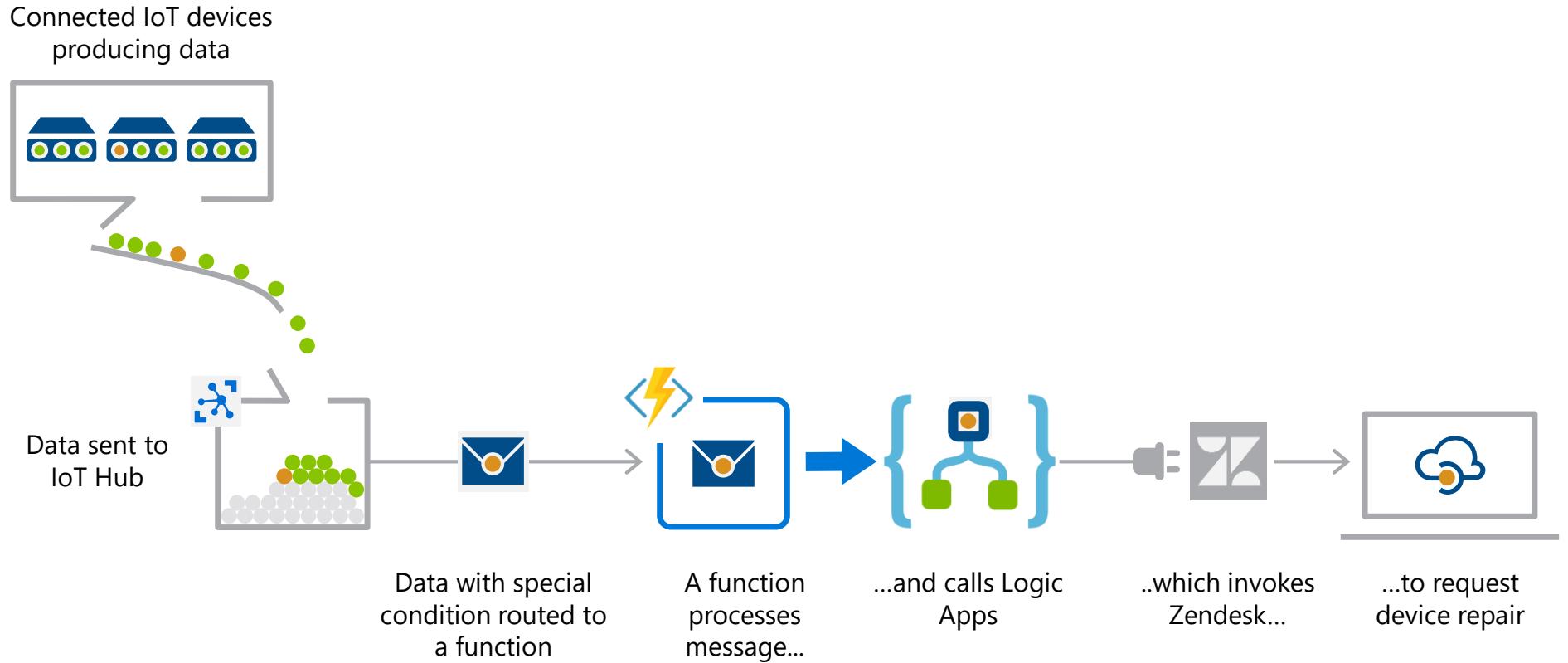
Online orders are picked up from a queue, processed and the resulting data is stored in a database



IoT-connected backends

Scenario Example: Manufacturing

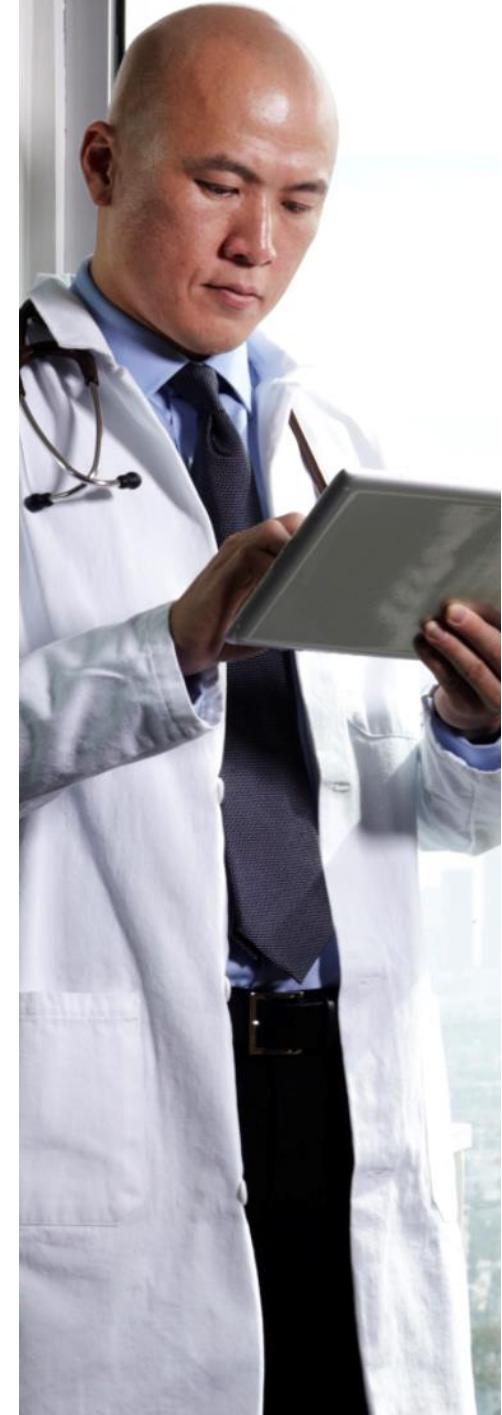
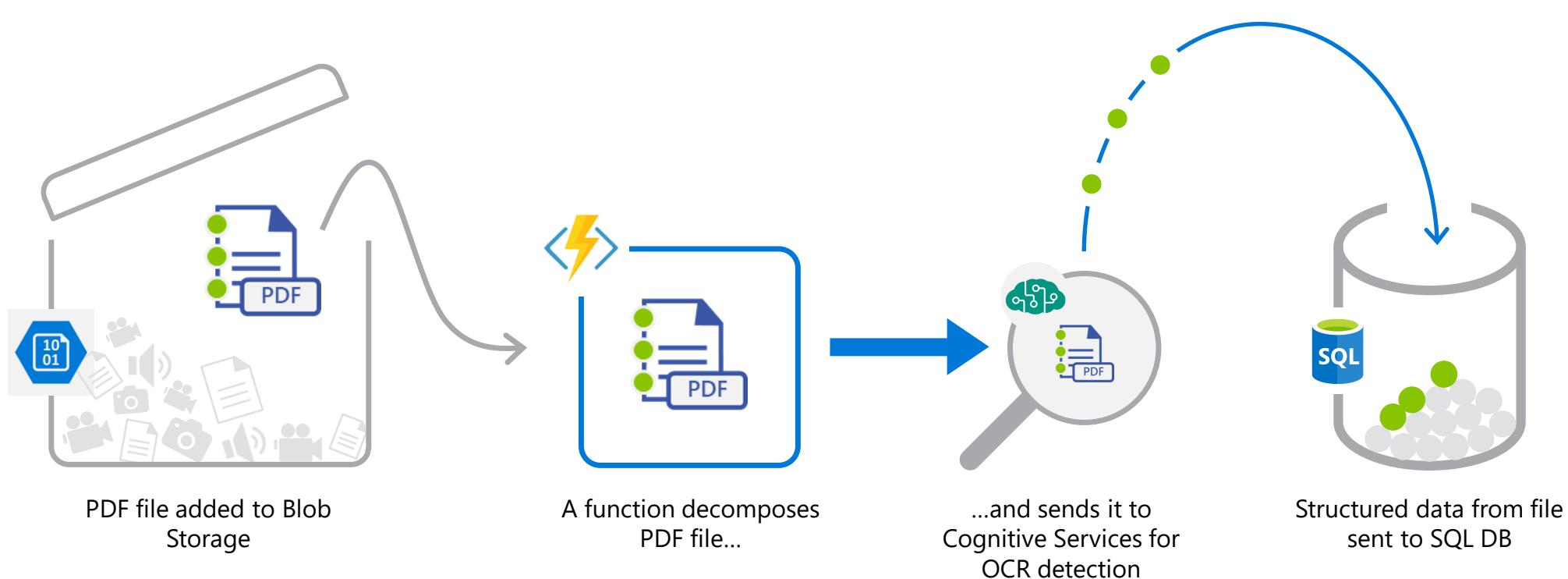
A manufacturing company uses IoT to monitor its machines. Functions detects anomalous data and triggers a message to Service department when repair is required.



Real-time file processing

Scenario Example: Healthcare

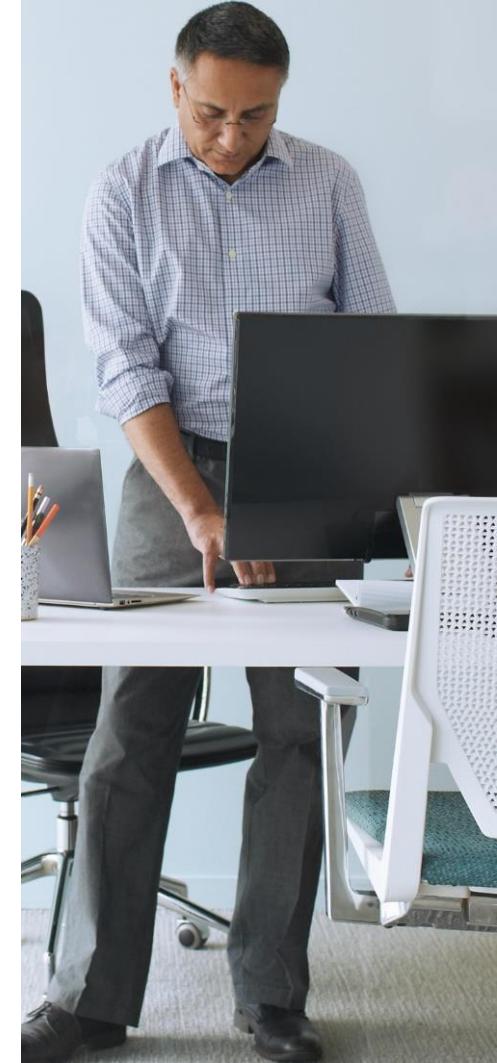
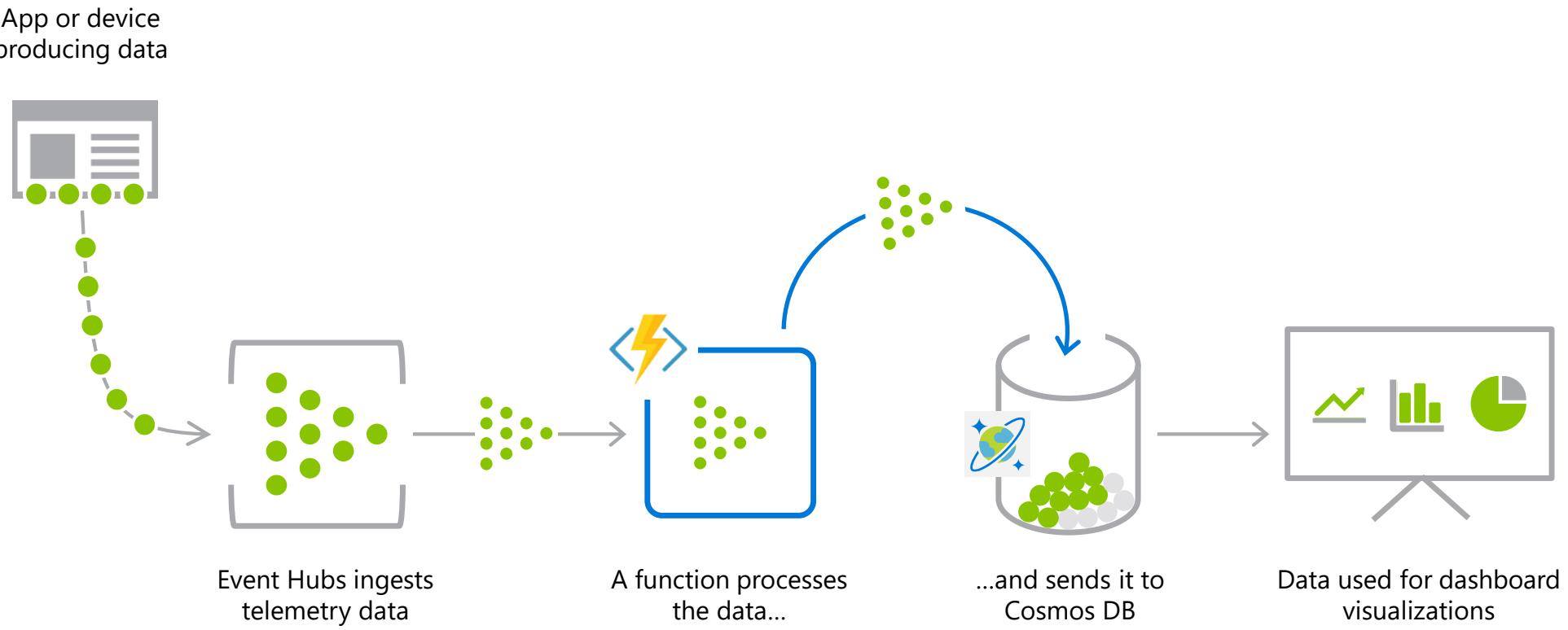
Patient records are securely uploaded as PDF files. That data is then decomposed, processed using OCR detection, and added to a database for easy queries.



Real-time stream processing

Scenario Example: ISV

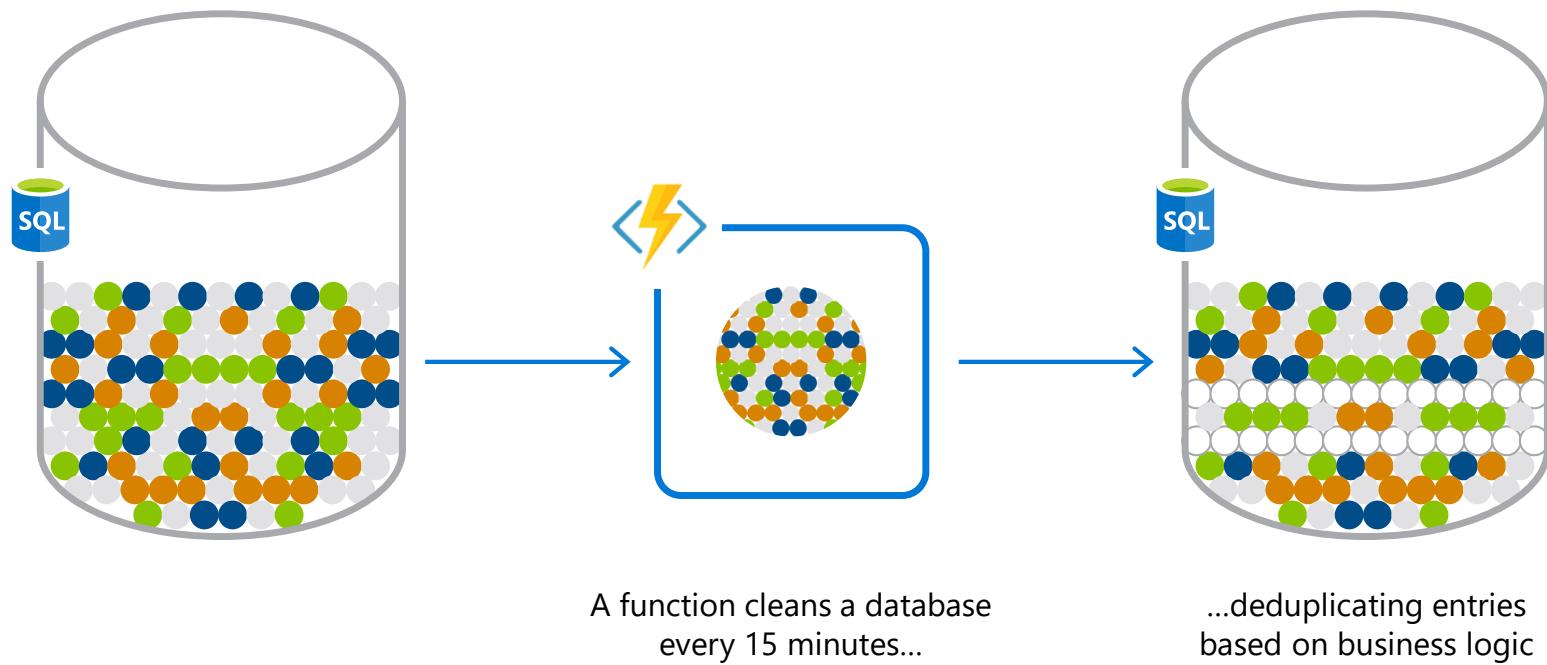
Huge amounts of telemetry data is collected from a massive cloud app. That data is processed in near real-time and stored in a DB for use in an analytics dashboard.



Automation of scheduled tasks

Scenario Example: Financial Services

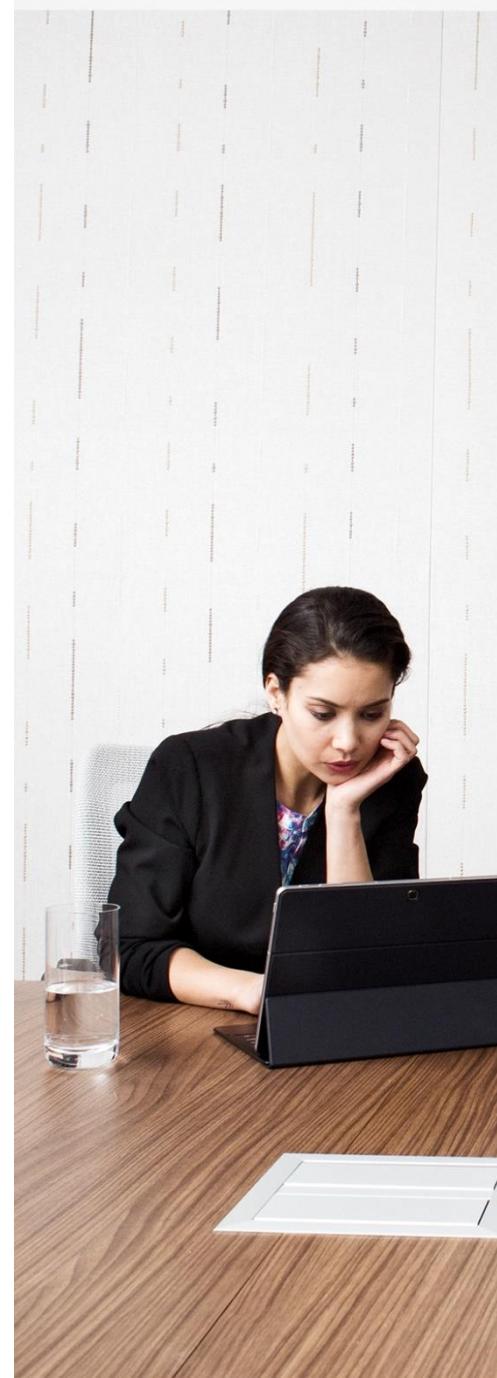
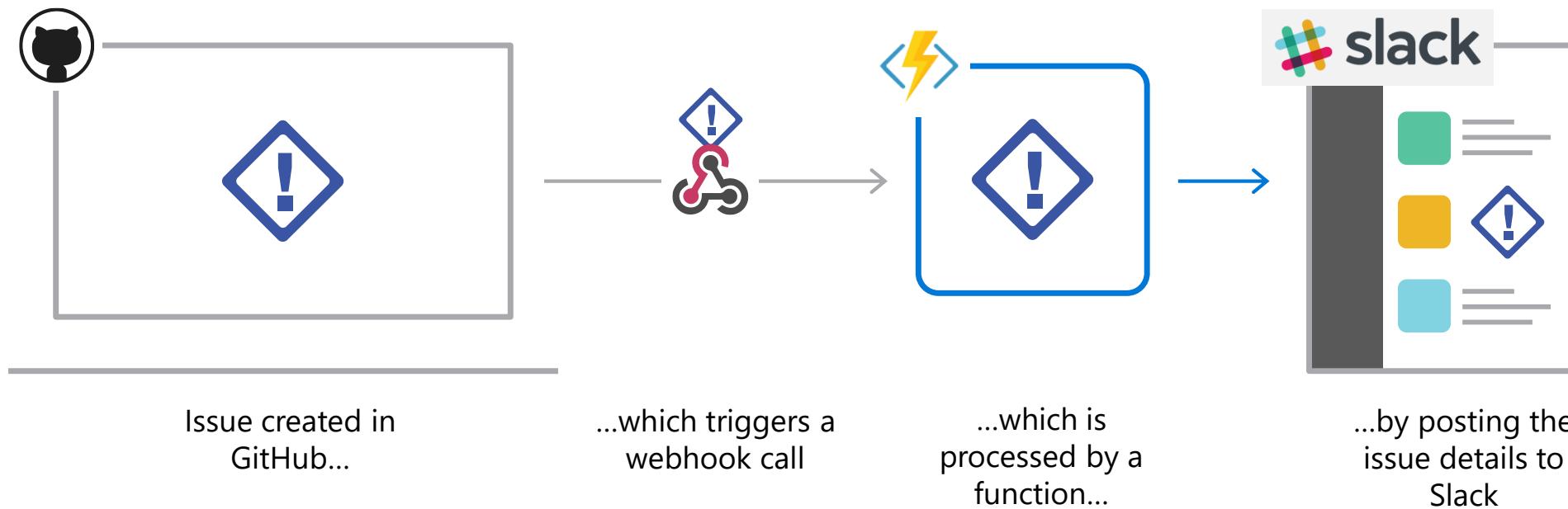
A customer database is analyzed for duplicate entries every 15 minutes, to avoid multiple communications being sent out to same customers.



Extending SaaS Applications

Scenario Example: Professional Services

A SaaS solution provides extensibility through webhooks, which can be implemented through Functions, to automate certain workflows

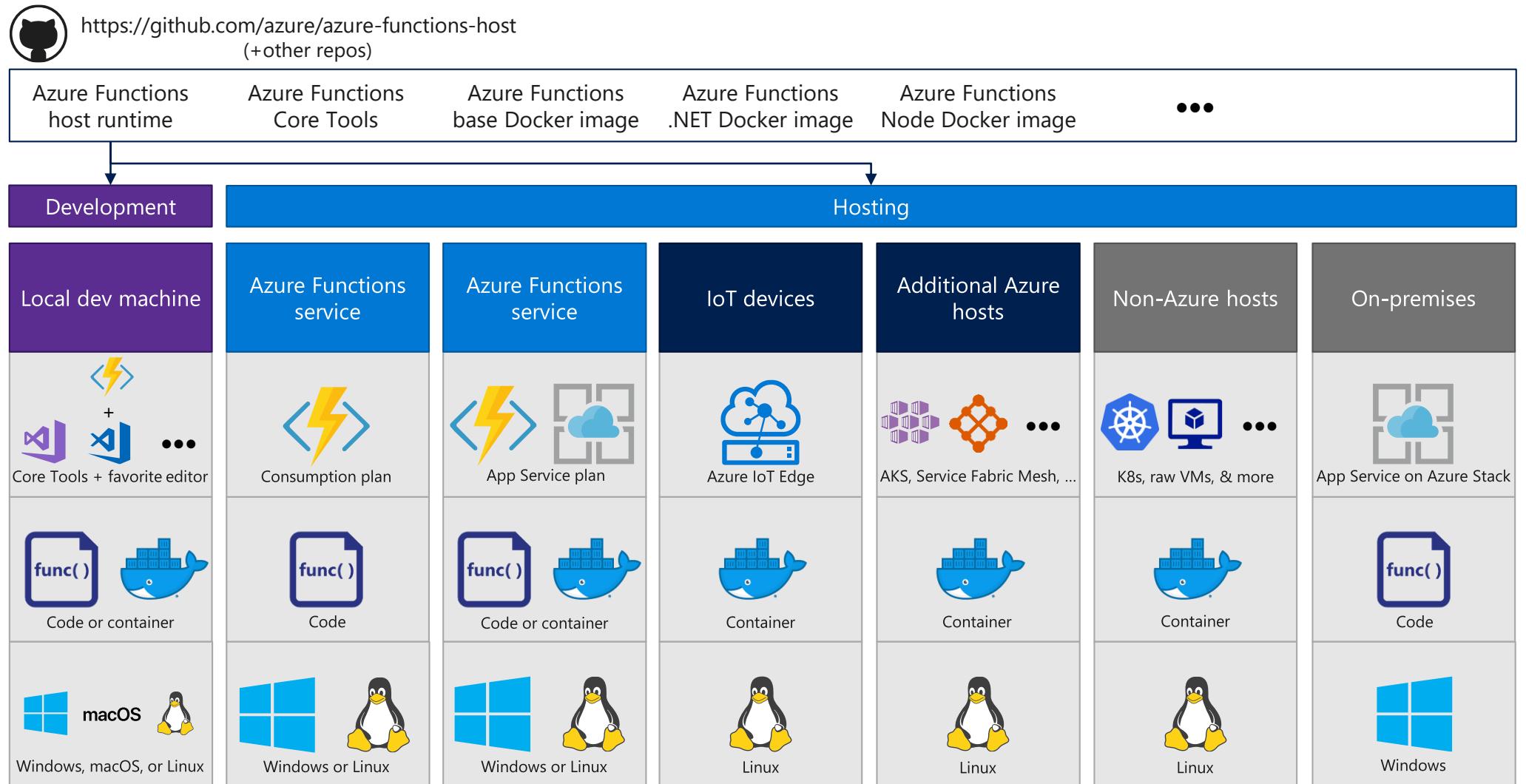


Azure Functions

Azure serverless application platform

Development		Platform					
	IDE support		Functions		Event Grid		Logic Apps
	Integrated DevOps	<ul style="list-style-type: none">Developer productivityTriggers and BindingsFlexible deployment options	Manage all events that can trigger code or logic	<ul style="list-style-type: none">Manage all events in one placeNear real-time deliveryBroad coverage	Design workflows and orchestrate processes	<ul style="list-style-type: none">Visual designer200+ connectors (e.g. Twitter, Blob storage)Functions orchestration	Execute your code based on events you specify
	Local development						
	Monitoring						
	Visual debug history						
	Automation		Storage		Intelligence		IoT

Functions everywhere



Language options



More on the way!

Triggers and Bindings

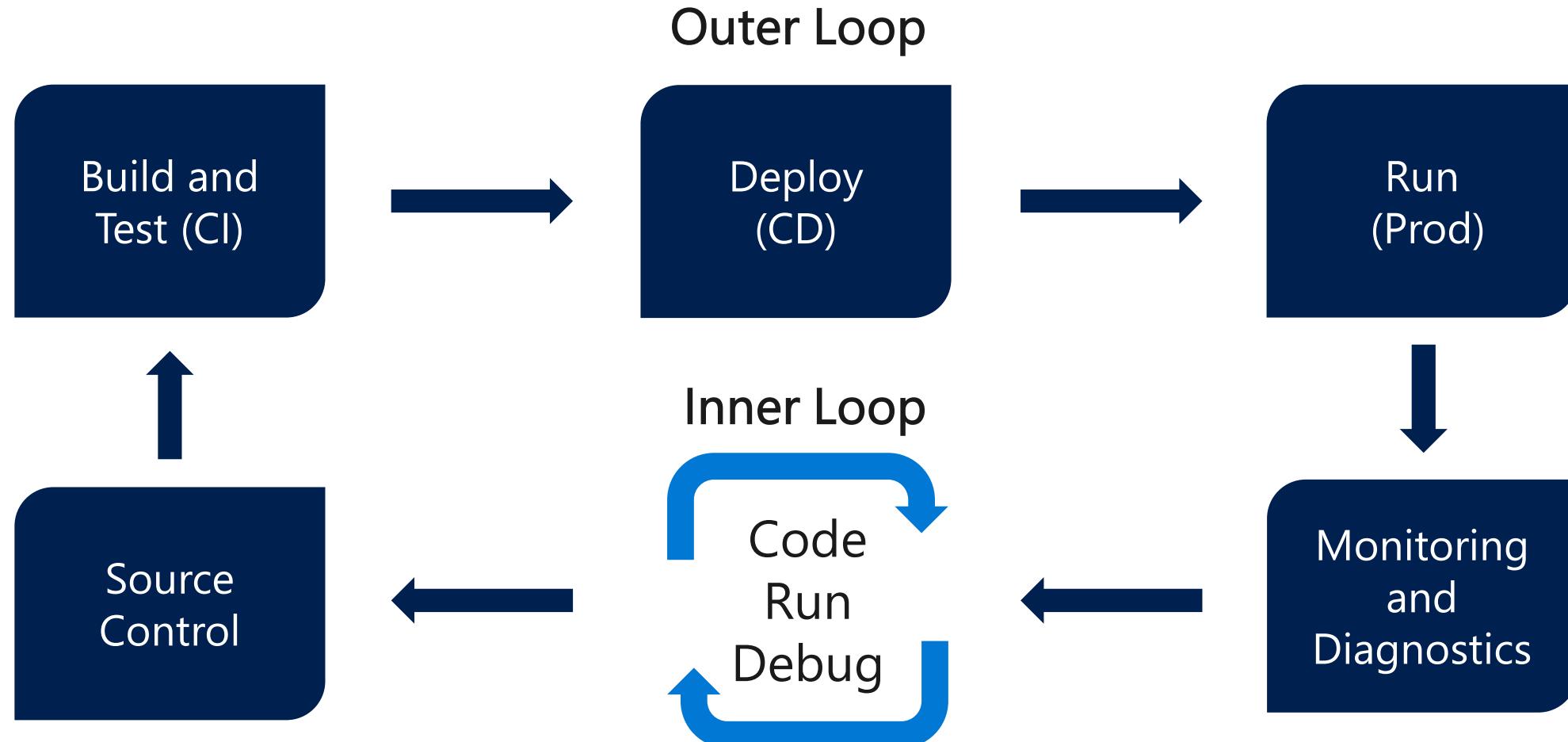
Type	Service	Trigger	Input	Output
Schedule	Azure Functions	✓		
HTTP (REST or webhook)	Azure Functions	✓		✓*
Blob Storage	Azure Storage	✓	✓	✓
Events	Azure Event Hubs	✓		✓
Queues	Azure Storage	✓		✓
Queues and topics	Azure Service Bus	✓		✓
Tables	Azure Storage	✓	✓	
Tables	Azure Mobile Apps	✓	✓	
No-SQL DB	Azure DocumentDB	✓	✓	
Push Notifications	Azure Notification Hubs			✓
Twilio SMS Text	Twilio			✓

Functions Programming Model



DevOps and Serverless

Inner and Outer Loop Development



Available tools of Azure Functions

Local Tools

Quickly publish to production

Best Suited – Quickly validate code works in the cloud

Watch out – “Friends don’t let friend right-click publish”

Tip – Use the ‘run from package’ feature

Deployment Center (Kudu)

App Services powered CI/CD

Best Suited – One-click deploy from GitHub/source

Watch out – Not as customizable as Azure DevOps pipelines

Tip – Use the new “Deployment Center” section

Azure DevOps

Fully managed CI/CD

Best Suited – Production CI/CD with various environments

Watch out – Web Deploy vs Run from Package

Tip – Can call functions as release gates

Other CI/CD

Any other CI/CD tool (Jenkins, Octopus, Travis)

Best Suited – Integrated serverless with existing tools and processes

Watch out – Documentation and samples are limited

Tip – Use the ‘run from package’ publish gesture

Inner Loop Best Practices

- Write unit tests for your functions
 - How should this behave on success?
 - How should this behave on failure?
 - Mock external systems and side-effects
- Environment variables == local.settings.json == Application Settings
 - local.settings.json when local
 - Application Settings when published

Outer Loop Best Practices

- Every function should be checked into source control
- Feature / Bug branches should be tested before merge into master
 - Code compiles
 - Unit tests pass
- Deploy to stages
 - Development, Pre-Prod, Prod
 - Use release gates and approval (auto or manual) between stages
- Use slots where appropriate
 - Optimized today for HTTP scenarios
 - Be aware of scaling impacts to consumption functions
- Enable Application Insights

Monitoring

Gain real-time observability

Analyze and debug traces and metrics

View dependencies and relationships with AppMap



Application
Insights



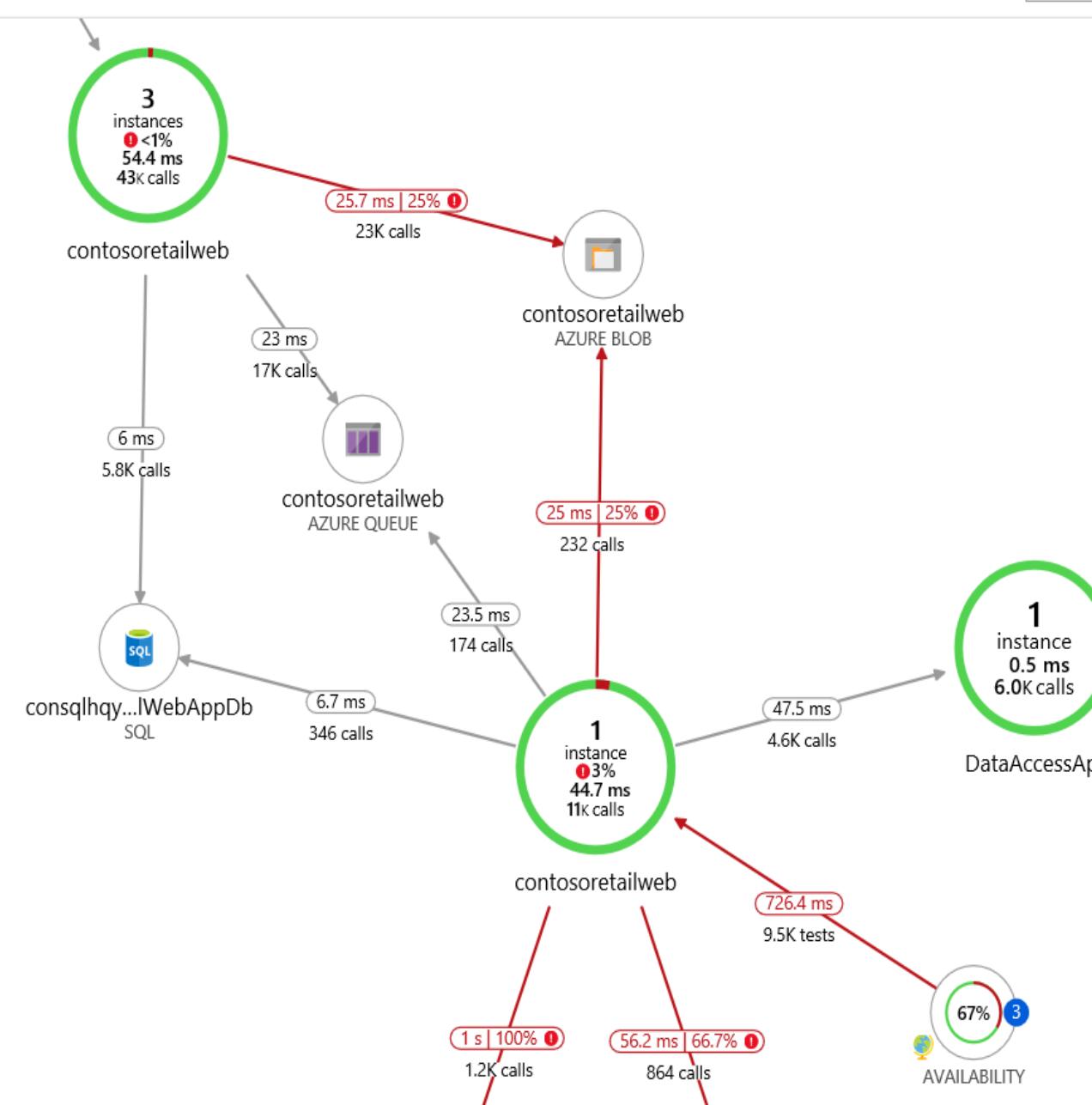
Last 24 hours

Feedback

Learn more

Refresh

Update map components

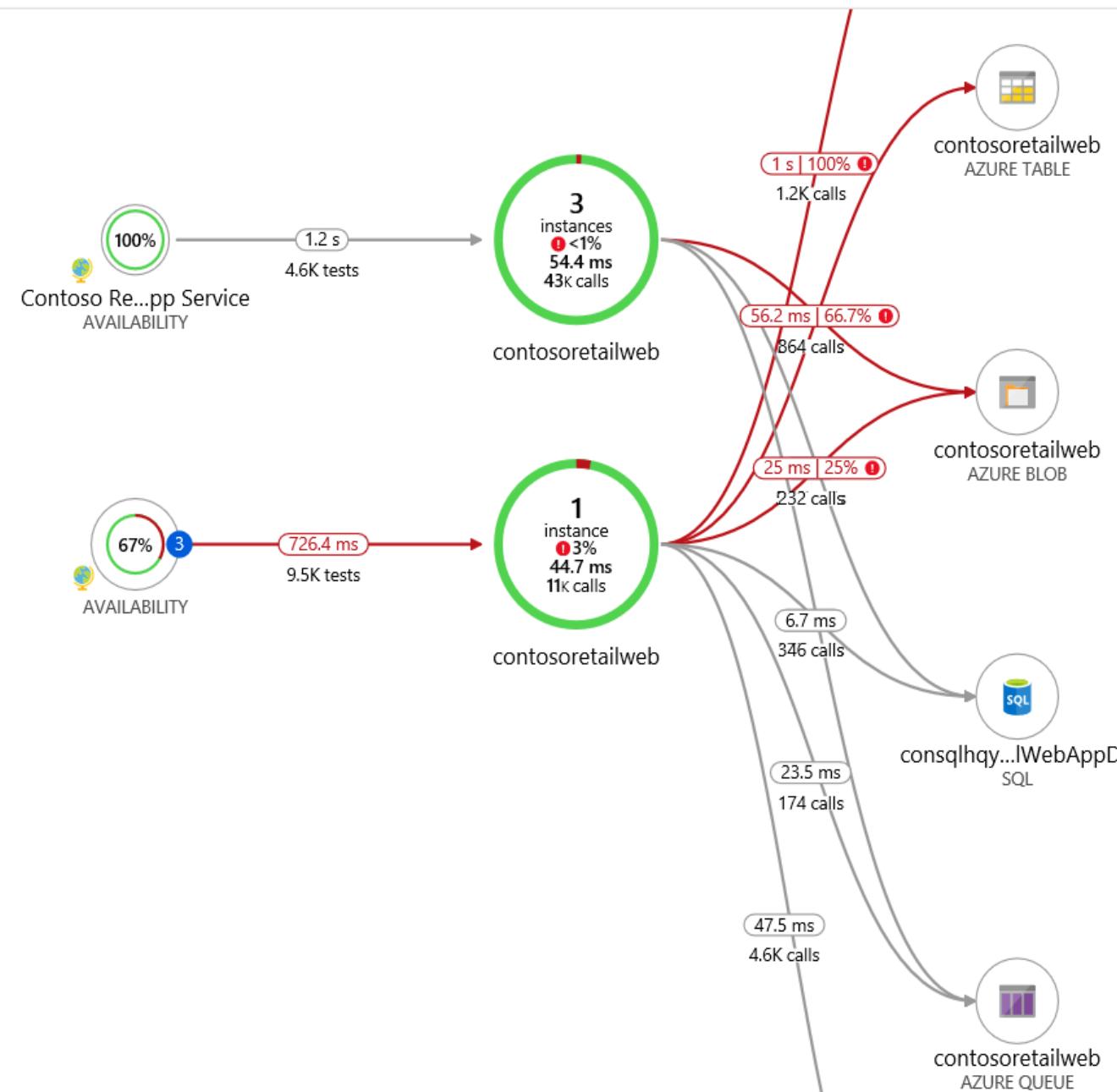


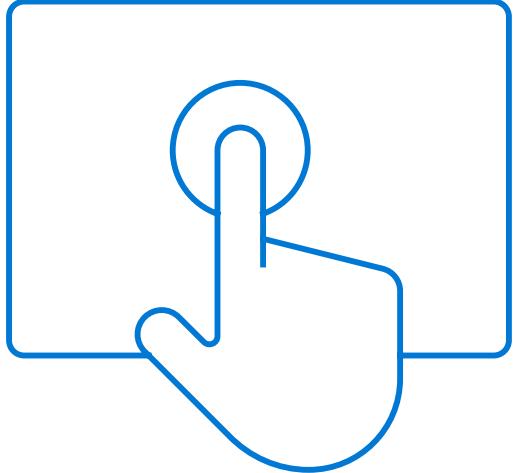
Last 24 hours

Feedback

 Learn more

Update map components





Getting Started Demo

Create Azure Functions using the Portal

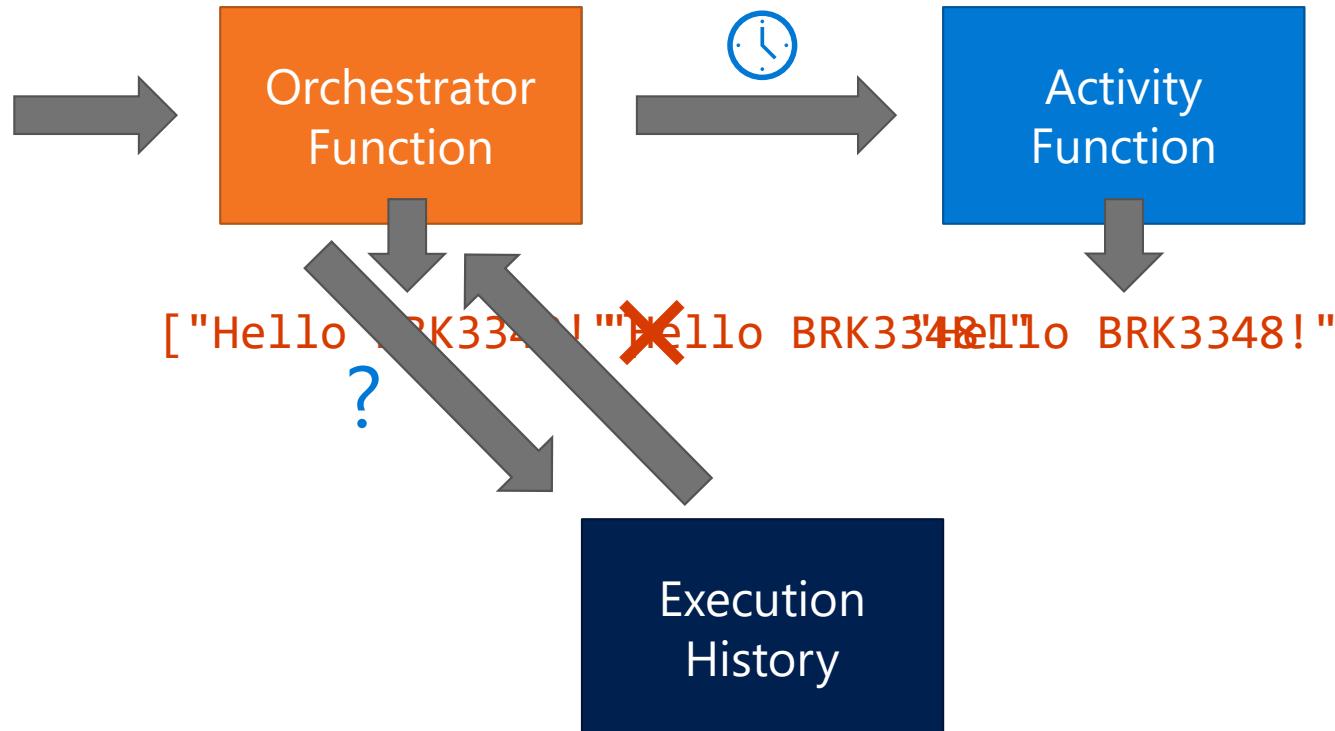
Create Azure Functions using Visual Studio Code

Create Azure Functions using Visual Studio

Inner & Outer Loop using Azure DevOps

Durable Functions

```
var outputs = new List<string>();  
  
outputs.Add(await context.CallActivityAsync<string>("SayHello", "BRK3348"));  
  
return outputs;
```



History Table

Orchestrator Started
Execution Started
Task Scheduled, SayHello, "BRK3348"
Orchestrator Completed
Task Completed, "Hello BRK3348!"
Orchestrator Started
Execution Completed, ["Hello BRK3348!"]
Orchestrator Completed

Hosting and Connectivity

Functions hosting options

Development		Hosting					
Platform	Local dev machine	Azure Functions service	Azure Functions service	IoT devices	Additional Azure hosts	Non-Azure hosts	On-premises
Application delivery	Core Tools + favorite editor	Consumption plan	App Service plan	Azure IoT Edge	AKS, Service Fabric Mesh, ...	K8s, raw VMs, & more	App Service on Azure Stack
Operating system	Windows, macOS, or Linux	Windows or Linux	Windows or Linux	Linux	Linux	Linux	Windows
Platform							
Application delivery							
Operating system							

Azure Functions Hosting Options

Consumption

- Rapid scale out
- “Unbounded” scale out
- No VNet connectivity available
- 10 minute execution
- Small instance size
- Scale to zero

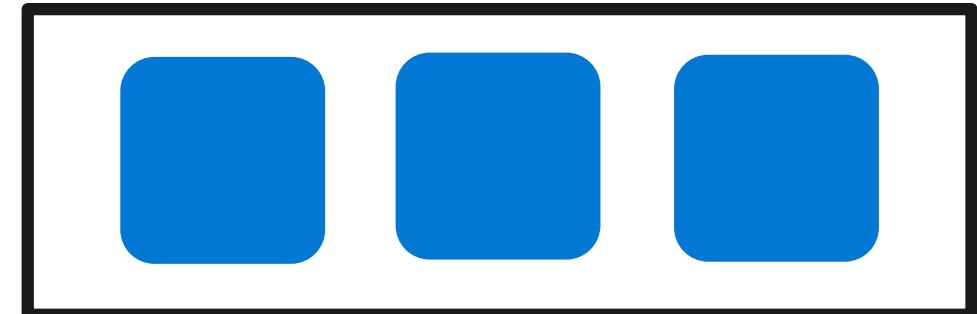
... ...



... ...

App Service Plan / Environment

- Auto-scale out (~5 min)
- Fixed scale out (Max=10 nodes)
- VNet connectivity / hybrid
- Unlimited execution duration
- Premium instance size
- Always on



Azure Functions Hosting Options

PRIVATE PREVIEW

Consumption

- Rapid scale out
- “Unbounded” scale out
- No VNet connectivity available
- 10 minute execution
- Small instance size
- Scale to zero (cold start)



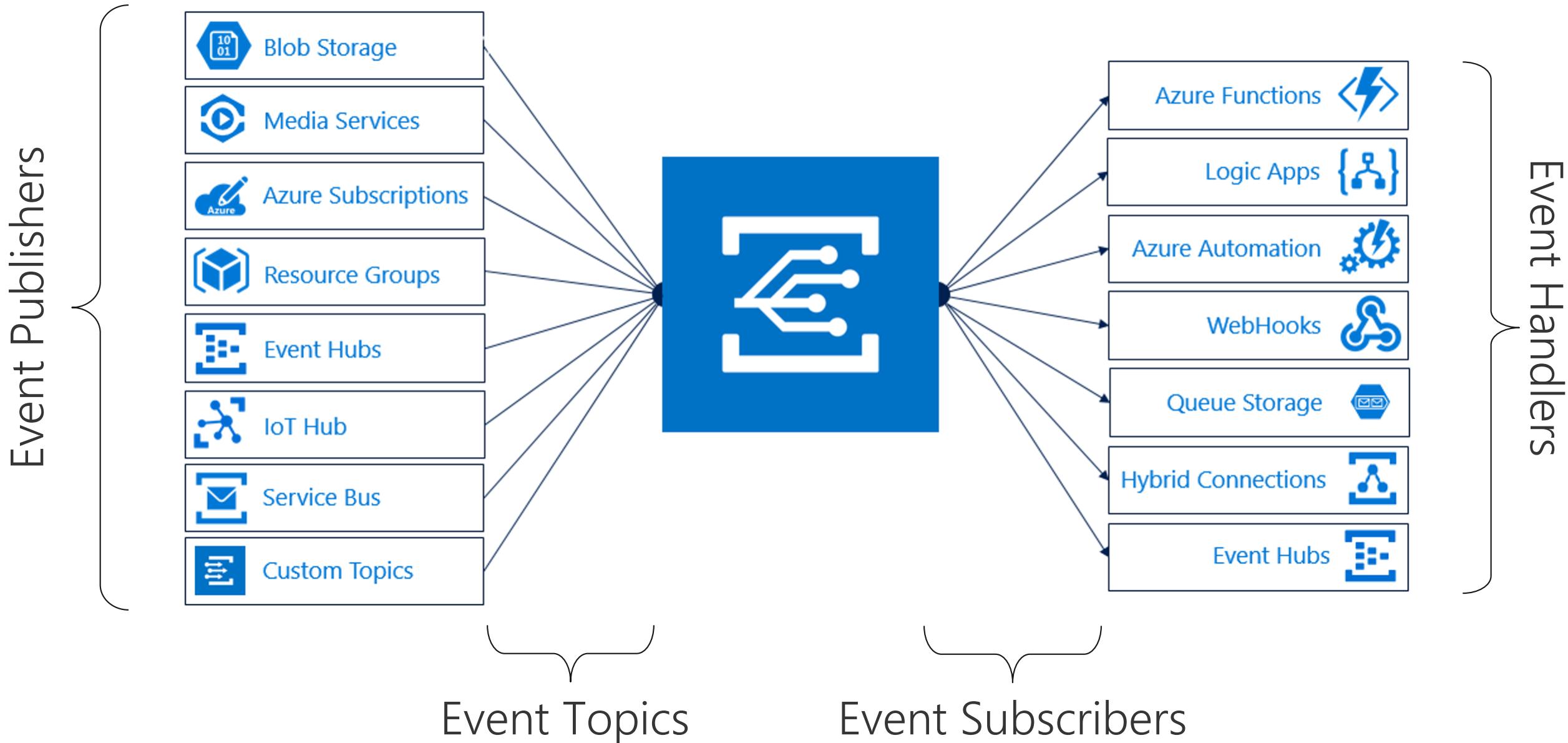
Functions premium plan

- Rapid scale out
- “Unbounded” scale out
- VNet connectivity / hybrid
- Unlimited execution duration
- Premium instance size
- Always on

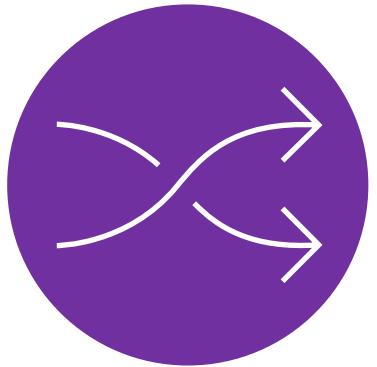


Event Grid

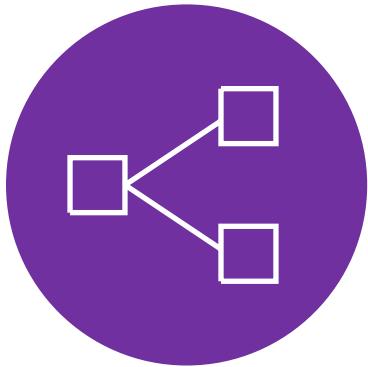
Manage all events in one place



Event Grid capabilities



Send events to
specific event
handlers



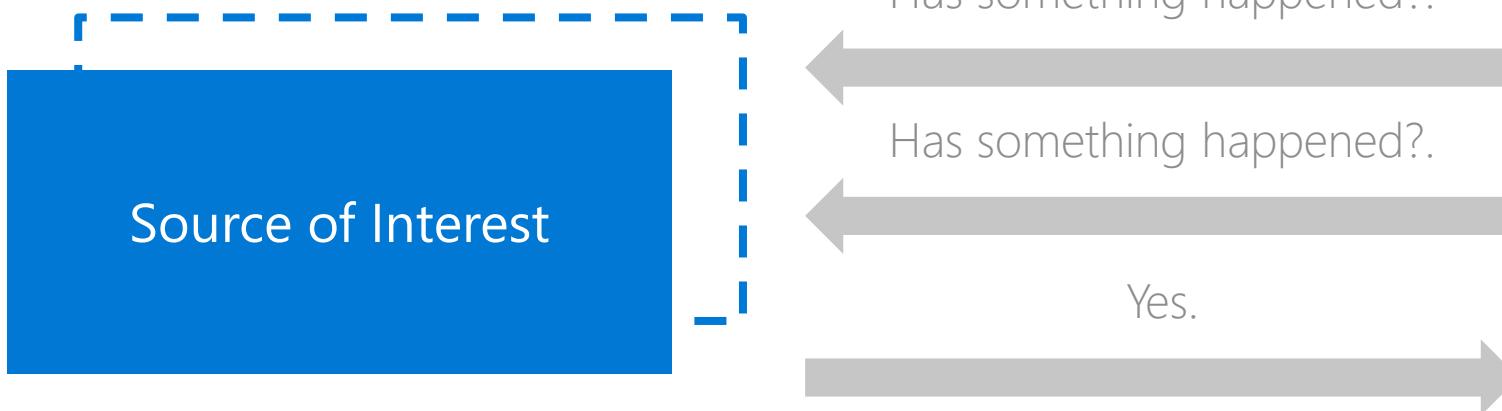
Send one event to
multiple event
handlers



Reliably deliver
events at scale

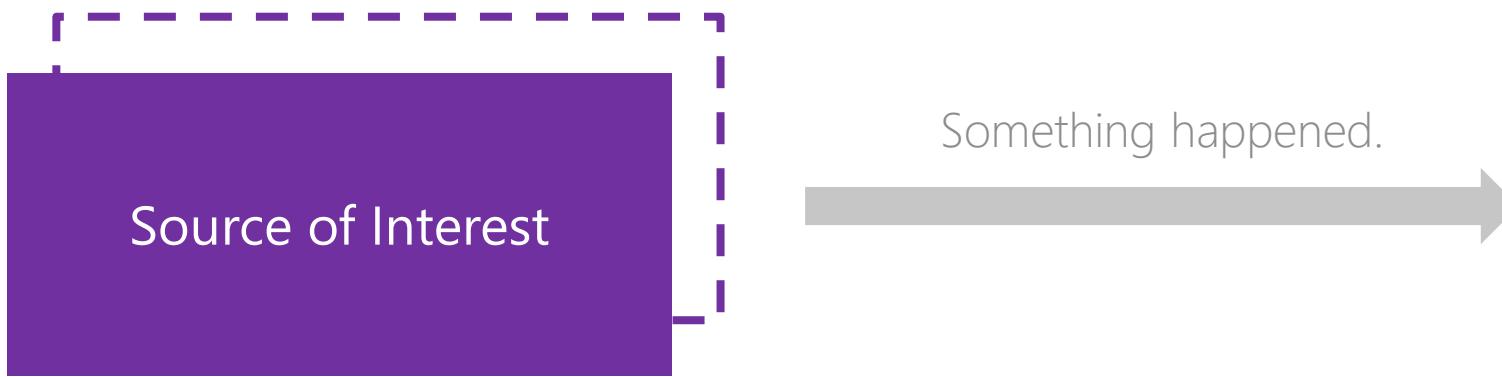
Pull vs. Push

Pull



Interested Party

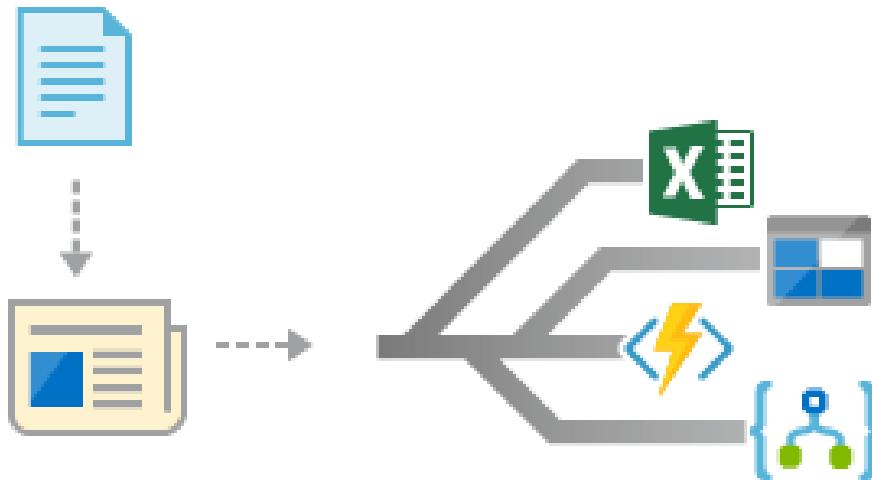
Push



Interested Party

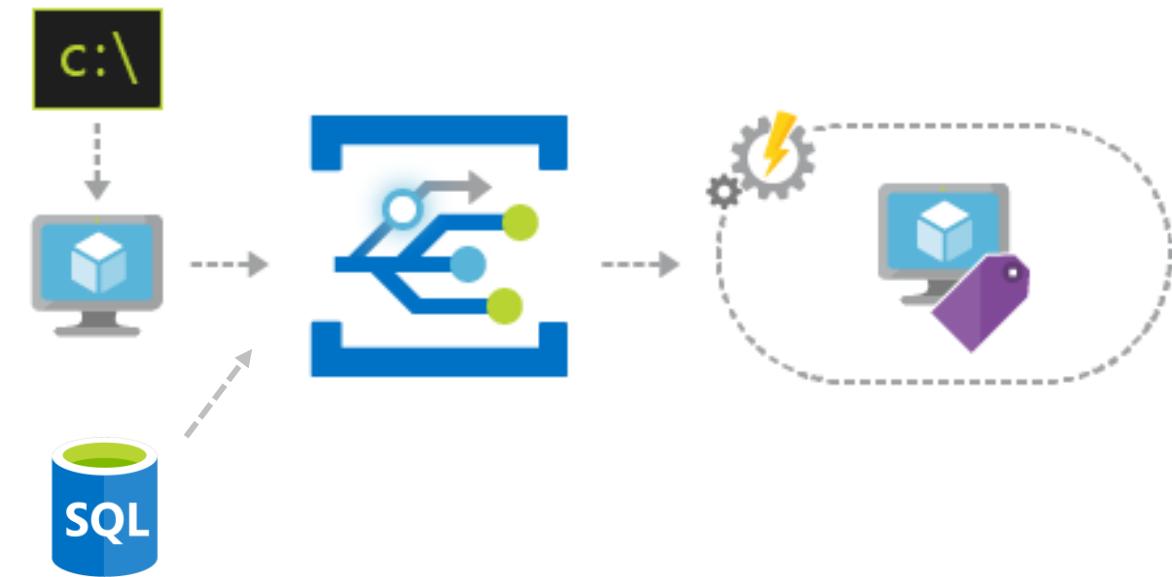
Examples using Event Grid

Serverless app integration



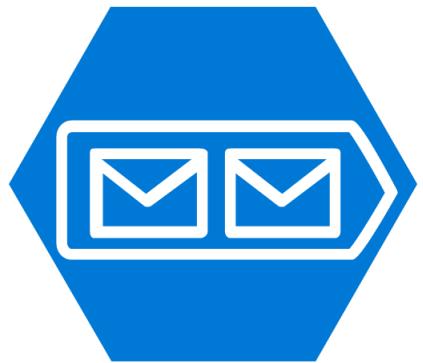
Event Grid connects your app with other services.

Automate operations



Automate and simplify policy enforcement.

Messaging in Azure



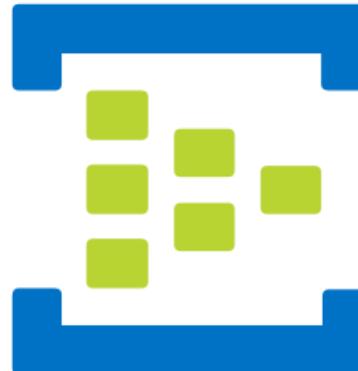
Storage Queues

Simple Queues
Background Processing



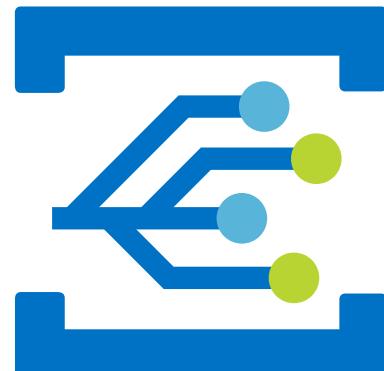
Service Bus

Enterprise Messaging
Ordered Delivery
De-Duplication



Event Hubs

Big Data Streaming (Fan-In)
Messaging at Scale



Event Grid

Reactive Eventing (Fan-Out)
Filtering & Routing
Multiple Subscribers
Push Model

Event Grid in Comparison



	Storage Queues	Service Bus	Event Hubs	Event Grid
Transactions/Atomicity	✗	✓	✗	✗
Ordering guarantee	✗	✓	✓	✗
Delivery Guarantee	At least Once	At least Once, At Most Once	At least Once	At least Once
Read	Pull/Destructive	Pull/Destructive	Pull/Repeatable	Push
Peek	✓	✓	✓ Repeatable reads	✗
Batch Send	✓	✓	✓	✓
Batch Receive	✗	✓	✗	✗
Filtering/Routing	✗	✓ Advanced	✗	✓ - Simple
In flight transformation	✗	✓	✗	✗
Message Size	64 KB	256KB – 1 MB	256 KB	64 KB
De-duplication	✗	✓	✗	✗

Benefit from broad coverage

Event Publishers

Immediately available

-  Blob Storage
-  Container Registry
-  Resource Groups
-  IoT Hub
-  Azure Subscriptions
-  Media Services
-  Event Hubs
-  Service Bus
-  Custom Events
-  Media Services
-  Storage General-V2

Coming soon

Azure Automation, Azure Active Directory, API Management, Logic Apps, Azure Data Lake Store, Cosmos DB

Event Handlers

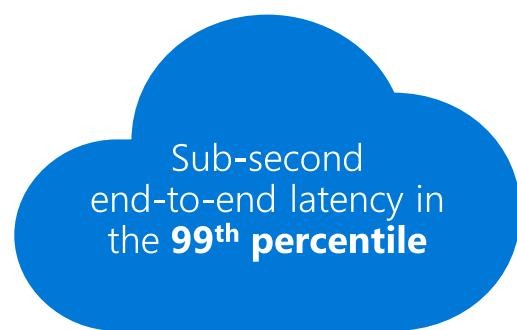
Immediately available

-  Azure Functions
-  Hybrid Connections
-  Logic Apps
-  Microsoft Flow
-  Azure Automation
-  WebHooks
-  Event Hubs
-  Storage Queues

Coming soon

Fabric Controller, Service Bus, Azure Data Factory

Event Grid delivers at massive scale



Sub-second
end-to-end latency in
the **99th percentile**

Near real-time



10,000,000 events
per second per
region

Massive scale-out

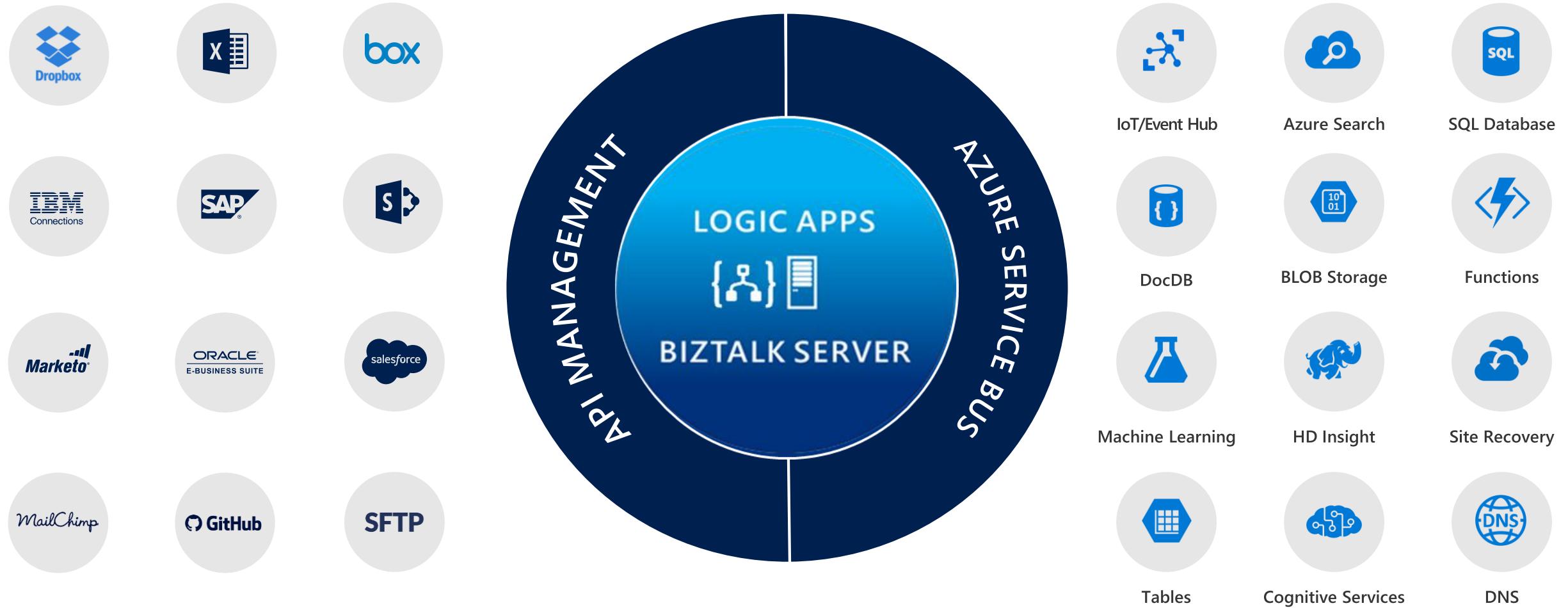


24-hour retry with
exponential back off
for events not
delivered

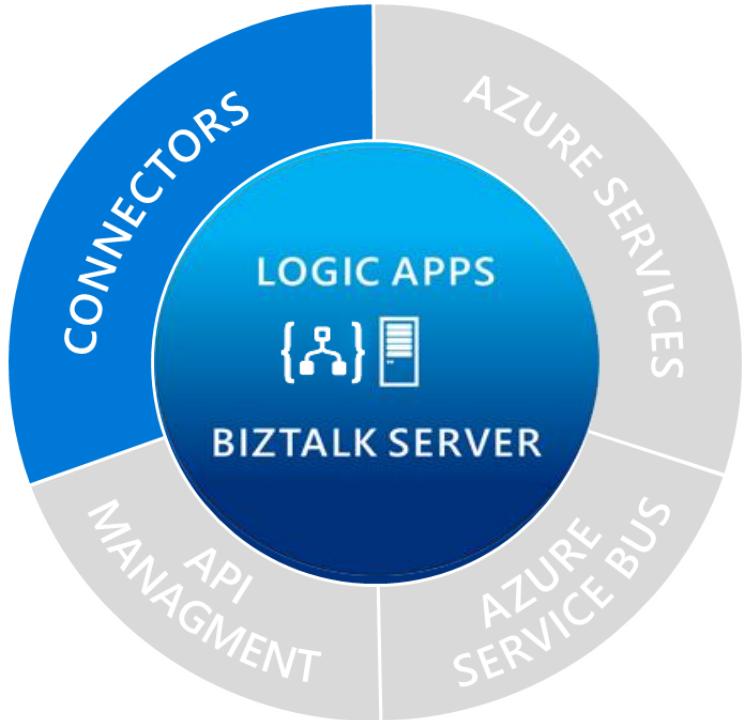
High reliability

Logic Apps

Microsoft's Hybrid Integration Platform



Spectrum of connectors



SaaS

- appFigures
- Asana
- Azure API Management
- Azure App Services
- Azure Cognitive Face API
- Azure Cognitive Text Analytics
- Azure Document DB
- Azure Functions
- Azure Machine Learning
- Azure Resource Manager
- Azure Service Bus
- Azure SQL
- Azure Storage Blob
- Azure Storage Queues
- Basecamp
- Bing Search
- Bitly
- Blogger
- Box
- Buffer
- Campfire
- Common Data Service
- Disqus
- DocuSign
- Dropbox
- Dynamics AX Online
- Dynamics CRM Online
- Dynamics CRM Service Bus
- Dynamics Financials
- Dynamics Operations
- Easy Redmine
- Facebook
- Freshdesk
- GitHub
- Google Calendar
- Google Drive
- Google Sheets
- Google Tasks
- Harvest
- JIRA
- Insightly
- Instagram
- Instapaper
- MailChimp
- Mandrill
- Medium
- Microsoft Project Online
- Microsoft Translator
- MSN Weather
- Office 365
- Office 365 Users
- Office 365 Video
- OneDrive
- OneDrive for Business
- OneNote
- Outlook.com
- PagerDuty
- Pinterest
- Power BI
- Project Online
- Redmine
- Salesforce
- Salesforce Chatter
- SendGrid
- SharePoint Online
- Slack
- SmartSheet
- SparkPost
- Survey Monkey
- Todoist
- Trello
- Twilio
- Twitter
- Typeform
- VS Team Services
- Wordpress
- Wunderlist
- Yammer
- YouTube

Protocols/Native

- HTTP, HTTPS
- HTTP Webhook
- FTP, SFTP
- SMTP
- RSS
- Compose, Query
- Wait
- Terminate
- Workflow
- XML Validation
- Transform XML (+Mapper)
- Flat File Encode
- Flat File Decode
- X12
- EDIFACT
- AS2
- Integration Account Artifact Lookup

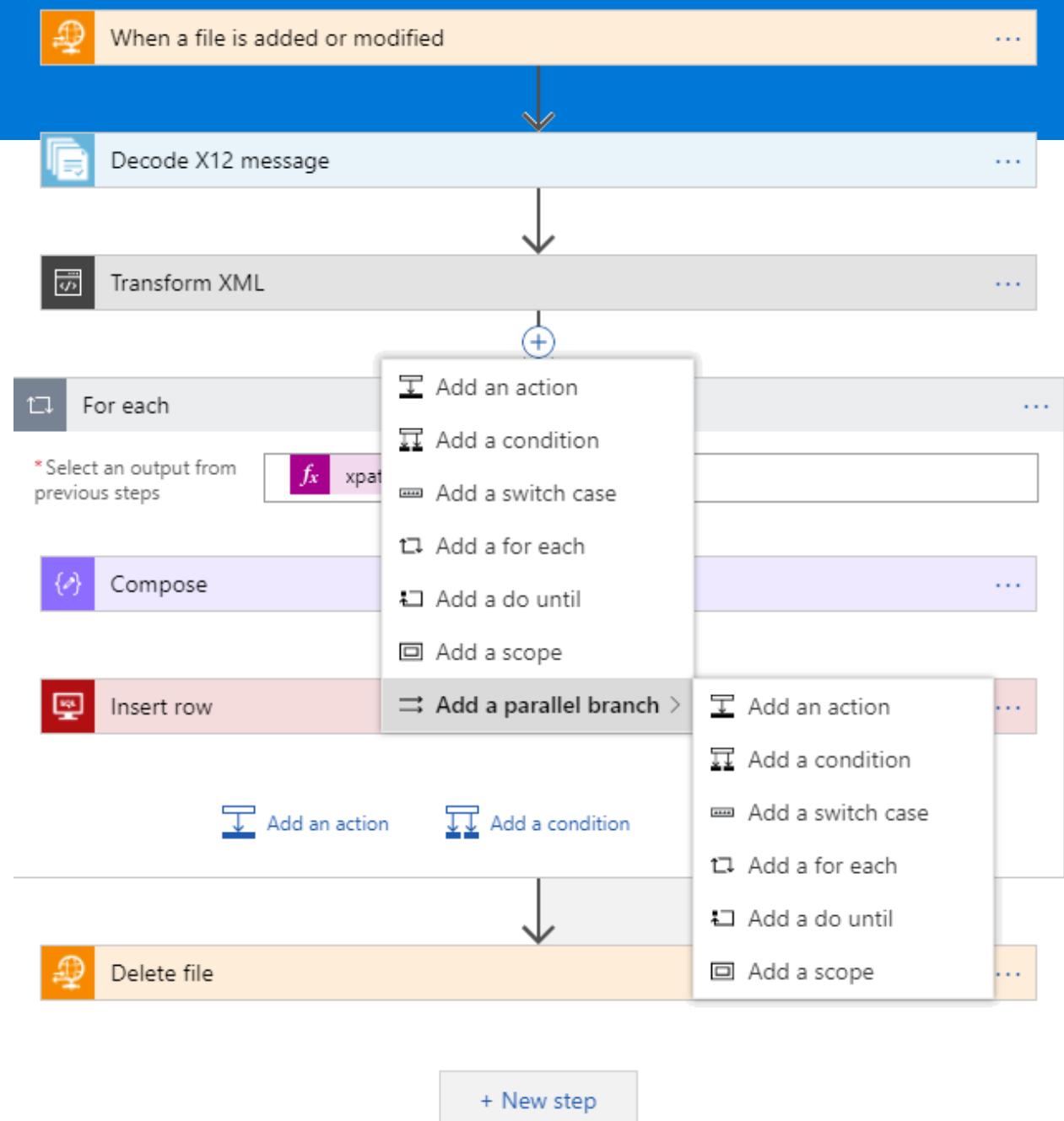
XML & EDI

- BizTalk Server
- File System
- IBM DB2
- Informix
- SharePoint Server
- SQL Server
- SAP
- Websphere MQ

Hybrid

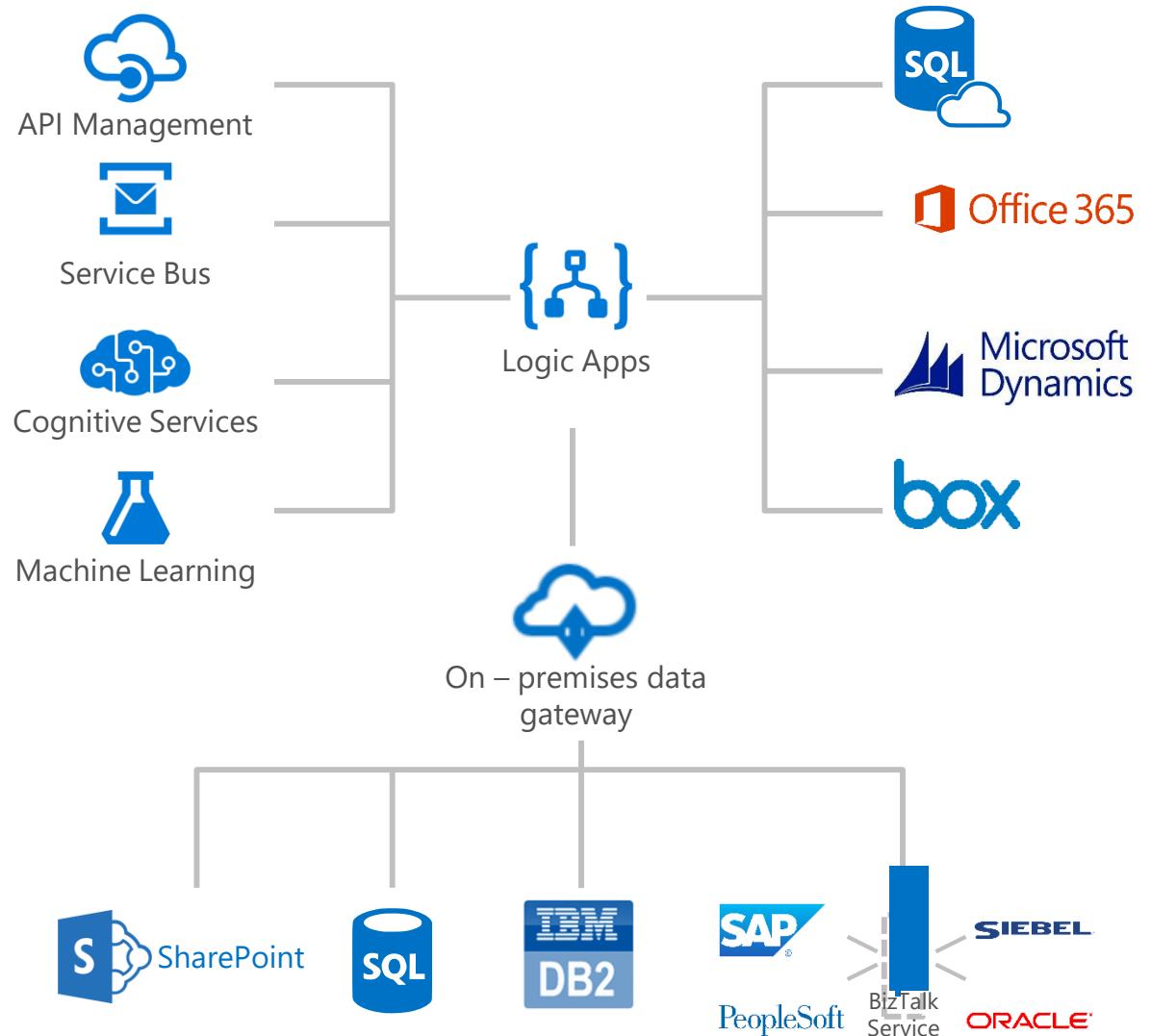
Azure Logic Apps

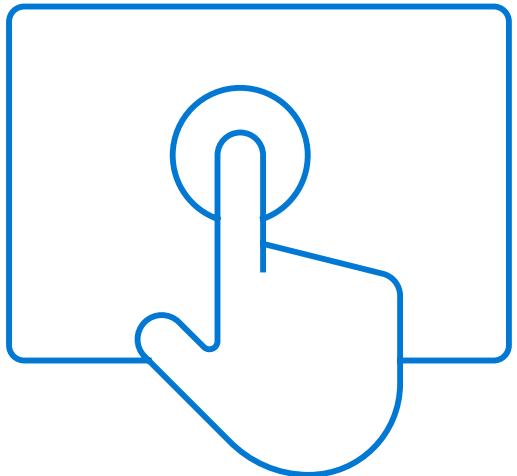
- Fast integration using innovative Visual Designer
- Easy workflow creation with triggers and actions
- More than 200+ connectors to mashup applications, data and services
- Built for mission critical 24x7 Enterprise Integration
- Devops built-in: Create, deploy, manage and monitor



Microsoft's Hybrid Integration Platform

- Connect on-premises, hybrid and cloud applications
- Run mission-critical, complex integration scenarios with ease
- Enhance business productivity by automating business processes where it makes sense, on-premises or in Azure
- Build “Smart” Integrations leveraging Machine Learning/Cognitive Services





Demo

Create Logic App using Azure Portal

Thank You!