Microsoft Azure
IoT and Advanced Cloud Services

Alon Fliess

Chief Software Architect

alonf@codevalue.net

http://blogs.microsoft.co.il/blogs/alon

MVP Microsoft Most Valuable Professional

Microsoft Regional Director

CodeValue
Inspiring Code, Creating Value.

Agenda

- Introduction to cloud computing
  - Microsoft Azure IaaS
  - Microsoft Azure PaaS
    - Azure Web Apps
    - Introduction to the MSA & Serverless
  - Microsoft Azure IoT
- Introduction to C# and Xamarin
- Advanced Azure Analysis Services

About Me

- Alon Fliess:
  - Chief Software Architect & Co-Founder at CodeValue Ltd.
  - More than 25 years of hands-on experience
  - Microsoft Regional Director & Microsoft MVP
  - Renowned speaker at both international and domestic events

About CodeValue

- Awesome software company!
- About 200 employees, most of which technology experts
- High quality software development solutions

- Cloud Computing
- Advanced Mobile Technologies
- UI/UX & Graphic Design
- Advanced Web Technologies
- Automated QA Testing
- Customer Support
- Performance & Debugging
- ALM & DevOps
- Software Architecture
- IoT & Embedded Software
- Training & Mentoring

ISO 9001 SGS

Israeli Azure Developer Community

- Curious about Azure? Join us!
- Meet every two months at Microsoft Ra’anana
- Next meeting – December 2015

- http://www.meetup.com/IsraeliAzureDevelopers
- https://www.facebook.com/IsraeliAzureDevelopers

Cloud Computing

Introduction to Cloud Computing

Cloud Computing - A Game Changing Technology

- Infinite shared resources & services
  - Infrastructure is not a limiting factor
  - No need to equip for peak-load requirements
- Elasticity on demand
  - Anytime, anywhere
- Efficient scalability and high availability
- Suitable pricing models
  - Pay for what you use

Cloud Computing

Evolution of Computing - The Next (Current) Big Thing

- Virtualization and Abstraction
  - Details are abstracted from consumers
    - Reduces complexity
- Not necessarily the Internet, can be on premises
  - Private cloud
- Automation, Monitoring, Deployment
  - Reduce cost, shift risk, shorten time-to-market, focus on business functionality

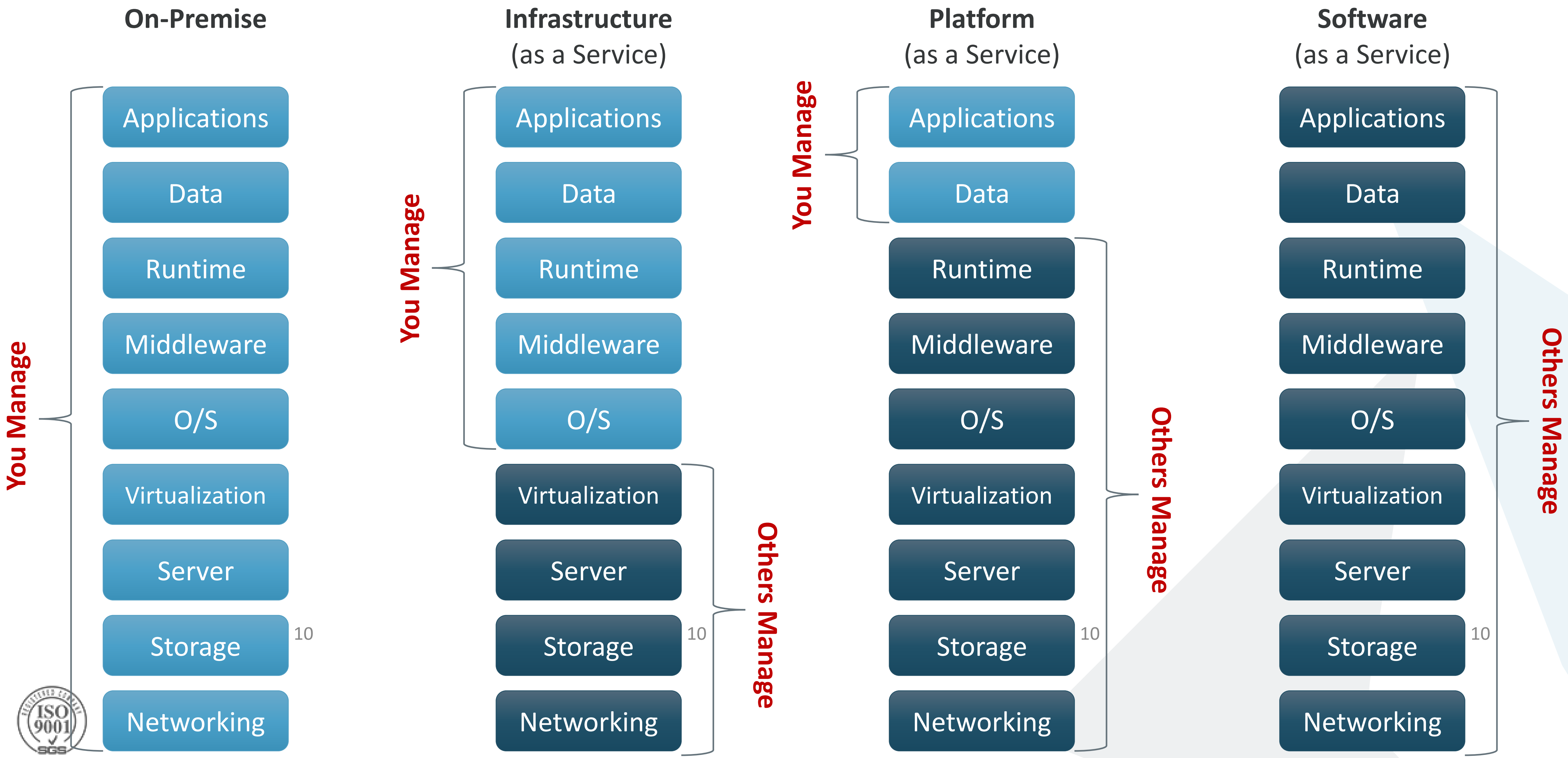
Less Worries

Focus on functionality

- Let others take care of:
  - Resource management
  - Security
  - Environments (staging, production)
  - High availability, scalability, load balancing
  - Fault tolerance
  - OS - installation, licensing, updates, patches
  - Network
  - Maintenance

ISO 9001 SGS

Why the Cloud? - IaaS, PaaS & SaaS



On-Premise

You Manage

Applications
Data
Runtime
Middleware
O/S
Virtualization
Server
Storage
Networking

Infrastructure
(as a Service)

You Manage

Applications
Data
Runtime
Middleware
O/S

Others Manage

Virtualization
Server
Storage
Networking

Platform
(as a Service)

You Manage

Applications
Data

Others Manage

Runtime
Middleware
O/S
Virtualization
Server
Storage
Networking

Software
(as a Service)

Others Manage

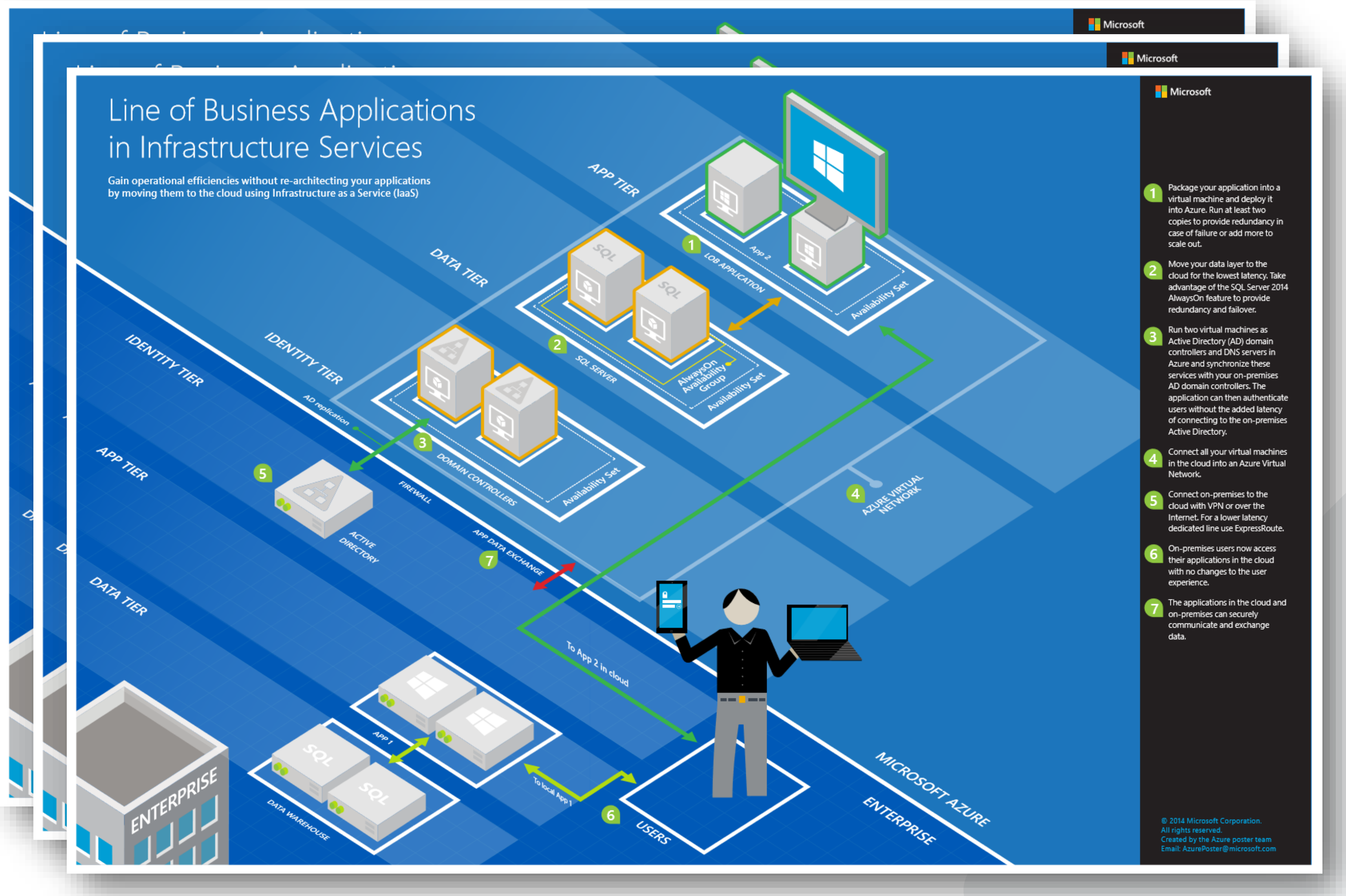
Applications
Data
Runtime
Middleware
O/S
Virtualization
Server
Storage
Networking

Introduction to Cloud Architecture

Cloud Computing - A Game Changing Technology

- Infinite shared resources & services
  - Infrastructure is not a limiting factor
  - No need to equip for peak-load requirements
- Elasticity on demand
  - Anytime, anywhere
- Efficient scalability and high availability
- Suitable pricing models
  - Pay for what you use
- **Less worries** about the foundation – **Better application quality!!!**

Azure Architecture Blueprints



Line of Business Applications in Infrastructure Services

Gain operational efficiencies without re-architecting your applications by moving them to the cloud using Infrastructure as a Service (IaaS)

1. Package your application into a virtual machine and deploy it into Azure. Run at least two copies to provide redundancy in case of failure or add more to scale out.
2. Move your data layer to the cloud for the lowest latency. Take advantage of the SQL Server 2014 AlwaysOn feature to provide redundancy and failover.
3. Run two virtual machines as Active Directory (AD) domain controllers and DNS servers in Azure and synchronize these services with your on-premises AD domain controllers. The application can then authenticate users without the added latency of connecting to the on-premises Active Directory.
4. Connect all your virtual machines in the cloud into an Azure Virtual Network.
5. Connect on-premises to the cloud with VPN or over the Internet. For a lower latency dedicated line use ExpressRoute.
6. On-premises users now access their applications in the cloud with no changes to the user experience.
7. The applications in the cloud and on-premises can securely communicate and exchange data.

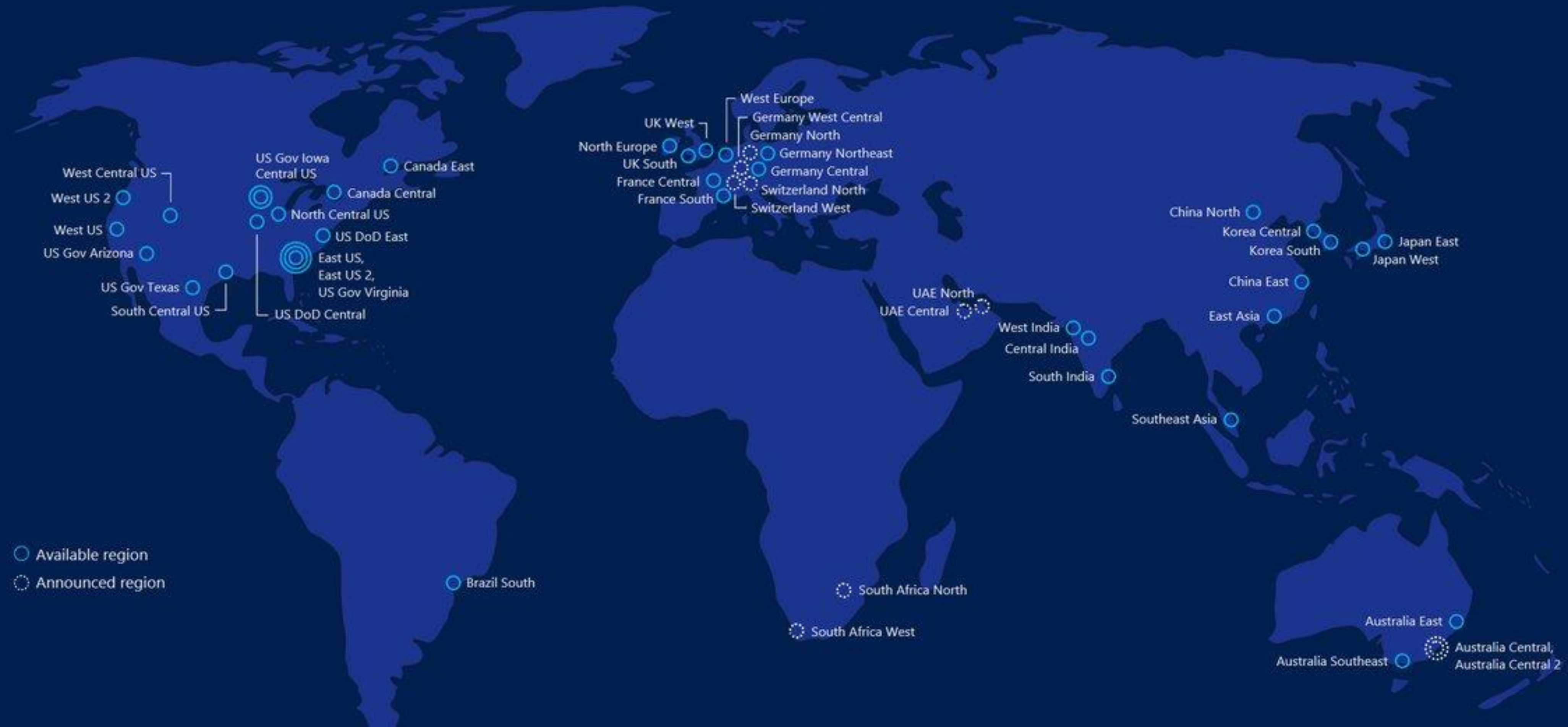
© 2014 Microsoft Corporation.
All rights reserved.
Created by the Azure poster team
Email: AzurePoster@microsoft.com

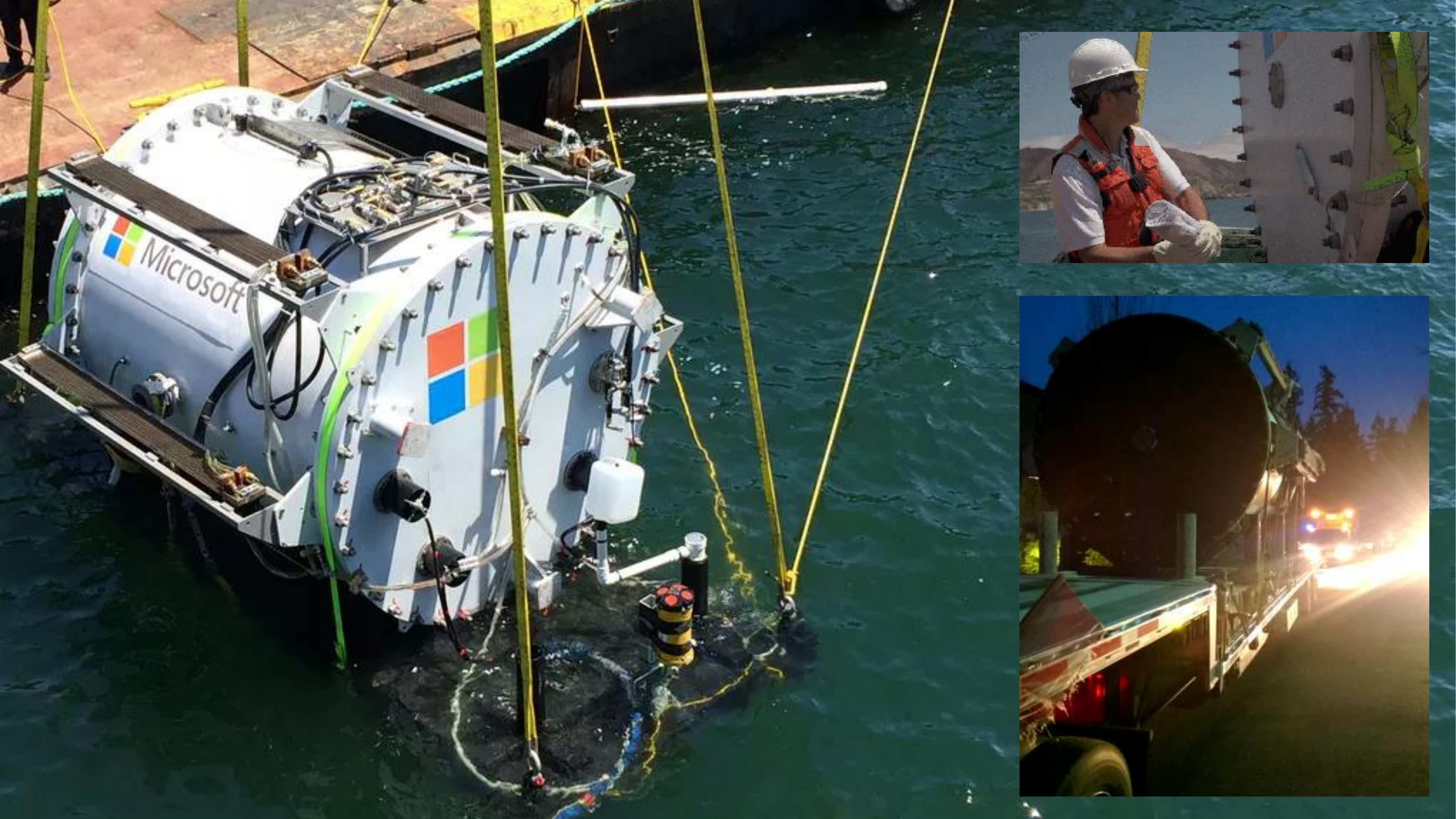
Microsoft Azure

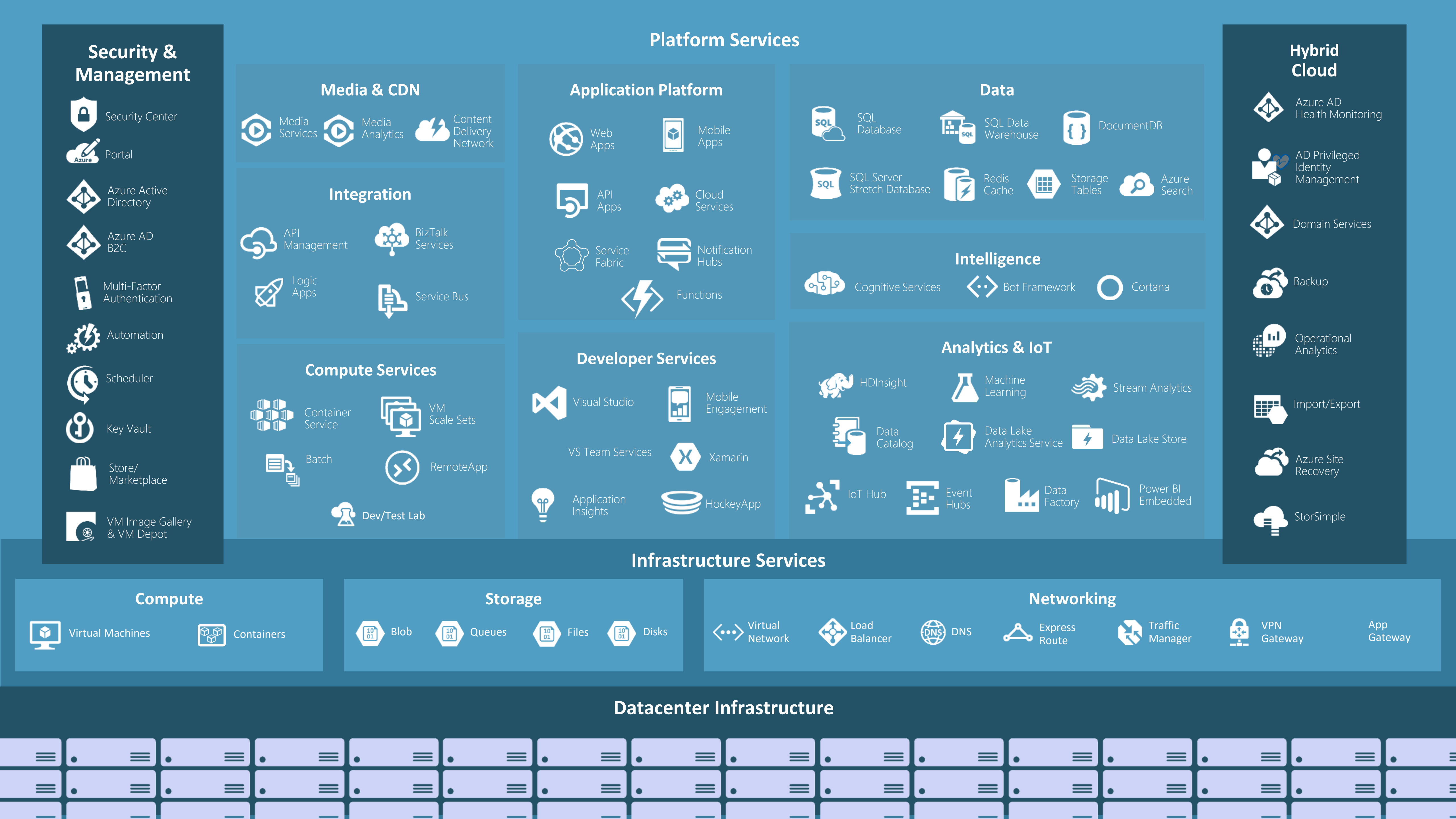
Azure regions

Azure has more global regions than any other cloud provider—offering the scale needed to bring applications closer to users around the world, preserving data residency, and offering comprehensive compliance and resiliency options for customers.

50 regions worldwide **140** available in 140 countries




Microsoft



Security & Management

- Security Center
- Portal
- Azure Active Directory
- Azure AD B2C
- Multi-Factor Authentication
- Automation
- Scheduler
- Key Vault
- Store/ Marketplace
- VM Image Gallery & VM Depot

Platform Services

Media & CDN

- Media Services
- Media Analytics
- Content Delivery Network

Integration

- API Management
- BizTalk Services
- Logic Apps
- Service Bus

Compute Services

- Container Service
- VM Scale Sets
- Batch
- RemoteApp
- Dev/Test Lab

Application Platform

- Web Apps
- Mobile Apps
- API Apps
- Cloud Services
- Service Fabric
- Notification Hubs
- Functions

Developer Services

- Visual Studio
- Mobile Engagement
- VS Team Services
- Xamarin
- Application Insights
- HockeyApp

Data

- SQL Database
- SQL Data Warehouse
- DocumentDB
- SQL Server Stretch Database
- Redis Cache
- Storage Tables
- Azure Search

Intelligence

- Cognitive Services
- Bot Framework
- Cortana

Analytics & IoT

- HDInsight
- Machine Learning
- Stream Analytics
- Data Catalog
- Data Lake Analytics Service
- Data Lake Store
- IoT Hub
- Event Hubs
- Data Factory
- Power BI Embedded

Hybrid Cloud

- Azure AD Health Monitoring
- AD Privileged Identity Management
- Domain Services
- Backup
- Operational Analytics
- Import/Export
- Azure Site Recovery
- StorSimple

Infrastructure Services

Compute

- Virtual Machines
- Containers

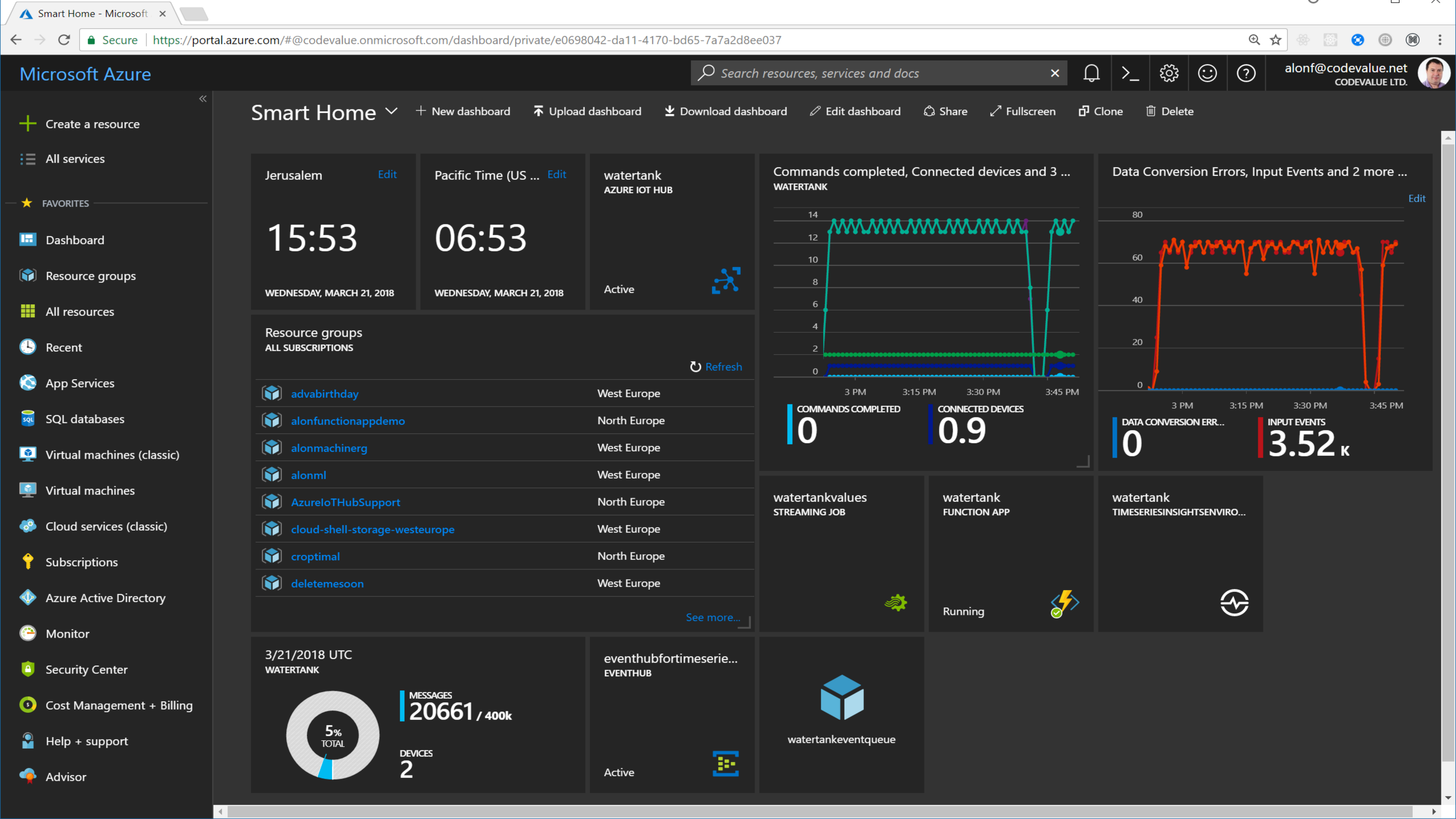
Storage

- Blob
- Queues
- Files
- Disks

Networking

- Virtual Network
- Load Balancer
- DNS
- Express Route
- Traffic Manager
- VPN Gateway
- App Gateway

Datacenter Infrastructure



Jerusalem

15:53

WEDNESDAY, MARCH 21, 2018

Pacific Time (US ...

06:53

WEDNESDAY, MARCH 21, 2018

watertank
AZURE IOT HUB

Active

Commands completed, Connected devices and 3 ...
WATERTANK

COMMANDS COMPLETED 0

CONNECTED DEVICES 0.9

Data Conversion Errors, Input Events and 2 more ...

DATA CONVERSION ERR... 0

INPUT EVENTS 3.52 K

Resource groups
ALL SUBSCRIPTIONS

advabirthday	West Europe
alonfunctionappdemo	North Europe
alonmachinerg	West Europe
alonml	West Europe
AzureIoTHubSupport	North Europe
cloud-shell-storage-westeurope	West Europe
croptimal	North Europe
deletemesoon	West Europe

watertankvalues
STREAMING JOB

watertank
FUNCTION APP

Running

watertank
TIMESERIESINSIGHTSENVIRO...

3/21/2018 UTC
WATERTANK

5% TOTAL

MESSAGES 20661 / 400k

DEVICES 2

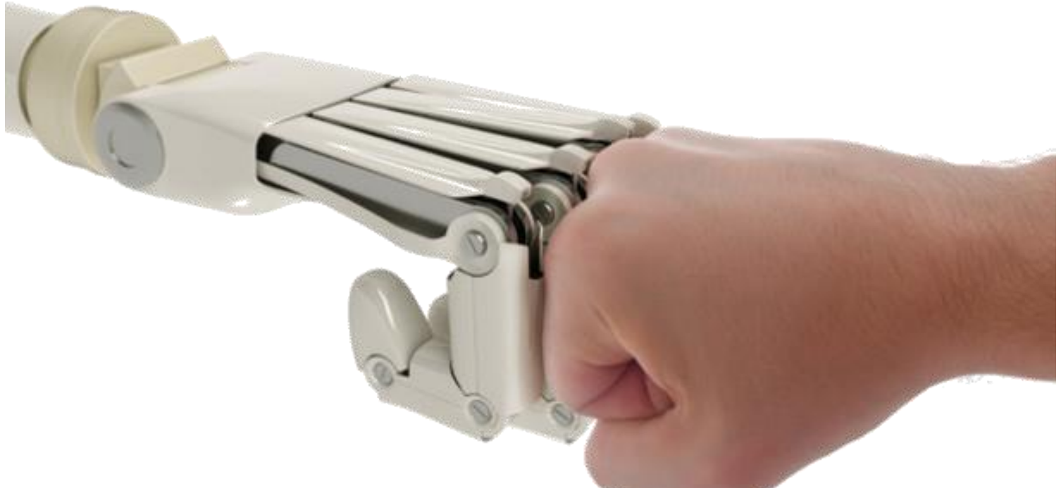
eventhubfortimeserie...
EVENTHUB

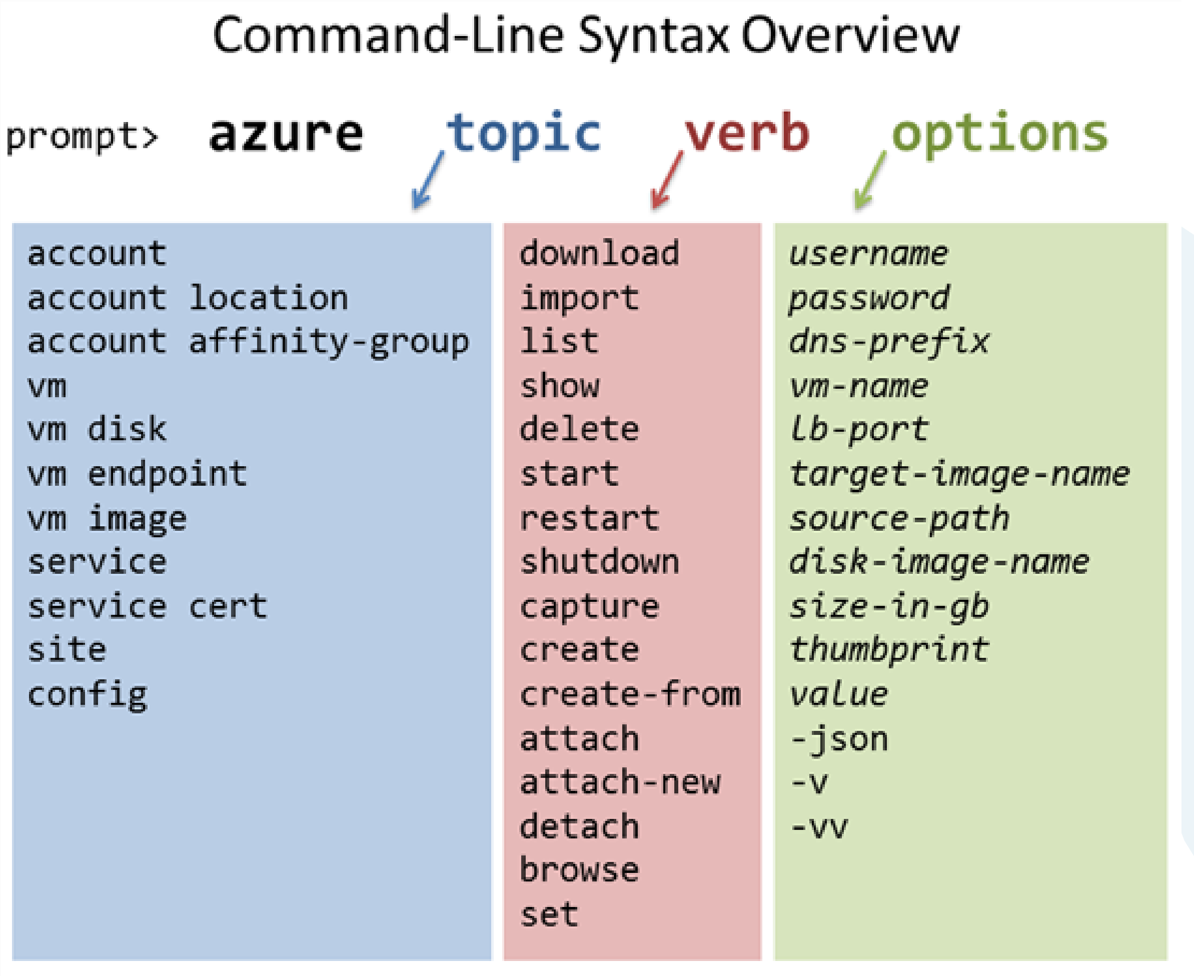
Active

watertankeventqueue

Automation is key

- Manual deployment, management and resources-definition can only take you so far
- Automate Azure with:
  - [Azure PowerShell]
  - [Azure CLI](Windows, Mac, Linux)
  - Azure Resource Manager (ARM)





Command-Line Syntax Overview

```
prompt> azure topic verb options
```

account
account location
account affinity-group
vm
vm disk
vm endpoint
vm image
service
service cert
site
config

download
import
list
show
delete
start
restart
shutdown
capture
create
create-from
attach
attach-new
detach
browse
set

username
password
dns-prefix
vm-name
lb-port
target-image-name
source-path
disk-image-name
size-in-gb
thumbprint
value
-json
-v
-vv

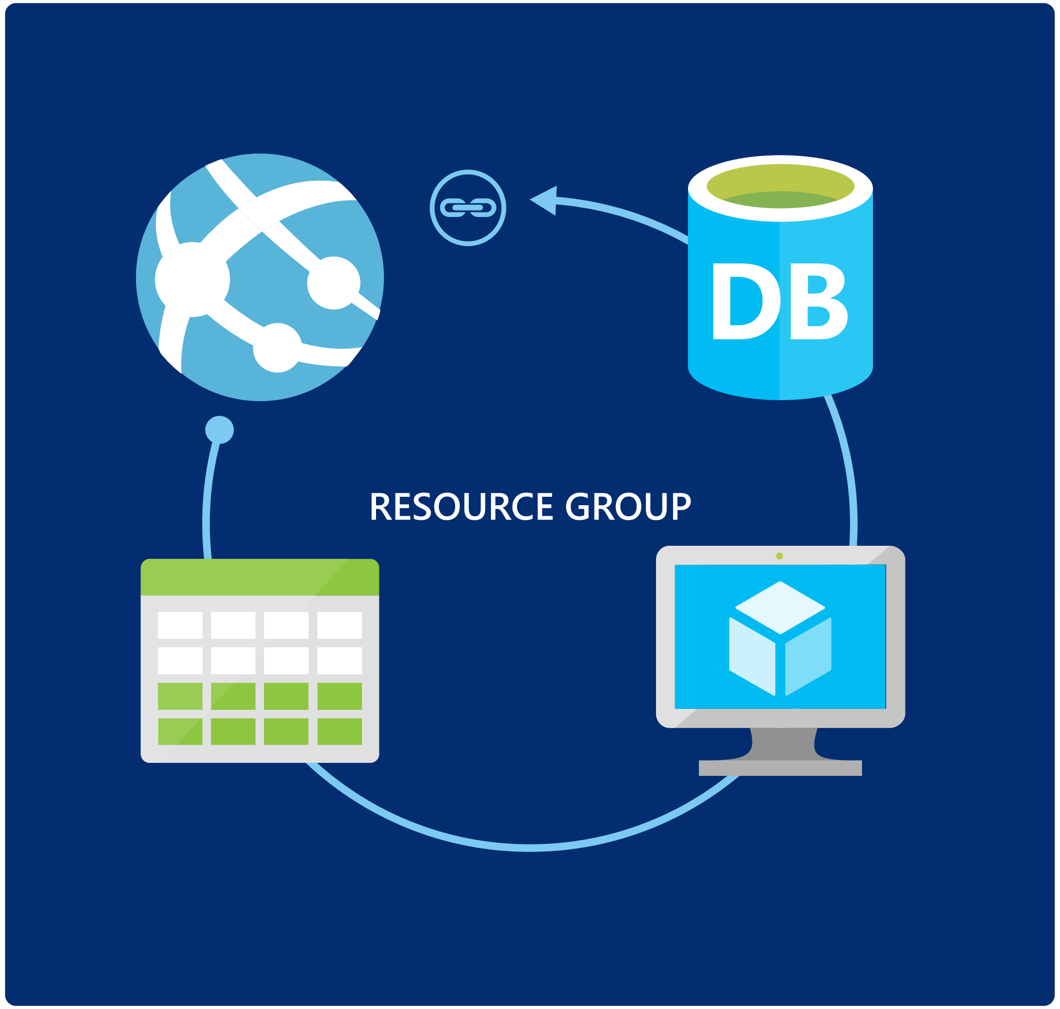
Azure Resource Manager (ARM)

- A holistic view of the entire cloud application resources
- Instead of looking at them as separate entities, they are part of a whole
- Benefits
  - Deploy, manage and monitor the entire solution group
  - Repeatedly deploy your solution
  - Declarative templates
  - Easy resource dependency management (ordered deployment)
  - Native RBAC support to all services in the resource group
  - Simple logical organization with tags
  - Clear billing

Resource Groups

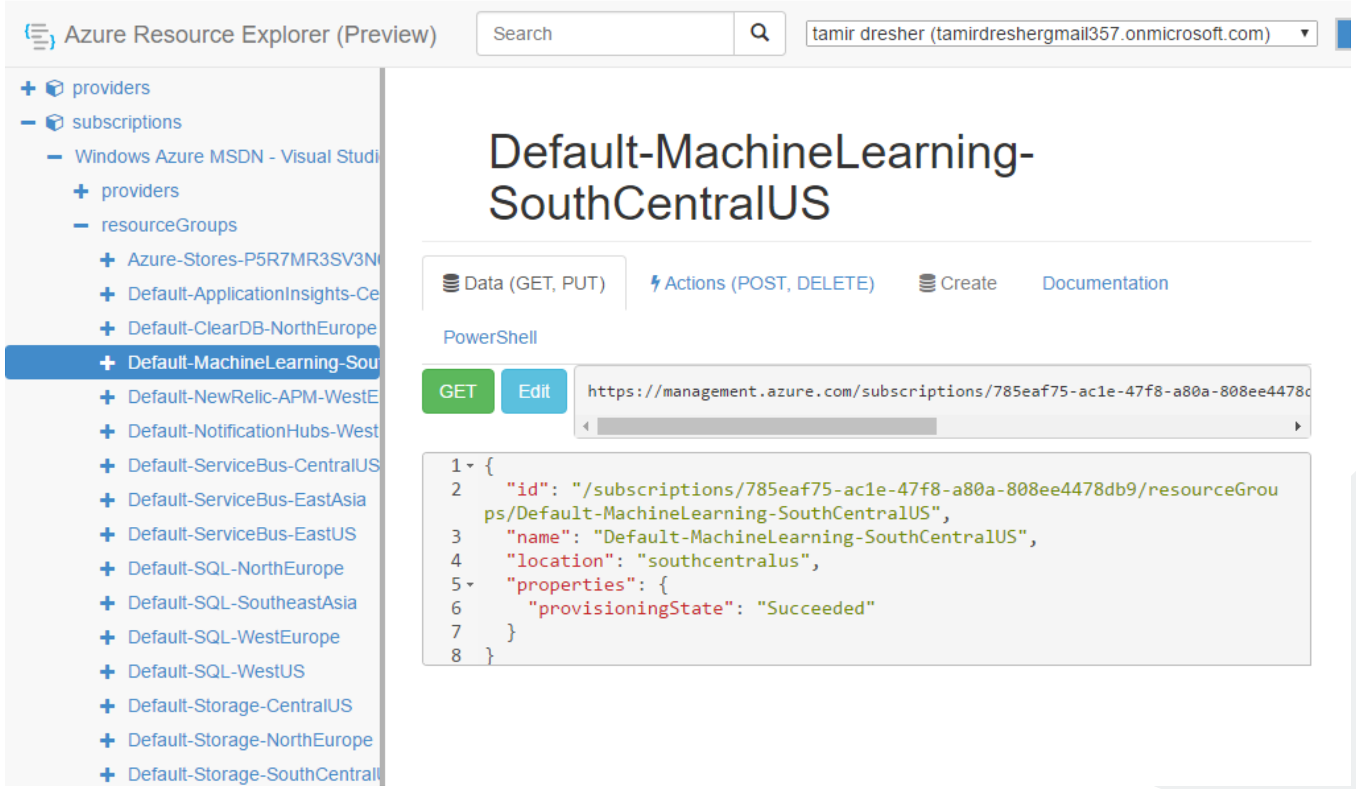
- Tightly coupled containers of multiple resources
- Every resource exist in one (and only one) resource group
- Resource groups can span regions

- You can deploy, update, delete a resource group as a whole
- Easy to understand billing



Azure Resource Explorer

- resources.azure.com
- Resources management site
- “Explorer like” experience for browsing subscriptions, providers, resources and templates



Azure Resource Explorer (Preview)
Search
tamir dresher (tamirdreshergmail357.onmicrosoft.com)
providers
subscriptions
Windows Azure MSDN - Visual Studi
providers
resourceGroups
Azure-Stores-P5R7MR3SV3N
Default-ApplicationInsights-Ce
Default-ClearDB-NorthEurope
Default-MachineLearning-Sou
Default-NewRelic-APM-WestE
Default-NotificationHubs-West
Default-ServiceBus-CentralUS
Default-ServiceBus-EastAsia
Default-ServiceBus-EastUS
Default-SQL-NorthEurope
Default-SQL-SoutheastAsia
Default-SQL-WestEurope
Default-SQL-WestUS
Default-Storage-CentralUS
Default-Storage-NorthEurope
Default-Storage-SouthCentral
Data (GET, PUT)
Actions (POST, DELETE)
Create
Documentation
PowerShell
GET
Edit

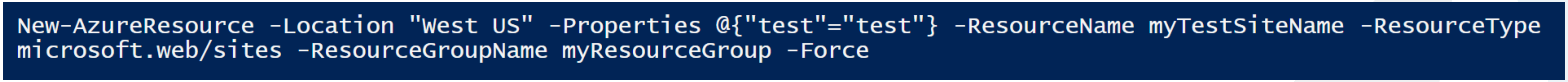
Default-MachineLearning-SouthCentralUS

```
https://management.azure.com/subscriptions/785eaf75-ac1e-47f8-a80a-808ee4478c
```

```
{
  "id": "/subscriptions/785eaf75-ac1e-47f8-a80a-808ee4478db9/resourceGroups/Default-MachineLearning-SouthCentralUS",
  "name": "Default-MachineLearning-SouthCentralUS",
  "location": "southcentralus",
  "properties": {
    "provisioningState": "Succeeded"
  }
}
```

ARM cmdlets

- Execute and manage deployments
  - `New-AzureResourceGroupDeployment`
- Create individual resources
  - `New-AzureRmResource`
- Invoke specific actions on existing resources
  - `Invoke-AzureRmResourceAction`



```
New-AzureResource -Location "West US" -Properties @{"test"="test"} -ResourceName myTestSiteName -ResourceType
microsoft.web/sites -ResourceGroupName myResourceGroup -Force
```

ARM Templates

- Declarative JSON files that specifies resource and their dependencies
- Idempotent
- Parametrized
- Source-control friendly

imperative

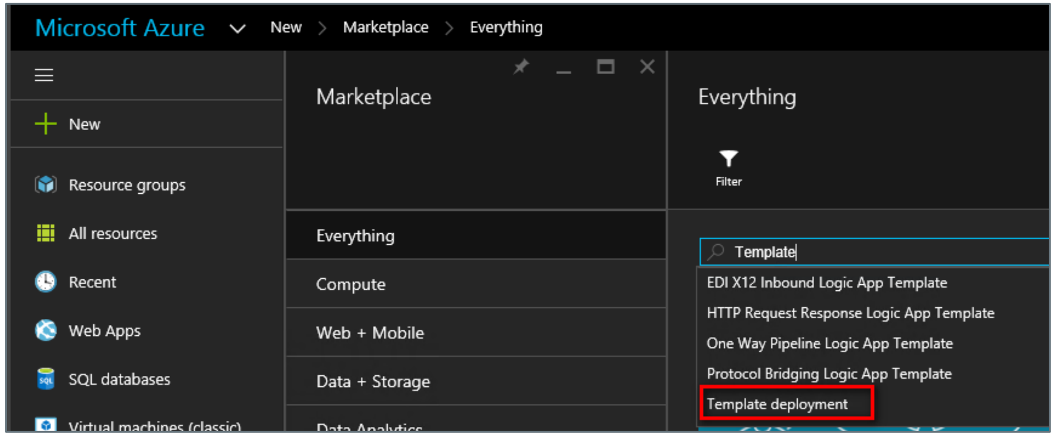
```
New-AzureVM -VM $myVM
New-AzureRmStorageAccount -StorageAccountName $acct
Set-AzureRmVNetConfig -ConfigurationPath -Path
```

declarative

```
{
 "$schema": "https://../deploymentTemplate.json#",
 "contentVersion": "1.0.0.0",
 "parameters": {},
 "variables": {},
 "resources": [],
 "outputs": {}
}
```

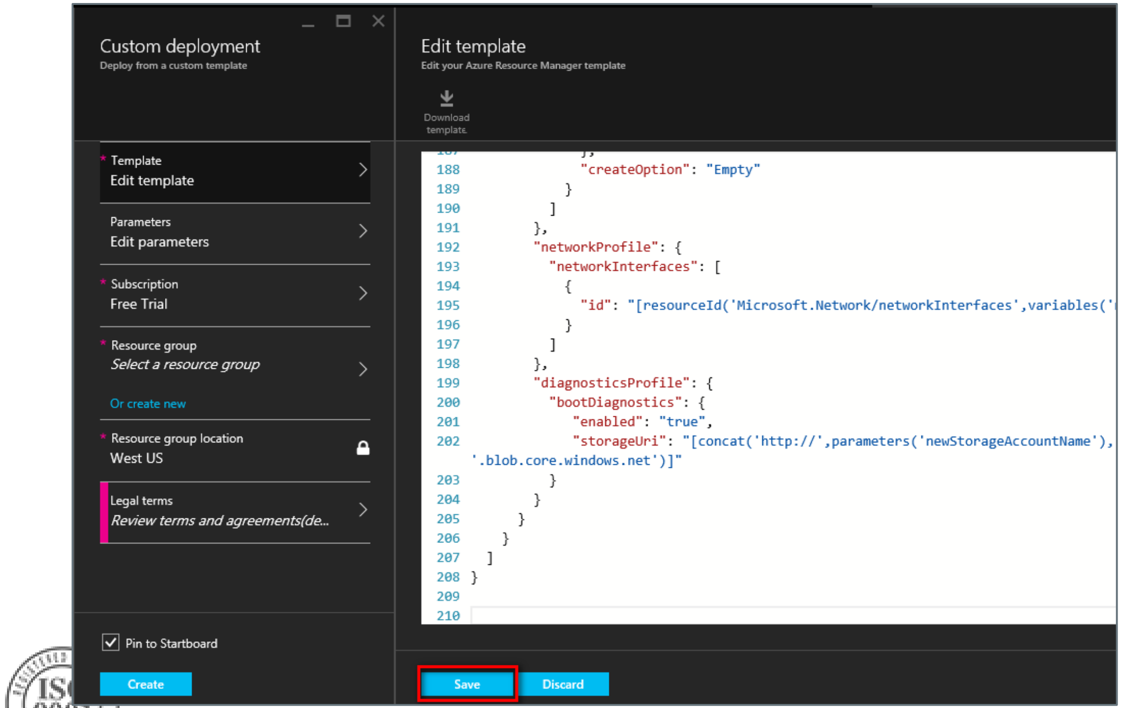
Http://github.com/rjmax/ArmExamples

Deploying Templates



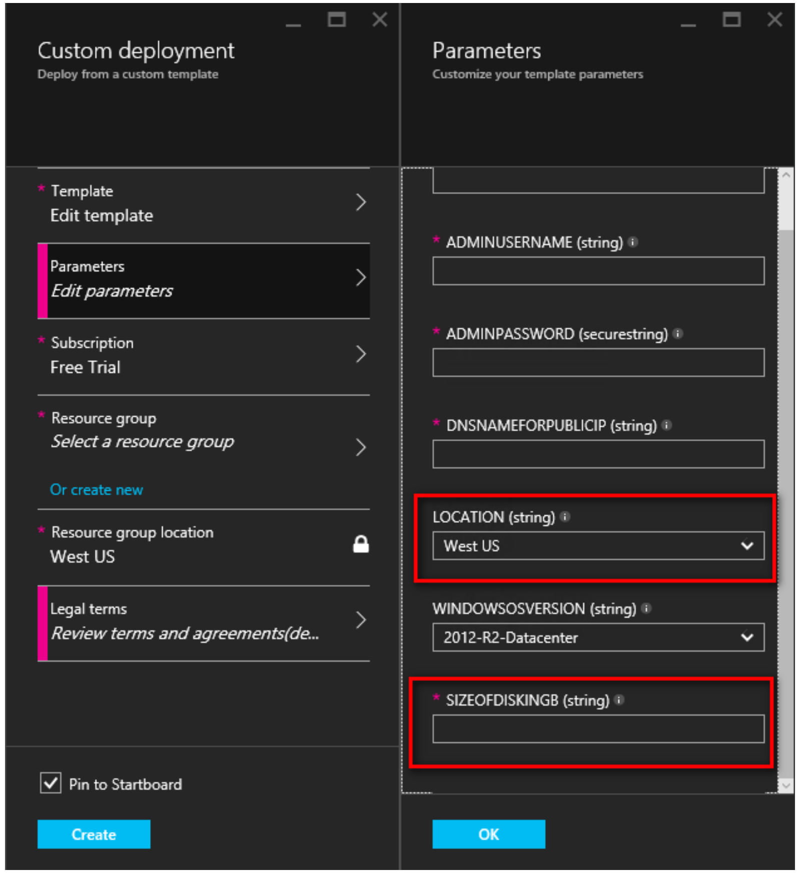
Template

- EDI X12 Inbound Logic App Template
- HTTP Request Response Logic App Template
- One Way Pipeline Logic App Template
- Protocol Bridging Logic App Template
- Template deployment



- Template: Edit template
- Parameters: Edit parameters
- Subscription: Free Trial
- Resource group: *Select a resource group*
- Resource group location: West US
- Legal terms: *Review terms and agreements(de...*
- ☑ Pin to Startboard

```
                    }],
                    "createOption": "Empty"
                  }
                ]
              },
              "networkProfile": {
                "networkInterfaces": [
                  {
                    "id": "[resourceId('Microsoft.Network/networkInterfaces',variables('
                  }
                ]
              },
              "diagnosticsProfile": {
                "bootDiagnostics": {
                   "enabled": "true",
                   "storageUri": "[concat('http://',parameters('newStorageAccountName'),
 '.blob.core.windows.net')]"
                }
              }
            }
          }
        ]
      }
```


Custom deployment
Deploy from a custom template
Parameters
Customize your template parameters
Or create new
Create
OK

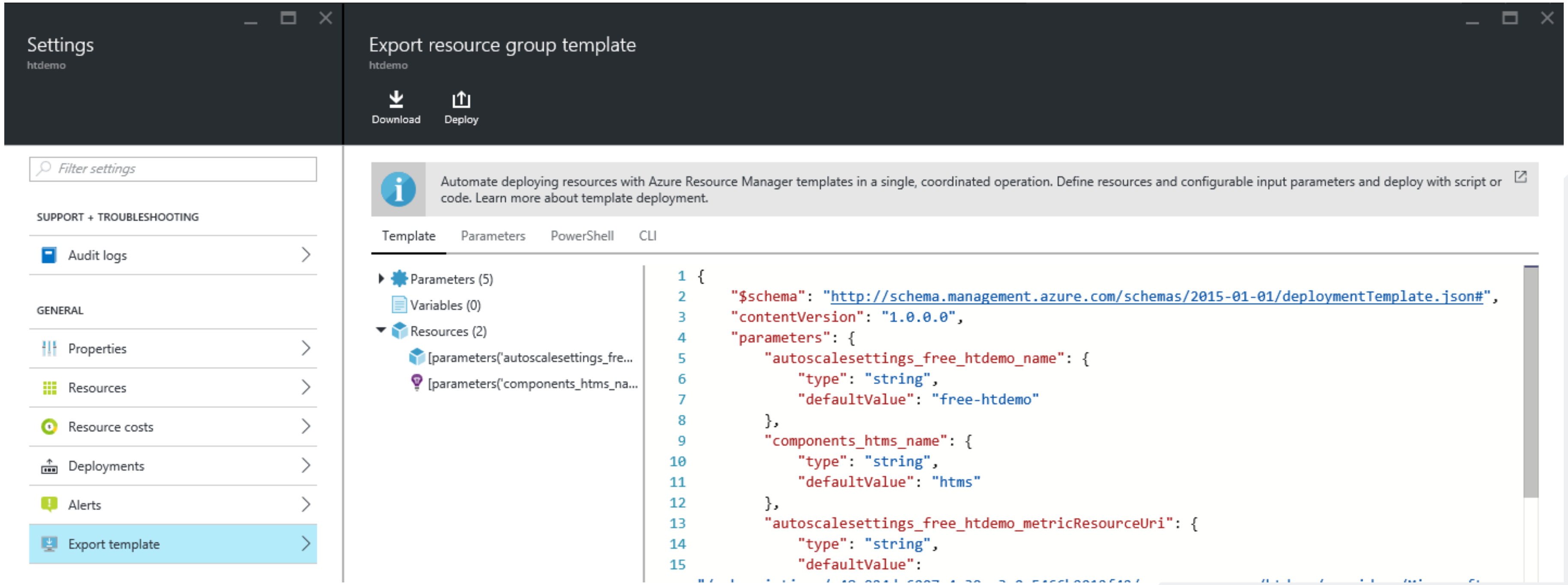
- Template: Edit template
- Parameters: *Edit parameters*
- Subscription: Free Trial
- Resource group: *Select a resource group*
- Resource group location: West US
- Legal terms: *Review terms and agreements(de...*
- ☑ Pin to Startboard

- ADMINUSERNAME (string):
- ADMINPASSWORD (securestring):
- DNSNAMEFORPUBLICIP (string):
- LOCATION (string): West US
- WINDOWSOSVERSION (string): 2012-R2-Datacenter
- SIZEOFDISKINGB (string):

```
New-AzureRmResourceGroupDeployment -DeploymentName
"Simple-VM" -ResourceGroupName
RG-AZITCAMP -TemplateFile "C:\GitHub\Templates\101-
simple-windows-vm\azuredeploy.json
```

Creating\Editing Templates

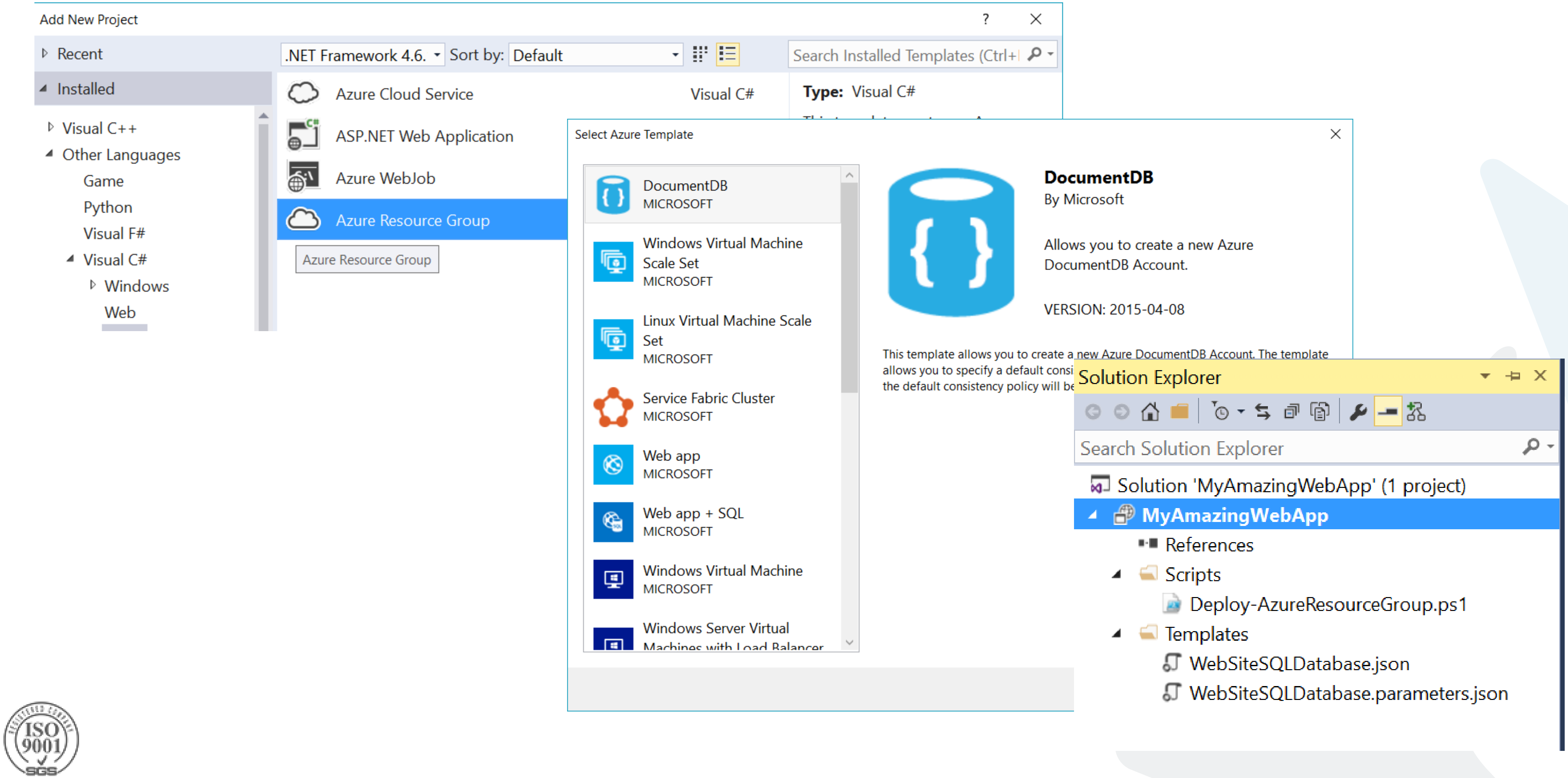
- Visual Studio
- Visual Studio Code
- http://armviz.io/
- [Export Resource Group Templates]()



Automate deploying resources with Azure Resource Manager templates in a single, coordinated operation. Define resources and configurable input parameters and deploy with script or code. Learn more about template deployment.

```
{
    "$schema": "http://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#",
    "contentVersion": "1.0.0.0",
    "parameters": {
        "autoscalesettings_free_htdemo_name": {
            "type": "string",
            "defaultValue": "free-htdemo"
        },
        "components_htms_name": {
            "type": "string",
            "defaultValue": "htms"
        },
        "autoscalesettings_free_htdemo_metricResourceUri": {
            "type": "string",
            "defaultValue":
```

Azure Resource Group Project



Add New Project

- Recent
- Installed
  - Visual C++
  - Other Languages
    - Game
    - Python
    - Visual F#
  - Visual C#
    - Windows
    - Web

.NET Framework 4.6. Sort by: Default

Search Installed Templates (Ctrl+

Azure Cloud Service	Visual C#
ASP.NET Web Application	
Azure WebJob	
Azure Resource Group	

Azure Resource Group

Type: Visual C#

Select Azure Template

- DocumentDB
MICROSOFT
- Windows Virtual Machine Scale Set
MICROSOFT
- Linux Virtual Machine Scale Set
MICROSOFT
- Service Fabric Cluster
MICROSOFT
- Web app
MICROSOFT
- Web app + SQL
MICROSOFT
- Windows Virtual Machine
MICROSOFT
- Windows Server Virtual Machines with Load Balancer

DocumentDB
By Microsoft

Allows you to create a new Azure DocumentDB Account.

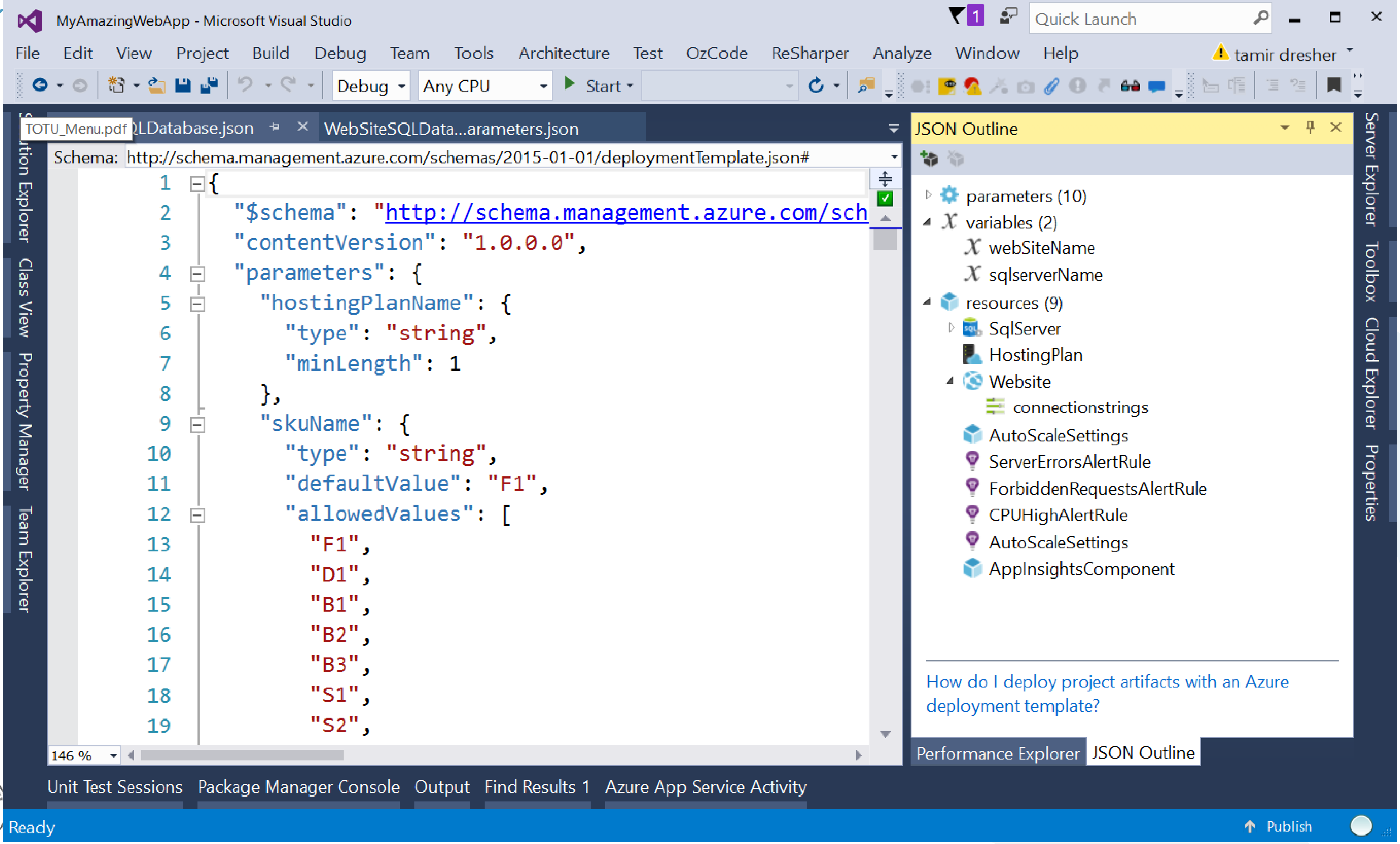
VERSION: 2015-04-08

This template allows you to create a new Azure DocumentDB Account. The template allows you to specify a default consi
the default consistency policy will be

Solution Explorer

Search Solution Explorer

- Solution 'MyAmazingWebApp' (1 project)
  - MyAmazingWebApp
    - References
    - Scripts
      - Deploy-AzureResourceGroup.ps1
    - Templates
      - WebSiteSQLDatabase.json
      - WebSiteSQLDatabase.parameters.json



Schema: http://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#

```
{
  "$schema": "http://schema.management.azure.com/sch
  "contentVersion": "1.0.0.0",
  "parameters": {
    "hostingPlanName": {
      "type": "string",
      "minLength": 1
    },
    "skuName": {
      "type": "string",
      "defaultValue": "F1",
      "allowedValues": [
        "F1",
        "D1",
        "B1",
        "B2",
        "B3",
        "S1",
        "S2",
```

- parameters (10)
- variables (2)
  - webSiteName
  - sqlserverName
- resources (9)
  - SqlServer
  - HostingPlan
  - Website
    - connectionstrings
  - AutoScaleSettings
  - ServerErrorsAlertRule
  - ForbiddenRequestsAlertRule
  - CPUHighAlertRule
  - AutoScaleSettings
  - AppInsightsComponent

How do I deploy project artifacts with an Azure deployment template?

More Azure Deployment Links

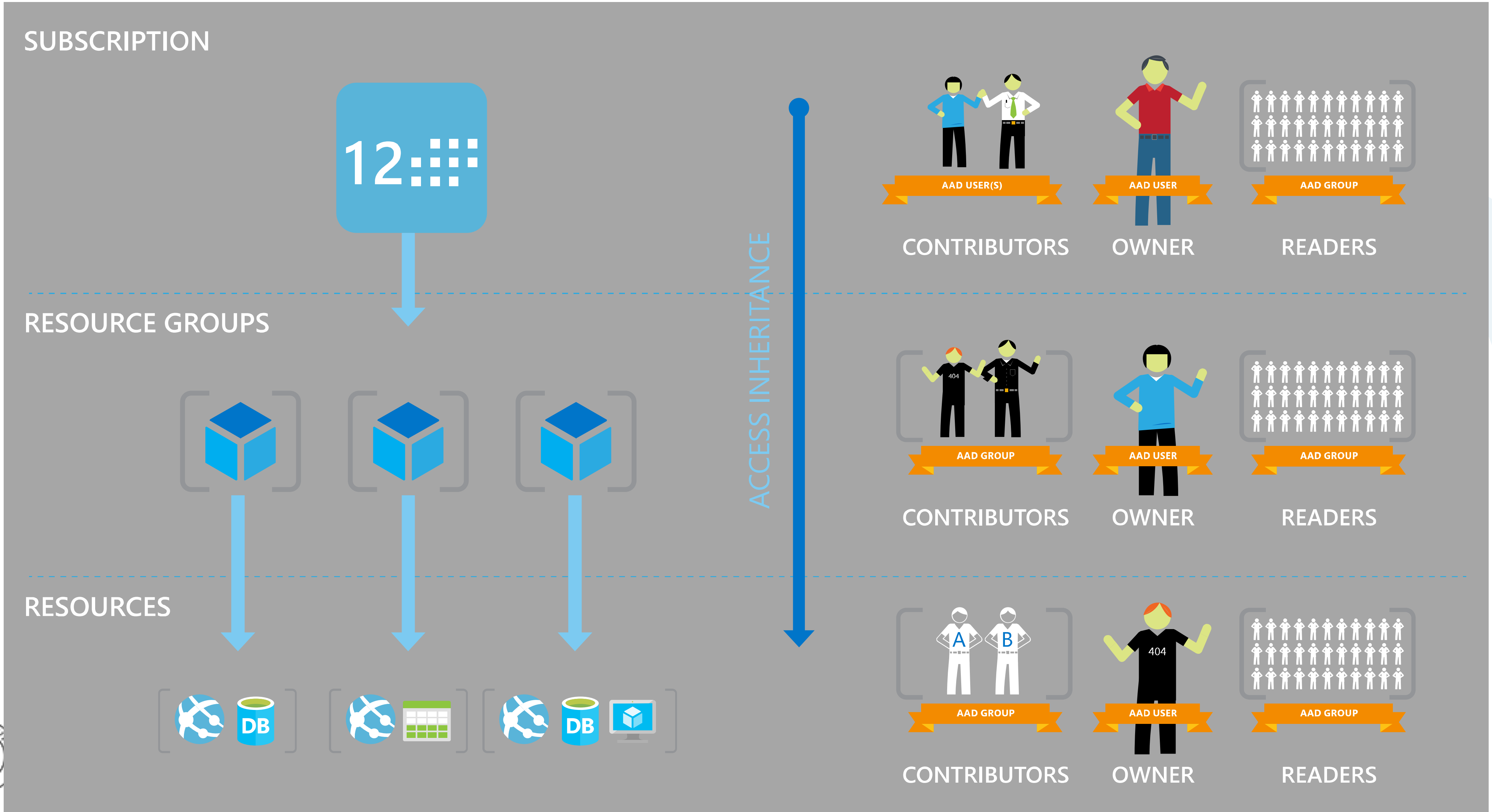
- https://github.com/Azure/azure-quickstart-templates/
- https://azure.microsoft.com/en-us/documentation/templates/
- http://azure.microsoft.com/en-us/documentation/articles/resource-group-overview/
- https://azure.microsoft.com/en-us/documentation/articles/best-practices-resource-manager-design-templates/
- https://github.com/Azure/azure-content/blob/master/articles/app-service-web/app-service-deploy-complex-application-predictably.md

Role Based Access Control

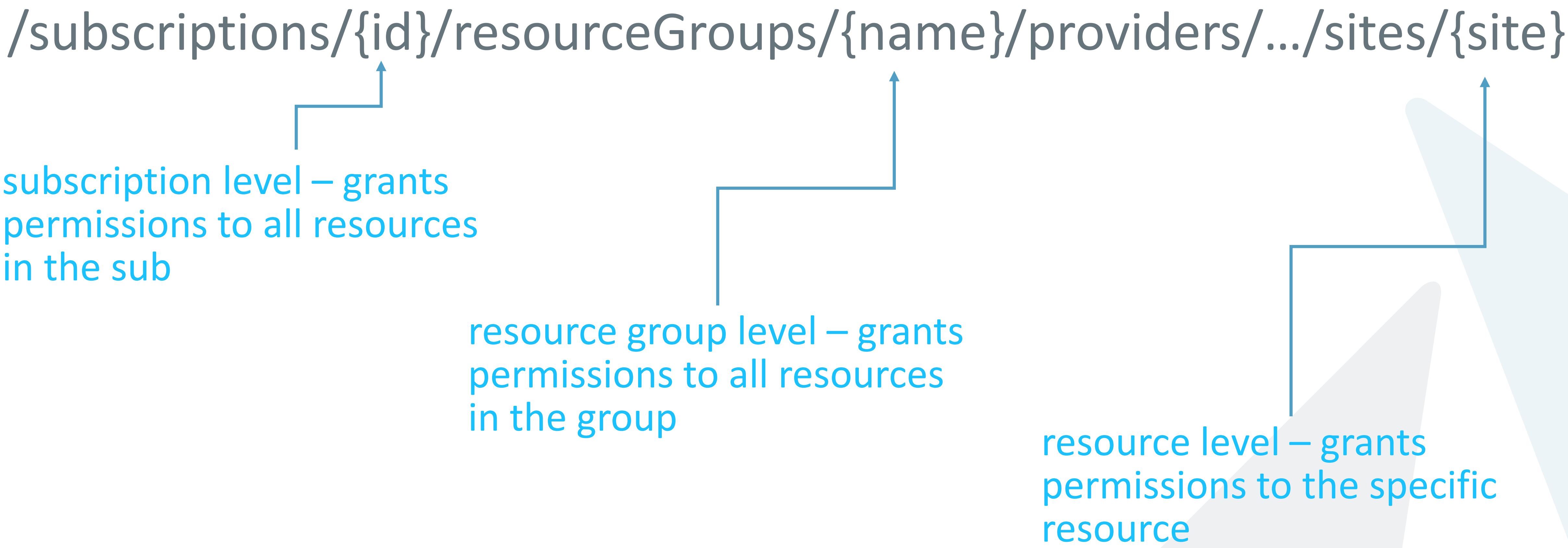
- Allows secure access with granular permissions
- Assignable to users, groups, or service principals
- Built-in roles make it easy to get started
- Key Concepts:
  - Role Definitions – the set of permissions
  - Role Assignments - associate role definitions with an identity
    - Identity == user/group
    - Assignemt is per scope (Directory/Subscription/Resource Group/Resource)
    - Inherited – subscription assignments apply to all resources

ISO 9001 SGS

Role Based Access Control



RBAC Scope



/subscriptions/{id}/resourceGroups/{name}/providers/…/sites/{site}

subscription level – grants permissions to all resources in the sub

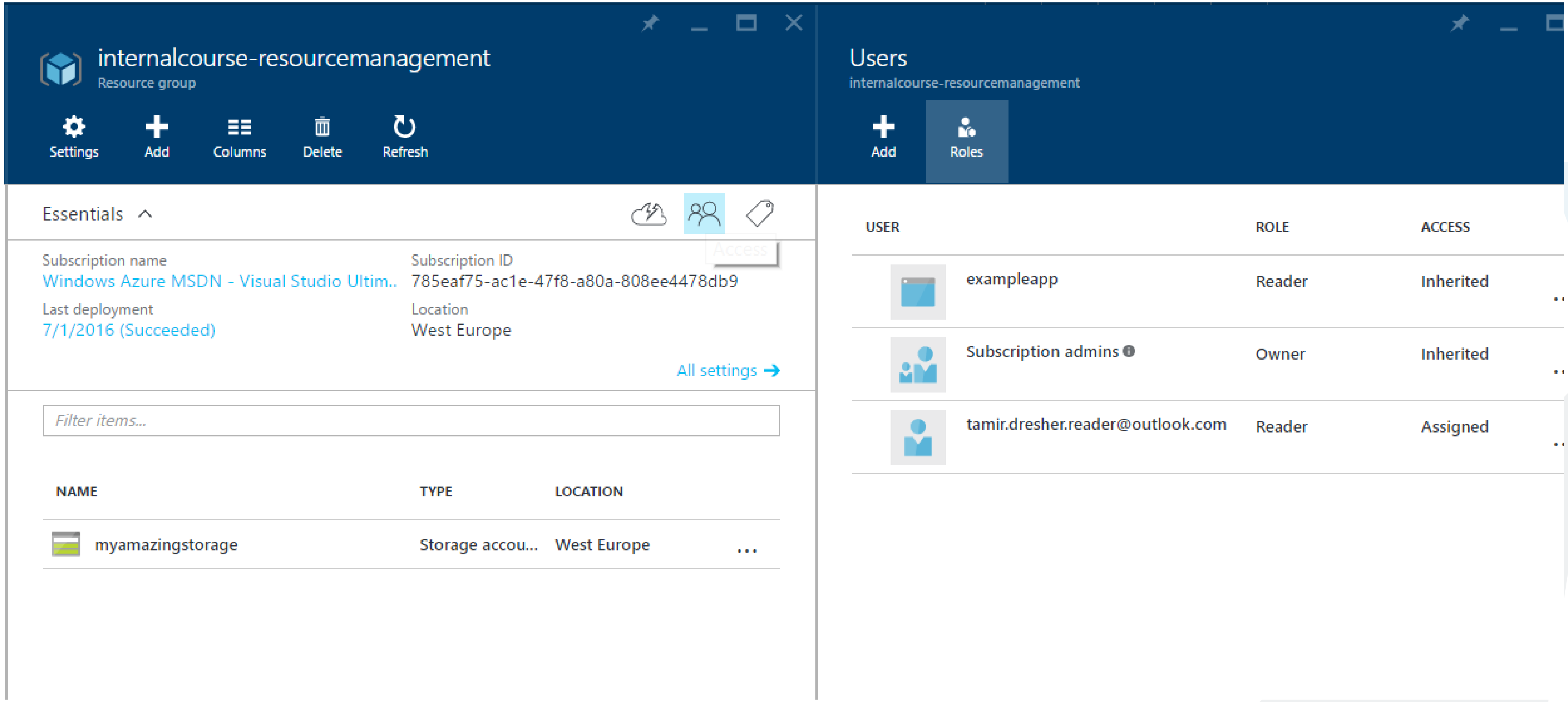
resource group level – grants permissions to all resources in the group

resource level – grants permissions to the specific resource

Built-in Roles

Role name	Permissions
Owner	Full management rights
Contributor	Full management rights except for user management
Reader	View resources and their settings
None	Does not see resources

RBAC in the portal



internalcourse-resourcemanagement
Resource group

Subscription name
Windows Azure MSDN - Visual Studio Ultim..

Subscription ID
785eaf75-ac1e-47f8-a80a-808ee4478db9

Last deployment
7/1/2016 (Succeeded)

Location
West Europe

Filter items...

NAME	TYPE	LOCATION
myamazingstorage	Storage accou...	West Europe

Users
internalcourse-resourcemanagement

USER	ROLE	ACCESS
exampleapp	Reader	Inherited
Subscription admins	Owner	Inherited
tamir.dresher.reader@outlook.com	Reader	Assigned

Azure Virtual machines

Azure Virtual Machines

- Launch Windows Server and Linux in minutes
- Scale from 1 to 1000s of VM Instances
- Save money with per-minute billing
- Open and extensible

Microsoft SUSE redhat OpenLogic

ubuntu ORACLE CoreOS

Provisioning VMs

Getting Started

Management Portal

Scripting
(Windows, Linux and Mac)

REST API

Select Image and VM Size

Windows Server

Linux

General Purpose
- Basic
- Standard

Optimized Compute

Performance Optimized

Network Optimized

New Disk Persisted in Storage

Boot VM from New Disk

Cloud

VM Gallery

- A collection of prebuilt images for various workloads

Windows Server 2012 R2

Ubuntu Server 14.04 LTS

CentOS 6.5

SUSE Linux Enterprise Server

openSUSE 13.1

Oracle Linux 6.4.0.0.0

SQL Server 2014 Standard

Oracle Database 11g R2

BizTalk Server 2013

SharePoint Server Farm

Microsoft Dynamics GP 2013

Zulu 8

SAP HANA Developer Edition

Puppet Enterprise 3.2.3

Barracuda Web Application

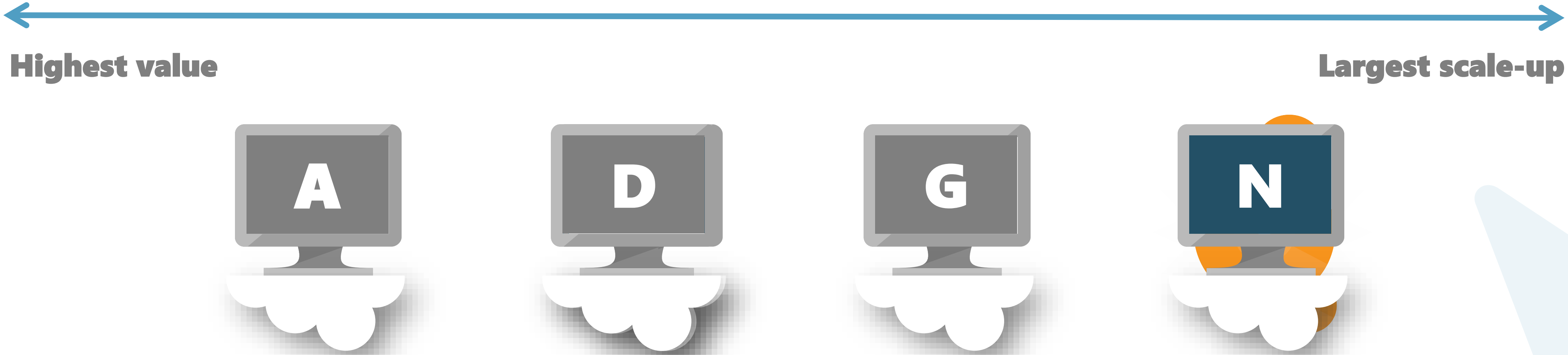
Oracle WebLogic Server 12.1.2

Visual Studio Ultimate 2013

Windows 8.1 Enterprise

ISO 9001 SGS

Scale Up Options



Highest value

New generation of D family VMs

SSD Storage
Fast CPUs

35% faster than D

Intel E5-2673 v3 CPUs

Most memory
fastest CPUs

GPU-enabled virtual machines

>80,000 IOPS
Premium Storage

NVIDIA GPUs

Remote visualization

Compute-intensive + RDMA

The G family

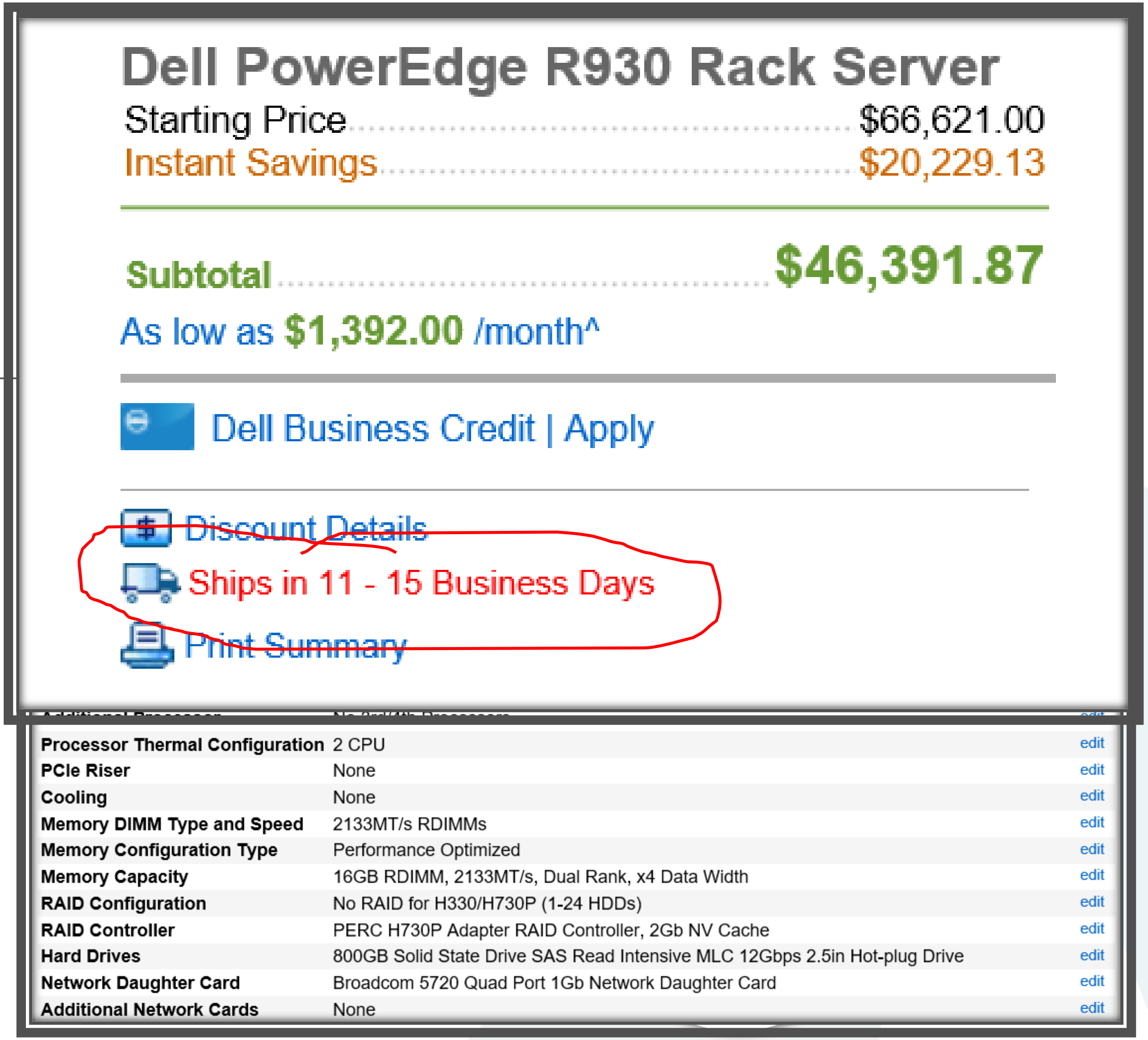
Optimized for data workloads

Up to 32 CPU cores, 448 GB RAM

6.5 TB local SSD

Latest generation Intel processor

Up to 64 attached disks!!



Dell PowerEdge R930 Rack Server

Starting Price	$66,621.00
Instant Savings	$20,229.13
Subtotal	$46,391.87

As low as $1,392.00 /month^

Dell Business Credit | Apply

Discount Details

Ships in 11 - 15 Business Days

Print Summary

Processor Thermal Configuration	2 CPU	edit
PCIe Riser	None	edit
Cooling	None	edit
Memory DIMM Type and Speed	2133MT/s RDIMMs	edit
Memory Configuration Type	Performance Optimized	edit
Memory Capacity	16GB RDIMM, 2133MT/s, Dual Rank, x4 Data Width	edit
RAID Configuration	No RAID for H330/H730P (1-24 HDDs)	edit
RAID Controller	PERC H730P Adapter RAID Controller, 2Gb NV Cache	edit
Hard Drives	800GB Solid State Drive SAS Read Intensive MLC 12Gbps 2.5in Hot-plug Drive	edit
Network Daughter Card	Broadcom 5720 Quad Port 1Gb Network Daughter Card	edit
Additional Network Cards	None	edit

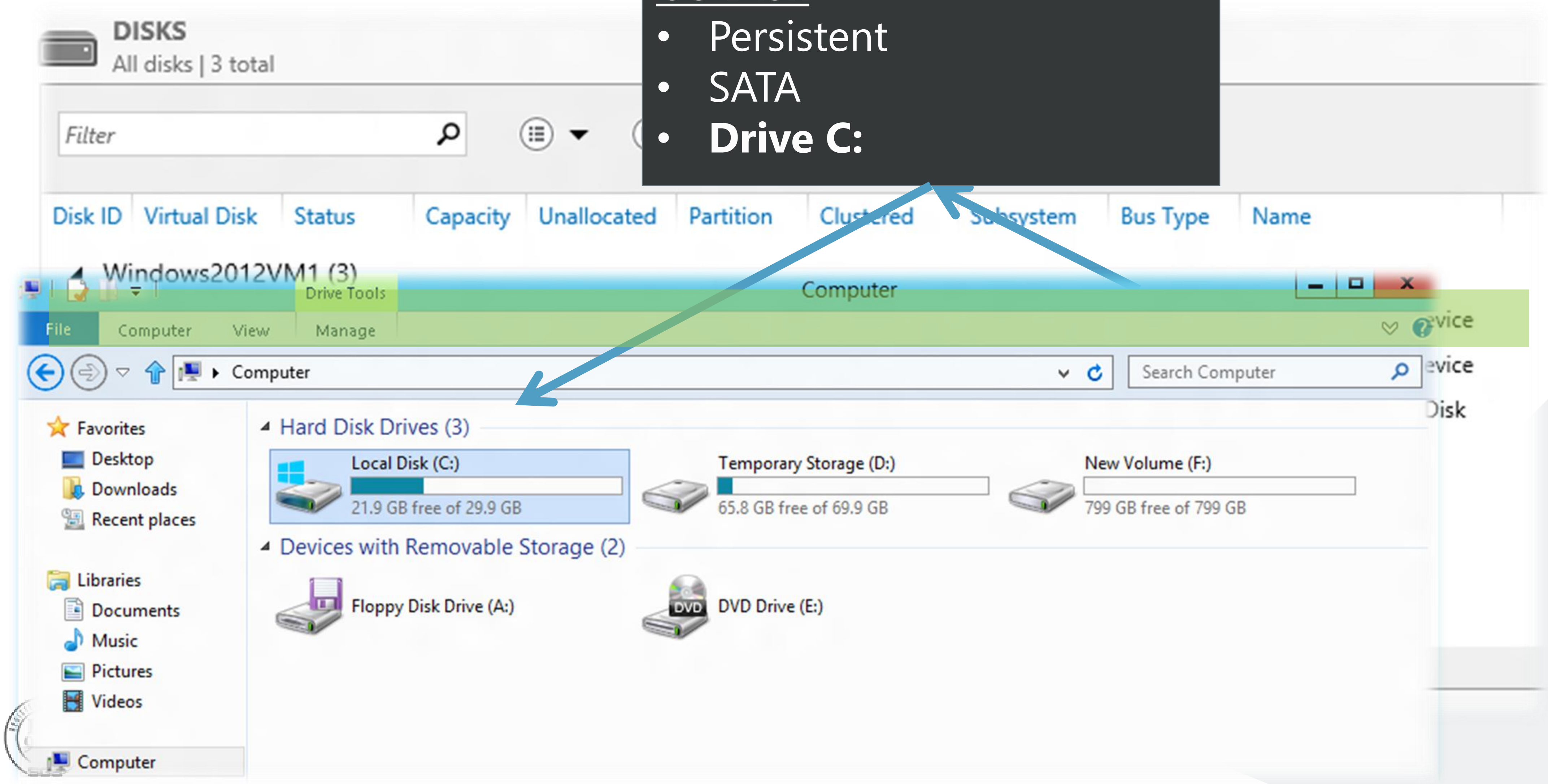
Demo

Creating a VM in the Portal

VM Disk Layout

OS Disk

- Persistent
- SATA
- **Drive C:**



DISKS
All disks | 3 total

Disk ID	Virtual Disk	Status	Capacity	Unallocated	Partition	Clustered	Subsystem	Bus Type	Name

Windows2012VM1 (3)

Computer

Hard Disk Drives (3)

Local Disk (C:)	Temporary Storage (D:)	New Volume (F:)
21.9 GB free of 29.9 GB	65.8 GB free of 69.9 GB	799 GB free of 799 GB

Devices with Removable Storage (2)

Floppy Disk Drive (A:)
DVD Drive (E:)

Azure Disks

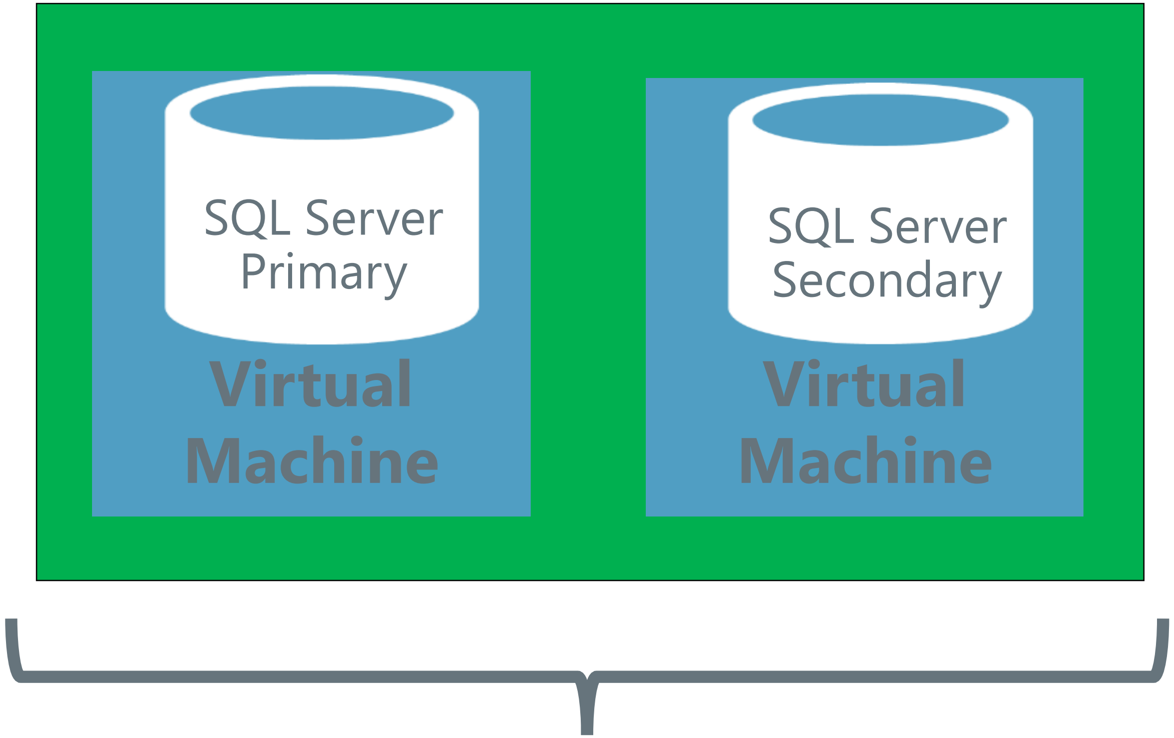
- Backed by Page BLOBs
  - Stored in an Azure Storage account
- Mounted by a single VM at a time
  - Lease is taken on BLOB itself
- Can be backed by Standard or Premium storage
  - Standard – up to 500 8KB IOPS per disk (60 MB/s)
  - Premium – up to 5000 8KB IOPS per disk (200 MB/s)
- Max disk size (per disk) – 1 TB
- Pay for actual storage only
  - Not paying for capacity itself!
- Premium storage can only be attached to DS & GS machines

Azure Files

- Shared Network File Storage for Azure
  - File Share as a Service
- Can be mounted by multiple VMs concurrently
- Availability, durability, scalability are managed automatically
- Supports two interfaces: SMB and REST
  - Individual files stored in share are accessible via REST
- Max file share storage (per share) – 5 TB
- Max file size (within share) – 1 TB
- Scale limit – up to 1000 8KB IOPS (up to 60 MB/s per file share)

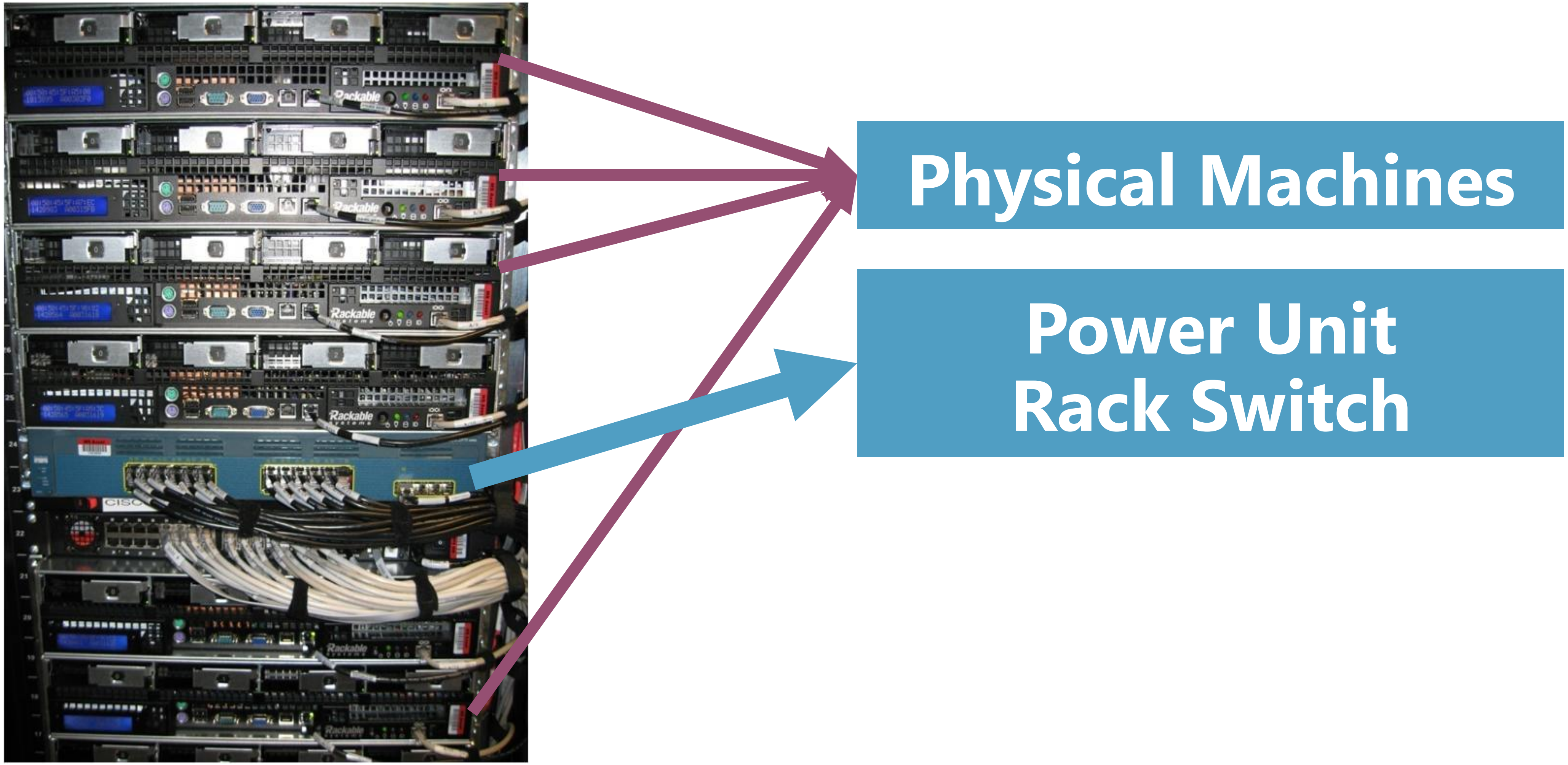
Availability Sets

Availability set



SLA 99.95

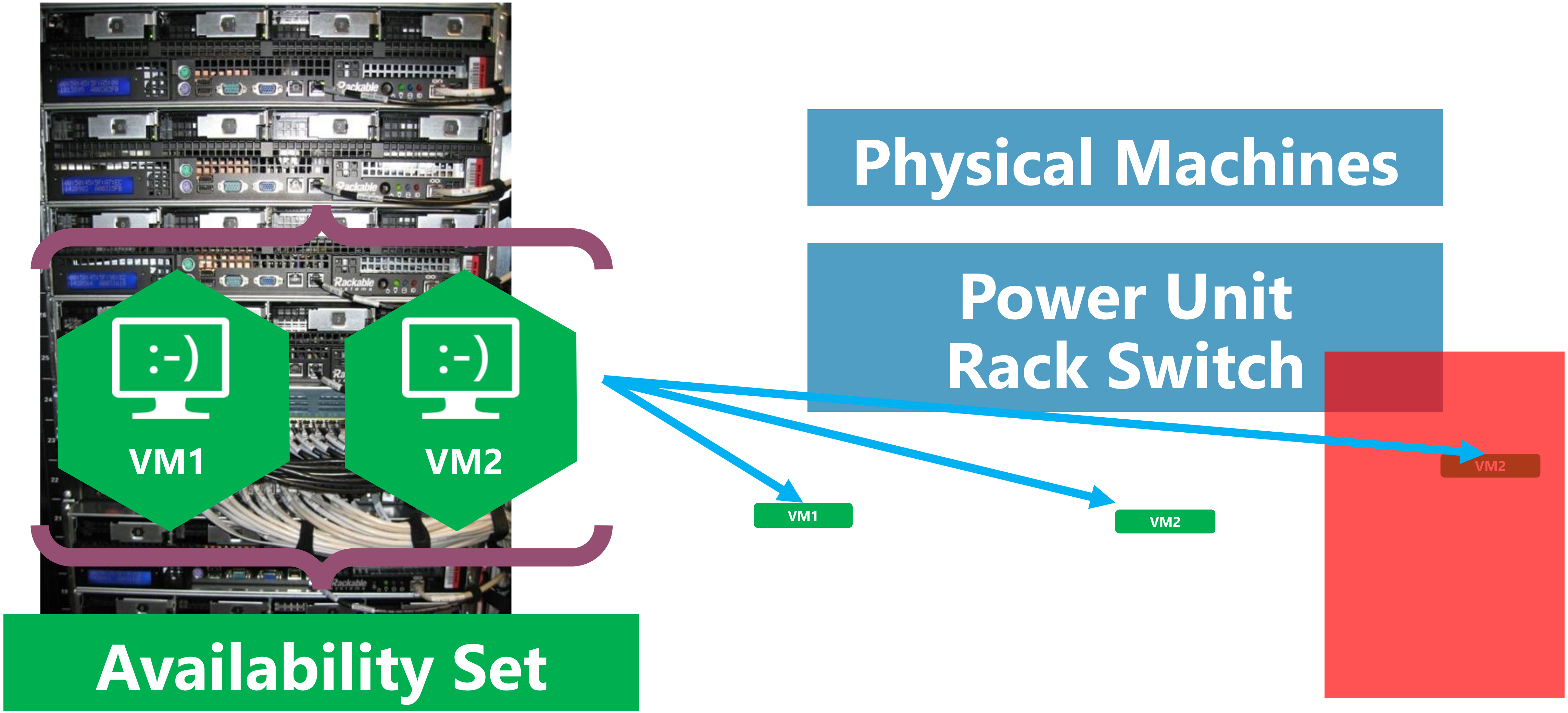
Availability Sets



Physical Machines

Power Unit
Rack Switch

Availability Sets



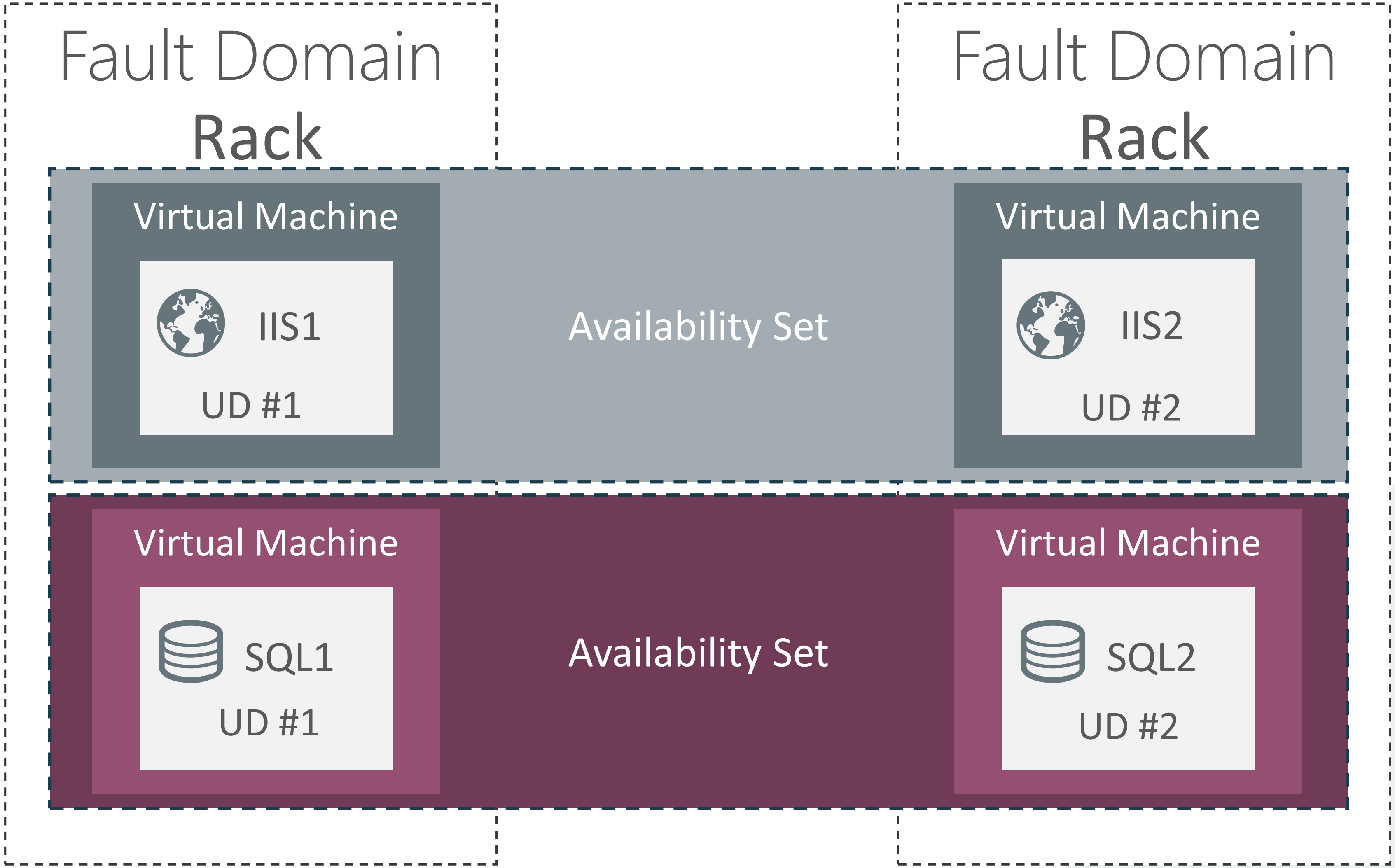
Physical Machines

Power Unit
Rack Switch

Availability Set

Virtual Machine Availability Sets

Update Domains are honored by host OS updates



Fault Domain
Rack

Fault Domain
Rack

Virtual Machine
IIS1
UD #1

Availability Set

Virtual Machine
IIS2
UD #2

Virtual Machine
SQL1
UD #1

Availability Set

Virtual Machine
SQL2
UD #2

Cloud Scale Compute Patterns

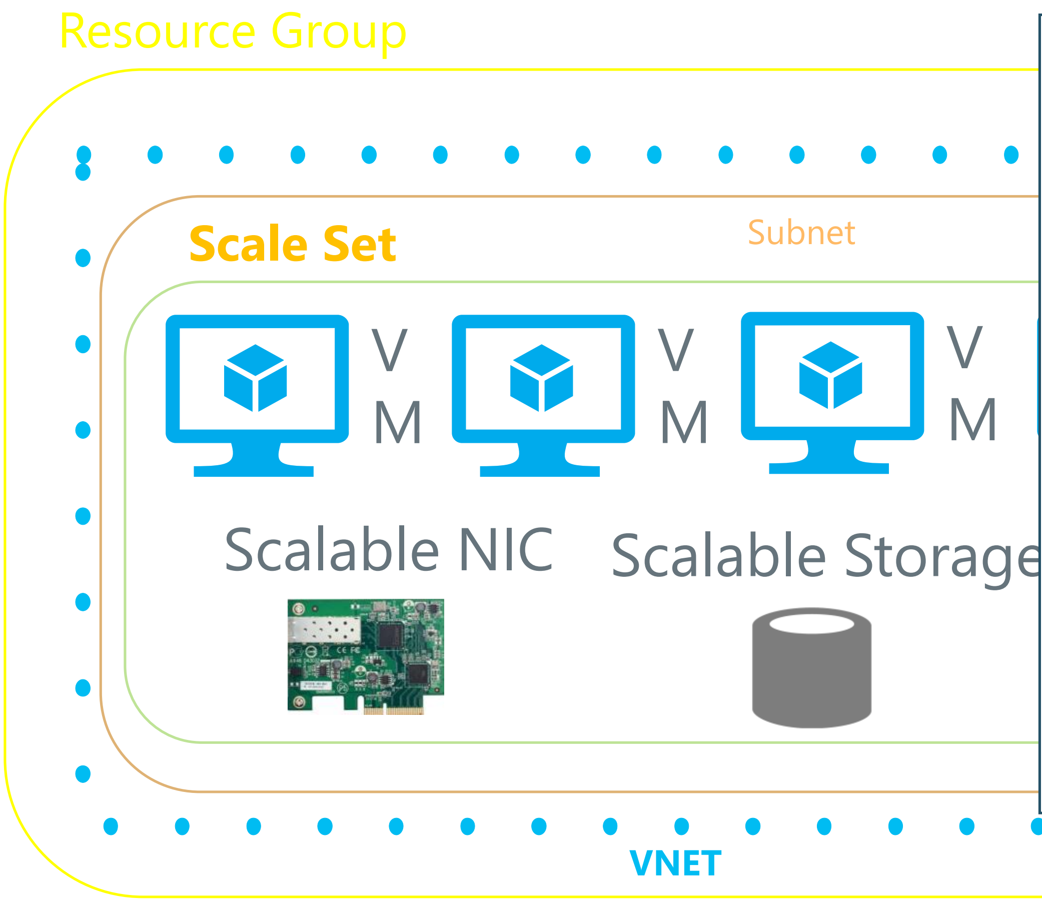
- Pets: Named resources with unique characteristics
- Cattle: Numbered, inherently replaceable, interchangeable

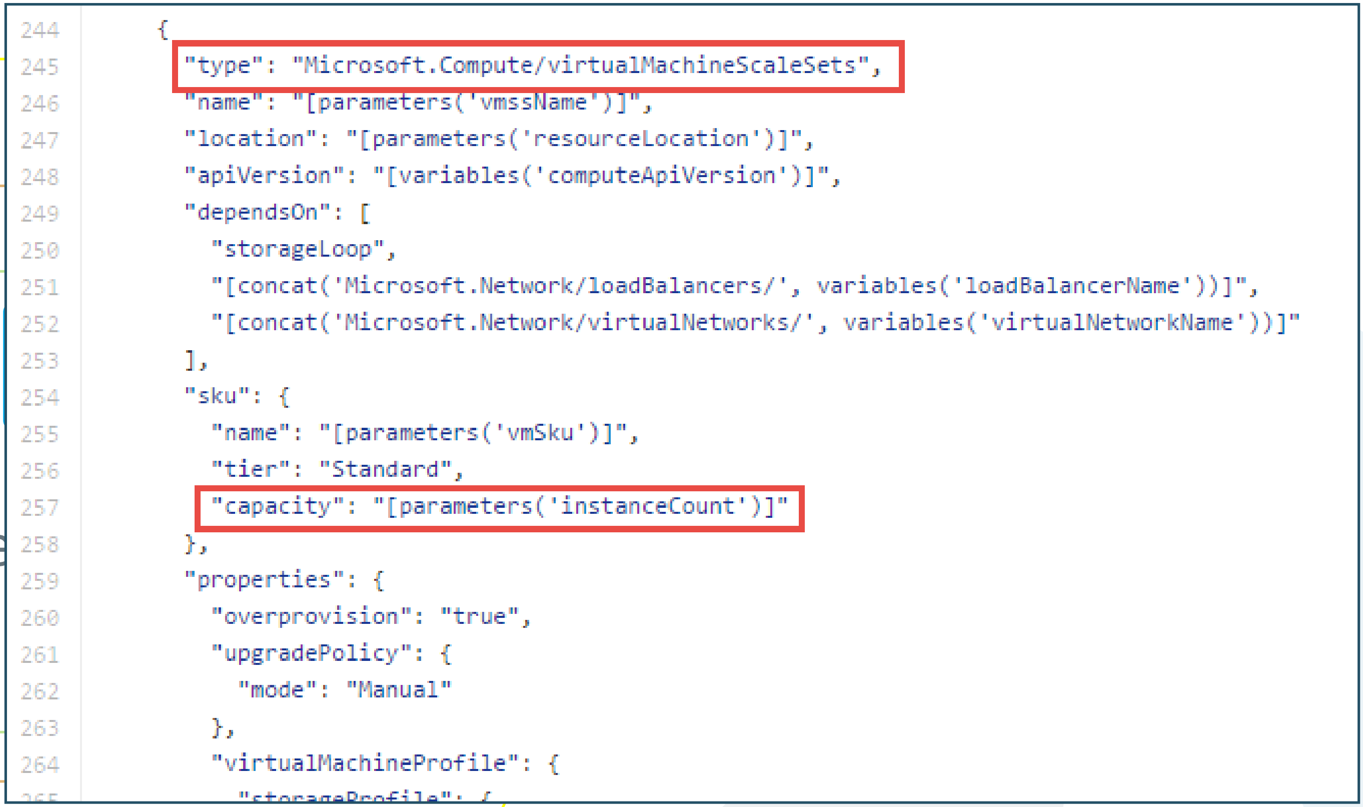
“Future application architectures should use Cattle but Pets with strong configuration management are viable and still needed.” – Gavin McCance (CERN)

Virtual Machine Scale Sets

- Deploy and manage VMs as a set
- An Azure Compute resource
  - Microsoft.Compute/virtualMachineScaleSets
- Integrated with Azure Autoscale
- Integrated with Azure Load Balancer

VM Scale Sets in ARM





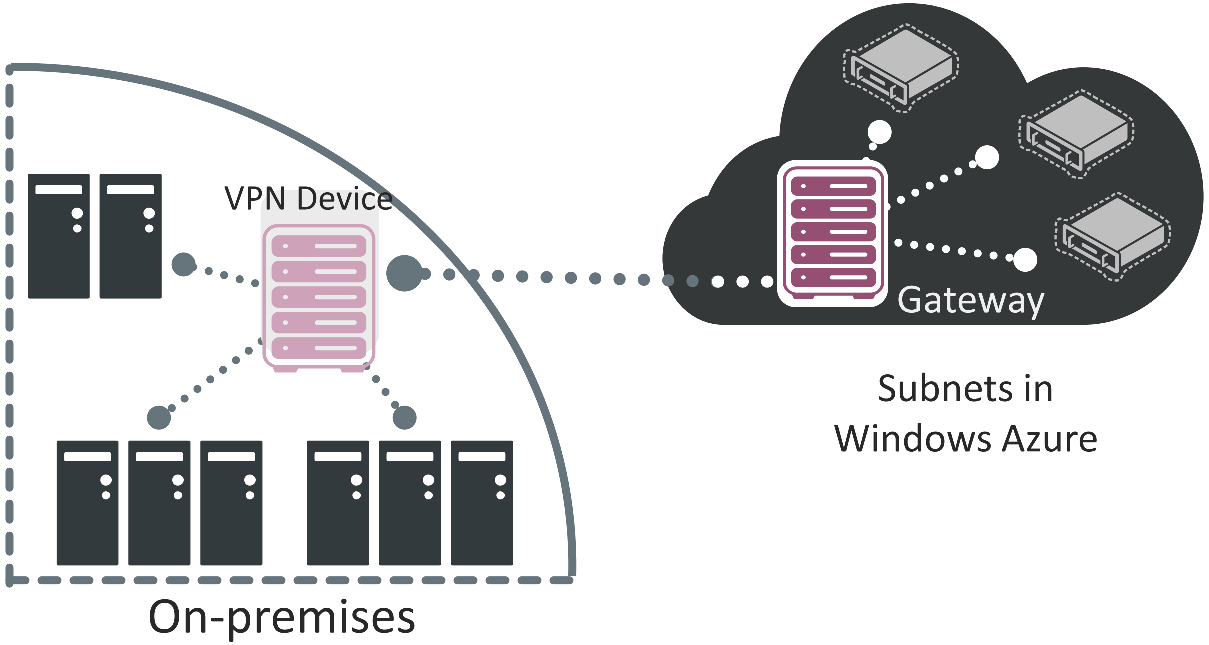
```
    {
      "type": "Microsoft.Compute/virtualMachineScaleSets",
      "name": "[parameters('vmssName')]",
      "location": "[parameters('resourceLocation')]",
      "apiVersion": "[variables('computeApiVersion')]",
      "dependsOn": [
        "storageLoop",
        "[concat('Microsoft.Network/loadBalancers/', variables('loadBalancerName'))]",
        "[concat('Microsoft.Network/virtualNetworks/', variables('virtualNetworkName'))]"
      ],
      "sku": {
        "name": "[parameters('vmSku')]",
        "tier": "Standard",
        "capacity": "[parameters('instanceCount')]"
      },
      "properties": {
        "overprovision": "true",
        "upgradePolicy": {
          "mode": "Manual"
        },
        "virtualMachineProfile": {
          "storageProfile": {
```

Network Interface Cards (NICs)

- NICs allow VMs to communicate with the world
- Each VM has at least one NIC attached
  - Can have more than one
- A single NIC is associated with a specific subnet
  - Receives a private IP address
  - Can be static or dynamic
- A single NIC can be associated with a public IP address
  - Not mandatory
  - A public IP is a separate entity
    - Can be static or dynamic
- Each NIC is associated with a Network Security Group

Azure Virtual Networks

- A protected private virtual network in cloud
- Extend enterprise networks into Azure
- Cross-premises connectivity



Summary

- Azure VMs can be provisioned on various tiers and sizes
- Data disks can be striped for extra performance
- Azure Files can be used for file share as a service
- Temporary disks are, well… temporary!
- Availability sets provide 99.95% uptime SLA
- NICs and Virtual Networks govern VM connectivity
- Network Security Groups can be used to configure network firewalls

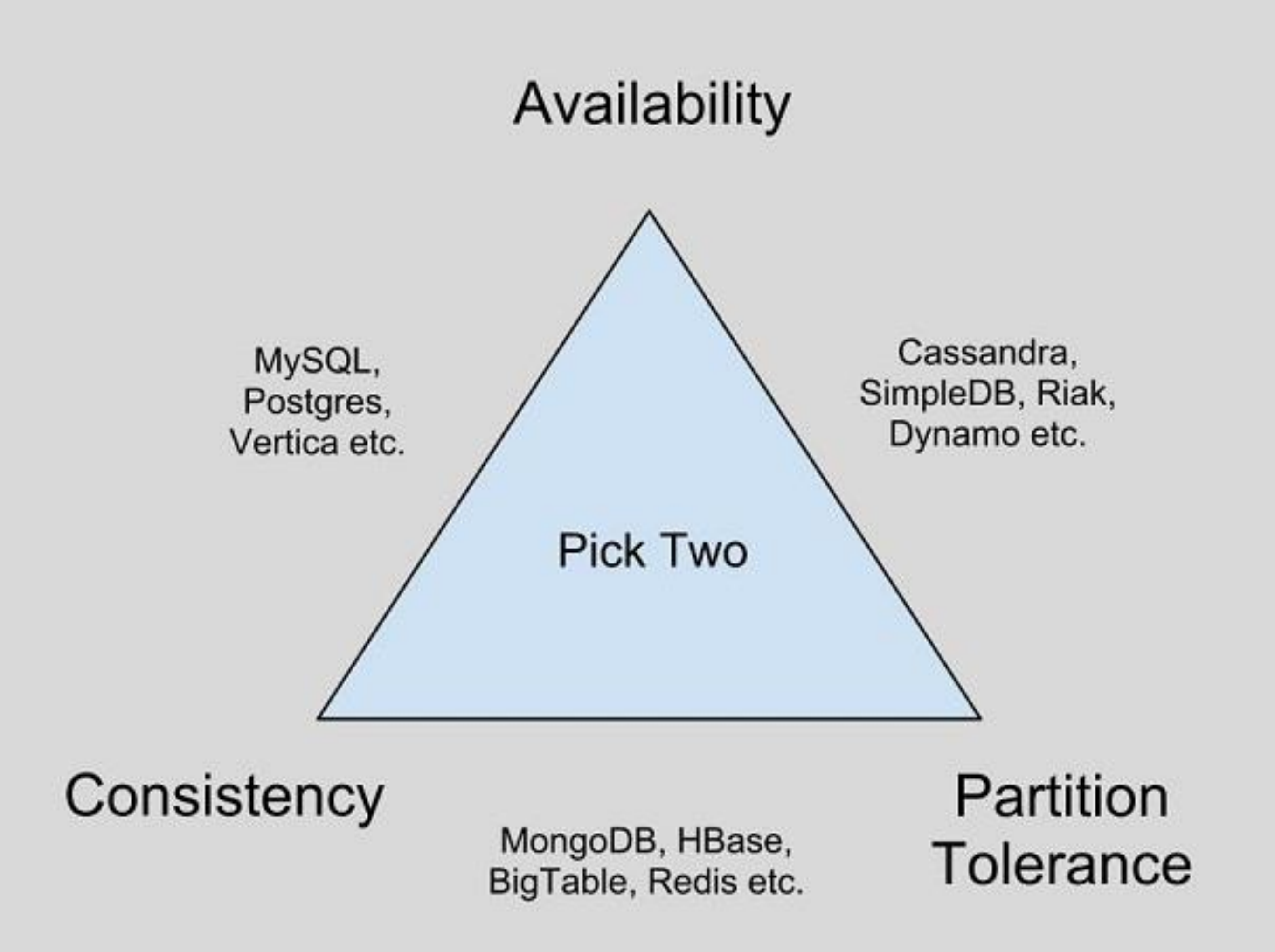
ISO 9001 SGS

Azure Storage Options

Relational					Files	Search
• Azure SQL Server • SQL Server • Postgres • MySQL • Oracle • SQLite	• Azure Blob Storage • Azure Table Storage • CosmosDB • Redis • Memcached • Riak	• Cassandra • HBase	• Cosmos DB (previously Document DB) • MongoDB • RavenDB • CouchDB	• Cosmos DB • Neo4J	• Azure Blob • Azure File Storage (SMB)	• Elasticsearch • Azure Search

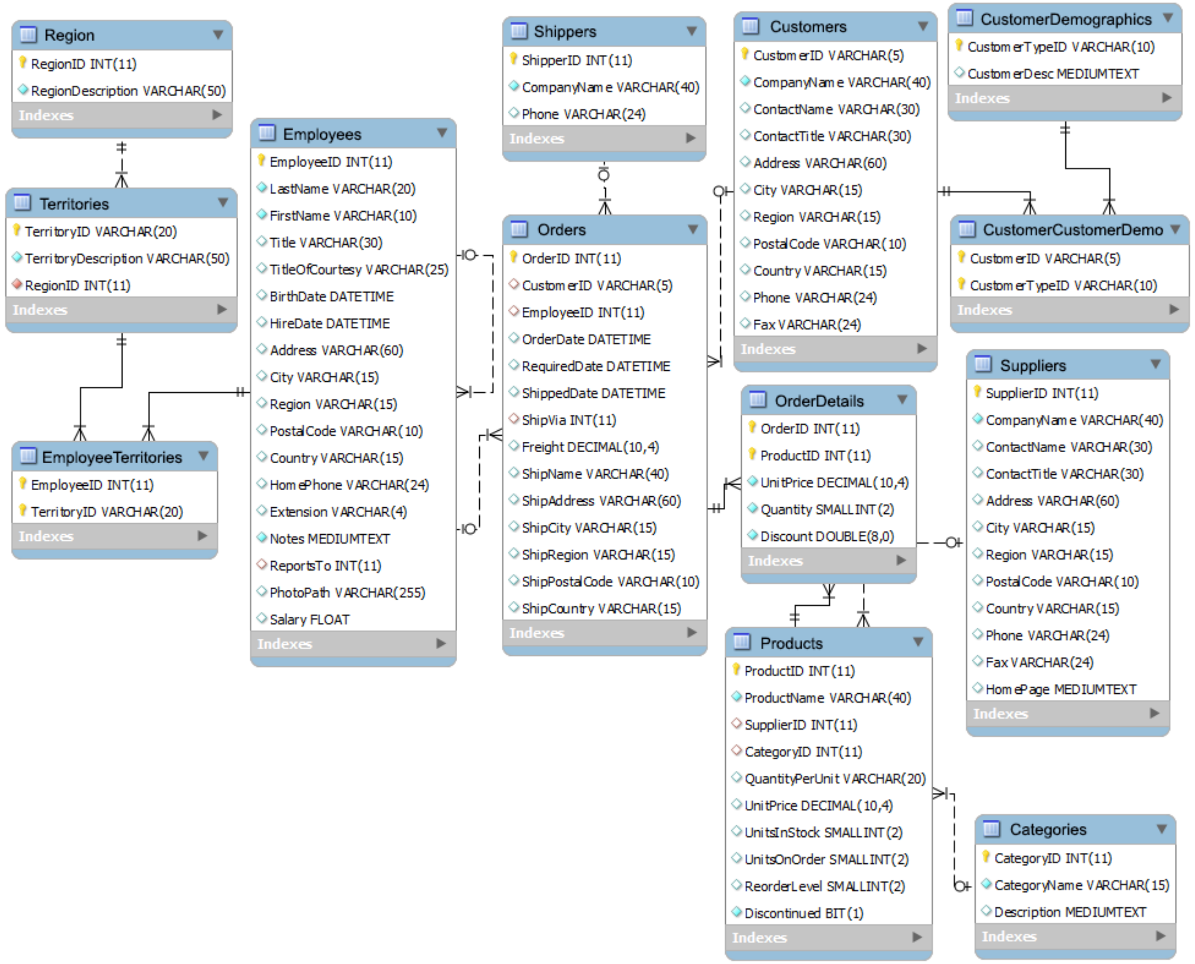
+ **Azure Queues**, which are part of the Azure storage infrastructure

CAP Theorem


Availability
MySQL, Postgres, Vertica etc.
Cassandra, SimpleDB, Riak, Dynamo etc.
Pick Two
Consistency
MongoDB, HBase, BigTable, Redis etc.
Partition Tolerance

RDBMS

- Relational model exist for many years
- Tables, columns, relationships and constraints
- Azure PaaS solutions:
  - MS SQL Server
  - PostgreSQL
  - MySQL



Region
RegionID INT(11)
RegionDescription VARCHAR(50)
Indexes

Territories
TerritoryID VARCHAR(20)
TerritoryDescription VARCHAR(50)
RegionID INT(11)
Indexes

EmployeeTerritories
EmployeeID INT(11)
TerritoryID VARCHAR(20)
Indexes

Employees
EmployeeID INT(11)
LastName VARCHAR(20)
FirstName VARCHAR(10)
Title VARCHAR(30)
TitleOfCourtesy VARCHAR(25)
BirthDate DATETIME
HireDate DATETIME
Address VARCHAR(60)
City VARCHAR(15)
Region VARCHAR(15)
PostalCode VARCHAR(10)
Country VARCHAR(15)
HomePhone VARCHAR(24)
Extension VARCHAR(4)
Notes MEDIUMTEXT
ReportsTo INT(11)
PhotoPath VARCHAR(255)
Salary FLOAT
Indexes

Shippers
ShipperID INT(11)
CompanyName VARCHAR(40)
Phone VARCHAR(24)
Indexes

Orders
OrderID INT(11)
CustomerID VARCHAR(5)
EmployeeID INT(11)
OrderDate DATETIME
RequiredDate DATETIME
ShippedDate DATETIME
ShipVia INT(11)
Freight DECIMAL(10,4)
ShipName VARCHAR(40)
ShipAddress VARCHAR(60)
ShipCity VARCHAR(15)
ShipRegion VARCHAR(15)
ShipPostalCode VARCHAR(10)
ShipCountry VARCHAR(15)
Indexes

Customers
CustomerID VARCHAR(5)
CompanyName VARCHAR(40)
ContactName VARCHAR(30)
ContactTitle VARCHAR(30)
Address VARCHAR(60)
City VARCHAR(15)
Region VARCHAR(15)
PostalCode VARCHAR(10)
Country VARCHAR(15)
Phone VARCHAR(24)
Fax VARCHAR(24)
Indexes

CustomerDemographics
CustomerTypeID VARCHAR(10)
CustomerDesc MEDIUMTEXT
Indexes

CustomerCustomerDemo
CustomerID VARCHAR(5)
CustomerTypeID VARCHAR(10)
Indexes

OrderDetails
OrderID INT(11)
ProductID INT(11)
UnitPrice DECIMAL(10,4)
Quantity SMALLINT(2)
Discount DOUBLE(8,0)
Indexes

Suppliers
SupplierID INT(11)
CompanyName VARCHAR(40)
ContactName VARCHAR(30)
ContactTitle VARCHAR(30)
Address VARCHAR(60)
City VARCHAR(15)
Region VARCHAR(15)
PostalCode VARCHAR(10)
Country VARCHAR(15)
Phone VARCHAR(24)
Fax VARCHAR(24)
HomePage MEDIUMTEXT
Indexes

Products
ProductID INT(11)
ProductName VARCHAR(40)
SupplierID INT(11)
CategoryID INT(11)
QuantityPerUnit VARCHAR(20)
UnitPrice DECIMAL(10,4)
UnitsInStock SMALLINT(2)
UnitsOnOrder SMALLINT(2)
ReorderLevel SMALLINT(2)
Discontinued BIT(1)
Indexes

Categories
CategoryID INT(11)
CategoryName VARCHAR(15)
Description MEDIUMTEXT
Indexes

Azure SQL - Basics

SQL Database

SQL Server database technology as a service

Fully Managed

Enterprise-ready with automatic support for HA

Designed to scale out elastically with demand

Ideal for simple and complex applications

Feature comparison with SQL Server –
https://docs.microsoft.com/en-us/azure/sql-database/sql-database-features

Server Provisioning

Server Defined

Service head that contains databases

Connect via automatically generated FQDN (xxx.database.windows.net)

Initially contains only a **master** database

Provision Servers Interactively

Log on to Windows Azure Management Portal

Create a SQL Database server

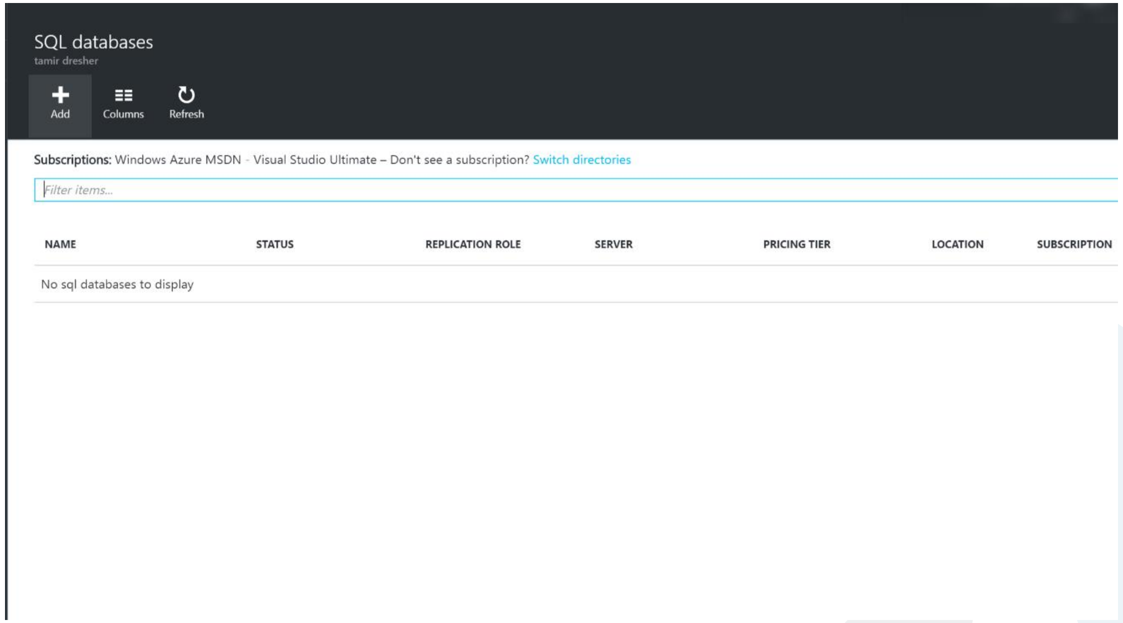
Specify admin login credentials

Add firewall rules and enable service access

Automate Server Provisioning

Use Windows Azure Platform PowerShell cmdlets
(or use REST API directly)

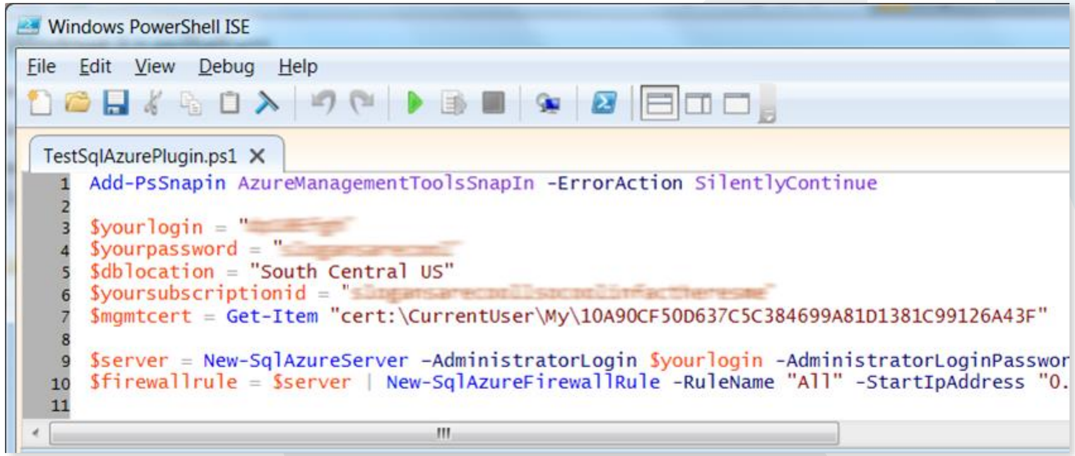
wappowershell.codeplex.com



Subscriptions: Windows Azure MSDN - Visual Studio Ultimate – Don't see a subscription? Switch directories

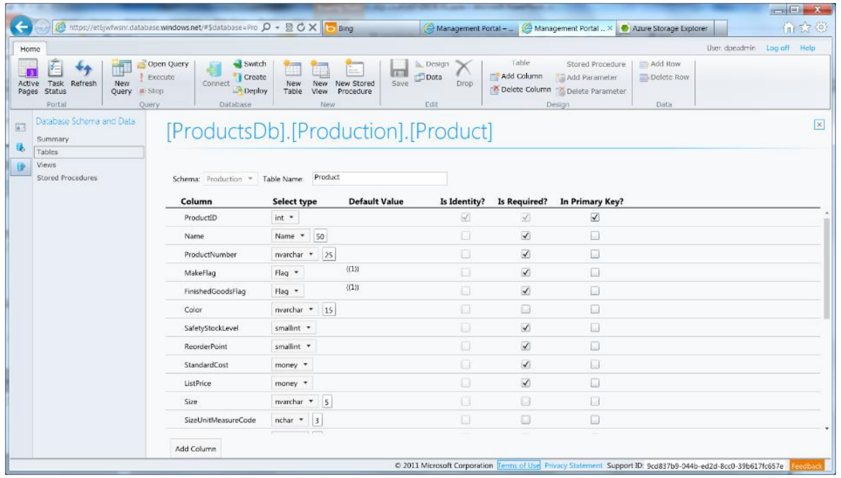
Filter items...

NAME	STATUS	REPLICATION ROLE	SERVER	PRICING TIER	LOCATION	SUBSCRIPTION
No sql databases to display						



```
Add-PsSnapin AzureManagementToolsSnapIn -ErrorAction SilentlyContinue

$yourlogin = "[redacted]
$yourpassword = "[redacted]
$dblocation = "South Central US"
$yoursubscriptionid = "[redacted]
$mgmtcert = Get-Item "cert:\CurrentUser\My\10A90CF50D637C5C384699A81D1381C99126A43F"

$server = New-SqlAzureServer -AdministratorLogin $yourlogin -AdministratorLoginPasswor
$firewallrule = $server | New-SqlAzureFirewallRule -RuleName "All" -StartIpAddress "0.

```

Enhanced Tooling

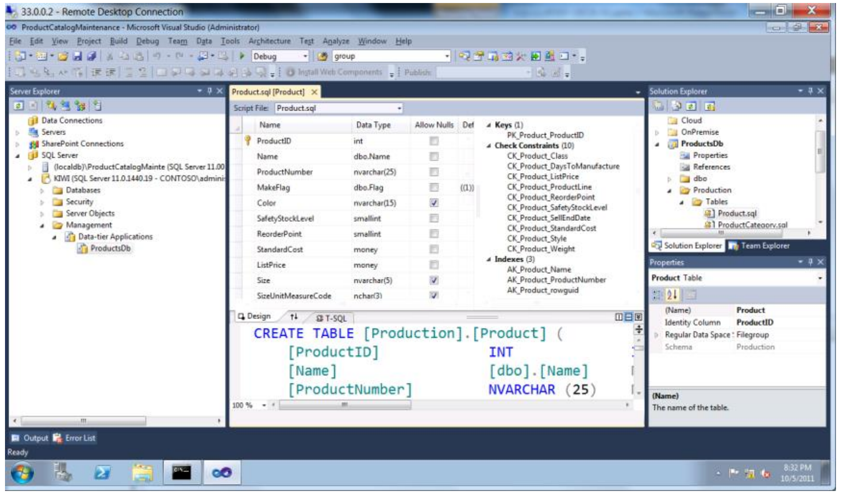


[ProductsDb].[Production].[Product]

Schema: Production Table Name: Product

Column	Select type		Default Value	Is Identity?	Is Required?	In Primary Key?
ProductID	int			☑	☑	☑
Name	Name	50		□	☑	□
ProductNumber	nvarchar	25		□	☑	□
MakeFlag	Flag		((1))	□	☑	□
FinishedGoodsFlag	Flag		((1))	□	☑	□
Color	nvarchar	15		□	□	□
SafetyStockLevel	smallint			□	☑	□
ReorderPoint	smallint			□	☑	□
StandardCost	money			□	☑	□
ListPrice	money			□	☑	□
Size	nvarchar	5		□	□	□
SizeUnitMeasureCode	nchar	3		□	□	□

© 2011 Microsoft Corporation Terms of Use Privacy Statement Support ID: 9cd837b9-044b-ed2d-8cc0-39b617fc657e



Product.sql [Product]

Script File: Product.sql

	Name	Data Type	Allow Nulls	Def
	ProductID	int	□	
	Name	dbo.Name	□	
	ProductNumber	nvarchar(25)	□	
	MakeFlag	dbo.Flag	□	((1))
	Color	nvarchar(15)	☑	
	SafetyStockLevel	smallint	□	
	ReorderPoint	smallint	□	
	StandardCost	money	□	
	ListPrice	money	□	
	Size	nvarchar(5)	☑	
	SizeUnitMeasureCode	nchar(3)	☑	

Keys (1)
PK_Product_ProductID
Check Constraints (10)
CK_Product_Class
CK_Product_DaysToManufacture
CK_Product_ListPrice
CK_Product_ProductLine
CK_Product_ReorderPoint
CK_Product_SafetyStockLevel
CK_Product_SellEndDate
CK_Product_StandardCost
CK_Product_Style
CK_Product_Weight
Indexes (3)
AK_Product_Name
AK_Product_ProductNumber
AK_Product_rowguid

```
CREATE TABLE [Production].[Product] (
    [ProductID]                INT
    [Name]                     [dbo].[Name]
    [ProductNumber]            NVARCHAR (25)
```

SQL Database Management Portal

Web designers for tables, views, stored procs

Interactive query editing and execution

SQL Server Data Tools (SSDT)

Visual Studio IDE for database development

Includes modern designers and projects with declarative, model-driven development

Develop and test in both connected and disconnected states

Platform targeting for both SQL Server (2005 and above) and SQL Database

Get it free with Web PI, with SQL Server 2012 and with Visual Studio 11

SQL Database editions

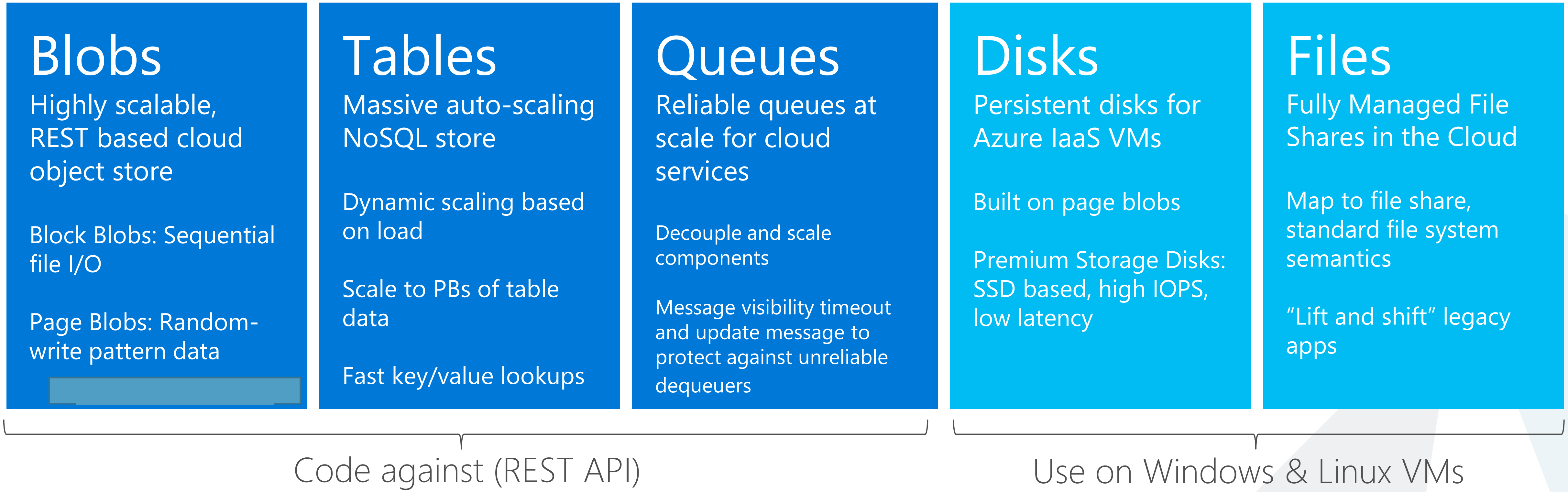
Service Tier	Performance Level	Common App Pattern	Performance			Business Continuity	
			Max DB Size	Transaction Perf. Objective	DTU	PITR	DR / GEO-Rep
Basic	Basic	Small DB, SQL opp	2 GB	Reliability / Hr.	5	7 Days	DB Copy + Manual Export
Standard	S0 S1 S2	Wrkgp/cloud app, multiple concurrent operations	250 GB	Reliability / Min.	10 20 50	14 Days	DB Copy + Manual Export
Premium	P1 P2 P3	Mission Critical, High volume, Many concurrent Users	500 GB	Reliability / sec.	100 200 800	35 Days	Active Geo-replication

http://dtucalculator.azurewebsites.net/

Backup and Restore

- Auto backups, transactional logs every 5 min
- Backups in Azure Storage and geo-replicated
- Creates a side-by-side copy, non-disruptive
- Backups retention policy: 7, 14 or 35 days
- Automated export of logical backups for long-term backup protection
- Geo-restore – Available in all tiers: Basic, Standard and Premium

Azure Storage Account Services



Blobs

Highly scalable, REST based cloud object store

Block Blobs: Sequential file I/O

Page Blobs: Random-write pattern data

Tables

Massive auto-scaling NoSQL store

Dynamic scaling based on load

Scale to PBs of table data

Fast key/value lookups

Queues

Reliable queues at scale for cloud services

Decouple and scale components

Message visibility timeout and update message to protect against unreliable dequeuers

Disks

Persistent disks for Azure IaaS VMs

Built on page blobs

Premium Storage Disks: SSD based, high IOPS, low latency

Files

Fully Managed File Shares in the Cloud

Map to file share, standard file system semantics

“Lift and shift” legacy apps

Code against (REST API)

Use on Windows & Linux VMs

Real Worlds Examples:

XBOX – Cloud Game Save, Halo 4, Music, Kinect data collection

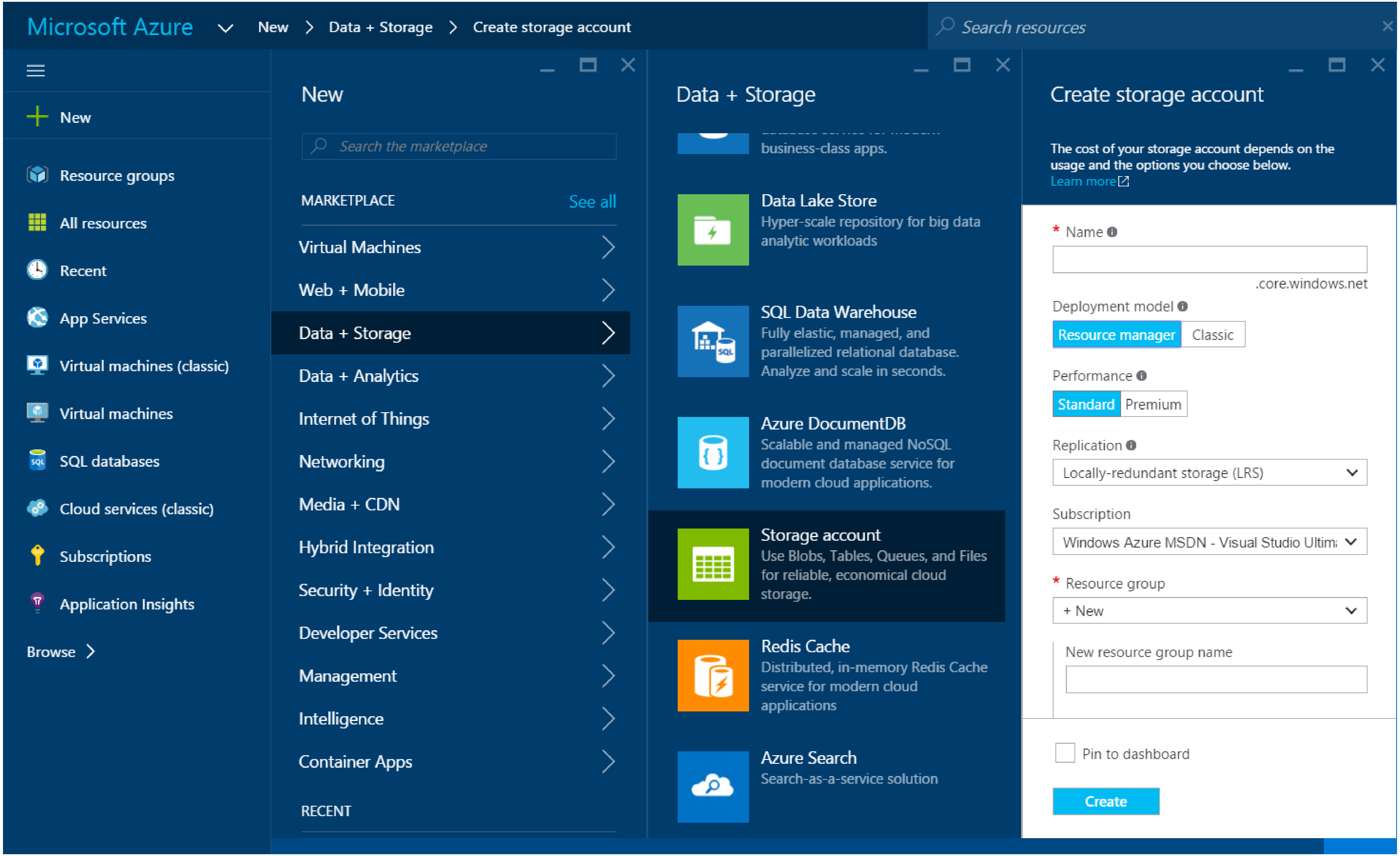
OneDrive

Bing – stores raw data from Twitter and Facebook to digest later

Skype – Video Messaging

Azure Storage Account types

- General-purpose Storage Accounts
  - Tables, Queues, Files, Blobs and Azure virtual machine
  - performance tiers:
    - **Standard storage performance tier** allows you to store Tables, Queues, Files, Blobs and Azure virtual machine disks.
    - **Premium storage performance tier** provides [High-Performance Storage for Azure Virtual Machine Workloads]
- Blob Storage Accounts
  - specialized storage for unstructured data as blobs (objects)
    - Only block and append blobs
  - Access tiers:
    - **Hot access tier** indicates that the objects in the storage account will be more frequently accessed.
    - **Cool access tier** indicates that the objects in the storage account will be less frequently accessed.



business-class apps.

Data Lake Store
Hyper-scale repository for big data analytic workloads

SQL Data Warehouse
Fully elastic, managed, and parallelized relational database. Analyze and scale in seconds.

Azure DocumentDB
Scalable and managed NoSQL document database service for modern cloud applications.

Storage account
Use Blobs, Tables, Queues, and Files for reliable, economical cloud storage.

Redis Cache
Distributed, in-memory Redis Cache service for modern cloud applications

Azure Search
Search-as-a-service solution

Create storage account

The cost of your storage account depends on the usage and the options you choose below.
Learn more

* Name
.core.windows.net

Deployment model: [✓ Resource manager] Classic

Performance: [✓ Standard] Premium

Replication: Locally-redundant storage (LRS)

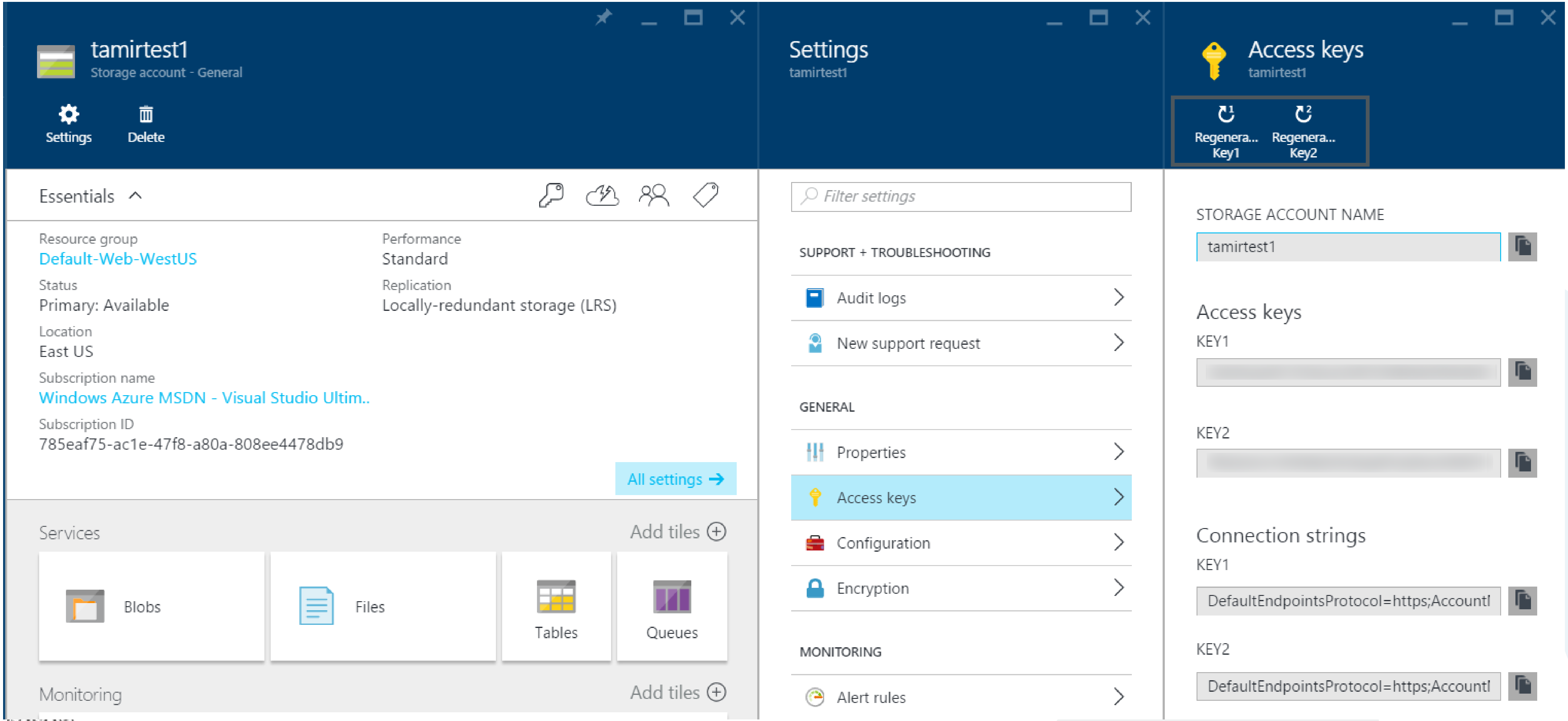
Subscription: Windows Azure MSDN - Visual Studio Ultim:

* Resource group: + New

New resource group name

□ Pin to dashboard

Storage account



tamirtest1
Storage account - General

Settings
Delete

Essentials ^

Resource group
Default-Web-WestUS

Status
Primary: Available

Location
East US

Subscription name
Windows Azure MSDN - Visual Studio Ultim..

Subscription ID
785eaf75-ac1e-47f8-a80a-808ee4478db9

Performance
Standard

Replication
Locally-redundant storage (LRS)

All settings →

Services
Add tiles ⊕

Blobs
Files
Tables
Queues

Monitoring
Add tiles ⊕

Settings
tamirtest1

Filter settings

SUPPORT + TROUBLESHOOTING
- Audit logs >
- New support request >

GENERAL
- Properties >
- Access keys >
- Configuration >
- Encryption >

MONITORING
- Alert rules >

Access keys
tamirtest1

Regenera... Key1
Regenera... Key2

STORAGE ACCOUNT NAME
tamirtest1

Access keys

KEY1
[redacted]

KEY2
[redacted]

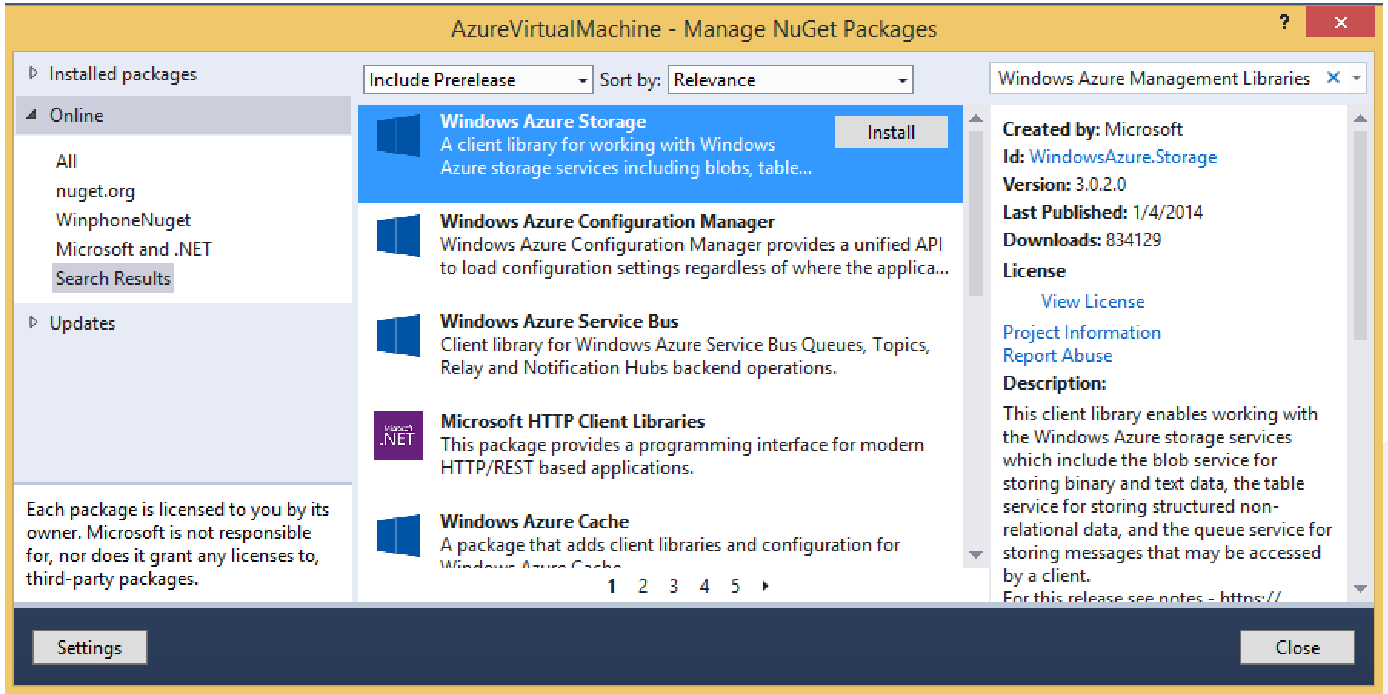
Connection strings

KEY1
DefaultEndpointsProtocol=https;AccountI

KEY2
DefaultEndpointsProtocol=https;AccountI

Storage API

- REST
- Client API from SDK: WindowsAzure.Storage namespace
  - A wrapper around the REST API
  - Hides many of the complexities of the service + Auto retries



AzureVirtualMachine - Manage NuGet Packages

Include Prerelease · Sort by: Relevance · Windows Azure Management Libraries

Windows Azure Storage
A client library for working with Windows Azure storage services including blobs, table...

Windows Azure Configuration Manager
Windows Azure Configuration Manager provides a unified API to load configuration settings regardless of where the applica...

Windows Azure Service Bus
Client library for Windows Azure Service Bus Queues, Topics, Relay and Notification Hubs backend operations.

Microsoft HTTP Client Libraries
This package provides a programming interface for modern HTTP/REST based applications.

Windows Azure Cache
A package that adds client libraries and configuration for Windows Azure Cache

Created by: Microsoft
Id: WindowsAzure.Storage
Version: 3.0.2.0
Last Published: 1/4/2014
Downloads: 834129
License
View License
Project Information
Report Abuse
Description:
This client library enables working with the Windows Azure storage services which include the blob service for storing binary and text data, the table service for storing structured non-relational data, and the queue service for storing messages that may be accessed by a client.
For this release see notes - https://

Each package is licensed to you by its owner. Microsoft is not responsible for, nor does it grant any licenses to, third-party packages.

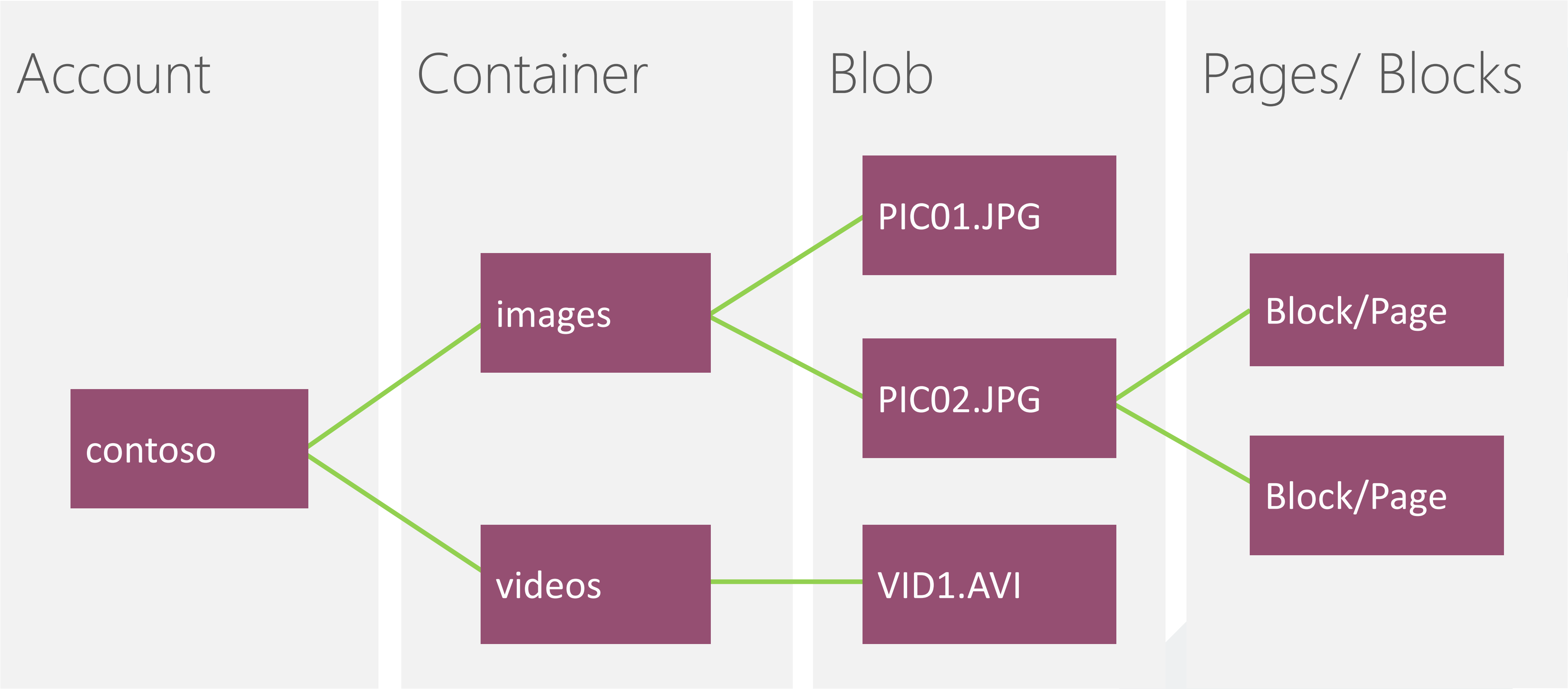
Cloud Storage - Azure BLOB Storage

- BLOB – Binary Large OBject
- Storage for any type of entity such as binary files and text documents
- Distributed File Service (DFS)
  - Scalability and High availability
- BLOB file is distributed between multiple server and replicated at least 3 times

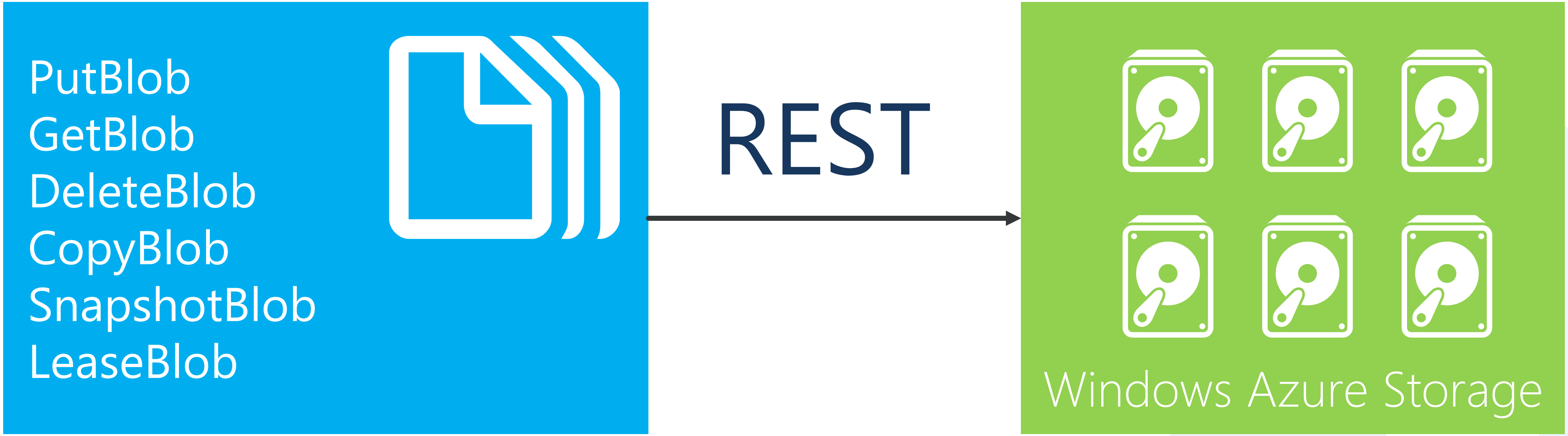
- Get Started with Storage Account
- Get Started with Blob Storage

Azure Blob Storage Concepts

`http://<account>.blob.core.windows.net/<container>/<blobname>`



Blob Operations



PutBlob
GetBlob
DeleteBlob
CopyBlob
SnapshotBlob
LeaseBlob

REST

Windows Azure Storage

Page Blob in code

Creating

```
using Microsoft.WindowsAzure.StorageClient ;

CloudStorageAccount storageAccount = CloudStorageAccount.Parse(
                  $"DefaultEndpointsProtocol=https;AccountName={accountName};AccountKey={accountKey}");
CloudBlobClient blobClient = storageAccount.CreateCloudBlobClient();
CloudBlobContainer container = blobClient.GetContainerReference(containerName);
container.CreateIfNotExist();
CloudPageBlob pageBlob = container.GetPageBlobReference(blobName);
pageBlob.Create(blobSize);
```

Writing

```
pageBlob.WritePages(dataStream, startingOffset) ;
```

Page Blob in code

- Reading

```
BlobStream blobStream = pageBlob.OpenRead();
byte[] buffer = new byte[rangeSize] ;
blobStream.Seek(blobOffset, SeekOrigin.Begin) ;
int numBytesRead = blobStream.Read(buffer, bufferOffset, rangeSize);
```

- Clear Pages

```
pageBlob.ClearPages(startOffset, length)
```

Page Blob in code

Reading

```
BlobStream blobStream = pageBlob.OpenRead();
byte[] buffer = new byte[rangeSize] ;
blobStream.Seek(blobOffset, SeekOrigin.Begin) ;
int numBytesRead = blobStream.Read(buffer, bufferOffset, rangeSize);
```

Clear Pages

```
pageBlob.ClearPages(startOffset, length)
```

Concurrency

- Optimistic concurrency – Timestamps/ETags
  - Timestamp based – If-Modified-Since and If-UnModified-Since
  - ETag based – If-Match and If-None-Match (can force update with *)
  - Conditional update with supplied Timestamp or ETag will fail if conditions not met
- Pessimistic Concurrency - Leases
  - Lease Blob for exclusive write and delete access
  - 15-60s lease duration (can be renewed) or infinite lease (locks)
  - Can change lease id to acquire ownership in a chain/workflow
  - Can also acquire on containers to prevent container deletion
- Last Writer wins
  - https://azure.microsoft.com/en-us/blog/managing-concurrency-in-microsoft-azure-storage-2/
  - https://msdn.microsoft.com/en-us/library/dd179371.aspx

Optimistic Concurrency

```
string orignalETag = blockBlob.Properties.ETag;
try
{
    blockBlob.UploadText(helloText,
       accessCondition: AccessCondition.GenerateIfMatchCondition(orignalETag));
}
catch (StorageException ex)
{
    if (ex.RequestInformation.HttpStatusCode == (int)HttpStatusCode.PreconditionFailed)
    {
        Console.WriteLine("Blob's orignal etag no longer matches");

    }
    else
        throw;
}
```

Passimistic Concurrency

```
string lease = blockBlob.AcquireLease(TimeSpan.FromSeconds(15), null);

// Update blob using lease. This operation will succeed
var accessCondition = AccessCondition.GenerateLeaseCondition(lease);
blockBlob.UploadText("update", accessCondition: accessCondition);

try
{
    // Below operation will fail as no valid lease provided
    blockBlob.UploadText("Update without lease, will fail");
}
catch (StorageException ex)
{
    if (ex.RequestInformation.HttpStatusCode == (int)HttpStatusCode.PreconditionFailed)
        Console.WriteLine("Blob's lease does not match");
    else
        throw;
}
```

Transient Faults

- “ **transient fault** is a fault that is no longer present if power is disconnected for a short time and then restored.” (http://en.wikipedia.org/wiki/Transient_fault#Transient_fault)
- Many faults in connectivity to cloud are transient by nature
- Commonly occur when connecting to service or database

ISO 9001 SGS

Transient Faults handling

- Retry Logic
  - Linear – every fixed amount of time
  - Exponential – if the server is heavy-used (throttling) we don’t want to flood it

    immediate….1 sec….5 seconds….etc.

- Idempotency
  - operations in mathematics and computer science, that can be applied multiple times without changing the result beyond the initial application (wikipedia)
  - Same messages could be sent more than once or out of sequence
  - Design for idempotency

Retry Policy Application

- Microsoft.WindowsAzure.Storage.RetryPolicies.IRetryPolicy Interface
- ExponentialRetry
- LinearRetry
- NoRetry

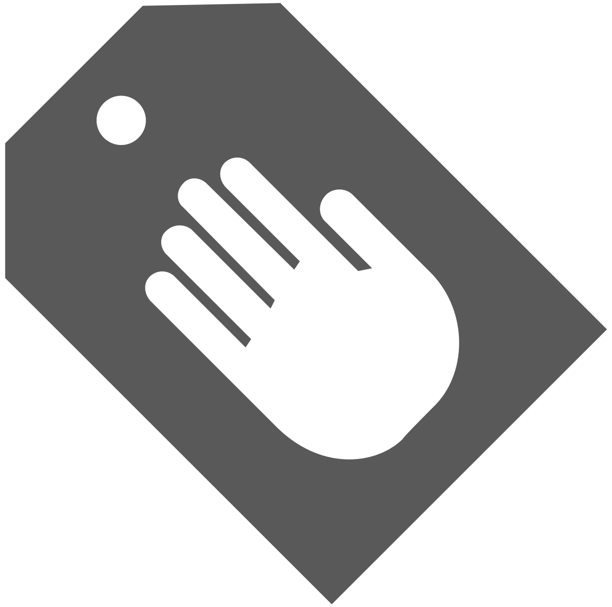
- Default is exponential – if you don't want any retry logic then you must override

Cloud Storage - Table Storage

- Not RDBMS
  - No relationships between entities
  - NoSql
- Entity can have up to 255 properties - Up to 1MB per entity
- Mandatory Properties for every entity
  - PartitionKey & RowKey (only indexed properties)
    - Uniquely identifies an entity
    - Same RowKey can be used in different PartitionKey
    - Defines the sort order
  - Timestamp - Optimistic Concurrency
- Strongly consistent
- Get Started with Table Storage

Shared Access Signatures

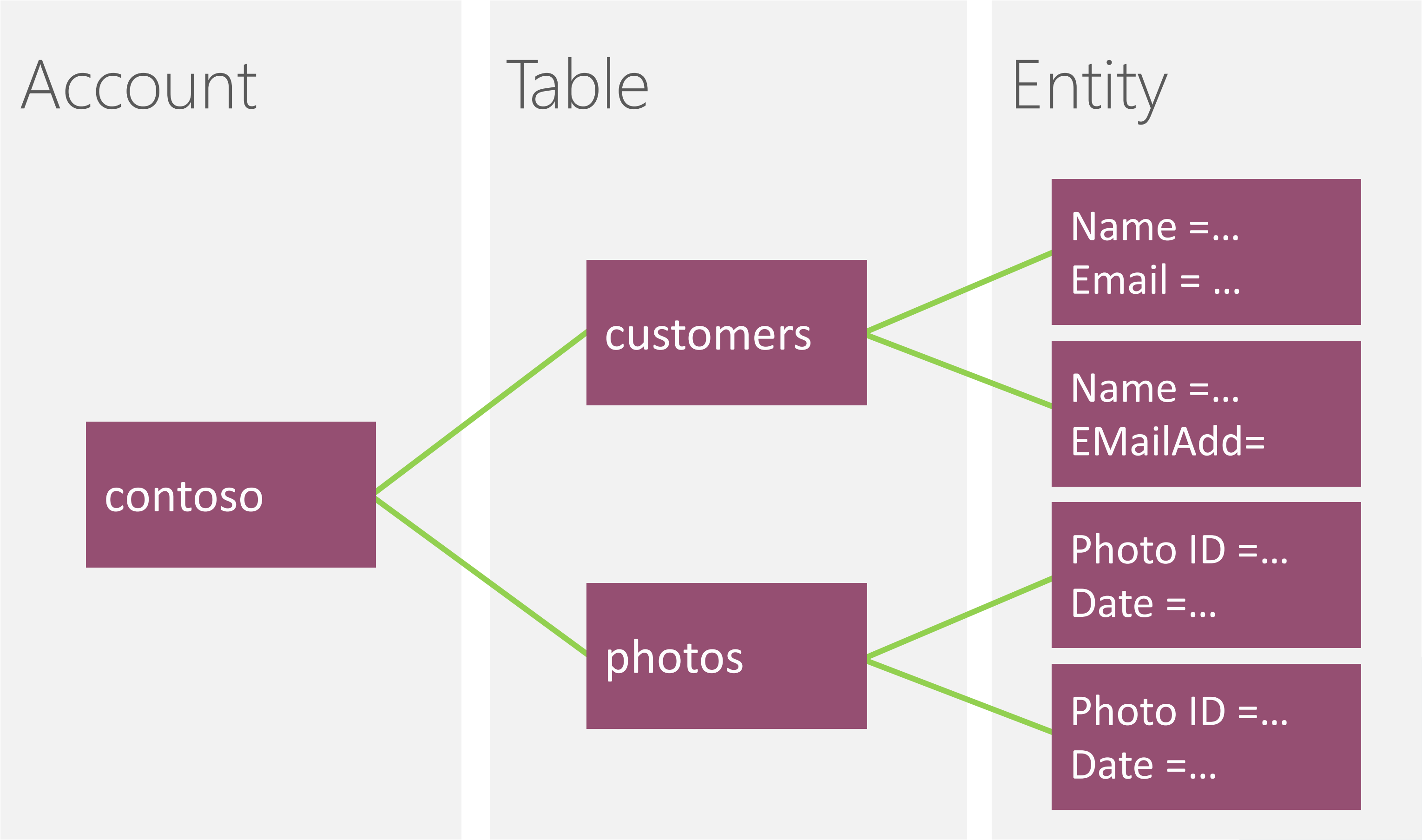
- Fine grain access rights to blobs and containers
- Sign URL with storage key – permit elevated rights
- Revocation
  - Use short time periods and re-issue
  - Use container level policy that can be deleted
- Two broad approaches
  - Ad-hoc
  - Policy based



Tables

Azure Storage

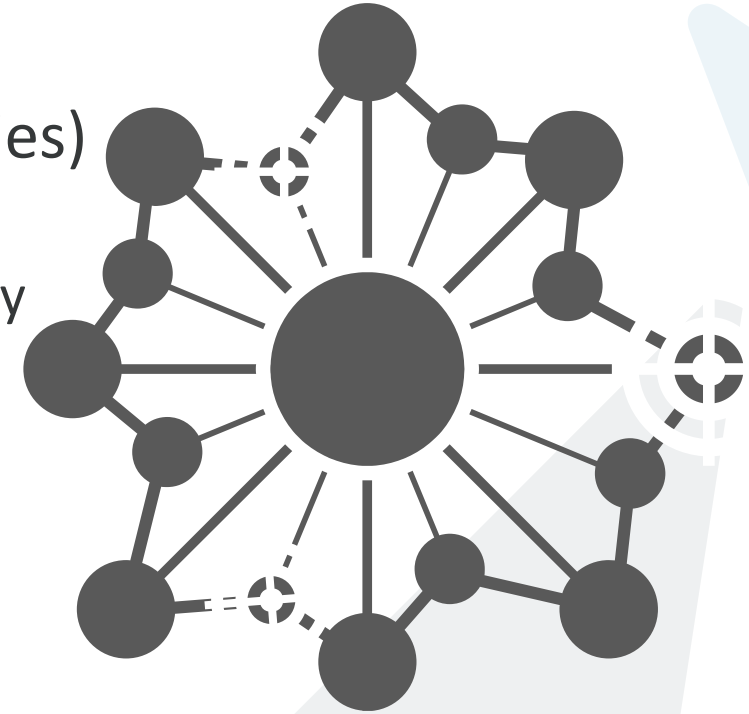
Table Storage Concepts



Entity Properties

- Entity can have up to 255 properties
  - Up to 1MB per entity

- Mandatory Properties for every entity
  - PartitionKey & RowKey (only indexed properties)
    - Uniquely identifies an entity
    - Same RowKey can be used in different PartitionKey
    - Defines the sort order
  - Timestamp
    - Optimistic Concurrency
    - Exposed as an HTTP Etag



Sample – Inserting an Entity into a Table

```
// You will need the following using statements
using Microsoft.WindowsAzure.Storage;
using Microsoft.WindowsAzure.Storage.Table;

// Create the table client.
CloudTableClient tableClient = storageAccount.CreateCloudTableClient();
CloudTable peopleTable = tableClient.GetTableReference("people");
peopleTable.CreateIfNotExists();

// Create a new customer entity.
CustomerEntity customer1 = new CustomerEntity("Harp", "Walter");
customer1.Email = "Walter@contoso.com";
customer1.PhoneNumber = "425-555-0101";

// Create an operation to add the new customer to the people table.
TableOperation insertCustomer1 = TableOperation.Insert(customer1);

// Submit the operation to the table service.
peopleTable.Execute(insertCustomer1);
```

Table Object Model

- *ITableEntity* interface –PartitionKey, RowKey, Timestamp, and Etag properties
  - Implemented by *TableEntity* and *DynamicTableEntity*

```
// This class defines one additional property of integer type,
// since it derives from TableEntity it will be automatically
// serialized and deserialized.
public class SampleEntity : TableEntity
{
    public int SampleProperty { get; set; }
}
```

Querying

- Retrieve(PartitionKey, RowKey) – retrieve single entity that satisfy the arguments
- TableQuery - lightweight object that represents a query for a given set of entities
- IQueryable (not efficient)

```
IQueryable<Footwear> query = table.CreateQuery<Footwear>()
             .Where(f => f.Gender == "Male" && (f.Size > 4 && f.Size < 7));

IEnumerable<Footwear> shoes = query.ToList();
```

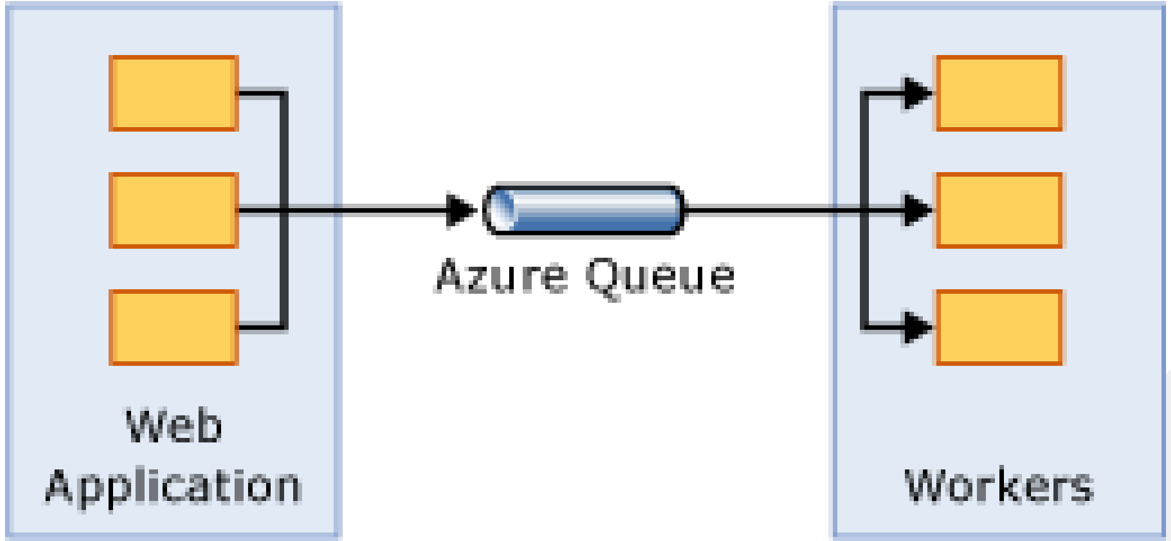
full table scan will be performed.
because no Partition key was specified, the query will be sent
to every Partition Server.

queues

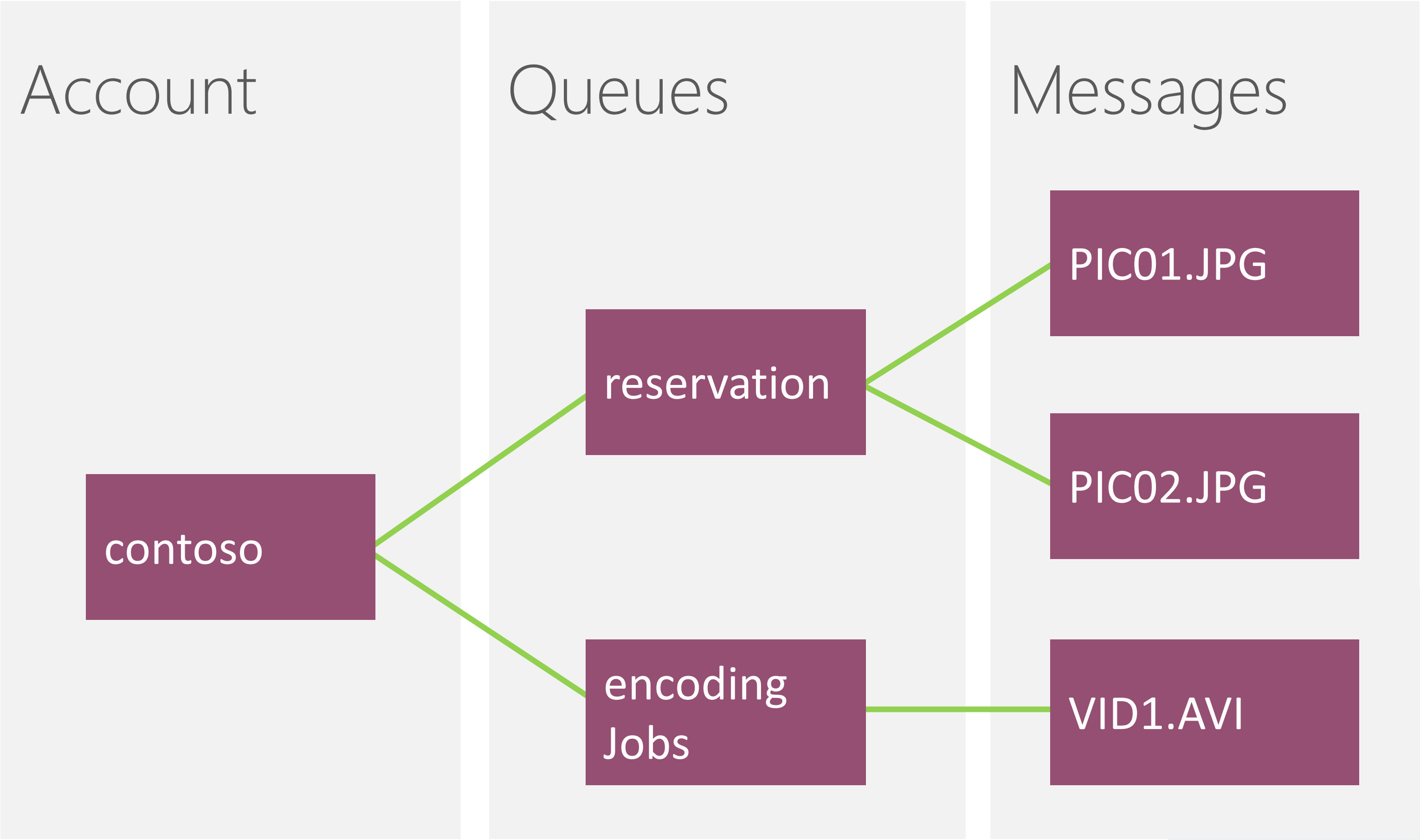
Azure Storage

Windows Azure Queues

- Queue – First In First Out (FIFO) – Not guaranteed
- Queue are performance efficient, highly available and provide reliable message delivery
  - Simple, asynchronous work dispatch
  - Programming semantics ensure that a message can be processed at least once
- Decouple Producers and Consumers



Storage Queue Concepts



Queue Operations

- Queue
  - Create Queue
  - Delete Queue
  - List Queues
  - Get/Set Queue Metadata

- Messages
  - Add Message (i.e. Enqueue Message)
  - Get Message(s) (i.e. Dequeue Message)
  - Peek Message(s)
  - Delete Message

Queue API

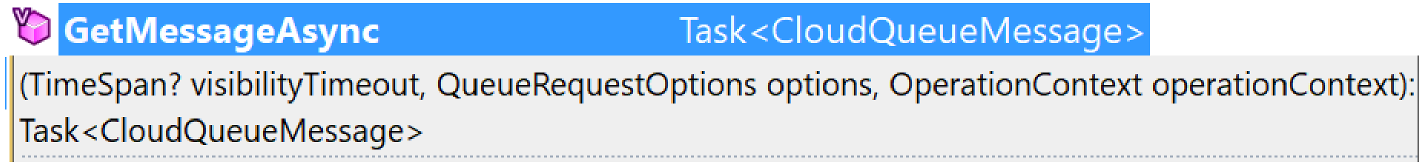
```
// Retrieve storage account from connection string
CloudStorageAccount storageAccount = CloudStorageAccount.Parse(
    CloudConfigurationManager.GetSetting("StorageConnectionString"));

// Create the queue client
CloudQueueClient queueClient = storageAccount.CreateCloudQueueClient();
// Retrieve a reference to a queue
CloudQueue queue = queueClient.GetQueueReference("myqueue");

// Create the queue if it doesn't already exist
queue.CreateIfNotExists();

// Create a message and add it to the queue.
CloudQueueMessage message = new CloudQueueMessage("Hello, World");
queue.AddMessage(message);
```

Queue API

```
// Peek at the next message
CloudQueueMessage peekedMessage = queue.PeekMessage();


// Get the next message
CloudQueueMessage retrievedMessage = queue.GetMessage();
//Process the message in less than 30 seconds, and then delete the message
queue.DeleteMessage(retrievedMessage);


// Get the message from the queue and update the message contents.
CloudQueueMessage message = queue.GetMessage();
message.SetMessageContent("Updated contents.") ;
queue.UpdateMessage(message,
        TimeSpan.FromSeconds(0.0), // Make it visible immediately.
        MessageUpdateFields.Content | MessageUpdateFields.Visibility);
```

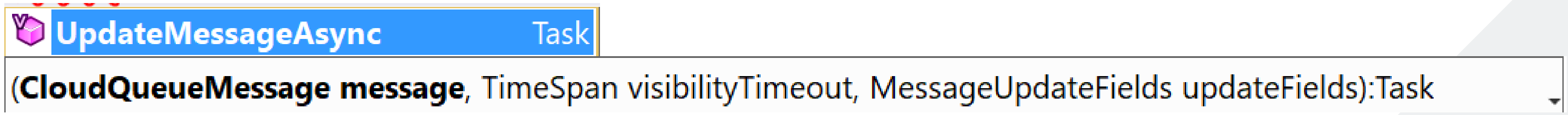
Message Visibility

- By default, after dequeuing, messages are invisible for 30 seconds
- While invisible, no other consumer can dequeue the message
- You can set the visibility-timeout when getting the message from the queue



```
GetMessageAsync                    Task<CloudQueueMessage>
(TimeSpan? visibilityTimeout, QueueRequestOptions options, OperationContext operationContext):
Task<CloudQueueMessage>
```

- You can extend the visibility-timeout by executing the `UpdateMessageAsync` method



```
UpdateMessageAsync                 Task
(CloudQueueMessage message, TimeSpan visibilityTimeout, MessageUpdateFields updateFields):Task
```

- Call the `DeleteMessageAsync` method to remove the message from the queue
- Use the `DequeueCount` property to validate the amount of times the message was dequeued

Poison Messages

- Message can cause the consumer to crash
- find “poison” messages when dequeuing by examining the DequeueCount property of the message.
- If DequeueCount is above a given threshold it is a potential “poison” message
- Two options
  1. Delete the message
  2. Store in Poison Queue/Table

Cosmos DB

Cosmos DB

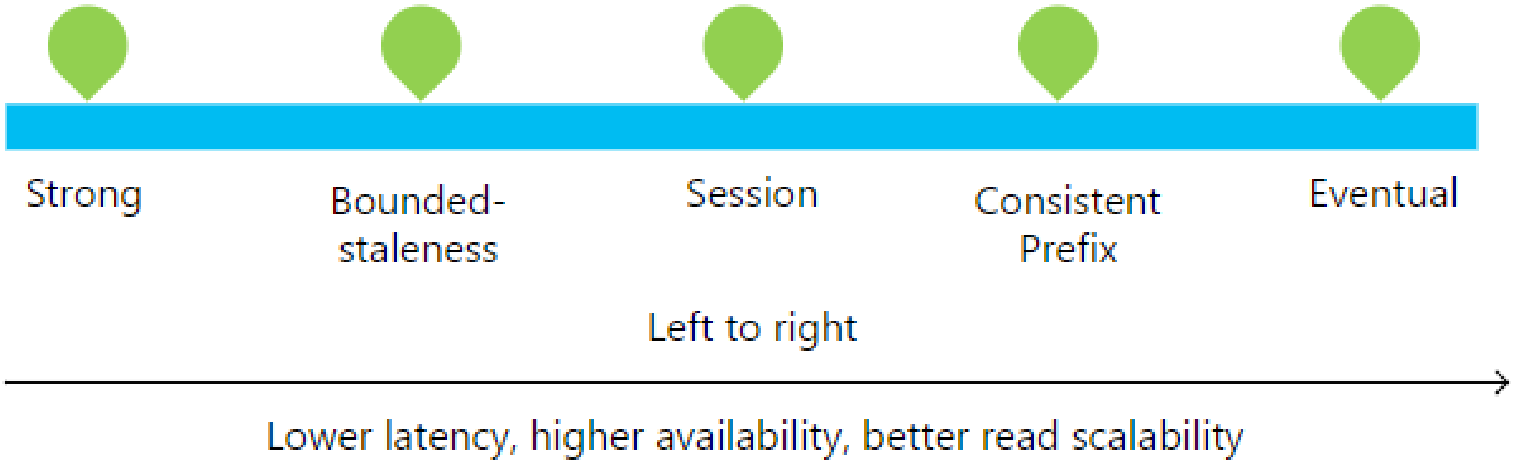
Microsoft's globally distributed, multi-model database service for mission-critical applications.

	.NET	☕	JS	🐍	
DocumentDB API	✓	✓	✓	✓	
MongoDB API	✓	✓	✓		
Graph API	✓	✓	✓		✓
Table API	✓				

CosomosDB Packages

- Nuget:
  - WindowsAzure.Storage-PremiumTable
  - Microsoft.Azure.DocumentDB
  - Microsoft.Azure.Graphs
- Emulator
  - https://docs.microsoft.com/en-us/azure/cosmos-db/local-emulator

Tunable Consistency – 5 levels



Consistency Level	Guarantees
Strong	Linearizability
Bounded Staleness	Consistent Prefix. Reads lag behind writes by k prefixes or t interval
Session	Consistent Prefix. Monotonic reads, monotonic writes, read-your-writes, write-follows-reads
Consistent Prefix	Updates returned are some prefix of all the updates, with no gaps
Eventual	Out of order reads

Tunable Consistency – 5 levels

- **Strong** - All writes are visible to all readers. Writes synchronously committed by a majority quorum of replicas and reads are acknowledged by the majority read quorum.
- **Bounded Stateless** - Guaranteed ordering of writes, reads adhere to minimum freshness. Writes are propagated asynchronously, reads are acknowledged by majority quorum lagging by at most K prefixes.
- **Session** - Read your own writes. Writes are propagated asynchronously while reads for a session are issued against the replica that can serve the requested version.
- **Consistent Prefix** - Updates returned are some prefix of all the updates, with no gaps
- **Eventual** - Reads eventually converge with writes. Writes are propagated asynchronously while reads can be acknowledged by any replica. Readers may view older data then previously observed.

Tunable Consistency

	Writes	*Reads*
Strong	sync quorum writes	quorum reads
Bounded	async replication	quorum reads
Session*	async replication	session bound replica
Eventual	async replication	any replica

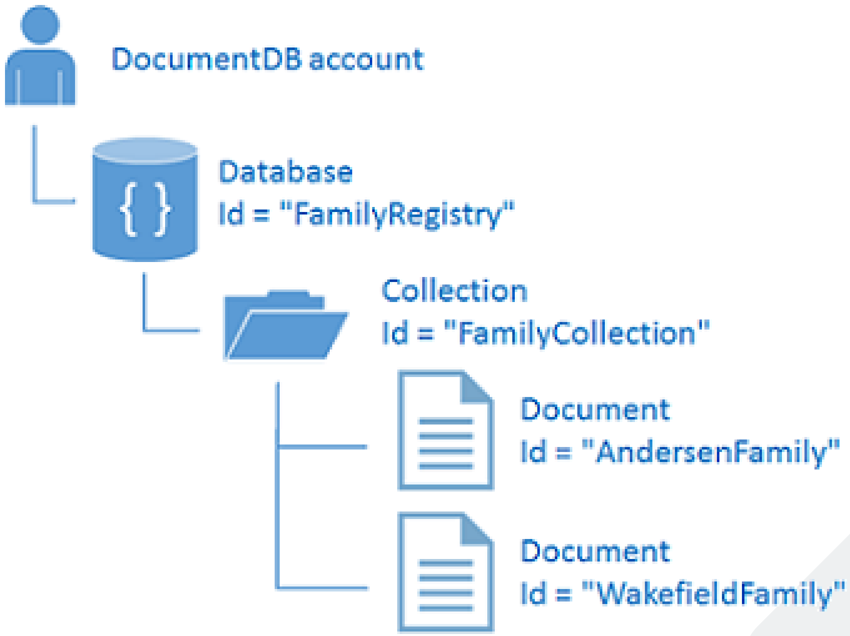
* Ideal consistency and performance tradeoff for many application scenarios. High performance writes and reads with predictable consistency.

Document database

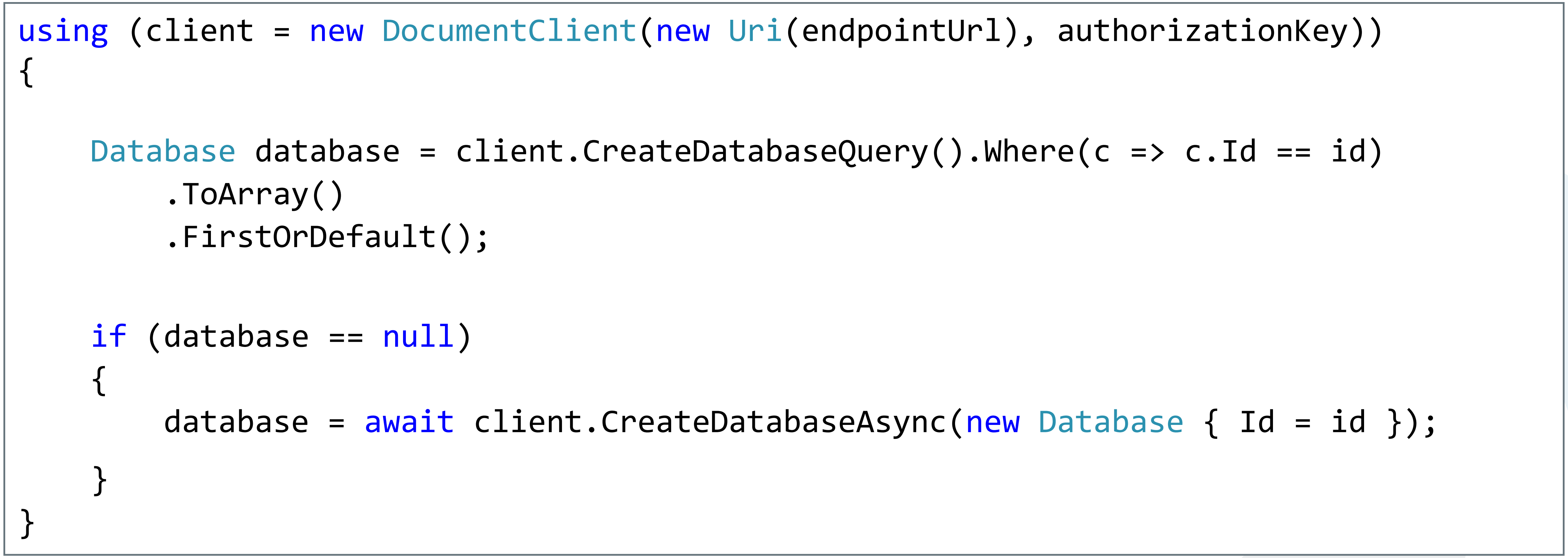
CosomosDB

CosmosDB - Document

- Fully managed, scalable JSON document database service
- Supports SQL as querying language
- Joins with nested documents is supported
- Everything is indexed



Creating DocumentDB Database



```
using (client = new DocumentClient(new Uri(endpointUrl), authorizationKey))
{

    Database database = client.CreateDatabaseQuery().Where(c => c.Id == id)
        .ToArray()
        .FirstOrDefault();

    if (database == null)
    {
        database = await client.CreateDatabaseAsync(new Database { Id = id });
    }
}
```

Creating DocumentDB Collection

```
DocumentCollection collection =
client.CreateDocumentCollectionQuery(UriFactory.CreateDatabaseUri(databaseId))
                .Where(c => c.Id == collectionId)
                .ToArray()
                .SingleOrDefault();

if (collection == null)
{
    DocumentCollection collectionDefinition = new DocumentCollection();
    collectionDefinition.Id = collectionId;
    collectionDefinition.IndexingPolicy = new IndexingPolicy(
       new RangeIndex(DataType.String) { Precision = -1 });
    collectionDefinition.PartitionKey.Paths.Add("/LastName");

    collection = await client.CreateDocumentCollectionAsync(
    UriFactory.CreateDatabaseUri(databaseId),
                        collectionDefinition,
                        new RequestOptions { OfferThroughput = 400 });
}
```

Creating DocumentDB Document

```
Family AndersonFamily = new Family() {...}
string collectionLink = collection.SelfLink;
await client.CreateDocumentAsync(collectionLink, AndersonFamily);
```

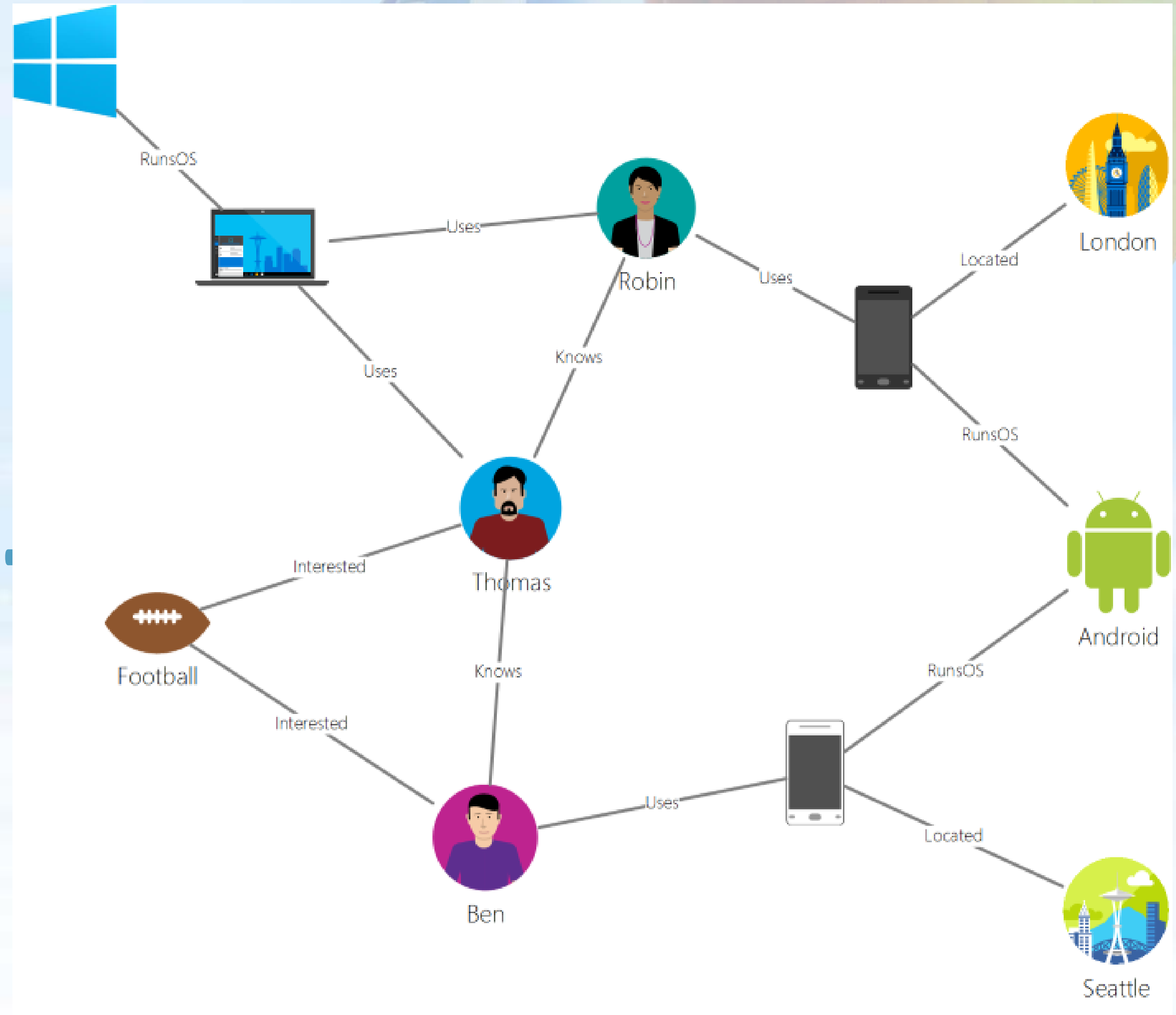
```
Document doc1 = await client.CreateDocumentAsync(col1.DocumentsLink,
       new { id = "doc1", partitionKey = "partitionKey1" });
```

Demo: Interacting with DocumentDB

```
var query = client.CreateDocumentQuery<Family>(
    collectionLink,
    new FeedOptions { MaxItemCount = 1, EnableCrossPartitionQuery = true })
    .Where(d => d.LastName == "Andersen")
    .Select(f => new { Name = f.LastName })
    .AsDocumentQuery();

foreach (var item in query.ToList())
{
    Console.WriteLine("The {0} family live in {1}", item.Name, item.City);
}
```

```
var query = client.CreateDocumentQuery<Family>(collectionLink, new SqlQuerySpec()
{
    QueryText = "SELECT * FROM Families f WHERE (f.id = @id)",
    Parameters = new SqlParameterCollection() { new SqlParameter("@id", "AndersenFamily")
    }
}, DefaultOptions);
```

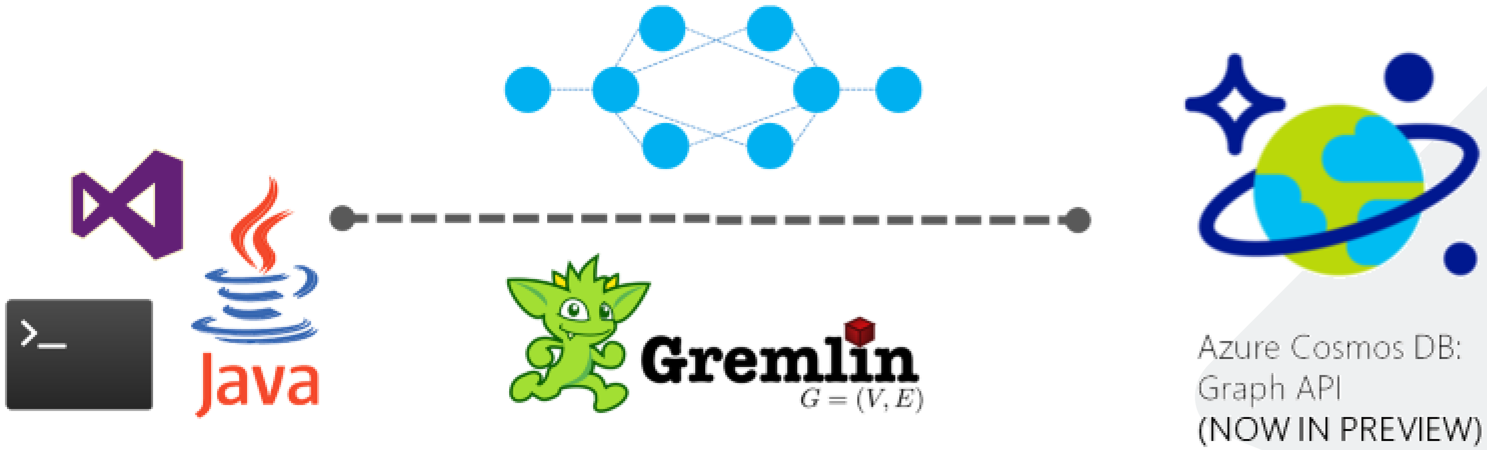
Graph da



CosomosDB

Gremlin

- Apache Tinkerpop's graph traversal language, Gremlin is a Graph API for creating graph entities, and performing graph query operations.
  - http://gremlindocs.spmallette.documentup.com/
  - https://docs.microsoft.com/en-us/azure/cosmos-db/graph-introduction



Gremlin

```
g.addV('person')
  .property('id', 'thomas')
  .property('firstName', 'Thomas')
  .property('age', 44)
```

```
g.addV('person')
  .property('id', ‘marry')
  .property('firstName', ‘Marry')
  .property('age', 35)
```

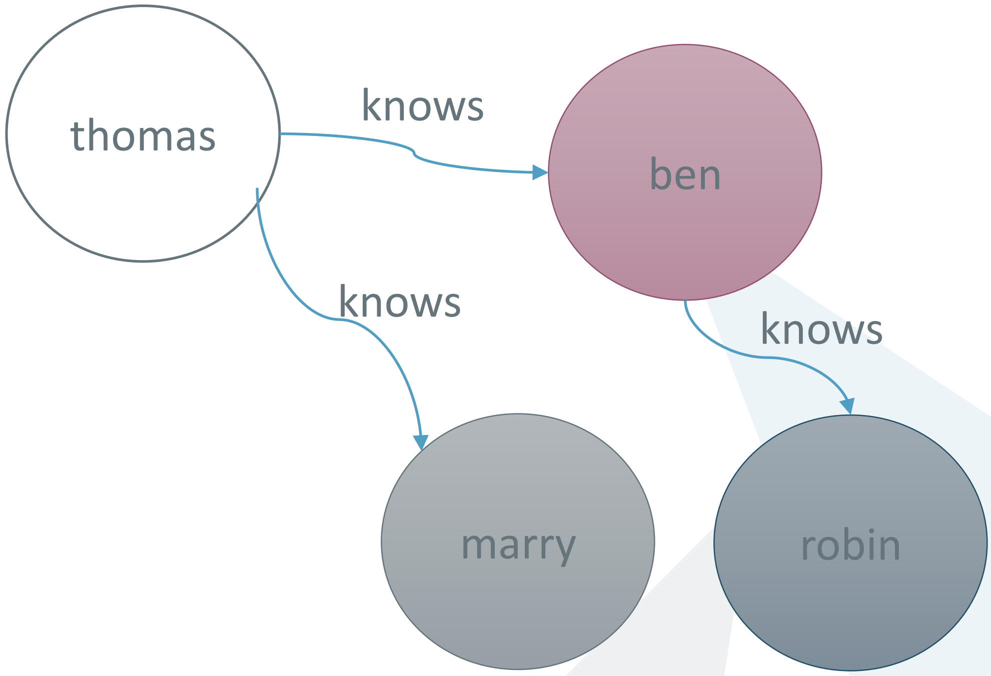
```
g.V('thomas')
  .addE('knows')
  .to(g.V('mary'))
```

```
g.addV('person')
  .property('id', ‘ben')
  .property('firstName', ‘Ben')
  .property('age', 36)
```

```
g.V('thomas')
  .addE('knows')
  .to(g.V(‘ben'))
```

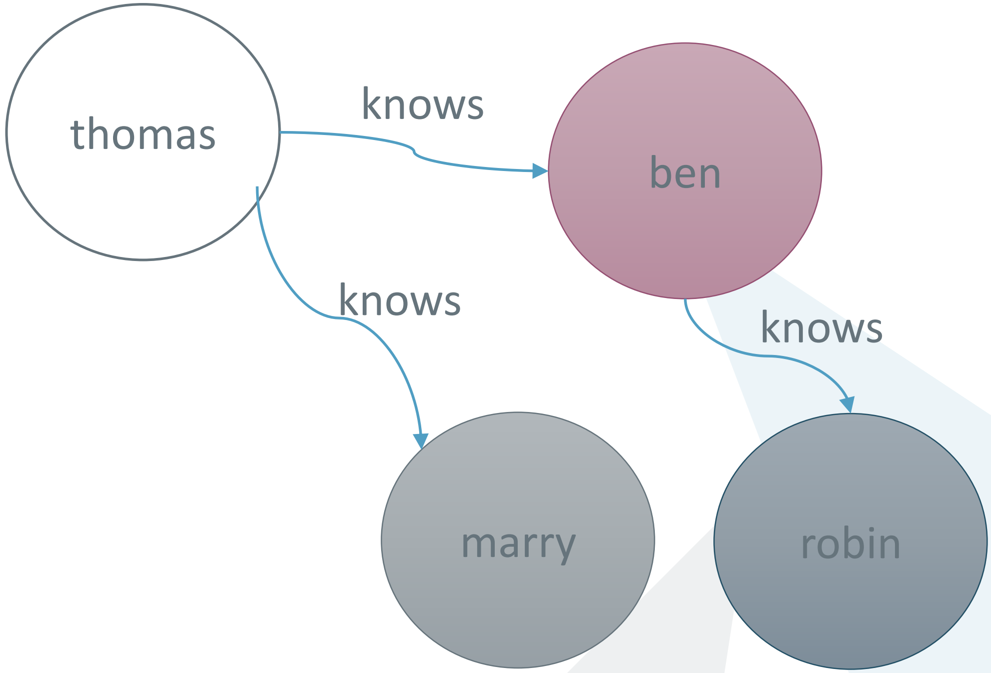
```
g.addV('person')
  .property('id', ‘robin')
  .property('firstName', ‘Robin')
  .property('age', 35)
```

```
g.V(‘ben')
  .addE('knows')
  .to(g.V(‘robin'))
```



Gremlin

```
g.V('thomas')
  .outE('knows')
  .inV().hasLabel('person')
  .outE('knows')
  .inV().hasLabel('person')
```



Summary

- **Choosing the correct persistence store is crucial for good architecture**
- **Realtional**
- **Storage Account**
  - **BLOBs**
  - **Tables**
  - **Queues**
  - **Files**
  - **Disks**
- **Shared Access Signatures (SAS) and Policies**
- **Partitioning and Consistency**
- **CosmosDB**
  - Document
  - Graph
  - Premium Tables

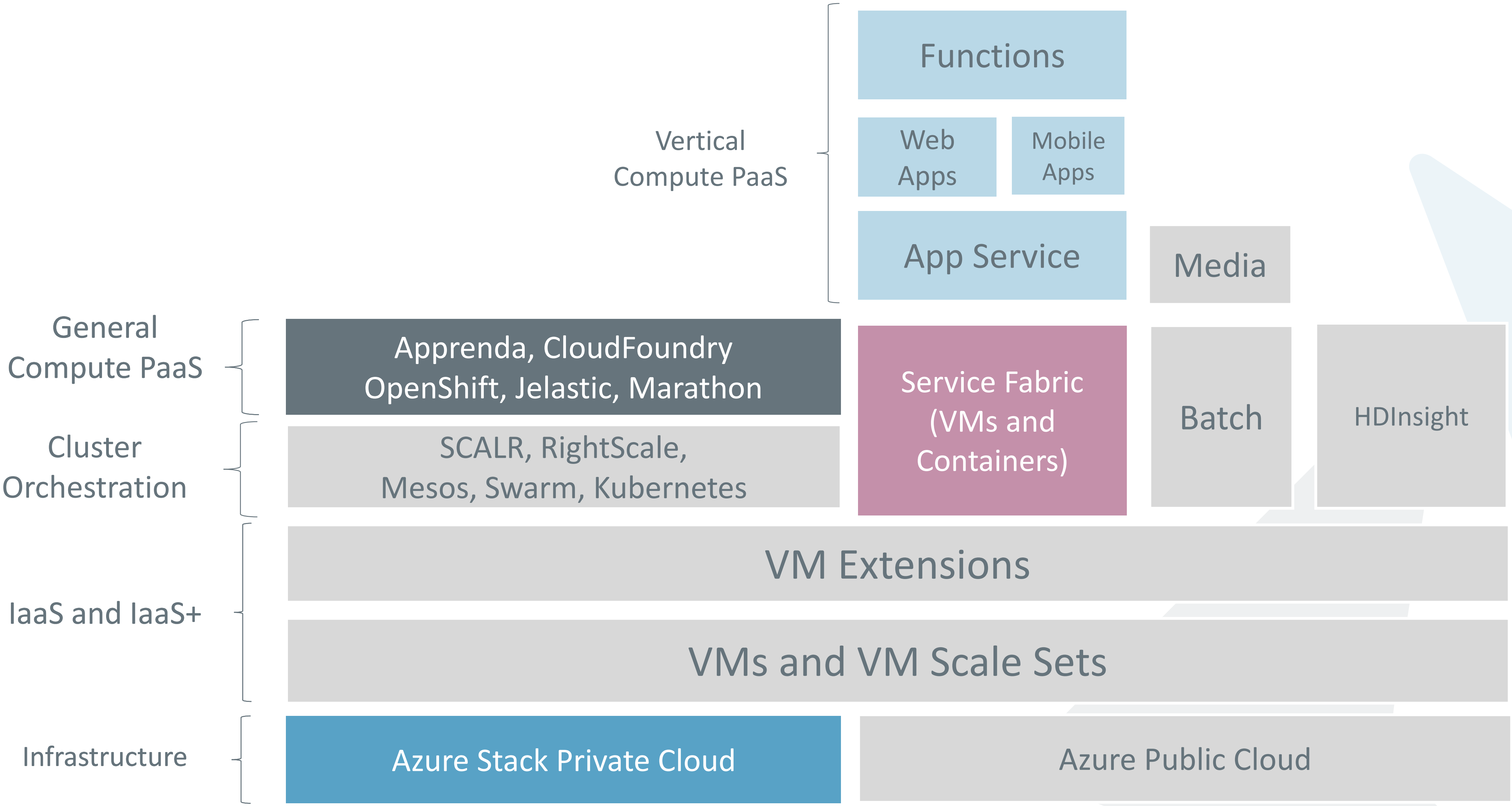
Other Data Services

- **Azure Search** - Full text search and text analysis, sophisticated data indexing

- **Azure Redis Cache** - gives you access to a secure, dedicated Redis cache, managed by Microsoft and accessible from any application within Azure.

PaaS - Azure App Service

Service hosting & management layers



Vertical Compute PaaS

Functions

Web Apps

Mobile Apps

App Service

Media

General Compute PaaS

Apprenda, CloudFoundry OpenShift, Jelastic, Marathon

Cluster Orchestration

SCALR, RightScale, Mesos, Swarm, Kubernetes

Service Fabric (VMs and Containers)

Batch

HDInsight

IaaS and IaaS+

VM Extensions

VMs and VM Scale Sets

Infrastructure

Azure Stack Private Cloud

Azure Public Cloud

Agenda

- Web Apps
  - WebJobs
- App Service Plans
- Deployment Slots
- Mobile Apps
- API Apps
- Logic Apps
- Azure Functions

Azure App Services

- Building and hosting web applications without managing infrastructure
- Offers auto-scaling and high availability
- Supports both Windows and Linux
- Enables automated deployments from GitHub, VSTS, or any Git repo
- Service App suite includes
  - **Web Apps** - websites and web applications
  - **API Apps** - RESTful APIs
  - **Mobile Apps** - mobile app back ends
  - **Logic Apps** - automating business processes and integrating systems and data across clouds without writing code
- **Azure Functions** are also based on the App Services infrastructure

Azure web APPS

Azure App Services

Azure Web Sites

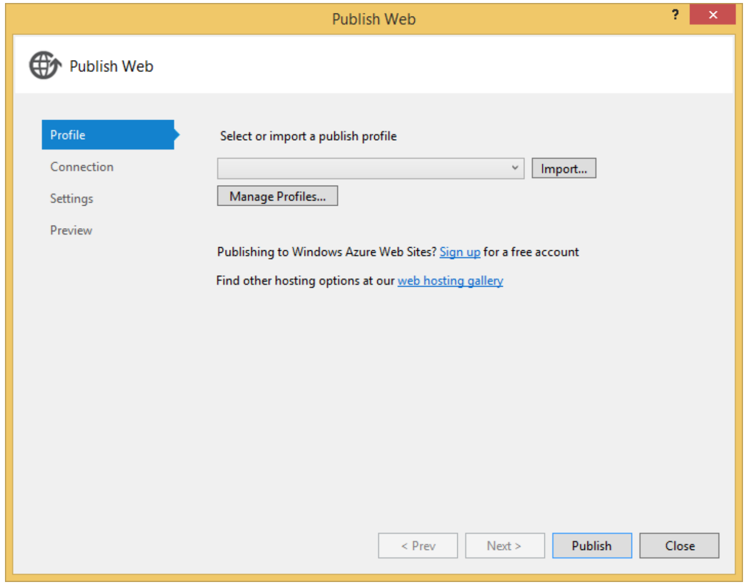
- Provision a Web Application Fast
- You can use IDE, PowerShell, Portal
- Deploy Easily via a Source Control

Creating a Web Site

Publish from VS

From Gallery

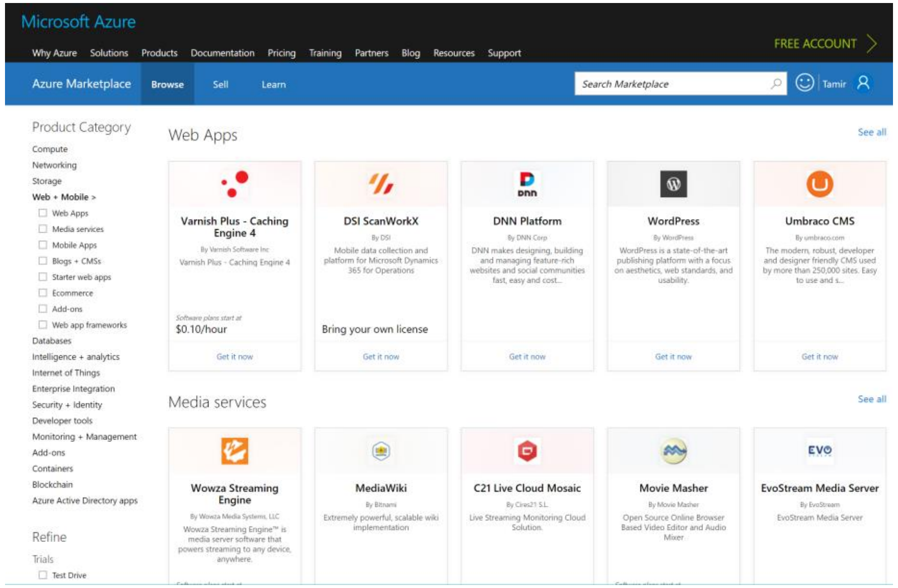
Sync with Source Control



Select or import a publish profile

Publishing to Windows Azure Web Sites? Sign up for a free account

Find other hosting options at our web hosting gallery



Web Apps

Varnish Plus - Caching Engine 4
By Varnish Software Inc
Varnish Plus - Caching Engine 4
Software plans start at
$0.10/hour

DSI ScanWorkX
By DSI
Mobile data collection and platform for Microsoft Dynamics 365 for Operations
Bring your own license

DNN Platform
By DNN Corp
DNN makes designing, building and managing feature-rich websites and social communities fast, easy and cost...

WordPress
By WordPress
WordPress is a state-of-the-art publishing platform with a focus on aesthetics, web standards, and usability.

Umbraco CMS
By umbraco.com
The modern, robust, developer and designer friendly CMS used by more than 250,000 sites. Easy to use and s...

Media services

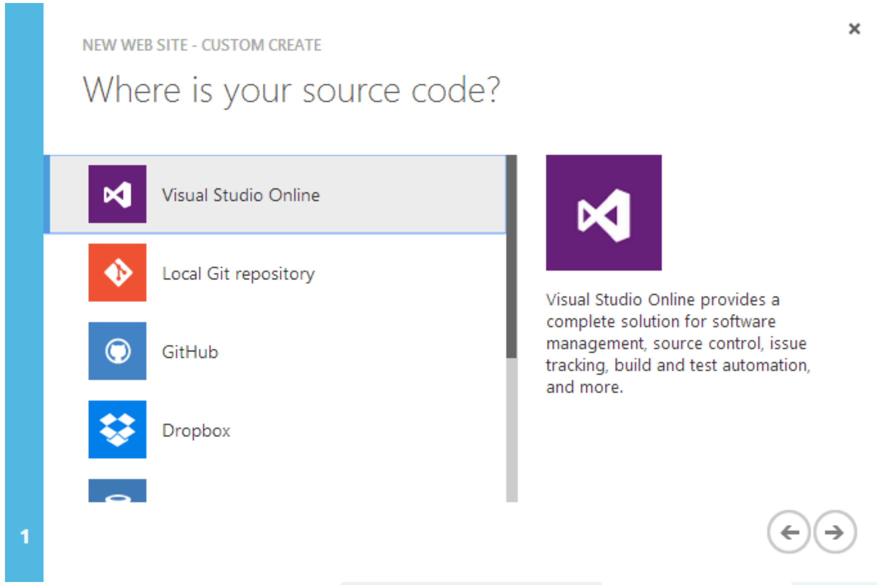
Wowza Streaming Engine
By Wowza Media Systems, LLC
Wowza Streaming Engine™ is media server software that powers streaming to any device, anywhere.

MediaWiki
By Bitnami
Extremely powerful, scalable wiki implementation

C21 Live Cloud Mosaic
By Cires21 S.L.
Live Streaming Monitoring Cloud Solution.

Movie Masher
By Movie Masher
Open Source Online Browser Based Video Editor and Audio Mixer

EvoStream Media Server
By EvoStream
EvoStream Media Server



NEW WEB SITE - CUSTOM CREATE

Where is your source code?

Visual Studio Online provides a complete solution for software management, source control, issue tracking, build and test automation, and more.

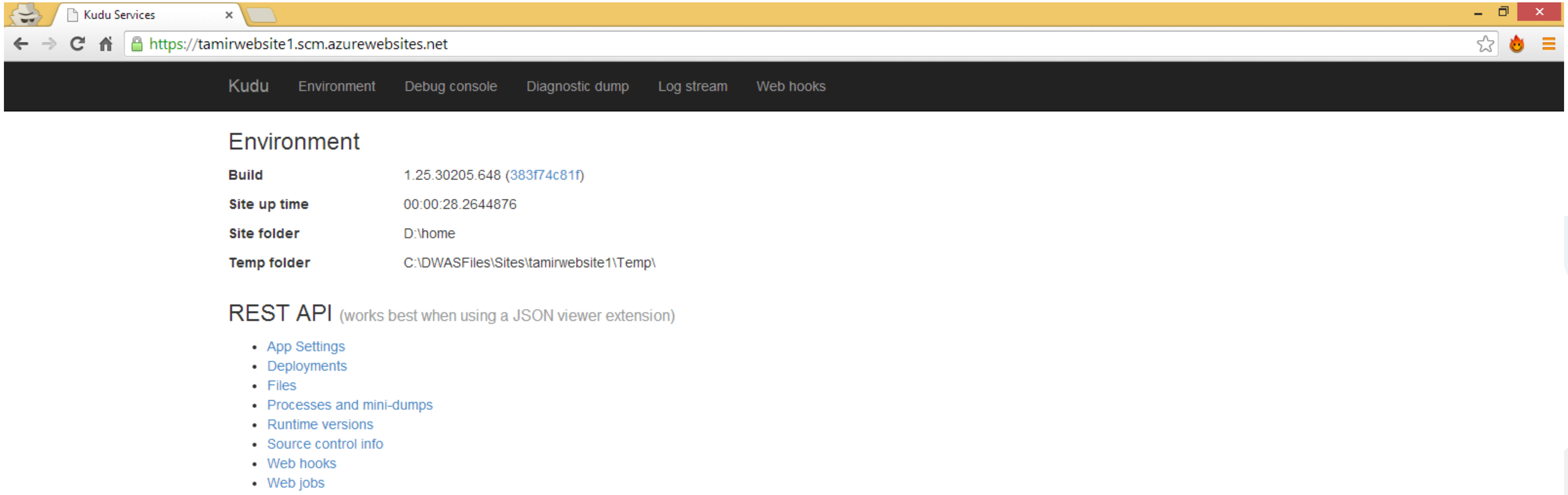
DEMO

Publishing from VS

kudu

- Every Azure web site has an associated Kudu service site
- Kudu is the engine behind git deployments in Azure Web Sites
- If your web site has URL
  http://mysite.azurewebsites.net/
  then the root URL of the Kudu service is
  https://mysite.**scm**.azurewebsites.net/.
- Gives monitoring utils for the deployment

kudu



Environment

Build 1.25.30205.648 (383f74c81f)

Site up time 00:00:28.2644876

Site folder D:\home

Temp folder C:\DWASFiles\Sites\tamirwebsite1\Temp\

REST API (works best when using a JSON viewer extension)

- App Settings
- Deployments
- Files
- Processes and mini-dumps
- Runtime versions
- Source control info
- Web hooks
- Web jobs

WebJobs

- Windows Azure Web App enables you to run custom jobs (running executables or scripts) on your web site
- The WebJobs SDK has a binding and trigger system which works with Windows Azure Storage Blobs, Queues and Tables.
- The trigger system calls a function in your code whenever any new data is received in a queue or blob.
- You can create your own binders and triggers
- Most of WebJobs functionality is now provided by Azure Functions

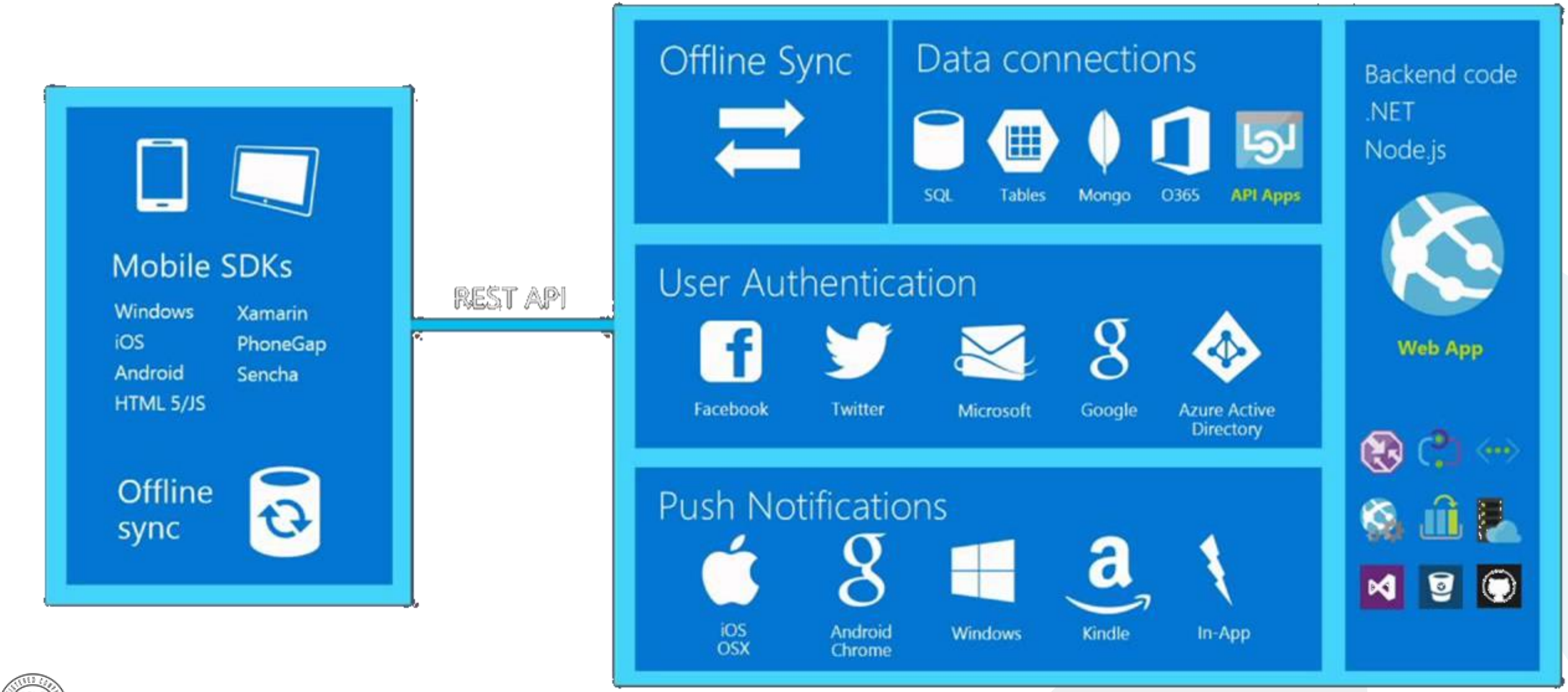
App Service Plan

- Represents a set of feature and capacity that can be shared across multiple apps in Azure App Service.
- This is the physical resources representative
- 5 pricing tiers – Free, Shared, Basic, Standard, Premium
- Apps can share the Service Plan if they are in the same subscription and same location
- A good usage for example is to share resources for each environment (DEV, TEST, PROD)

Mobile Apps

Azure App Services

What is Mobile Apps?



Structured Storage

- Powered by SQL Database
- Same DB – Multiple Mobile Services
- Data management in
  - Windows Azure Portal
  - SQL Portal
  - SQL Management Studio
  - REST API
  - CLI Tools
- JSON to SQL Type Mappings

The REST API

Base REST API Endpoint URL

https://Mobileservice.azure-mobile.net/tables/*

Data Operations and their REST Equivalents

Action	HTTP Verb	URL Suffix
Create	POST	/TodoItem
Read	GET	/TodoItem?$filter=id%3D42
Update	PATCH	/TodoItem/id
Delete	DELETE	/TodoItem/id

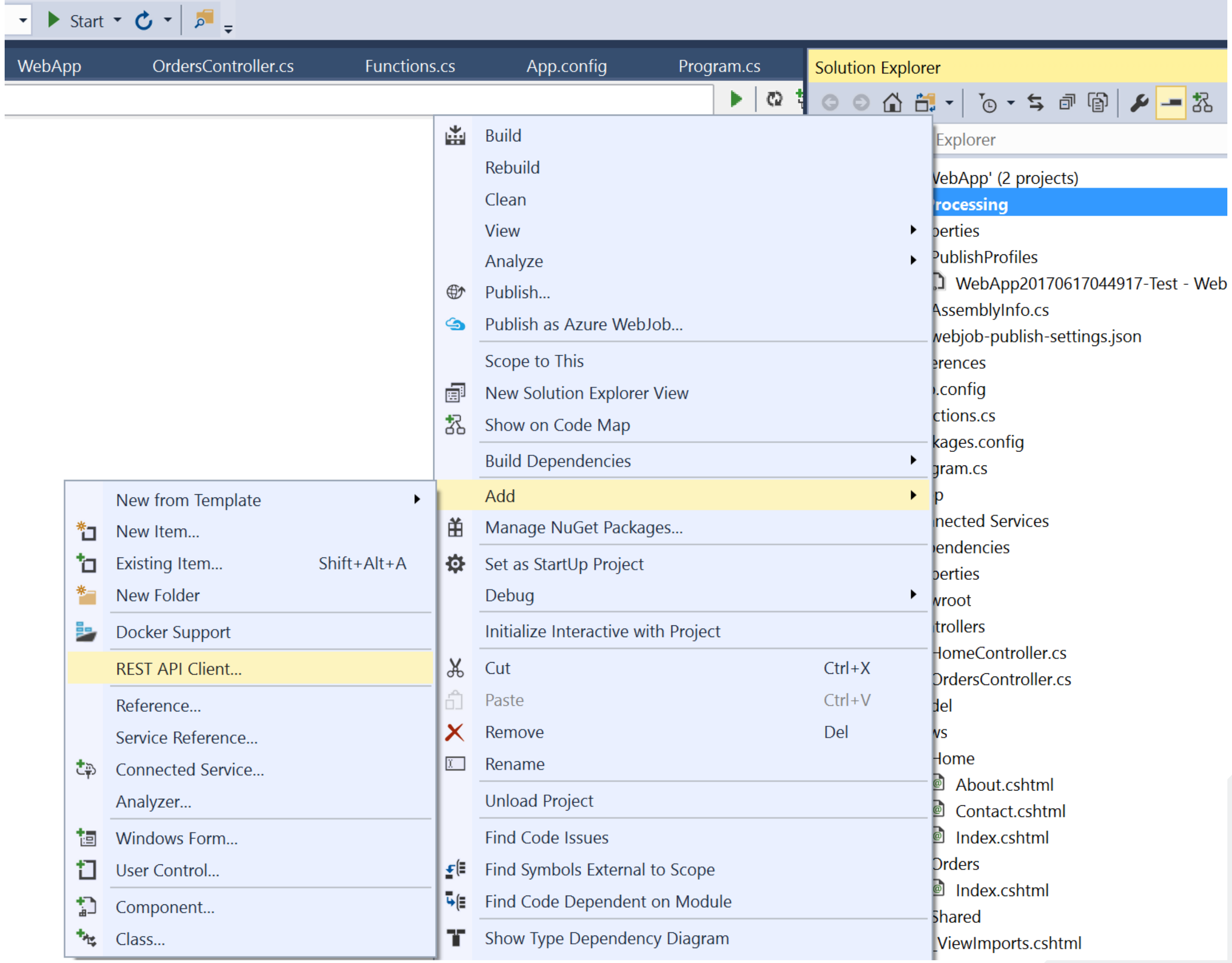
API Apps

Azure App Services

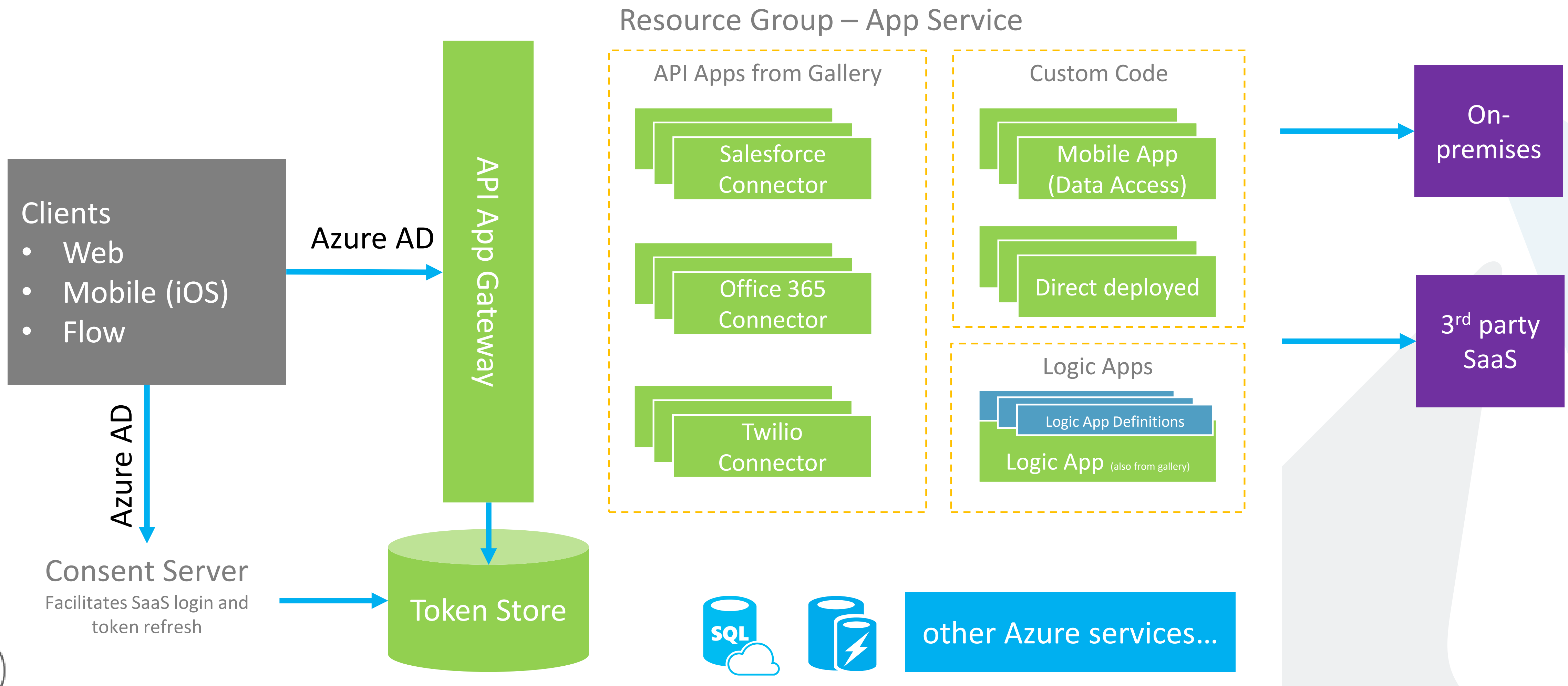
API Apps

- Basically Web Apps for Web Api’s
- Simple access control
- Swagger metadata
- Logic App Integration
- Marketplace support for connectors
- VS tooling and support (for client side as well)

Creating REST Client from Swagger



API Apps Architecture Example



Logic Apps

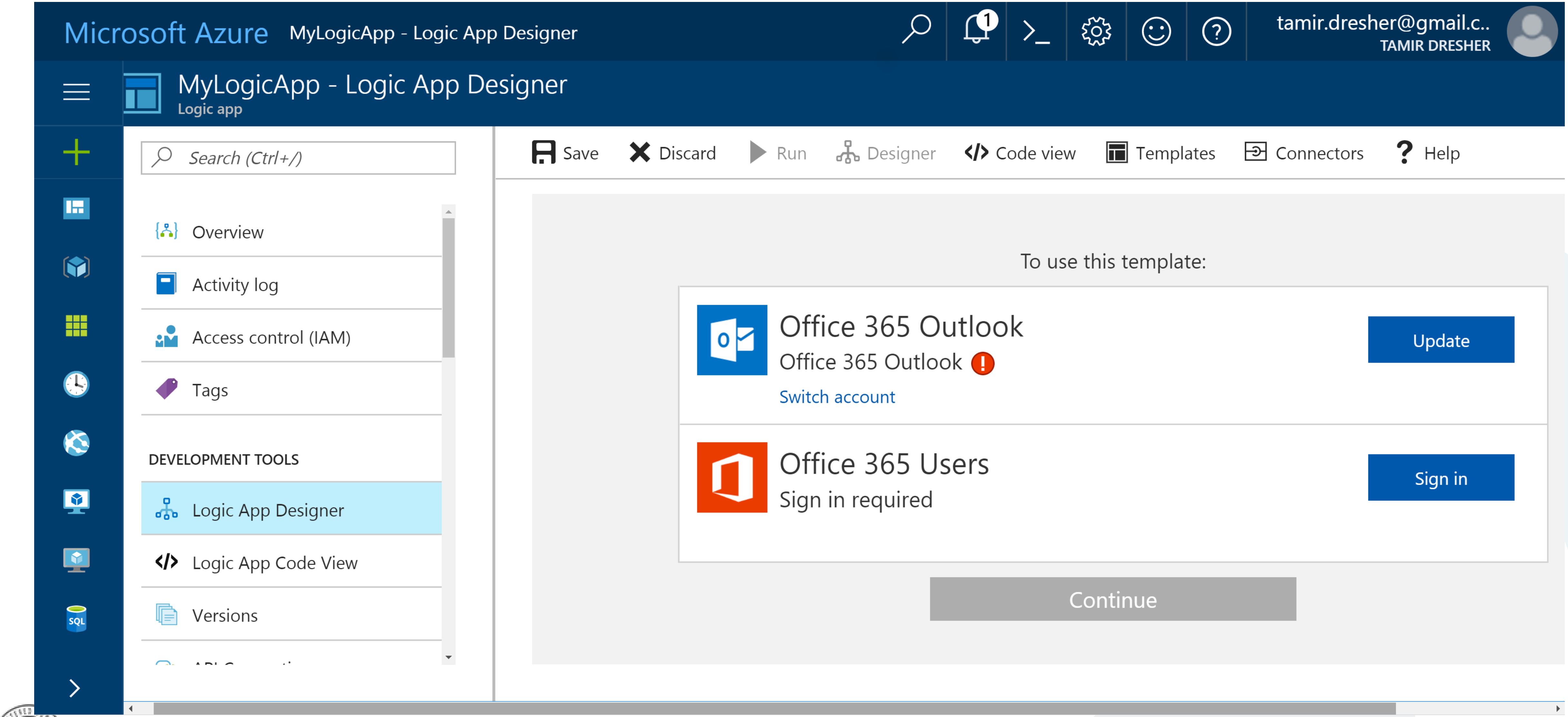
Azure App Services

Logic Apps

- Visually create business process and workflows based on Triggers and Actions
- Deliver integration capabilities in Web, Mobile, and API Apps
- Integrate with your SaaS and enterprise applications
- Automate EAI/B2B and business processes
- Connect to on-premises data

ISO 9001 SGS

Logic App Designer



Microsoft Azure MyLogicApp - Logic App Designer

tamir.dresher@gmail.c..
TAMIR DRESHER

MyLogicApp - Logic App Designer
Logic app

Search (Ctrl+/)

- Overview
- Activity log
- Access control (IAM)
- Tags

DEVELOPMENT TOOLS

- Logic App Designer
- Logic App Code View
- Versions

Save | Discard | Run | Designer | Code view | Templates | Connectors | Help

To use this template:

Office 365 Outlook
Office 365 Outlook
Switch account
Update

Office 365 Users
Sign in required
Sign in

Continue

Built-in API Connectors

Connectors

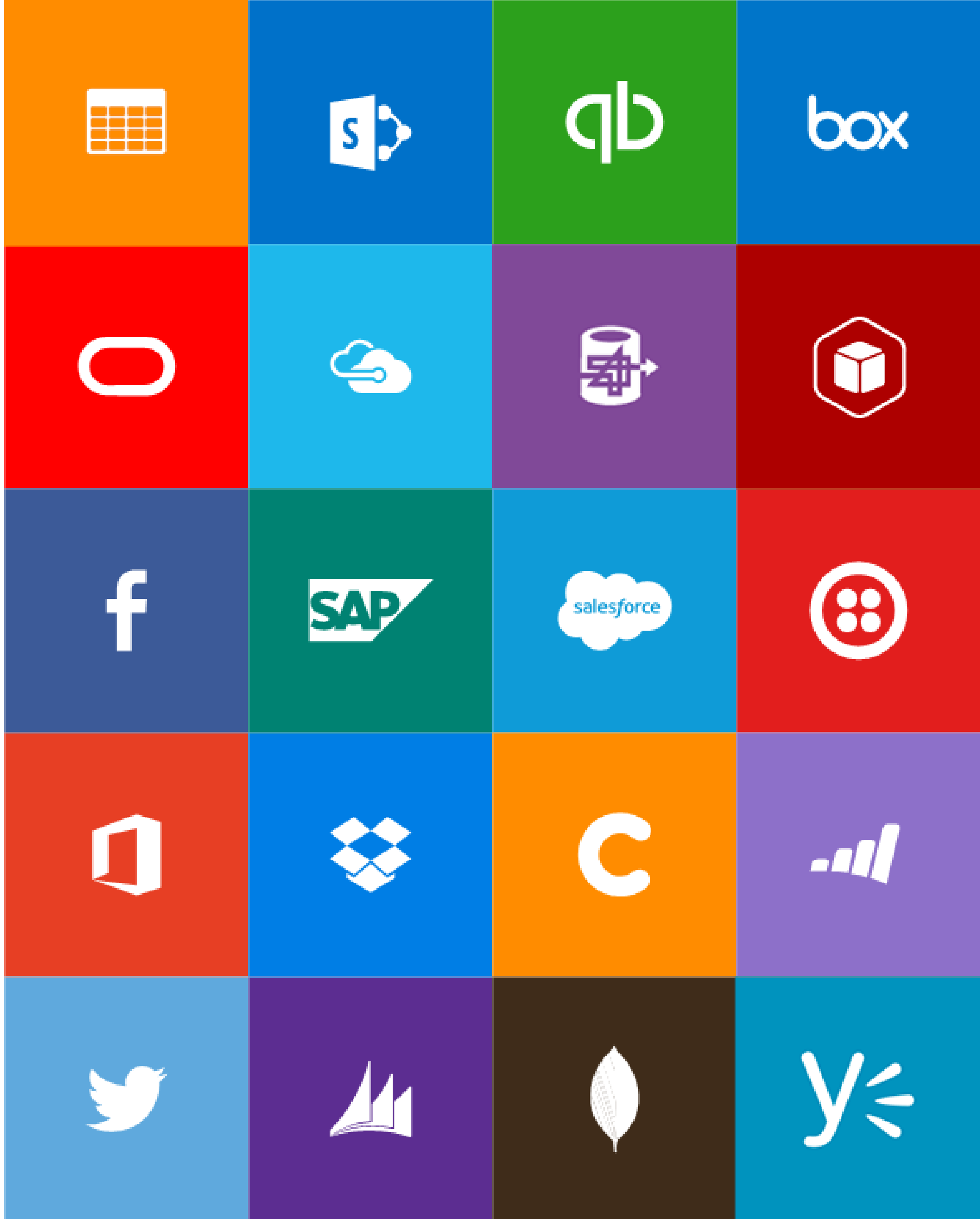
- Box
- Chatter
- Delay
- Dropbox
- Azure HD Insight
- Marketo
- Azure Media Services
- OneDrive
- SharePoint
- SQL Server
- Office 365
- Oracle
- QuickBooks
- SalesForce
- Sugar CRM
- SAP
- Azure Service Bus
- Azure Storage
- Timer / Recurrence
- Twilio
- Twitter
- IBM DB2
- Informix
- Websphere MQ
- Azure Web Jobs
- Yammer
- Dynamics CRM
- Dynamics AX
- Hybrid Connectivity

Protocols

- HTTP, HTTPS
- File
- Flat File
- FTP, SFTP
- POP3/IMAP
- SMTP
- SOAP + WCF

BizTalk Services

- Batching / Debatching
- Validate
- Extract (XPath)
- Transform (+Mapper)
- Convert (XML-JSON)
- Convert (XML-FF)
- X12
- EDIFACT
- AS2
- TPMOM
- Rules Engine



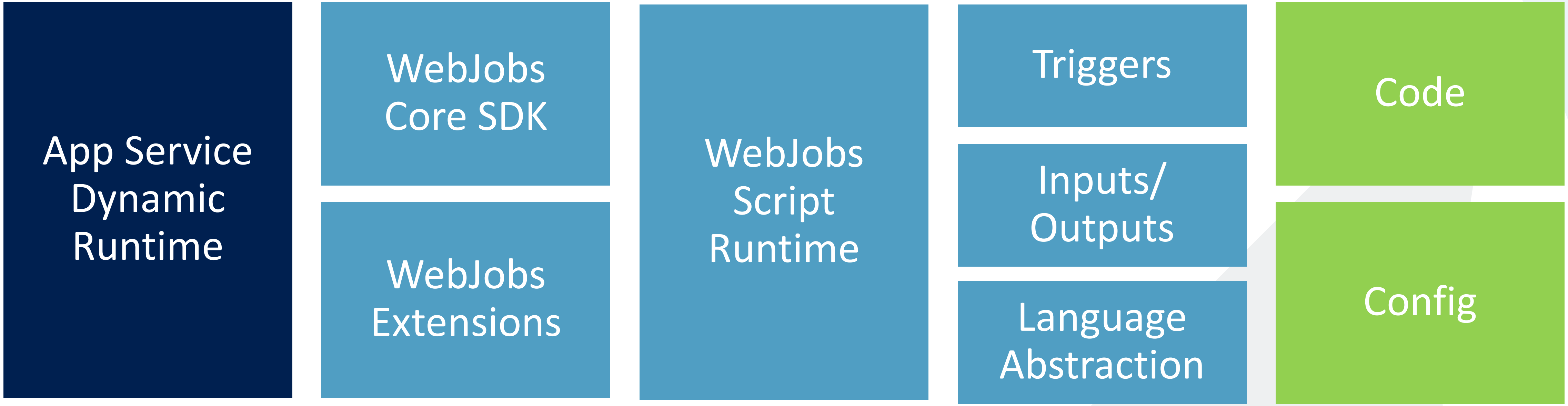
Azure Functions

- Azure Functions is an event driven, compute-on-demand experience
- Azure Functions scale based on demand and you pay only for the resources you consume.
- Function can be written in C# or nodeJS
- The runtime, otherwise known as the script host, is the underlying WebJobs SDK host which listens for events, gathers and sends data, and ultimately runs your code.

ISO 9001 SGS

Azure Functions architecture

- Azure Functions is built around the WebJobs SDK runtime. The WebJobs SDK makes it easy to react to events and work with data in a consistent abstracted fashion.



App Service Dynamic Runtime

WebJobs Core SDK

WebJobs Extensions

WebJobs Script Runtime

Triggers

Inputs/ Outputs

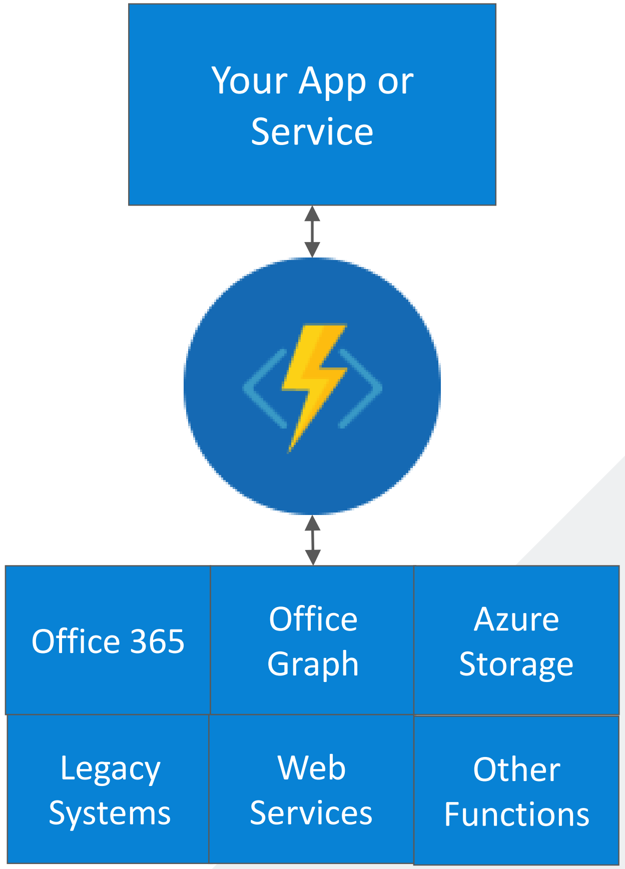
Language Abstraction

Code

Config

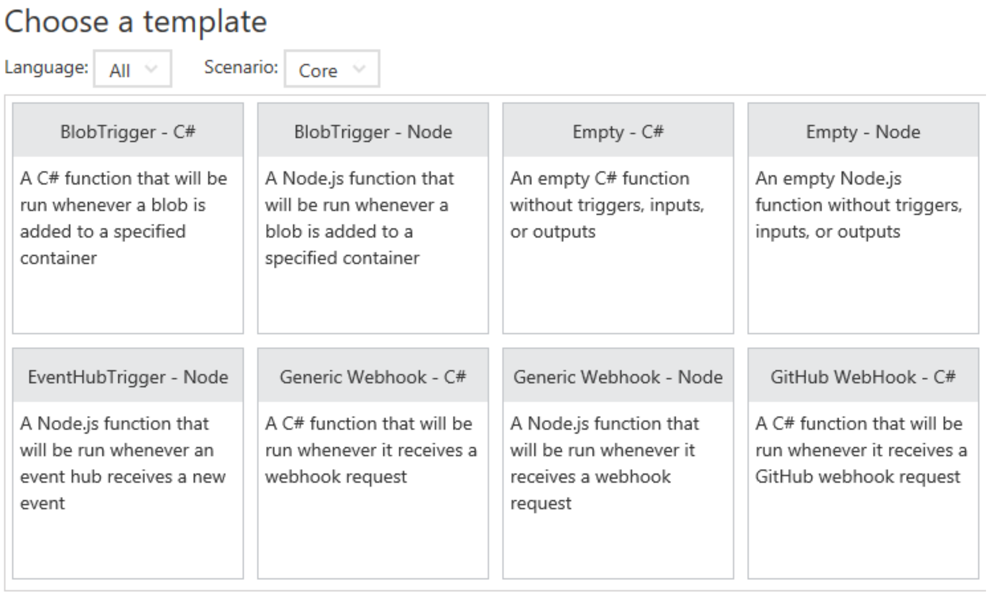
Common Scenarios

- Timer-based processing
- Azure service event processing
- SaaS event processing
- Serverless web application architectures
- Serverless mobile backends
- Real-time stream processing
- Real-time bot messaging



Function App Templates

Function App templates are categorized into general areas of Timer, Data Processing, and Webhook & API



Choose a template

Language: All Scenario: Core

BlobTrigger - C#	BlobTrigger - Node	Empty - C#	Empty - Node
A C# function that will be run whenever a blob is added to a specified container	A Node.js function that will be run whenever a blob is added to a specified container	An empty C# function without triggers, inputs, or outputs	An empty Node.js function without triggers, inputs, or outputs
EventHubTrigger - Node	Generic Webhook - C#	Generic Webhook - Node	GitHub WebHook - C#
A Node.js function that will be run whenever an event hub receives a new event	A C# function that will be run whenever it receives a webhook request	A Node.js function that will be run whenever it receives a webhook request	A C# function that will be run whenever it receives a GitHub webhook request

- BlobTrigger
- EventHubTrigger
- Generic webhook
- GitHub webhook
- HTTPTrigger
- QueueTrigger
- ServiceBusQueueTrigger
- ServiceBusTopicTrigger
- TimerTrigger
- Blank & Experimental

Azure Functions folder structure

- nodeJS

```
mynodefunction
| - function.json
| - index.js
| - node_modules
| | - ... packages ...
| - package.json
```

- C#

```
mycsharpfunction
| - function.json
| - run.csx
```

function.json

```
{
    "disabled":false,
    "bindings":[
        // ... bindings here
        {
            "type": "bindingType",
            "direction": "in",
            "name": "myParamName",
            // ... more depending on binding
        }
    ]
}
```

- Type - Binding type. For example, queueTrigger.
- direction - 'in', 'out'' Indicates whether the binding is for receiving data into the function or sending data from the function.
- name - The name that will be used for the bound data in the function. For C# this will be an argument name; for JavaScript it will be the key in a key/value list.

logging

- To log output to your streaming logs in C#, you can include a TraceWriter typed argument. We recommend that you name it **log** or **logger**. It’s recommend to avoid using Console.Write in Azure Functions.

```
 public static void Run(string myBlob, TraceWriter log)
{
    log.Verbose($"C# Blob trigger function processed: {myBlob}");
}
```

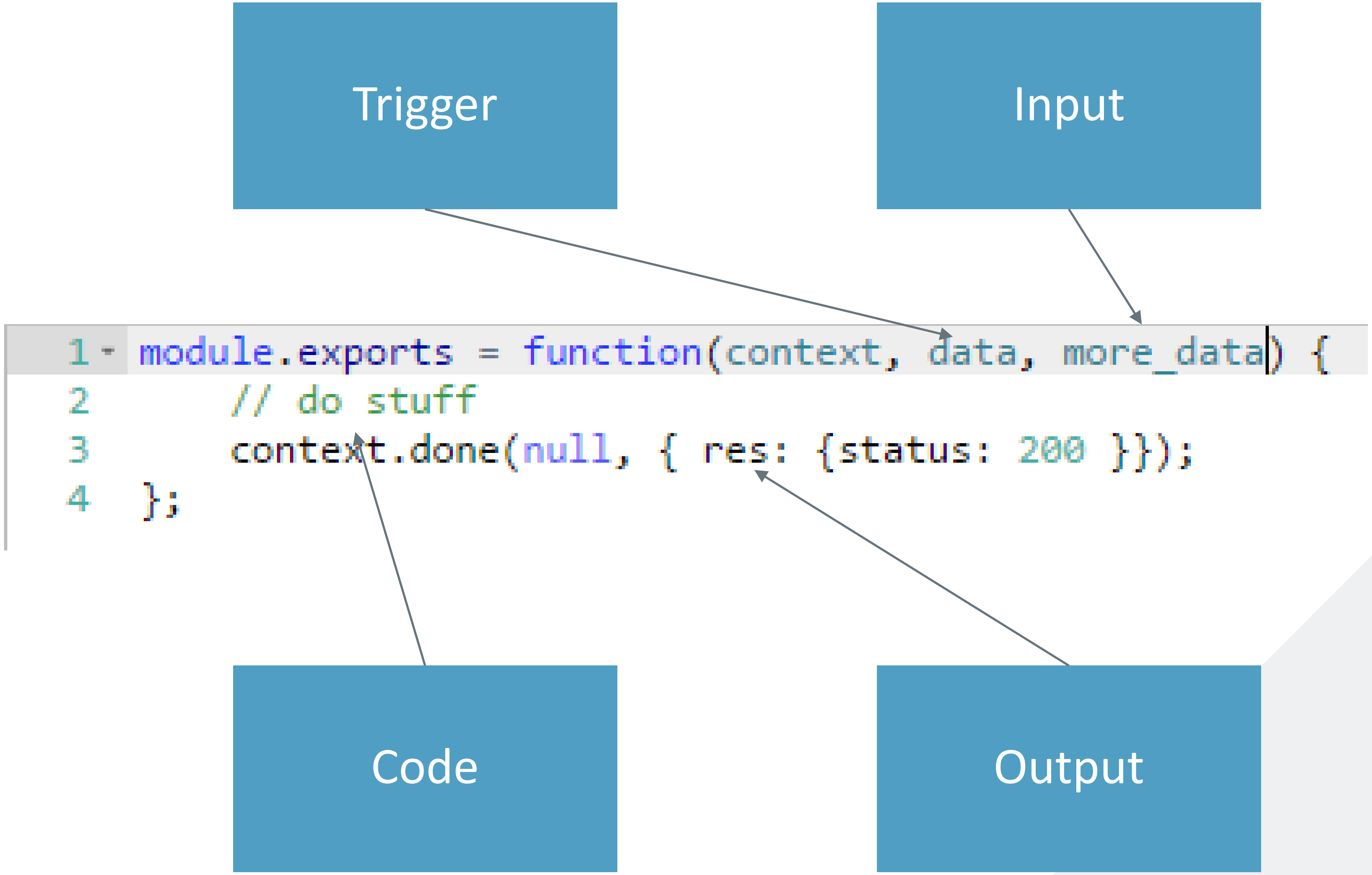
CSX

- A lightweight C# script
- Only the .NET Framework 4.6 is supported
- If you need to reference a private assembly, you can upload the assembly file into a bin folder relative to your function and reference it by using the file name
  - `#r "AssemblyName“`
- Supports Nuget by adding the `packages.json`
  - When you upload a *project.json* file, the runtime gets the packages and automatically adds references to the package assemblies
- Other *.csx files can be reused by adding #load "myfile.csx"

Dynamic tier pricing

- Pay per execution model - two meters, three units
- Number of executions
- Duration of execution x reserved memory

Functions programming concepts



```
module.exports = function(context, data, more_data) {
      // do stuff
      context.done(null, { res: {status: 200 }});
};
```

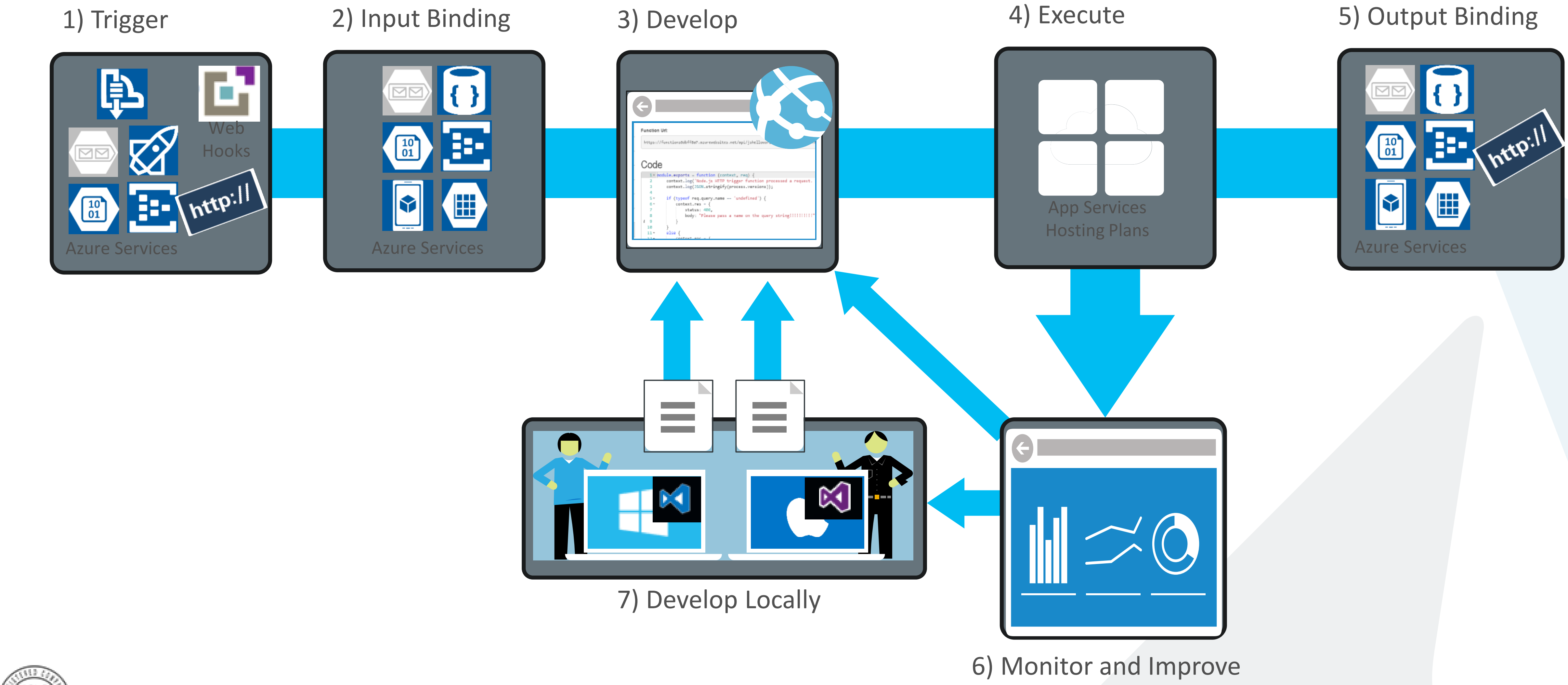
Trigger

Input

Code

Output

Azure Functions



Function Url:

https://functions8dbff8a7.azurewebsites.net/api/jshellowo

Code

```
module.exports = function (context, req) {
    context.log('Node.js HTTP trigger function processed a request.
    context.log(JSON.stringify(process.versions));

    if (typeof req.query.name == 'undefined') {
        context.res = {
            status: 400,
            body: "Please pass a name on the query string!!!!!!!!!!"
        }
    }
    else {
```

Benefits of “serverless”

- “Pinnacle of PaaS compute”
- Not just hardware “servers”, but software servers are also **managed for you**
- Focus on **business logic**, not solving technical problems not **core to business**
- Lower effort to get started makes it easier to experiment (bots, etc.)

Signs that a serverless pattern might be useful for a given scenario

1. Stateless → Scale
2. Too complicated to deploy a traditional backend
3. Workload is sporadic (very low & high scale)
4. (Human) Operational costs need to stay low
5. Lots of different services involved

ISO 9001 SGS

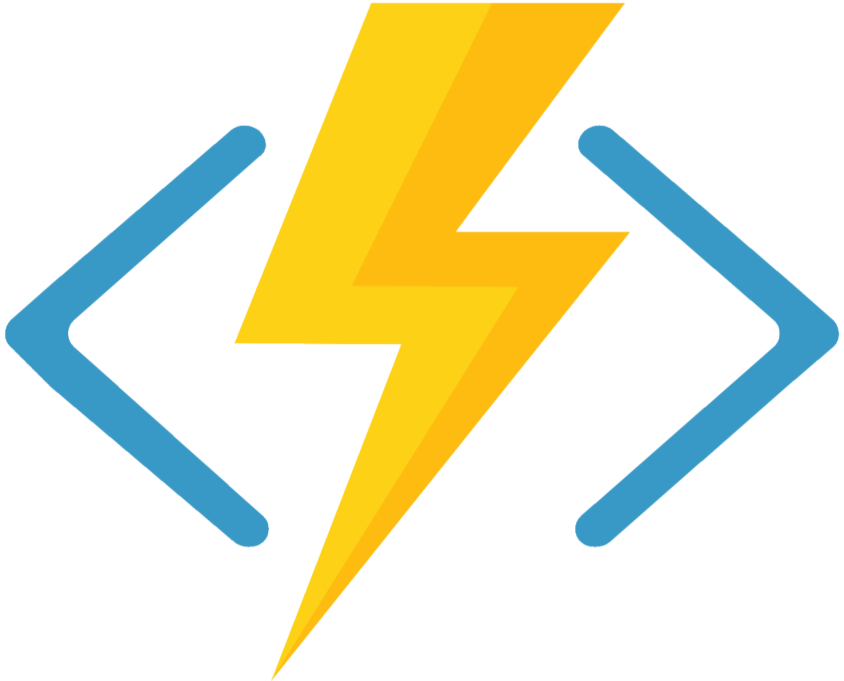
Suggestions for getting started

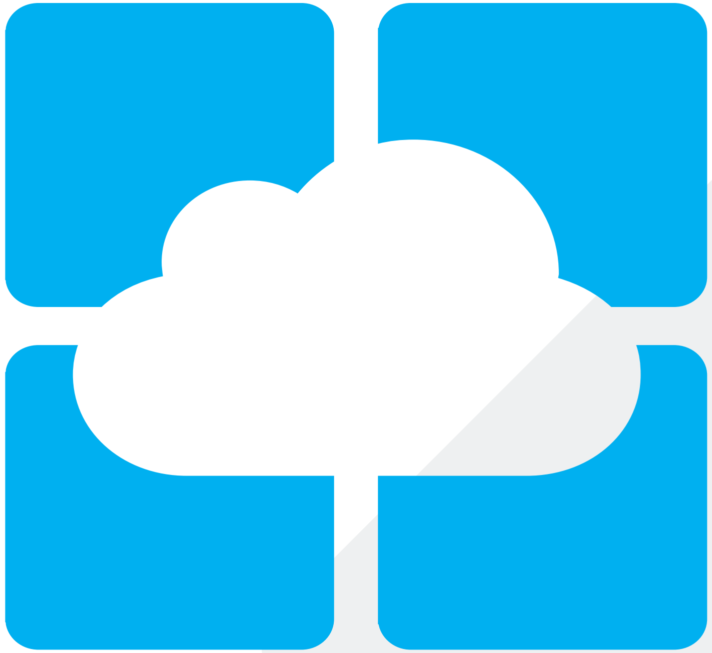
1. For existing services, start small. Replace 1 API or background processing item
2. Integration is a great place to introduce serverless, because it is often a new layer on top of old layers
3. For new services, establish a pattern early and stick with it. Lack of tooling/established patterns mean you pay an early adopter tax. Build automation asap

ISO 9001 SGS

Get started and reach out!

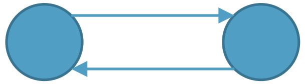
- Try Azure - https://azure.microsoft.com/en-us/free/
- Try Functions – https://functions.azure.com
- Try App Service – https://tryappservice.azure.com





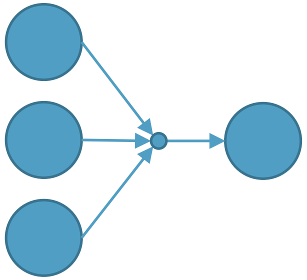
Advanced Messaging

Communication Patterns



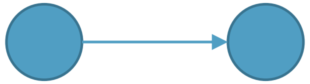
Synchronous Request-Reply

- A.K.A RPC – Remote Procedure Call
- Client synchronously wait for the server response
- Connection remains open -> Increase load on server
- Sent message is not durable



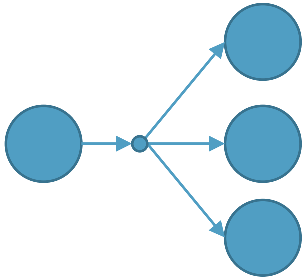
Fan in

- Server receives messages asynchronously from multiple producers
- Decoupling of client and server



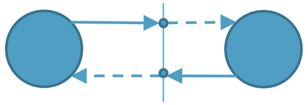
Fire and Forget

- After the server acknowledge, client continues without waiting for response (not even for operation completion)
- Sent message is not durable



Fan out

- A.K.A Publish-Subscribe (PubSub)
- The producer broadcasts a message
- Decoupling of client and server

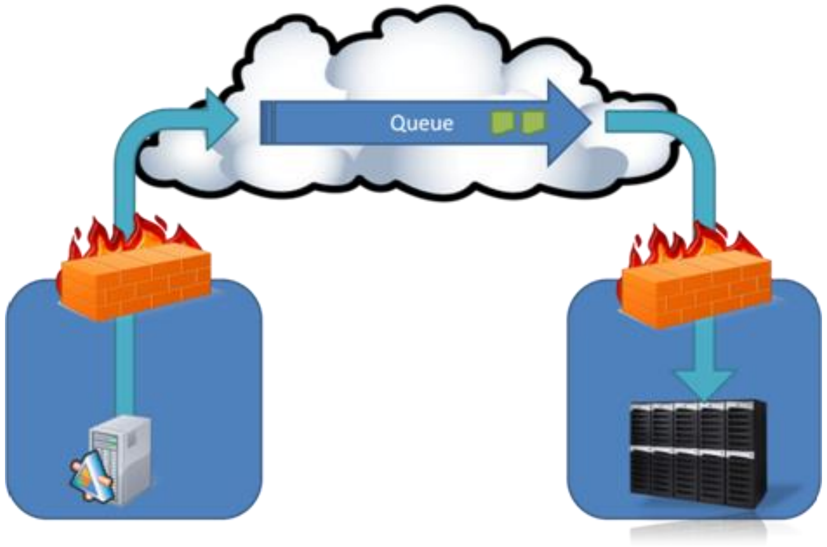


Asynchronous Request-Reply

- Decoupling of client and server
- The server asynchronously process the message and post a response
- The client asynchronously process the response

Service Bus/Queue Scenarios

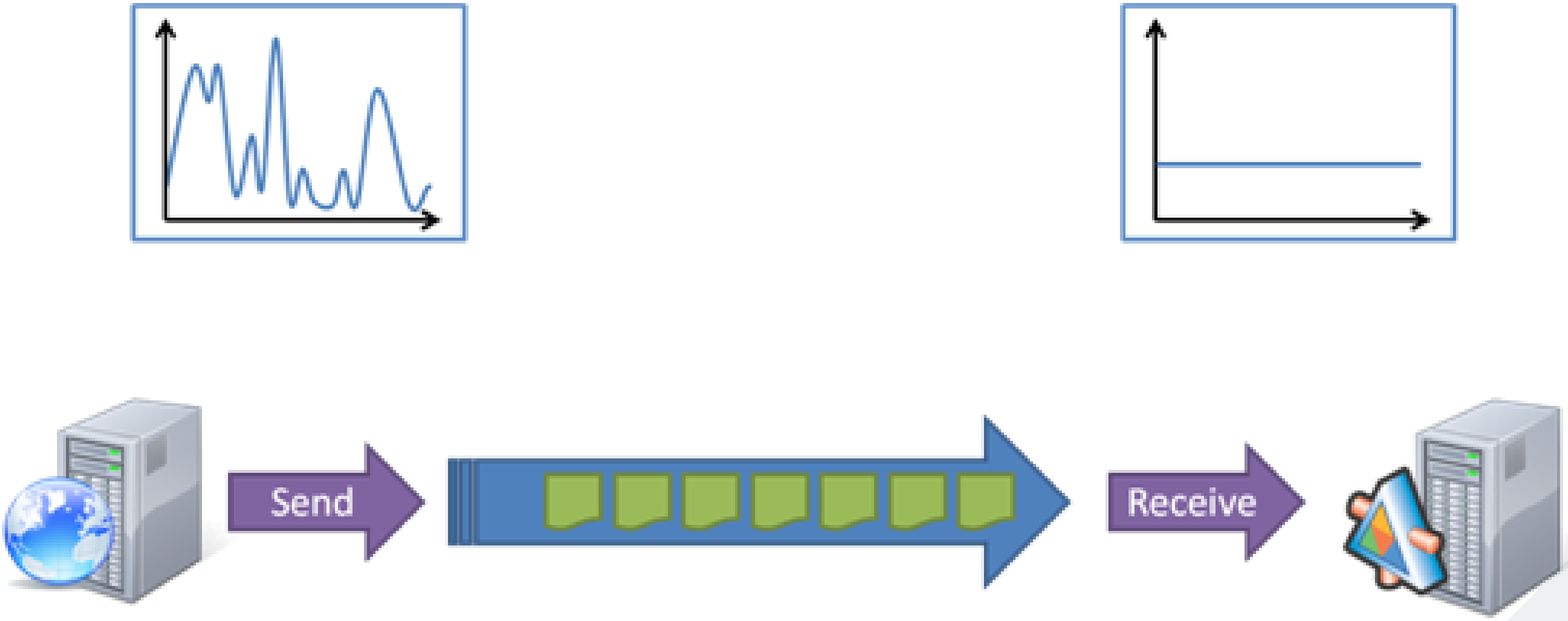
- Secure Network Traversal



http://www.cloudcasts.net/devguide

Service Bus/Queue Scenarios

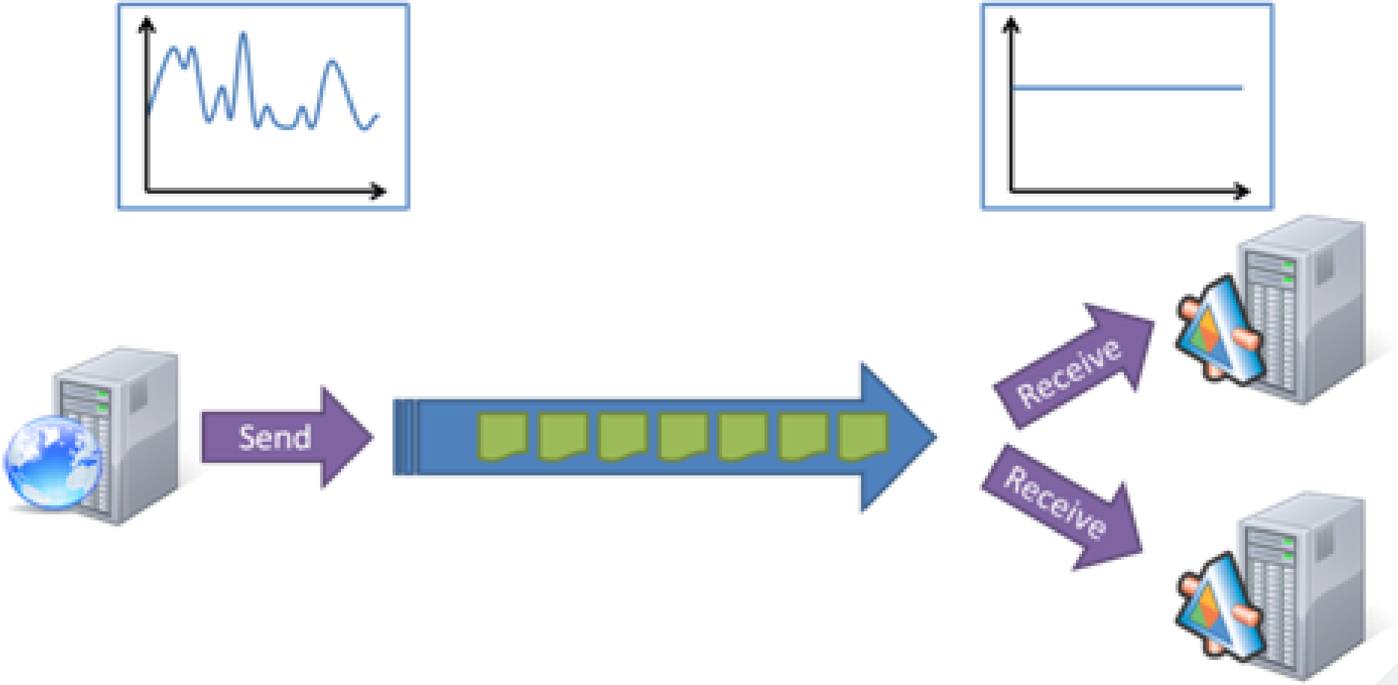
Load Leveling



http://www.cloudcasts.net/devguide

Service Bus/Queue Scenarios

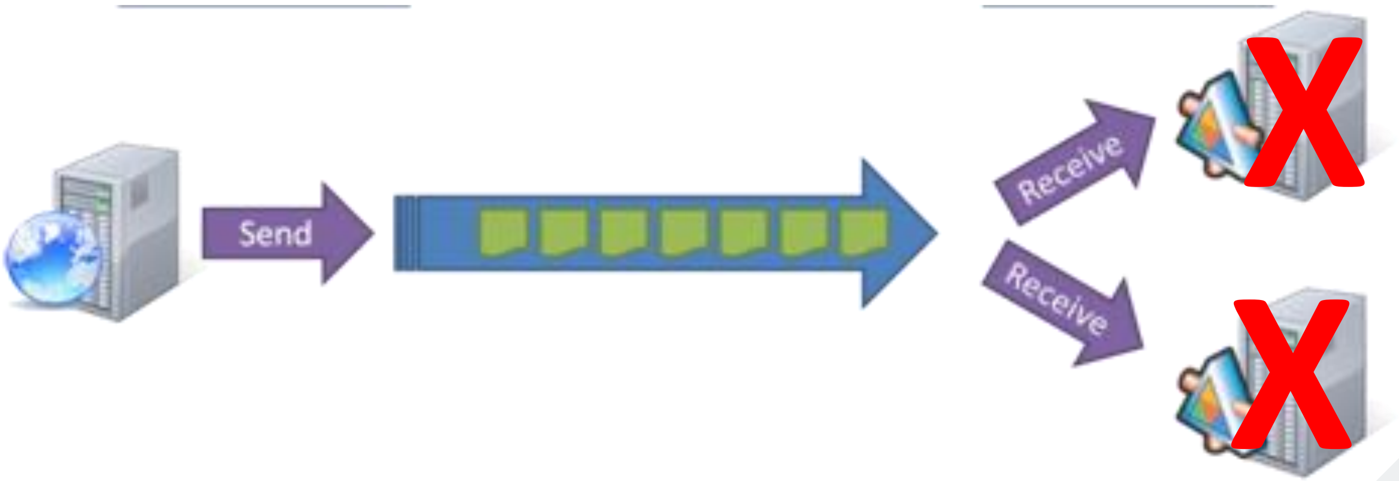
- Load Balancing



http://www.cloudcasts.net/devguide

Service Bus/Queue Scenarios

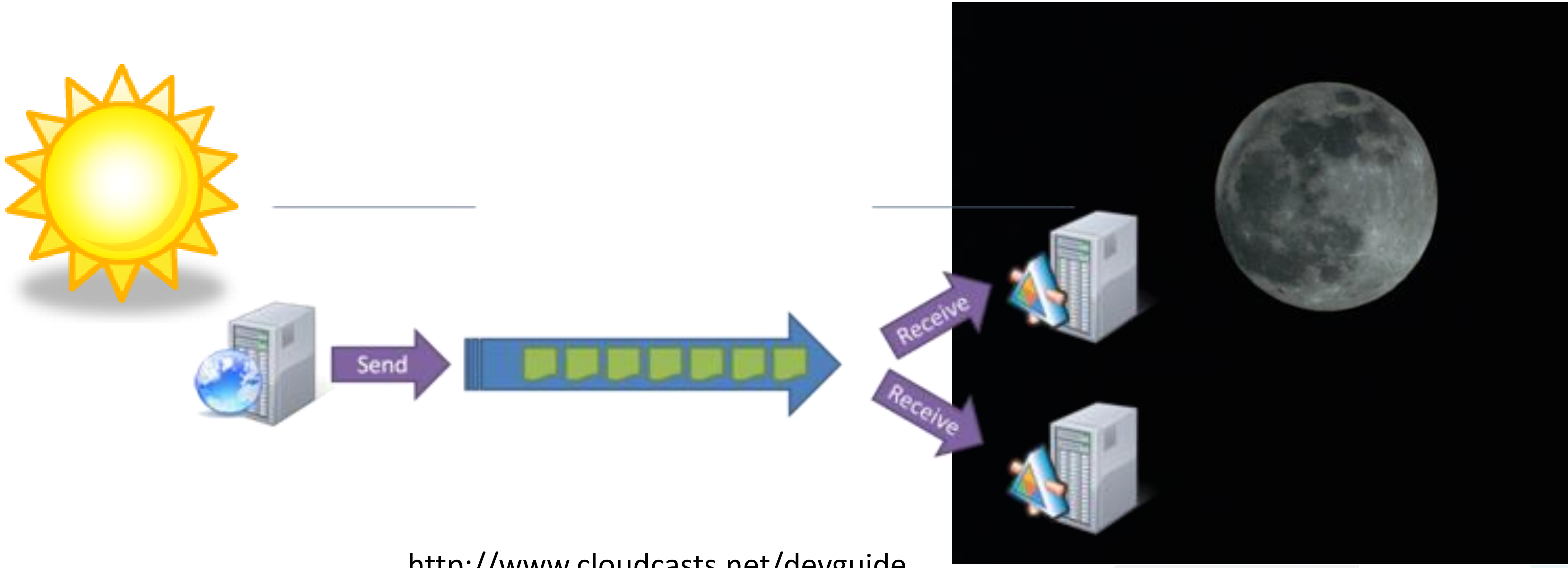
- **Resilience against Service Failure**



http://www.cloudcasts.net/devguide

Service Bus/Queue Scenarios

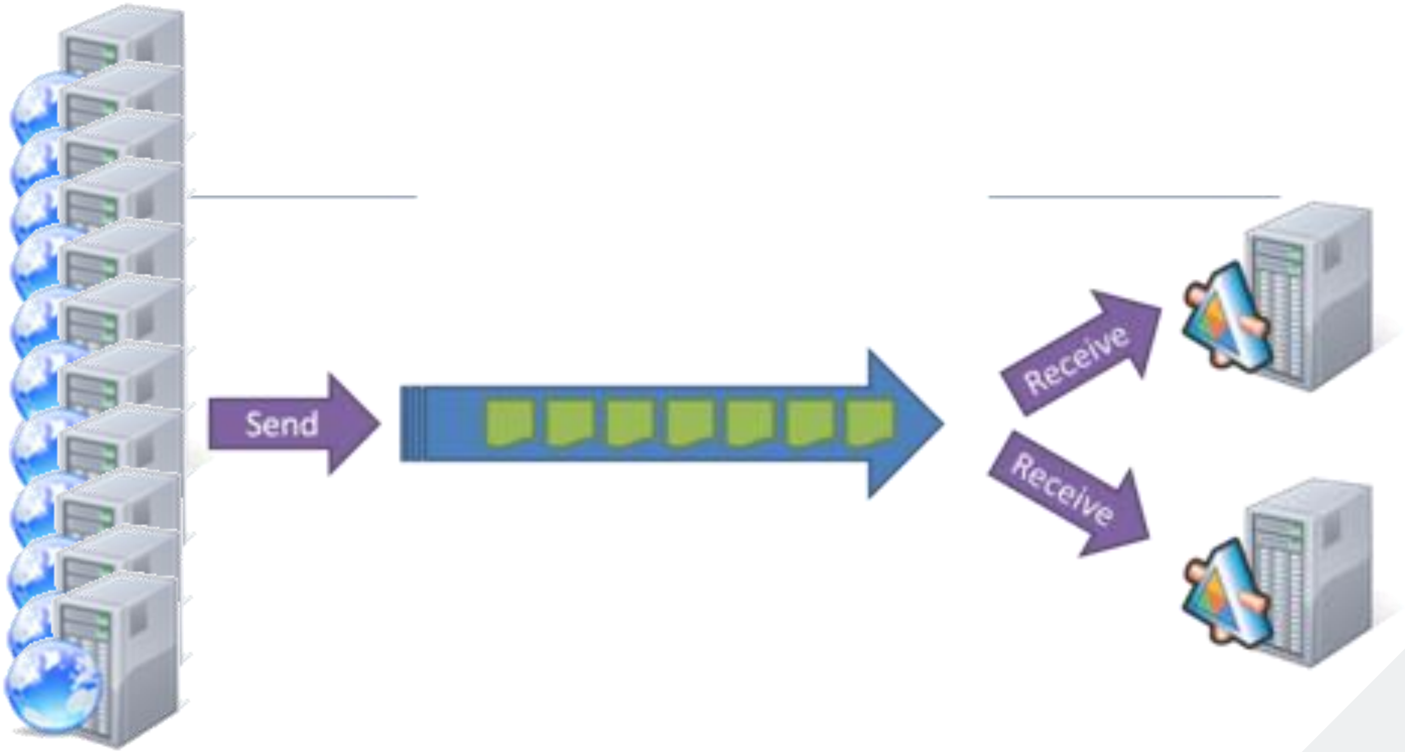
- End of Day Processing



http://www.cloudcasts.net/devguide

Service Bus/Queue Scenarios

- Hyper scale data ingress (Event Hub)

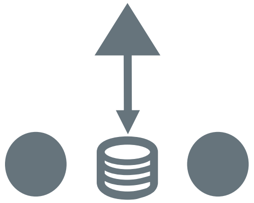


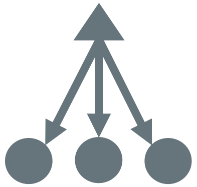
Azure Service Bus

Communication Patterns

Azure Service Bus

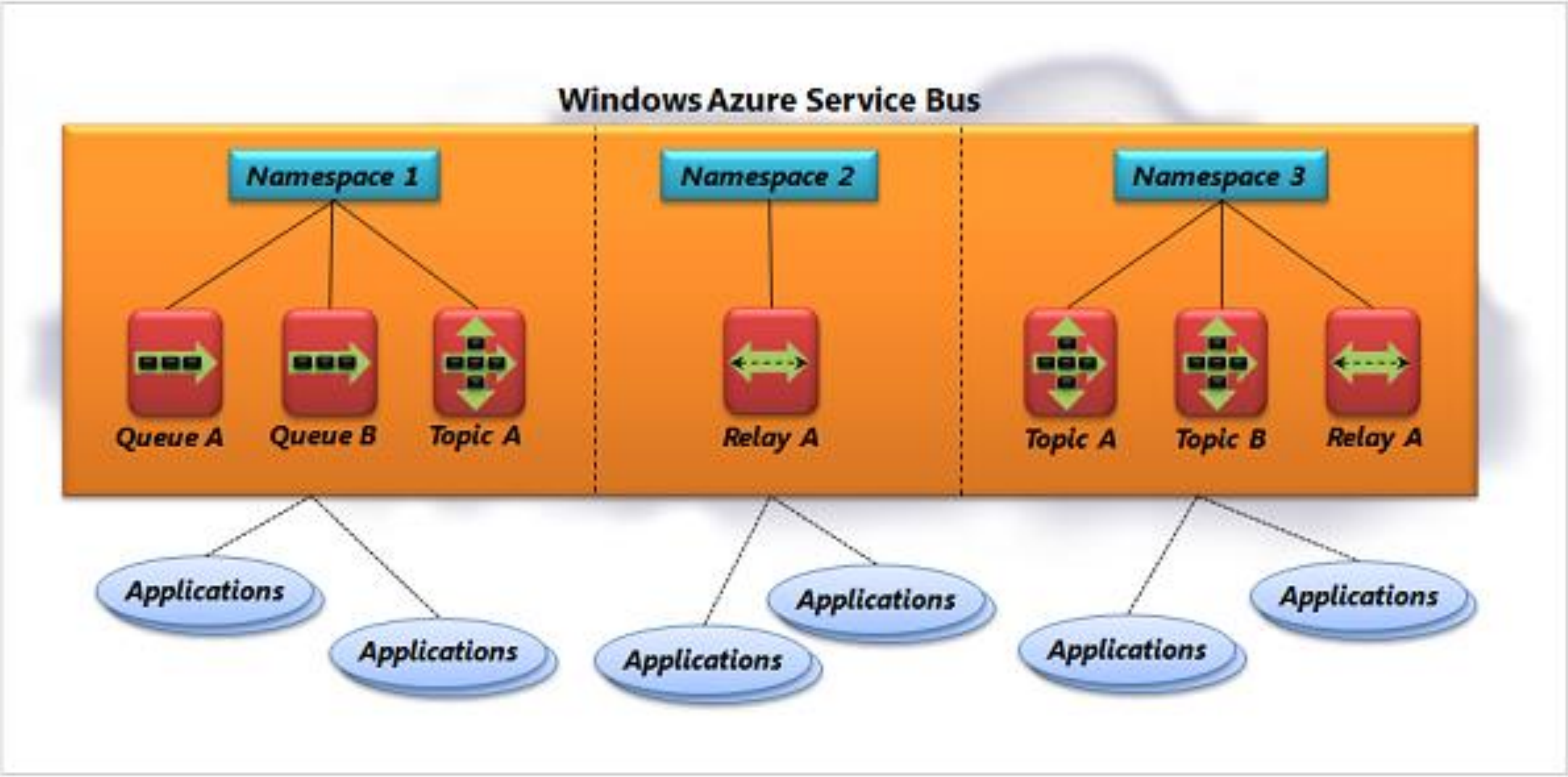
- Messaging
  - Queuing
  - Topics (Pub/Sub)
  - Reliable Transfer
- Connectivity
  - Service Relay
  - Protocol tunneling
- Hyper scale data ingestion
- Notification Hub
  - Scalable Push notifications
  - Multi platform



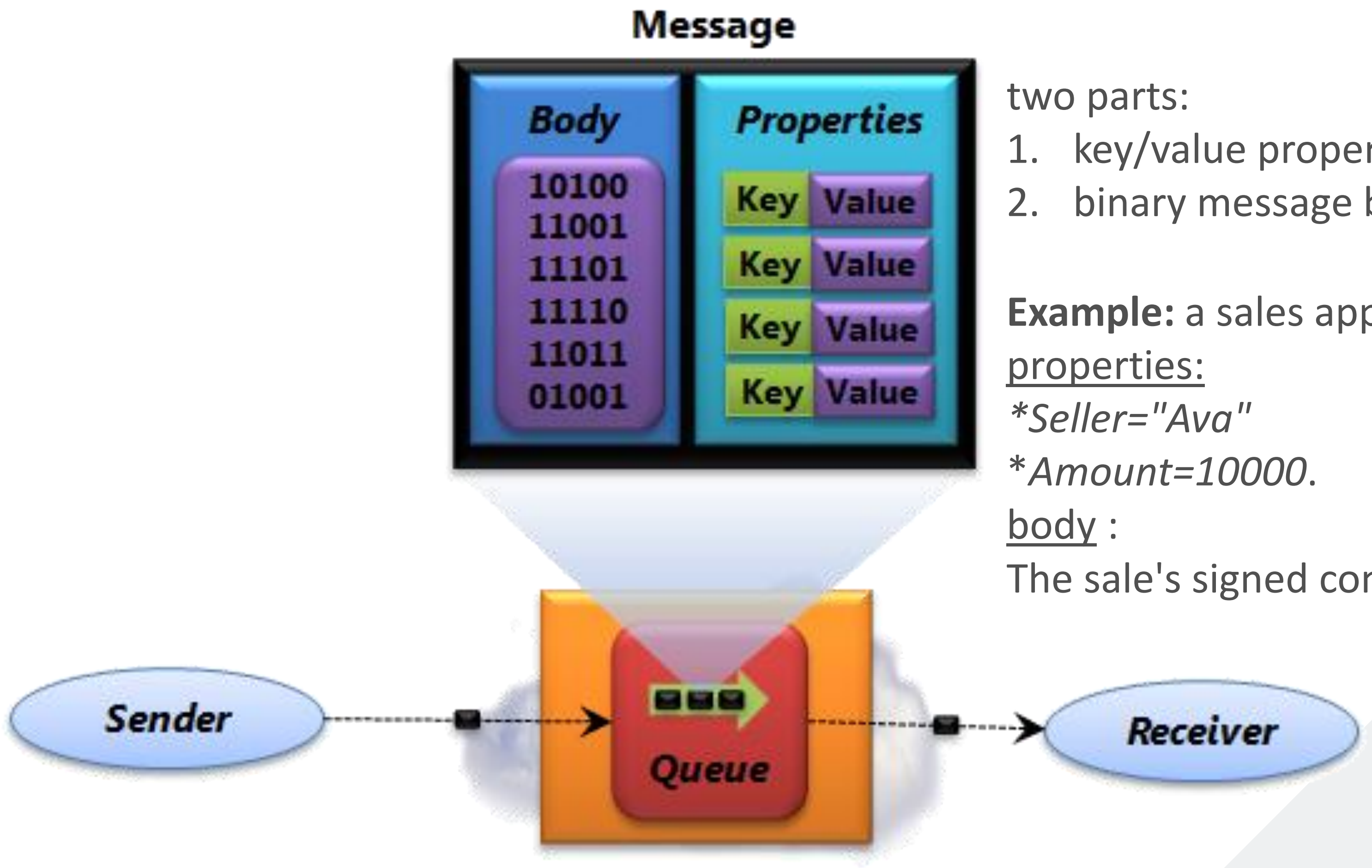


Service Bus Messaging

Service Bus Messaging



Service Bus Messaging - Queue



two parts:

1. key/value properties
2. binary message body.

Example: a sales application sends message

<u>properties:</u>

**Seller="Ava"*

**Amount=10000*.

<u>body</u> :

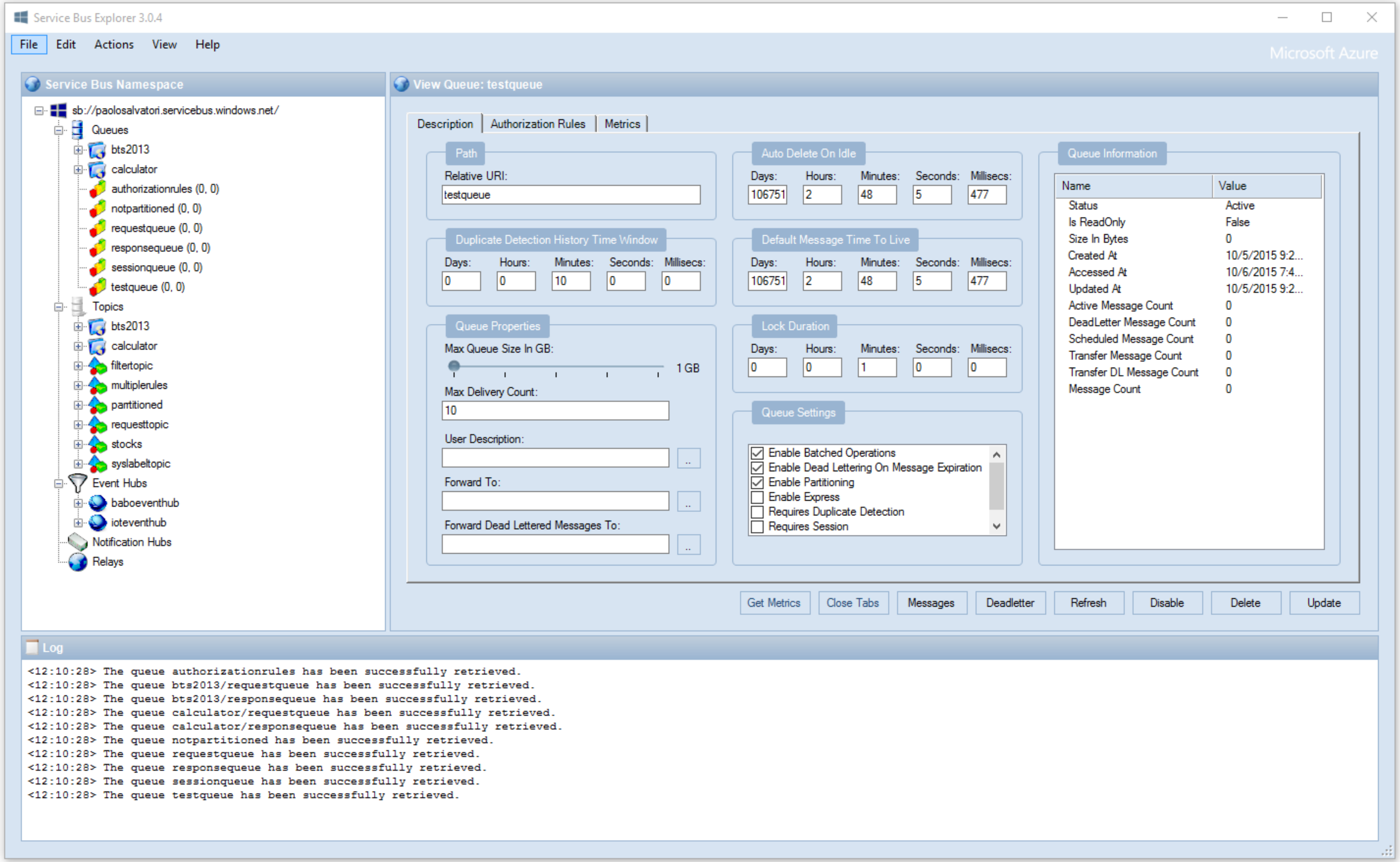
The sale's signed contract scanned image

Service Bus Messaging - Queue

```
Uri managementUri =
       ServiceBusEnvironment.CreateServiceUri("sb", "ServiceBusNamespace", string.Empty);
var sharedSecretTokenProvider =
   TokenProvider.CreateSharedSecretTokenProvider(  "[ServiceBusIssuerName]",
                                                   "[ServiceBusIssuerKey]");
var namespaceManager = new NamespaceManager(managementUri, sharedSecretTokenProvider);

var queuDescription=namespaceManager.CreateQueue("[QueuePath]");
var queueClient = QueueClient.Create(queuDescription.Path);

var someSerializableObject = new SomeSerializableType();
var brokeredMessageToSend = new BrokeredMessage(someSerializableObject);
brokeredMessageToSend.Properties["key"] = "val";
queueClient.Send(brokeredMessageToSend);

var recievedBrokerdMessage = queueClient.Receive();
var someSerializableType = recievedBrokerdMessage.GetBody<SomeSerializableType>();
var peekedBrokeredMessage = queueClient.Peek();
```

Service Bus Explorer

- https://github.com/paolosalvatori/ServiceBusExplorer



Path

Relative URI: testqueue

Auto Delete On Idle

Days:	Hours:	Minutes:	Seconds:	Millisecs:
106751	2	48	5	477

Duplicate Detection History Time Window

Days:	Hours:	Minutes:	Seconds:	Millisecs:
0	0	10	0	0

Default Message Time To Live

Days:	Hours:	Minutes:	Seconds:	Millisecs:
106751	2	48	5	477

Queue Properties

Max Queue Size In GB: 1 GB

Max Delivery Count: 10

User Description:

Forward To:

Forward Dead Lettered Messages To:

Lock Duration

Days:	Hours:	Minutes:	Seconds:	Millisecs:
0	0	1	0	0

Queue Settings

- ☑ Enable Batched Operations
- ☑ Enable Dead Lettering On Message Expiration
- ☑ Enable Partitioning
- □ Enable Express
- □ Requires Duplicate Detection
- □ Requires Session

Queue Information

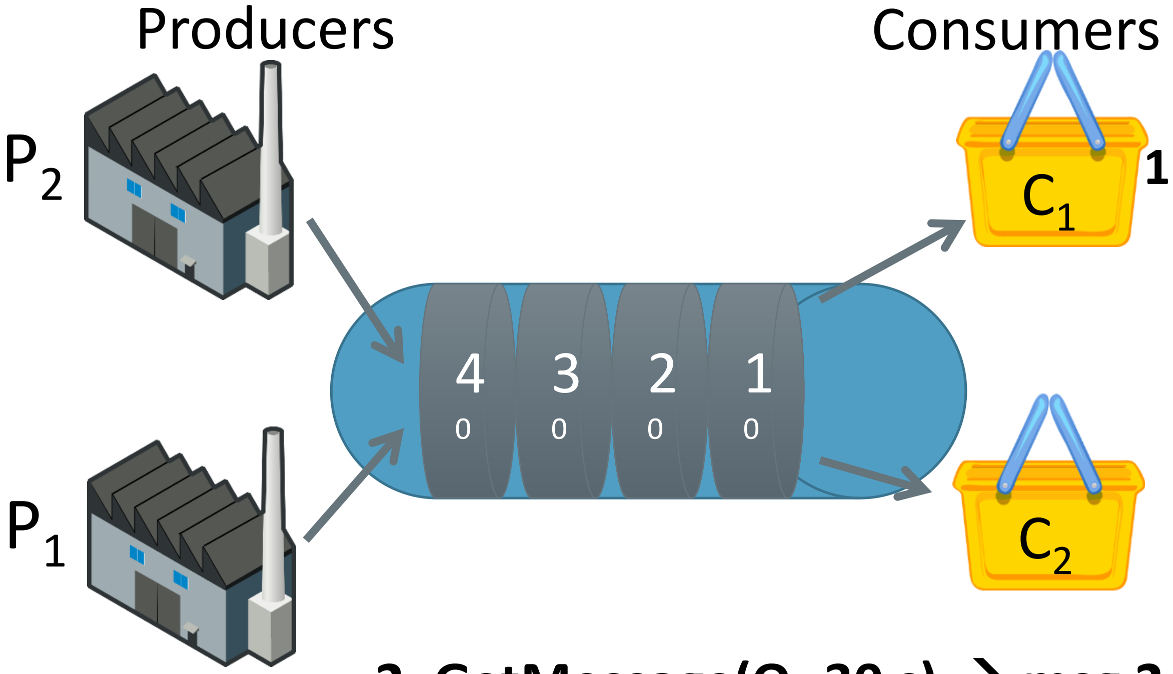
Name	Value
Status	Active
Is ReadOnly	False
Size In Bytes	0
Created At	10/5/2015 9:2...
Accessed At	10/6/2015 7:4...
Updated At	10/5/2015 9:2...
Active Message Count	0
DeadLetter Message Count	0
Scheduled Message Count	0
Transfer Message Count	0
Transfer DL Message Count	0
Message Count	0

```
<12:10:28> The queue authorizationrules has been successfully retrieved.
<12:10:28> The queue bts2013/requestqueue has been successfully retrieved.
<12:10:28> The queue bts2013/responsequeue has been successfully retrieved.
<12:10:28> The queue calculator/requestqueue has been successfully retrieved.
<12:10:28> The queue calculator/responsequeue has been successfully retrieved.
<12:10:28> The queue notpartitioned has been successfully retrieved.
<12:10:28> The queue requestqueue has been successfully retrieved.
<12:10:28> The queue responsequeue has been successfully retrieved.
<12:10:28> The queue sessionqueue has been successfully retrieved.
<12:10:28> The queue testqueue has been successfully retrieved.
```

Service Bus Messaging – Queue – Event Driven

```
var eventDrivenMessagingOptions = new OnMessageOptions
{
    AutoComplete = true,
    MaxConcurrentCalls = 5
};
eventDrivenMessagingOptions.ExceptionReceived += OnExceptionReceived;
queueClient.OnMessage(OnMessageArrived, eventDrivenMessagingOptions);

private void OnMessageArrived(BrokeredMessage obj)
}
    //do something
{

private void OnExceptionReceived(object sender, ExceptionReceivedEventArgs e)
}
   //do something
{
```

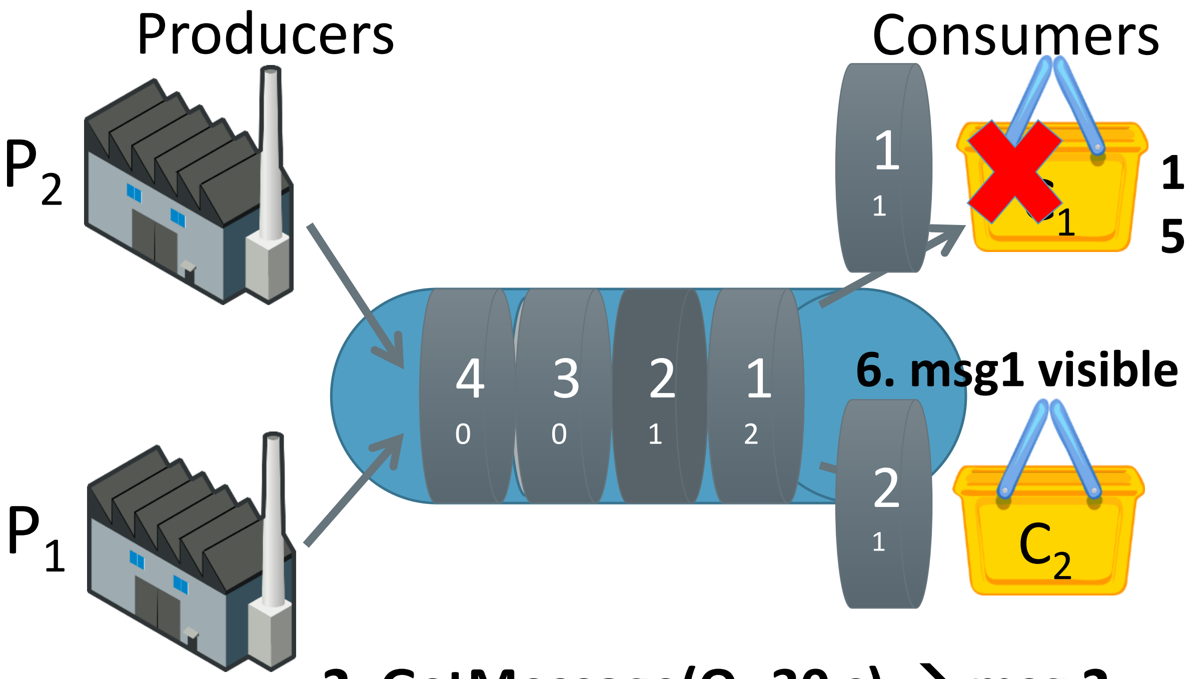
Removing Poison Messages



1. GetMessage(Q, 30 s) → msg 1

2. GetMessage(Q, 30 s) → msg 2

Removing Poison Messages

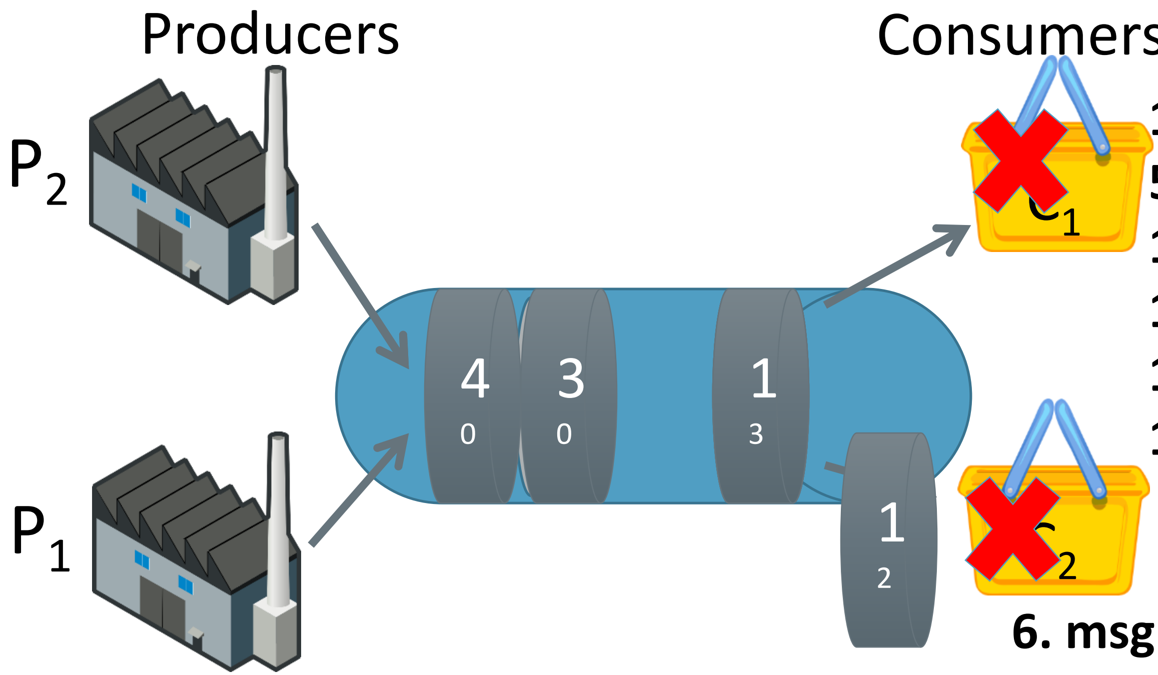


1. GetMessage(Q, 30 s) → msg 1
5. C_1 crashed

6. msg1 visible 30 s after Dequeue

2. GetMessage(Q, 30 s) → msg 2
3. C2 consumed msg 2
4. DeleteMessage(Q, msg 2)
7. GetMessage(Q, 30 s) → msg 1

Removing Poison Messages



1. Dequeue(Q, 30 sec) → msg 1
5. C_1 crashed
10. C1 restarted
11. Dequeue(Q, 30 sec) → msg 1
12. DequeueCount > 2
13. Delete (Q, msg1)

6. msg1 visible 30s after Dequeue
9. msg1 visible 30s after Dequeue

2. Dequeue(Q, 30 sec) → msg 2
3. C2 consumed msg 2
4. Delete(Q, msg 2)
7. Dequeue(Q, 30 sec) → msg 1
8. C2 crashed

Poison Messages

- Message can cause the consumer to crash
- Detecting “Poison Messages”
  - For Storage Queues - examine the DequeueCount property of the message.
    - Two options
      1. Delete the message
      2. Store in Poison Queue/Table
  1. Azure Service bus
     1. automatically done by setting the QueueDescription.MaxDeliveryCount and SubscriptionDescription.MaxDeliveryCount properties
     2. Explicitily calling the **DeadLetter()** method

Service Bus Messaging – Dead letters

- If a receiver finishes processing a message successfully then it should call **Completed** method
- If a receiver application is unable to process the message for some reason, then it can call the **Abandon** method
    - This will cause the Service Bus to unlock the message within the queue and make it available to be received again
- If the application crashes after processing the message but before the **Complete** request is issued, then the message will be redelivered to the application when it restarts
- We can mark failing message as dead-letter using the **DeadLetter()** method

Service Bus Messaging – Dead letters

- When we mark the message as dead letter. The message is moved to dead letter queue.
- the name of the sub-queue is [queueName]/$DeadLetterQueue
- The path can be obtained using the **FormatDeadLetterPath** method of the QueueClient
- This sub-queue can be consumed by any other system or consumer and check the messages, log them and so on.
- It is not possible to add the message to the original queue.

ISO 9001 SGS

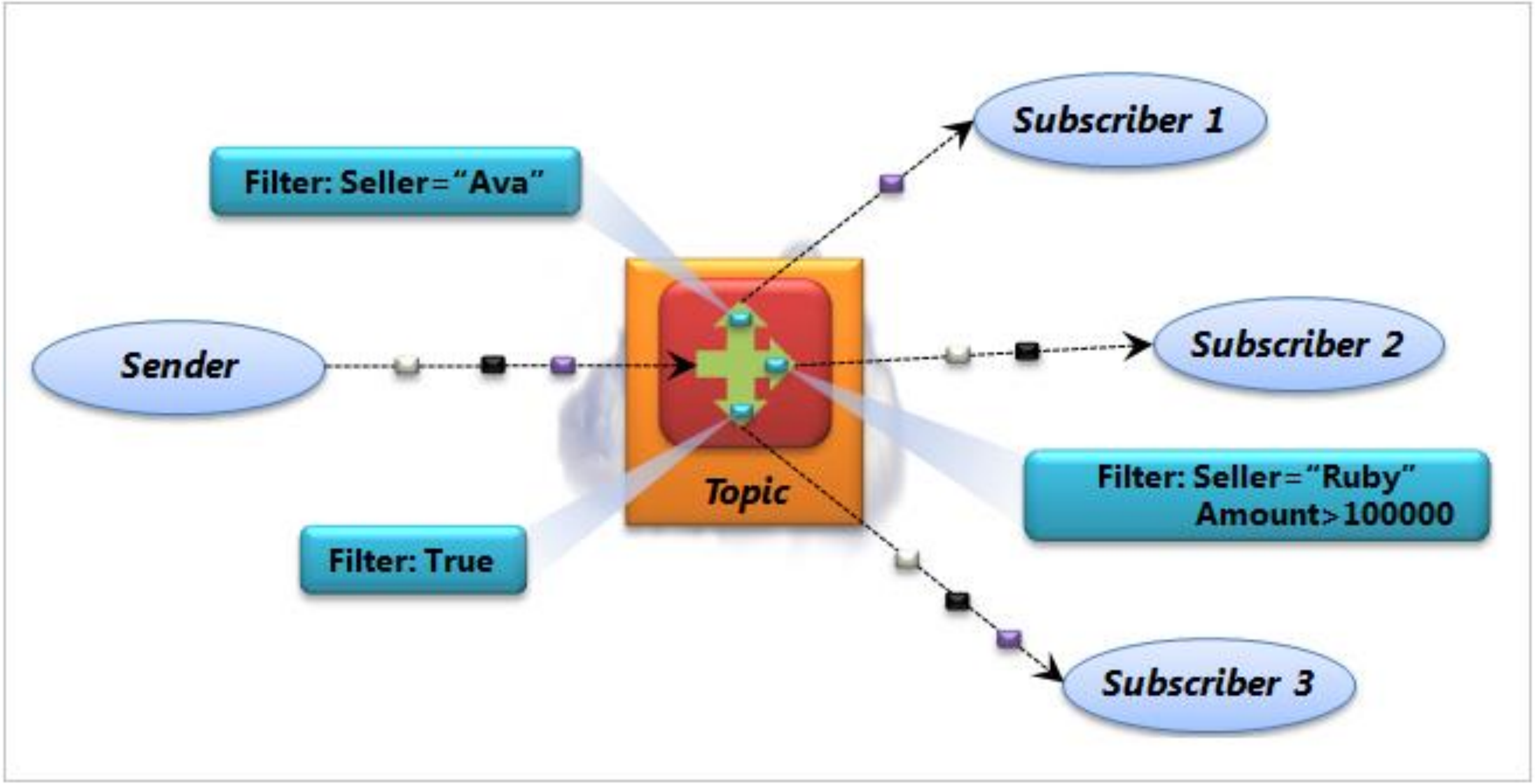
Service Bus Messaging – Dead letters

```
BrokeredMessage receivedMessage;
while ((receivedMessage = queueClient.Receive(TimeSpan.FromSeconds(10))) != null)
{
    int retryCount = 0;
    while (retryCount < MaxRetryCount)
    {
        if (ProcessOrder(receivedMessage))
            break;
        else
            retryCount++;
    }

    if (retryCount == MaxRetryCount)
    {
        receivedMessage.DeadLetter(  "UnableToProcess",
                                     "Failed to process in reasonable attempts");
    }
}
```

Service Bus Messaging – Dead letters

```
// Log the dead-lettered messages that could not be processed:
var deadQueuePath=
       QueueClient.Create(QueueClient.FormatDeadLetterPath(queueClient.Path)

QueueClient deadLetterClnt =
       QueueClient.Create(deadQueuePath, ReceiveMode.ReceiveAndDelete);

BrokeredMessage receivedDeadLetterMessage;
while ((receivedDeadLetterMessage = deadLetterClnt.Receive(TimeSpan.FromSeconds(10))) != null)
{
    LogOrder(receivedDeadLetterMessage);
}
```

Service Bus Messaging – Topics



Service Bus Messaging – Topics

- A topic is similar in many ways to a queue.
- topics let each receiving application create its own subscription by defining a *filter*.
- A subscriber will then see only the messages that match that filter.
- Unlike queues, however, a single message sent to a topic can be received by multiple subscribers.
- ***publish and subscribe***

Service Bus Messaging – Topics

- Up to 2000 rules per topic
- Each matched rule yield a message copy
- Types:
  - SqlFilter - SQL92 expressions over message properties
  - CorrelationFilter
  - FlaseFilter
  - TrueFilter
- Filter can help for
  - Routing based on content
  - Partitioning aware message distribution – without sender involvement

Service Bus Messaging – Topics

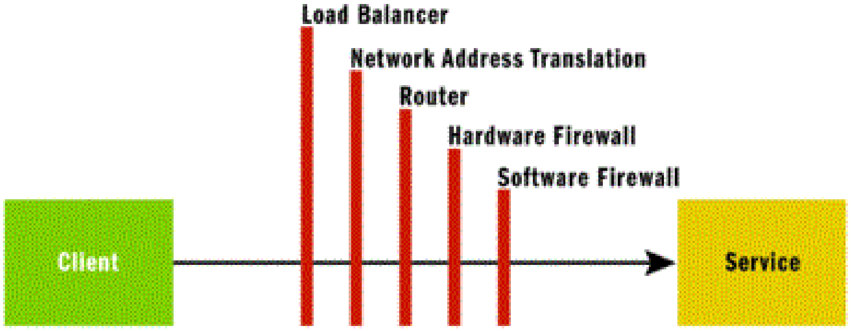
```
TopicDescription myTopic = namespaceManager.CreateTopic("DataCollectionTopic");
Uri serviceUri = ServiceBusEnvironment.CreateServiceUri("sb", "ServiceBusNamespace",
string.Empty);
var tokenProvider = TokenProvider.CreateSharedSecretTokenProvider("[ServiceBusIssuerName]",
                                                                  "[ServiceBusIssuerKey]");
MessagingFactory factory = MessagingFactory.Create(serviceUri, tokenProvider);
TopicClient myTopicClient = factory.CreateTopicClient(myTopic.Path);

var brokeredMessage = new BrokeredMessage(someSerializableObject);
brokeredMessage.Properties["MessageCategory"] = "Inventory";
myTopicClient.Send(brokeredMessage);
```

Service Bus Messaging – Topics

```
SqlFilter inventoryFilter =
   new SqlFilter("MessageCategory = Inventory");//MessageCategory is a property in the message
SubscriptionDescription myAgentSubscription =
    namespaceManager.CreateSubscription(myTopic.Path, "Inventory");

var subscriptionClient = SubscriptionClient.Create(myTopic.Path, "Inventory“, inventoryFilter);
BrokeredMessage message = subscriptionClient.Receive();
```

Service Bus– Relay

- IPv4 is running out
  - Dynamic DNS
  - Network Address Translation (NAT)
- Load Balancers
- Routers
- Hardware Firewall
- Software Firewall

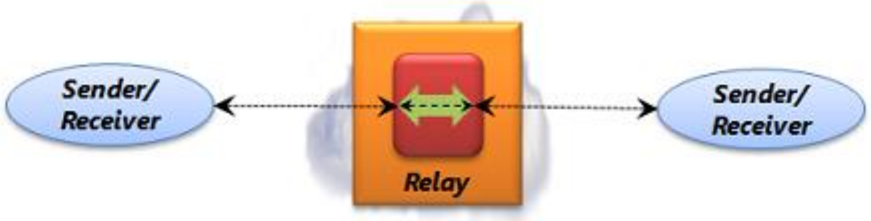


* From MSDN Magazine

- Web Services simply don’t work
- What about calling back to the client?

Service Bus – Relay

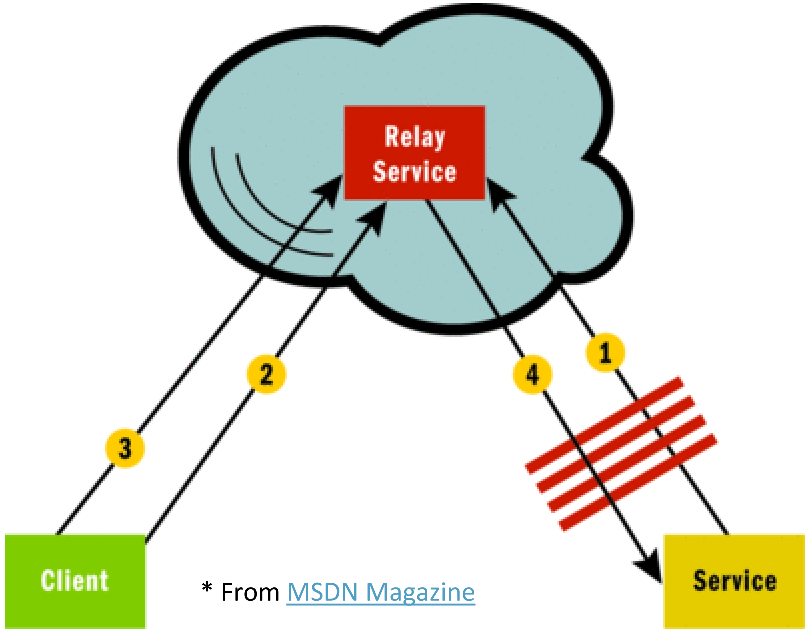
- Intermediary Pass-Through Service (Push)
  - Overcome connectivity challenges
- Uses outbound connections only
  - Defaults to TCP with fallback to HTTP
- Relays client calls to service
- Hosted in the cloud
  - Scalability
  - Security
  - Management Portal



* From MSDN

Connectivity Relay

- Service connects and authenticates against the relay
  - Relay determines how to communicate with the service
- Client authenticates and calls the service
  - Relay forwards the message to the service



* From MSDN Magazine

Service Address and Registry

- Service Bus Address
  - **[scheme]**:**[ns]**.servicebus.windows.net/{uri}
  - **sb**://**MyCompany**.servicebus.windows.net/CalcService
- Service Bus Registry
  - ATOM-based feed of online services
  - http://MyCompany.servicebus.windows.net/
- Need to enable publishing to registry
  - Add ServiceRegsitrySettings endpoint behavior configured with public discovery type

Summary

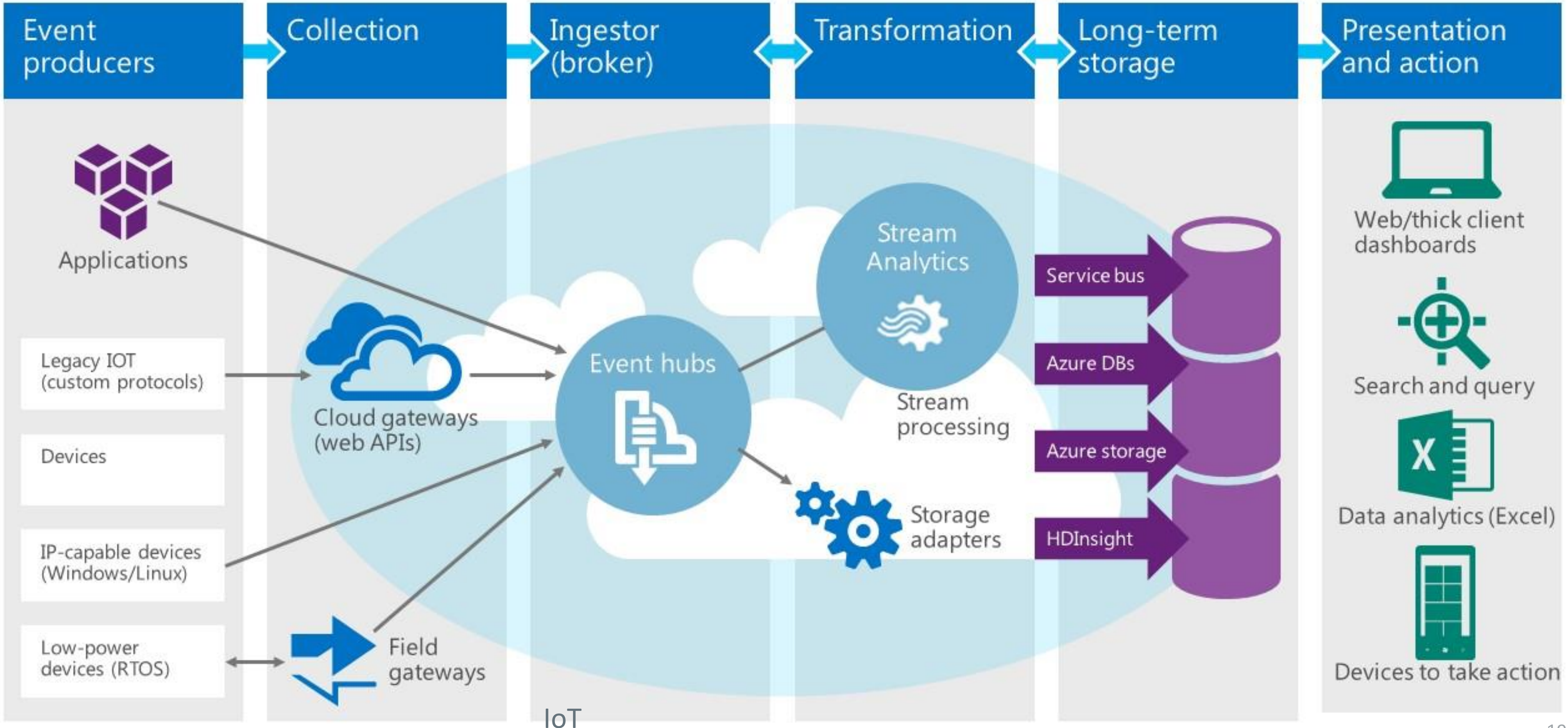
- Azure Service Bus is a key component for many connectivity scenarios
- Highly flexible and robust messaging & connectivity solution
  - Skype in a box
- Can be installed on premise
  - Service Bus for Windows Server
- PaaS – no maintenance or setup issues
  - Simply configure and use

Event Hubs

Event Hubs

- Highly scalable data ingress service
- Can ingest millions of events per second
- Act as the "front door" for an event pipeline
  - Once data is collected into an Event Hub, it can be transformed and stored using any real-time analytics provider or batching/storage adapters.
- Decouples the production of a stream of events from the consumption of those events
  - Event consumers can access the events on their own schedule
- Different from traditional queues
  - Journal Logging
  - Similar to Apache Kafka

Azure Based High-Throughput Ingest Architecture



IoT

Event Hubs

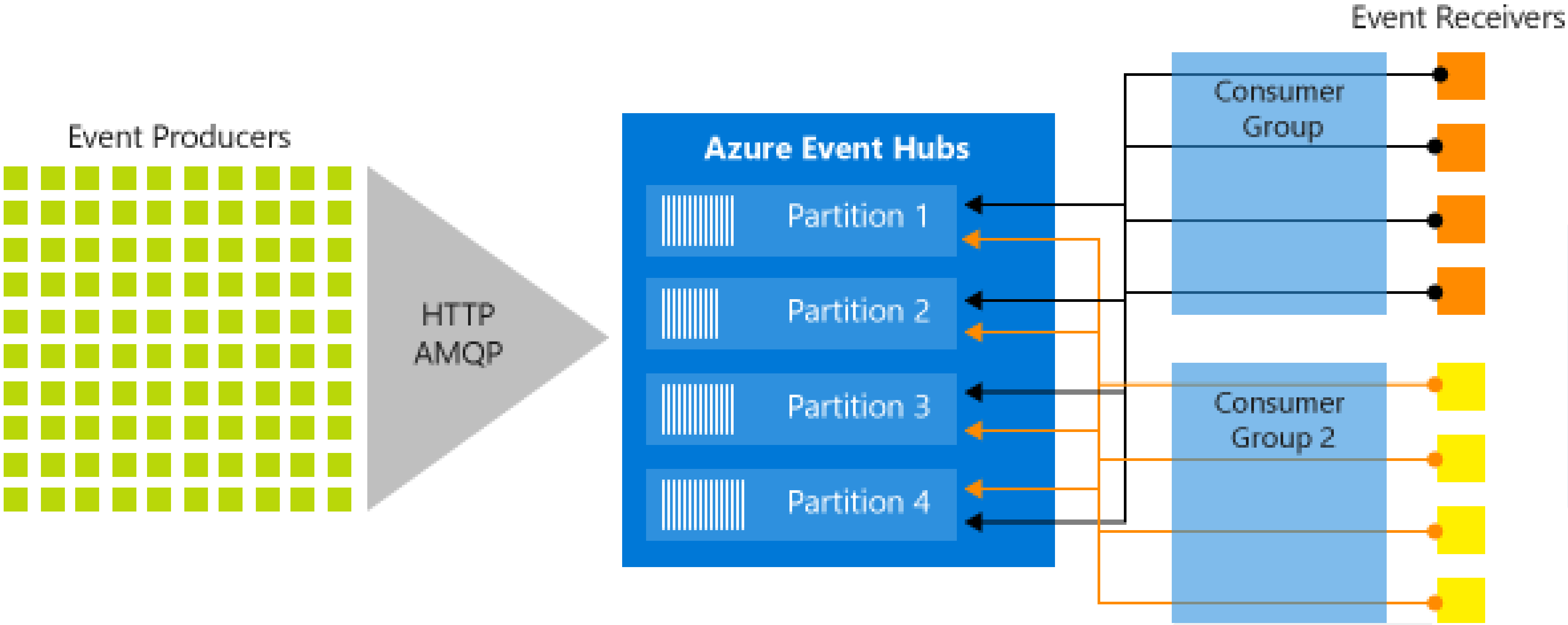
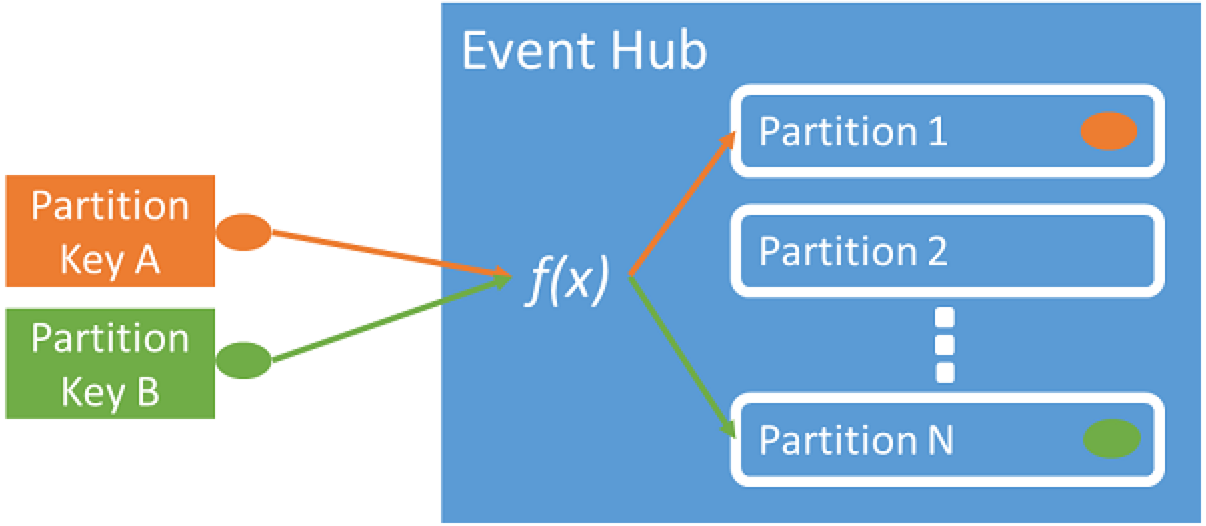
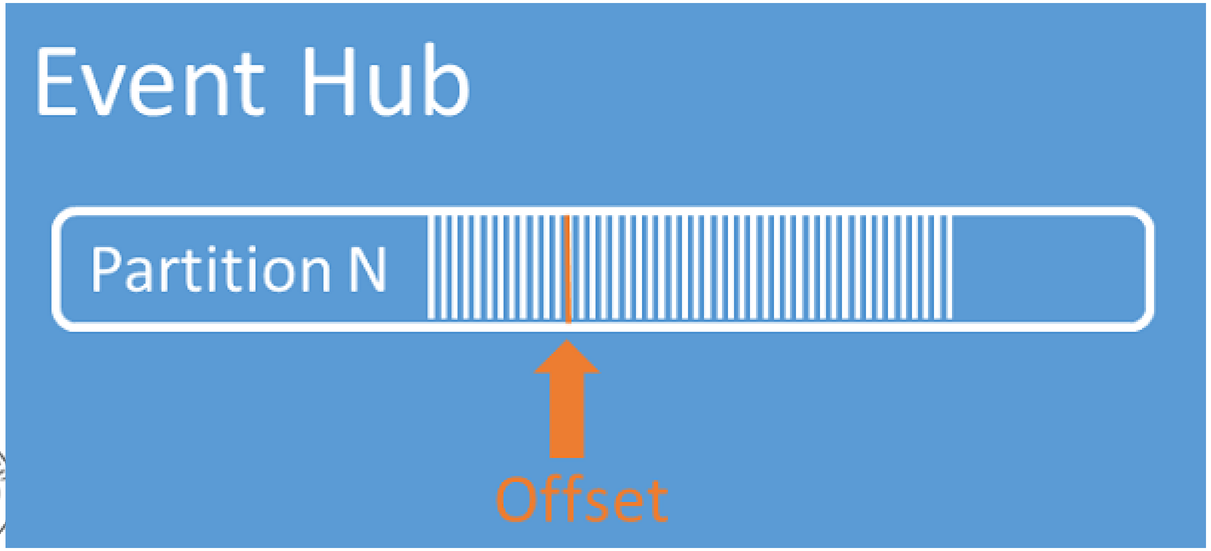


Image from https://docs.microsoft.com/en-us/azure/event-hubs/event-hubs-features

Event Hub Partition





Basic Programming model

Creating

```
var manager = new Microsoft.ServiceBus.NamespaceManager("mynamespace.servicebus.windows.net");
var description = manager.CreateEventHub("MyEventHub");
var client = EventHubClient.Create(description.Path);
```

Sending

```
var partitionedSender = client.CreatePartitionedSender();
var partitionedSender = client.CreatePartitionedSender(description.PartitionIds[0]);
```

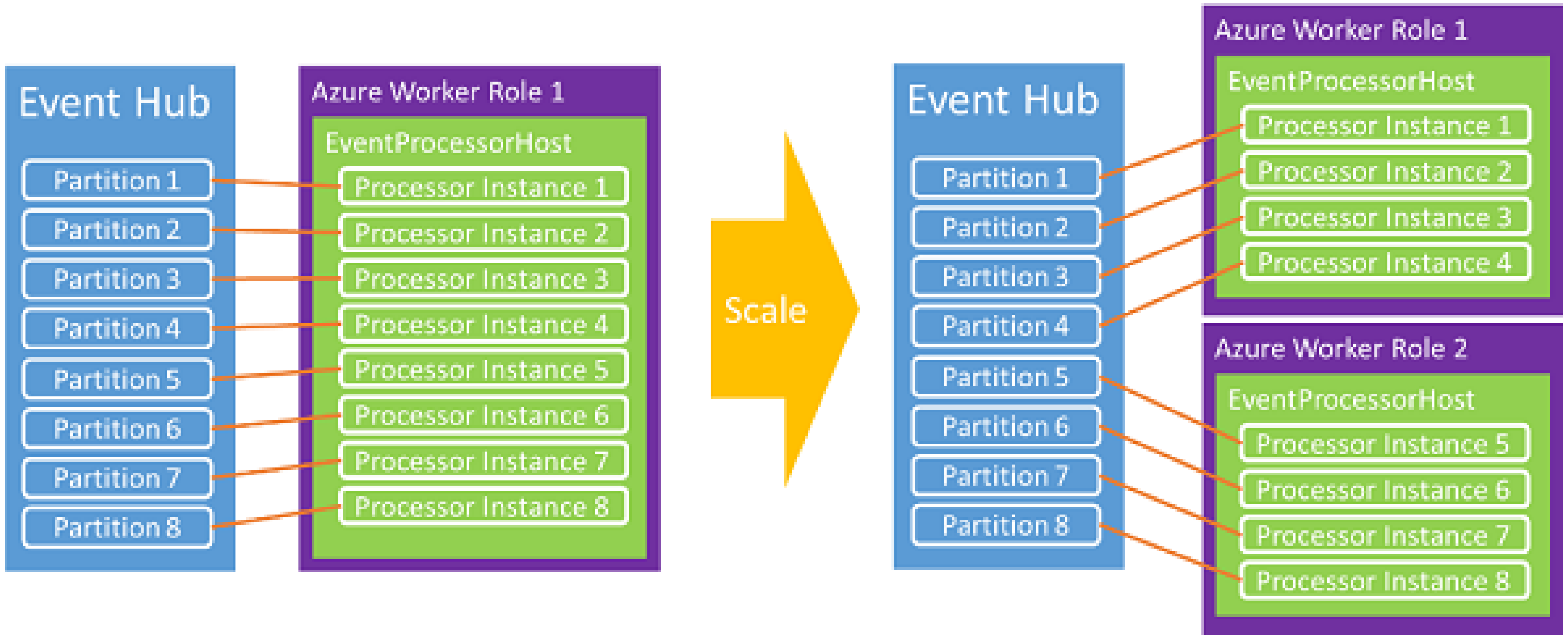
Receving

```
EventHubConsumerGroup group = client.GetDefaultConsumerGroup();
var receiver = group.CreateReceiver(client.GetRuntimeInformation().PartitionIds[0]);
```

Event Processor Host

- EventProcessorHost provides a thread-safe, multi-process, safe runtime environment for event processor with checkpointing and partition lease
- Reside in `Microsoft Azure Service Bus Event Hub – EventProcessorHost` nuget package
- How it works:
  - Implement IEventProcessor with the logic of your message-processing:
  - Use EventProcessorHost .RegisterEventProcessorAsync to register IEventProcessor
  - The host will attempt to acquire a lease on every partition in the event hub using a "greedy" algorithm.
  - As new nodes,(worker instances), come online, they place lease reservations and over time the load shifts between nodes as each attempts to acquire more leases.

Event Processor Host



- The EventProcessorHost class also implements an Azure storage-based checkpointing mechanism.
- This mechanism stores the offset on a per partition basis, so that each consumer can determine what the last checkpoint from the previous consumer was.

Event Processor Host - IEventProcessor

```
class SimpleEventProcessor : IEventProcessor
{
    async Task IEventProcessor.CloseAsync(PartitionContext context, CloseReason reason)
    {
        Console.WriteLine("Processor Shutting Down. Partition '{context.Lease.PartitionId}'."
        if (reason == CloseReason.Shutdown){
            await context.CheckpointAsync();
        }
    }
    Task IEventProcessor.OpenAsync(PartitionContext context){
        Console.WriteLine($"SimpleEventProcessor initialized. Partition:'{context.Lease.PartitionId}’”);
        return Task.CompletedTask;
    }
    async Task IEventProcessor.ProcessEventsAsync(PartitionContext context,
                                                  IEnumerable<EventData> messages){
        foreach (EventData eventData in messages) {
            string data = Encoding.UTF8.GetString(eventData.GetBytes());
            Console.WriteLine($"Message received.
                Partition: '{context.Lease.PartitionId}', Data: '{data}'"));
        }
    }
}
```

Event Processor Host – Register IEventProcessor

```
string eventProcessorHostName = Guid.NewGuid().ToString();
EventProcessorHost eventProcessorHost =
       new EventProcessorHost(eventProcessorHostName,
                              eventHubName,
                              EventHubConsumerGroup.DefaultGroupName,
                              eventHubConnectionString,
                              storageConnectionString);
Console.WriteLine("Registering EventProcessor...");
var options = new EventProcessorOptions();
options.ExceptionReceived += (sender, e) => { Console.WriteLine(e.Exception); };
eventProcessorHost.RegisterEventProcessorAsync<SimpleEventProcessor>(options).Wait();

Console.WriteLine("Receiving. Press enter key to stop worker.");
Console.ReadLine();
eventProcessorHost.UnregisterEventProcessorAsync().Wait();
```

Notification Hubs

Push is Transforming Businesses

Broadcast breaking news to millions of customers using their preferences

Send notifications based on account changes or actions

Engage customer to improve your brand, customer satisfaction, and business metrics

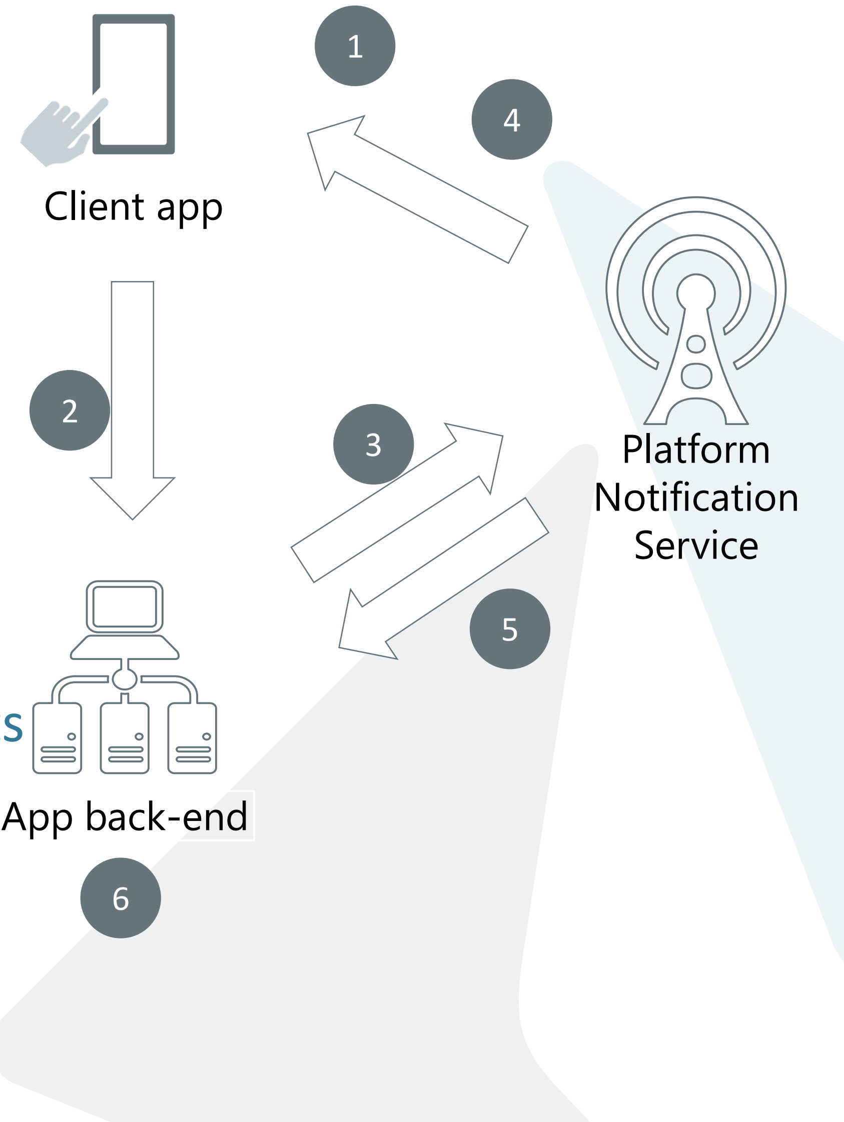
Increase employee productivity and
responsiveness

Push Notification 101

- Register device handle at app launch
  1. Client app retrieves handle from Platform Notification Service (PNS)
  2. Client app sends handle to your custom backend

- Send Notification
  3. **Your backend** connects to PNS and requests push **Your code** has to map between logical users and device handles
  4. PNS pushes notification to device

- Maintain backend device handles
  5. **Your code** must delete expired handles when PNS rejects them
  6. **Your code** must map between logical users and device handles

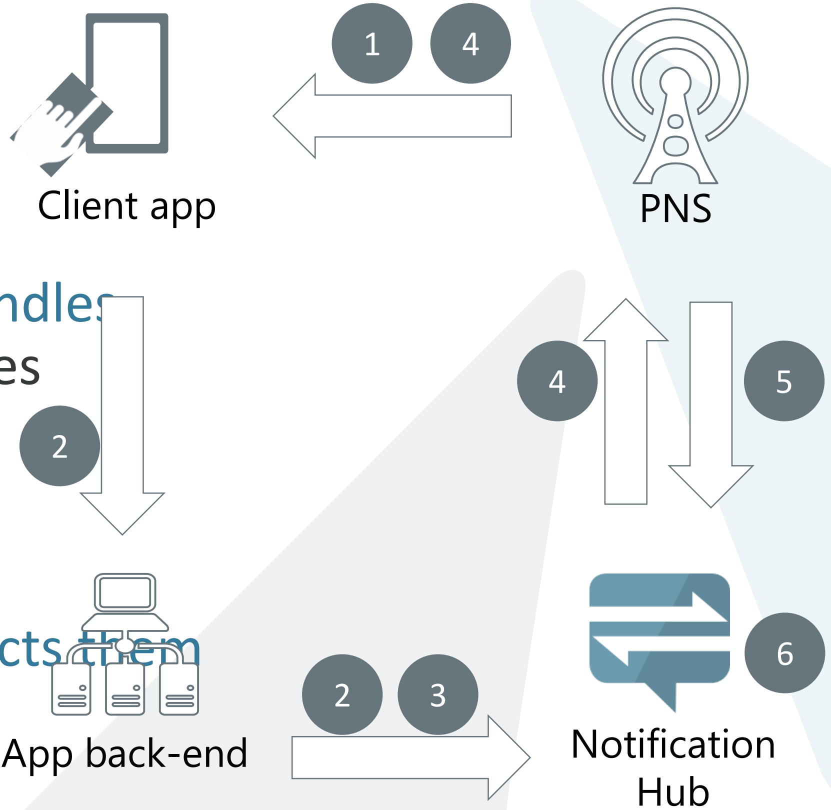


Azure Notification Hub

- Register device handle at app launch
  1. Client app retrieves handle from Platform Notification Service
  2. Client sends handle to your backend
     Backend **registers with Notification Hub** using tags to represent logical users and groups

- Send Notification
  3. Backend sends request to Notification Hub using a tag
     Notification Hub manages scale
     Notification Hub maps logical users/groups to device handles
  4. Notification Hub delivers notifications to matching devices via PNS

- Maintain backend device handles
  5. **Notification Hub** deletes expired handles when PNS rejects them
  6. **Notification Hub** maintains mapping between logical users/groups and device handles



Advantages of Notification Hub

- X-plat: one API to notify on any mobile platform
  - Backend can be on-prem or in the cloud, .NET, Java, PHP, Node, you name it
  - Support iOS, Android, Windows Phone, Windows, Kindle
- Avoid storing device information in your tables
- Work with logical users and segments
- Personalization and localization
  - Templates
- Broadcast at scale, multicast, unicast
- Rich Telemetry

ISO 9001 SGS

Push Notification Patterns

- Notify a single logical user
- Broadcast to a segment based on interest
  - Tag Expressions
- Notify all users
- Notify all users, in batches
- Personalization and localization
  - Template Expressions
  - Geo-Targeting Tags

Event Grid

Modern computing is all about events

Managing events is important but cumbersome

What if all these events could be managed and directed from one place?

Azure Event Grid

Fully-managed
event routing

Near real-time event
delivery at scale

Broad coverage within
Azure and beyond

Backbone of event-driven computing

Benefits

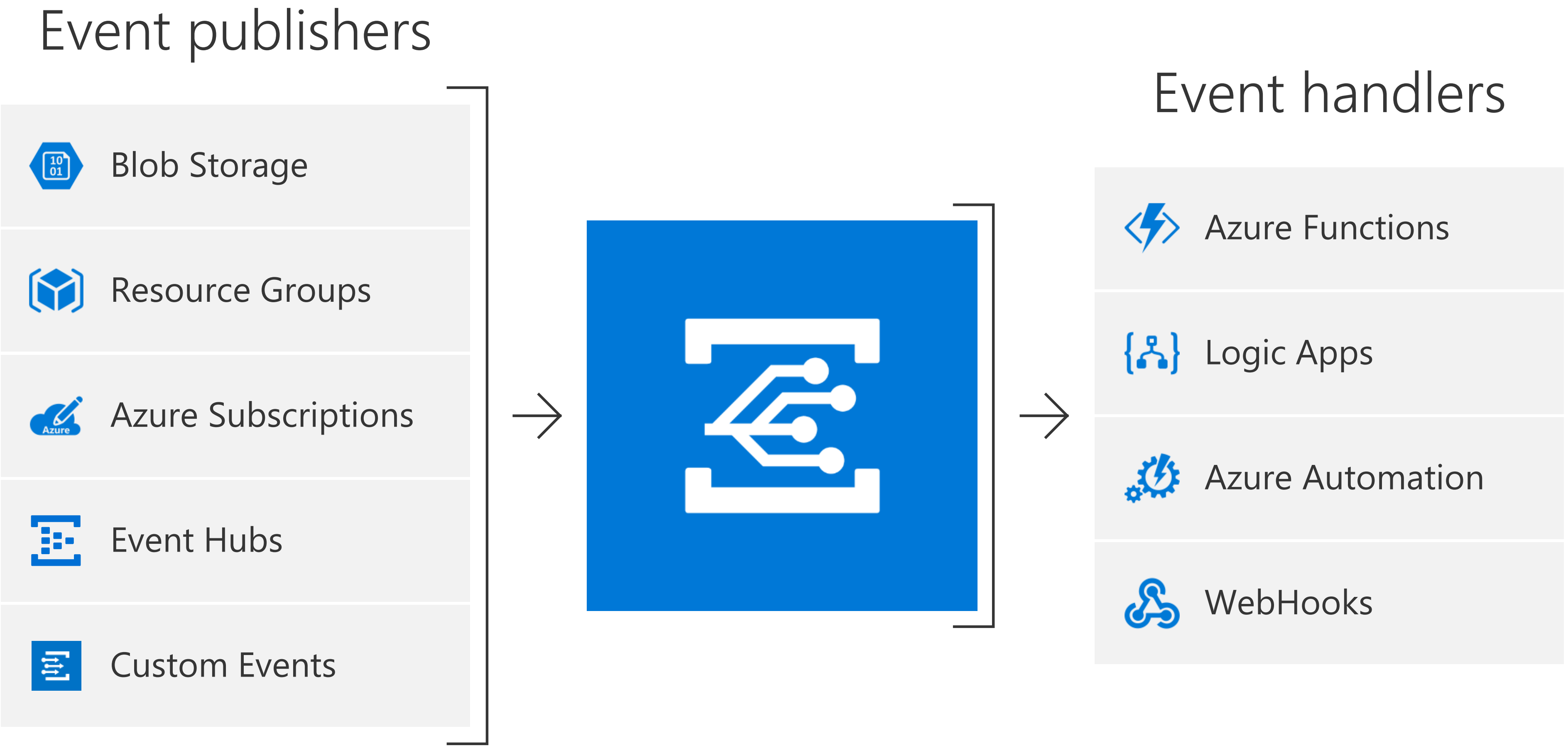
Focus on innovation and pay per event

Ensure reliability and performance for your apps

Unlock new scenarios for your apps

Manage all events in one place

Manage all events in one place



Event publishers

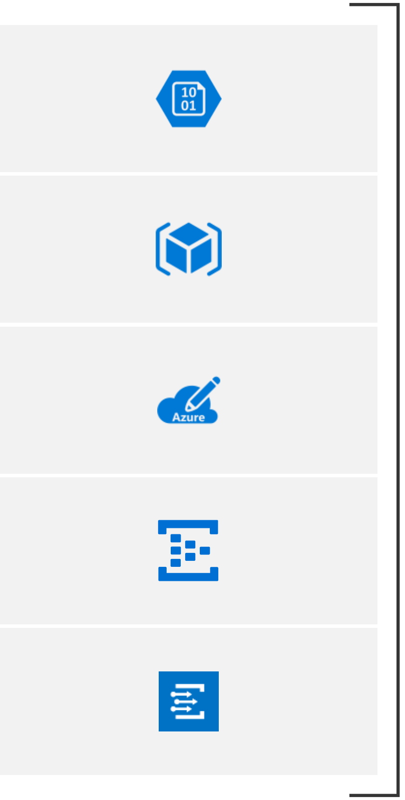
- Blob Storage
- Resource Groups
- Azure Subscriptions
- Event Hubs
- Custom Events

Event handlers

- Azure Functions
- Logic Apps
- Azure Automation
- WebHooks

Manage all events in one place

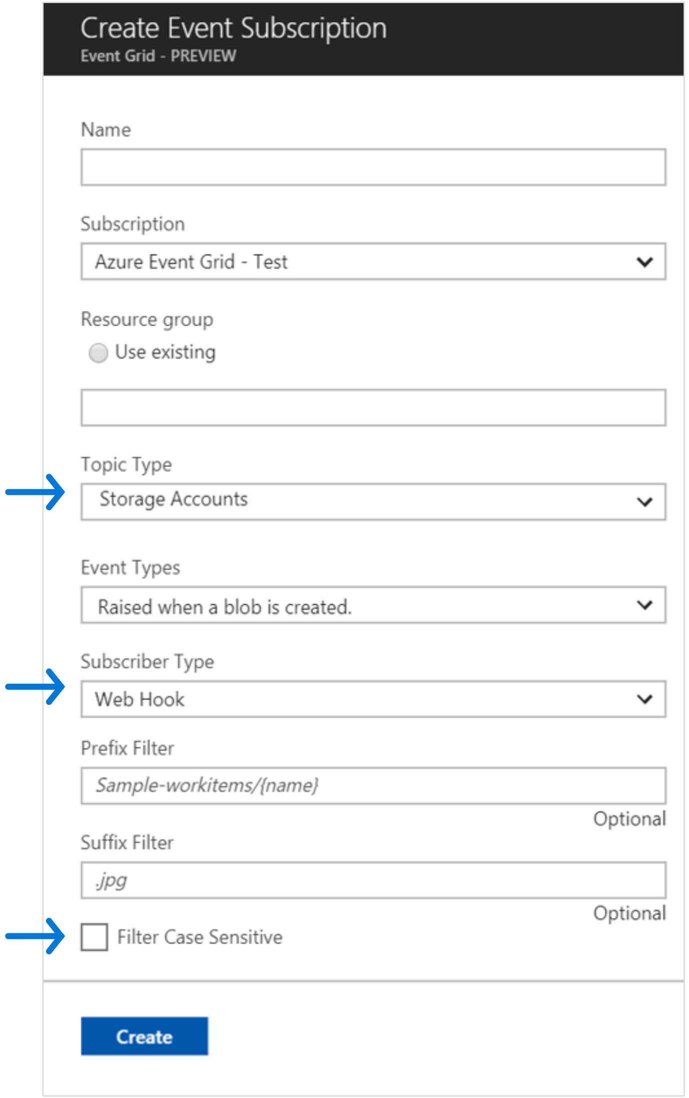
Event publishers



Subscribe to pre-defined system events in Azure or create your own custom topics

Route events to any end-points, Azure or even beyond

Enable filtering and efficient routing of events



Create Event Subscription
Event Grid - PREVIEW

Name

Subscription
Azure Event Grid - Test

Resource group
○ Use existing

Topic Type
Storage Accounts

Event Types
Raised when a blob is created.

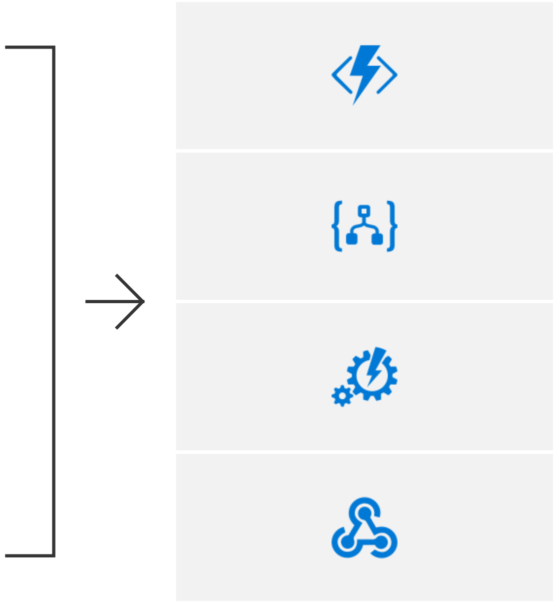
Subscriber Type
Web Hook

Prefix Filter
Sample-workitems/{name}
Optional

Suffix Filter
.jpg
Optional

□ Filter Case Sensitive

Event handlers



Ensure reliability and performance in your apps

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

Benefit from broad coverage

Publishers

Immediately available

- ✓ Blob Storage
- ✓ Resource Groups
- ✓ Azure Subscriptions
- ✓ Event Hubs
- ✓ Custom Events

Coming soon

Azure Automation, Azure Active Directory, API Management, Logic Apps, IoT Hub, Service Bus, Azure Data Lake Store, Cosmos DB

Subscribers

Immediately available

- ✓ Azure Functions
- ✓ Logic Apps
- ✓ Azure Automation
- ✓ WebHooks

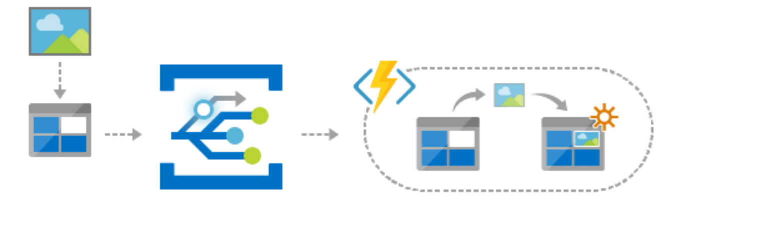
Coming soon

Fabric Controller, Service Bus, Event Hubs, Azure Data Factory, Storage Queues

Scenarios

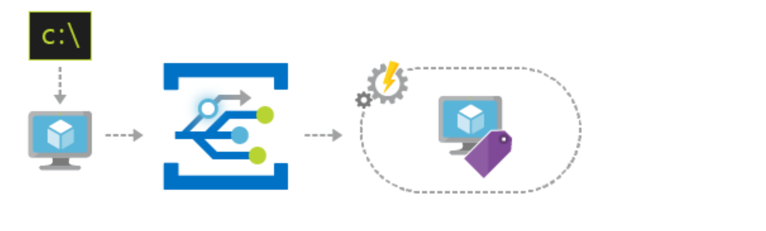
Serverless apps

Instantly trigger a serverless function to run analysis when a new file is added to a blob storage container.



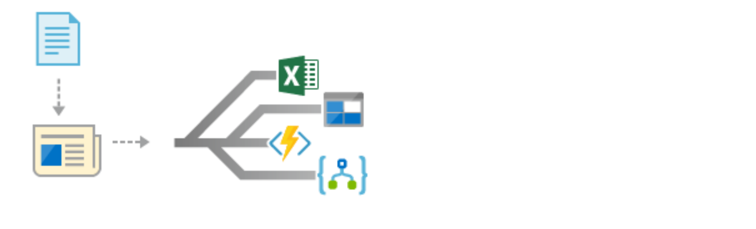
Ops automation

Speed up automation and simplify policy enforcement by notifying Azure Automation when underlying infrastructure is provisioned

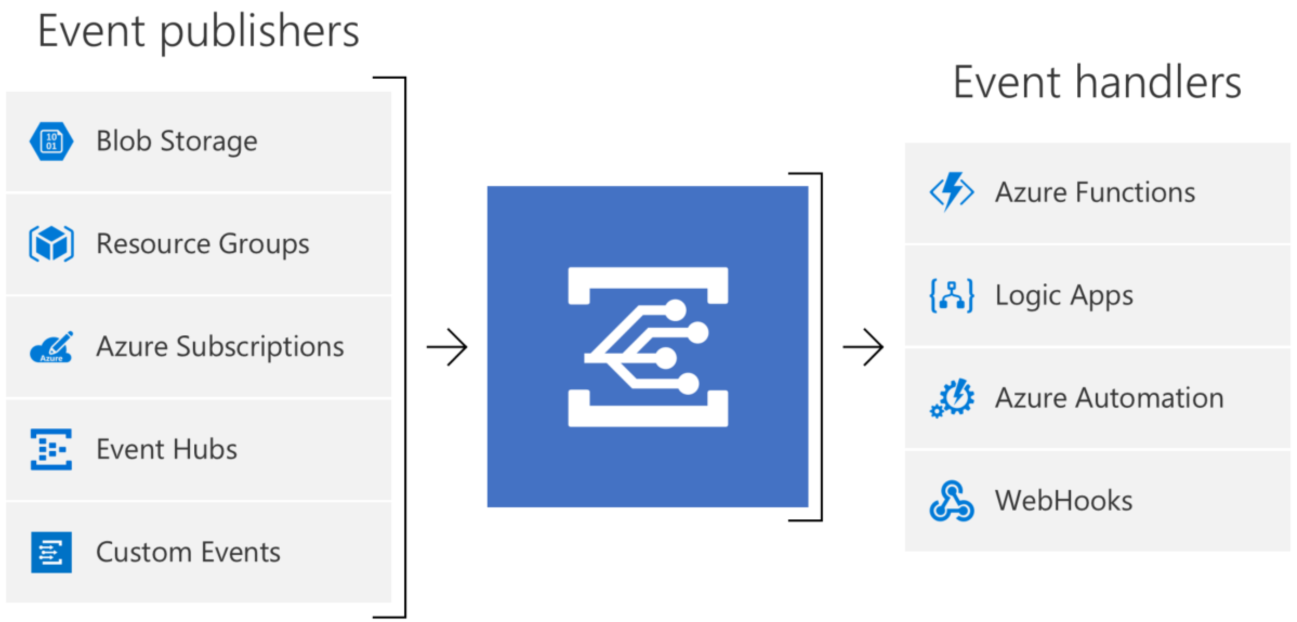


Application integration

Connects your app with other services. Create an application topic to route your app's event data to any desired destination

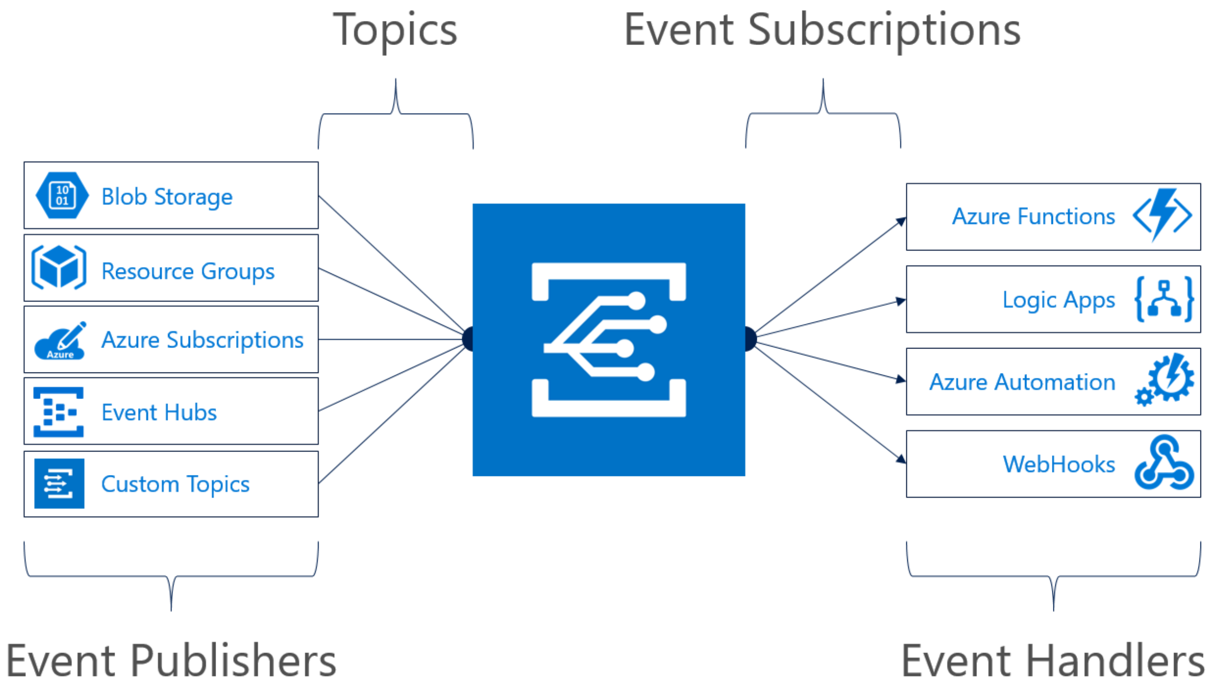


Concepts



1. Events: what happened
2. Event Publishers: where it took place
3. Topics: where publishers send events
4. Event Subscriptions: how you receive events
5. Event Handlers: the app or service reacting to the event

Concepts



1. Events: what happened
2. Event Publishers: where it took place
3. Topics: where publishers send events
4. Event Subscriptions: how you receive events
5. Event Handlers: the app or service reacting to the event

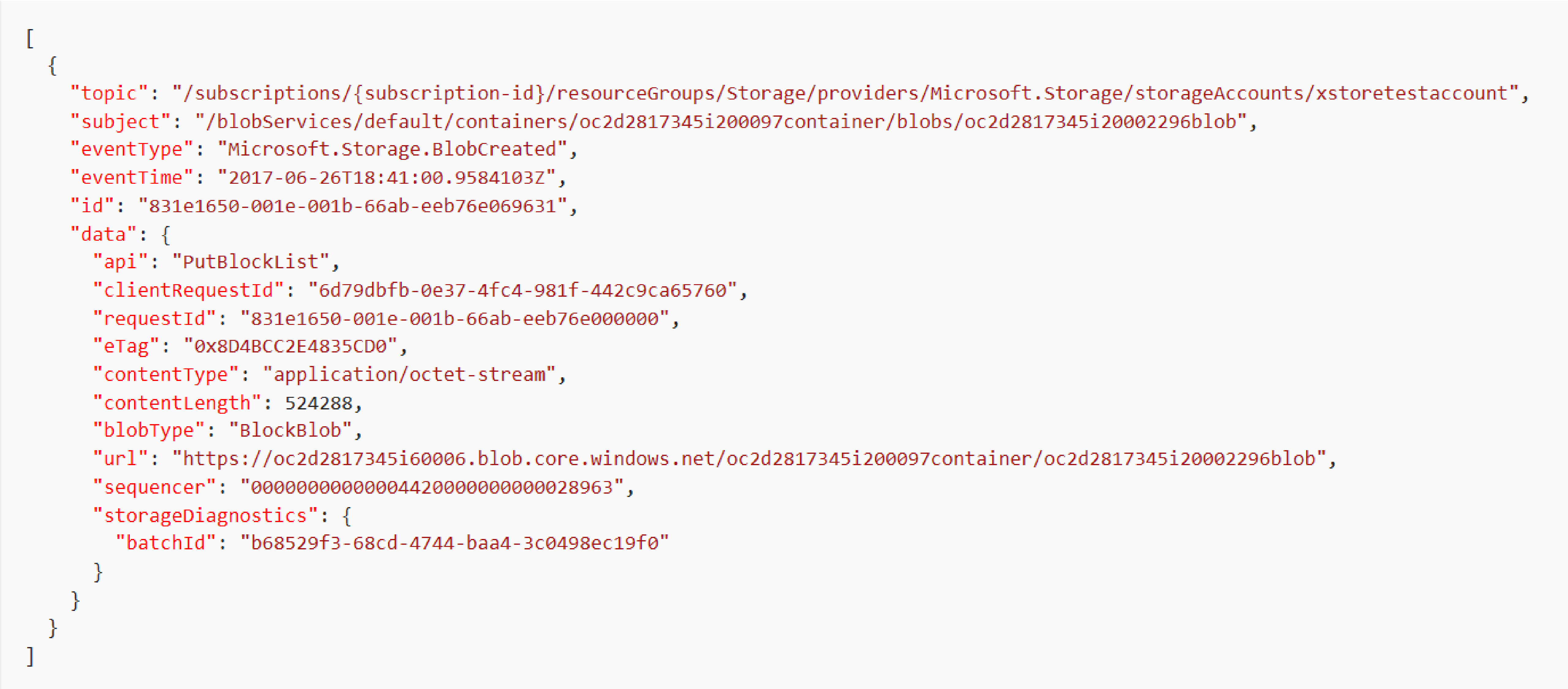
Event Grid guiding principles

- Always available
- Near real-time event delivery
- At least once delivery
- Dynamic scale
- Platform agnostic (WebHook)
- Language agnostic (HTTP protocol)

Target performance

- Sub-second end-to-end latency in the 99th percentile
- 99.99% availability
- 10,000,000 events per second per region
- 100,000,000 subscriptions per region
- 50 ms publisher latency
- 24 hour retry with exponential back off for events not delivered
- Transparent regional failover

Event Schema

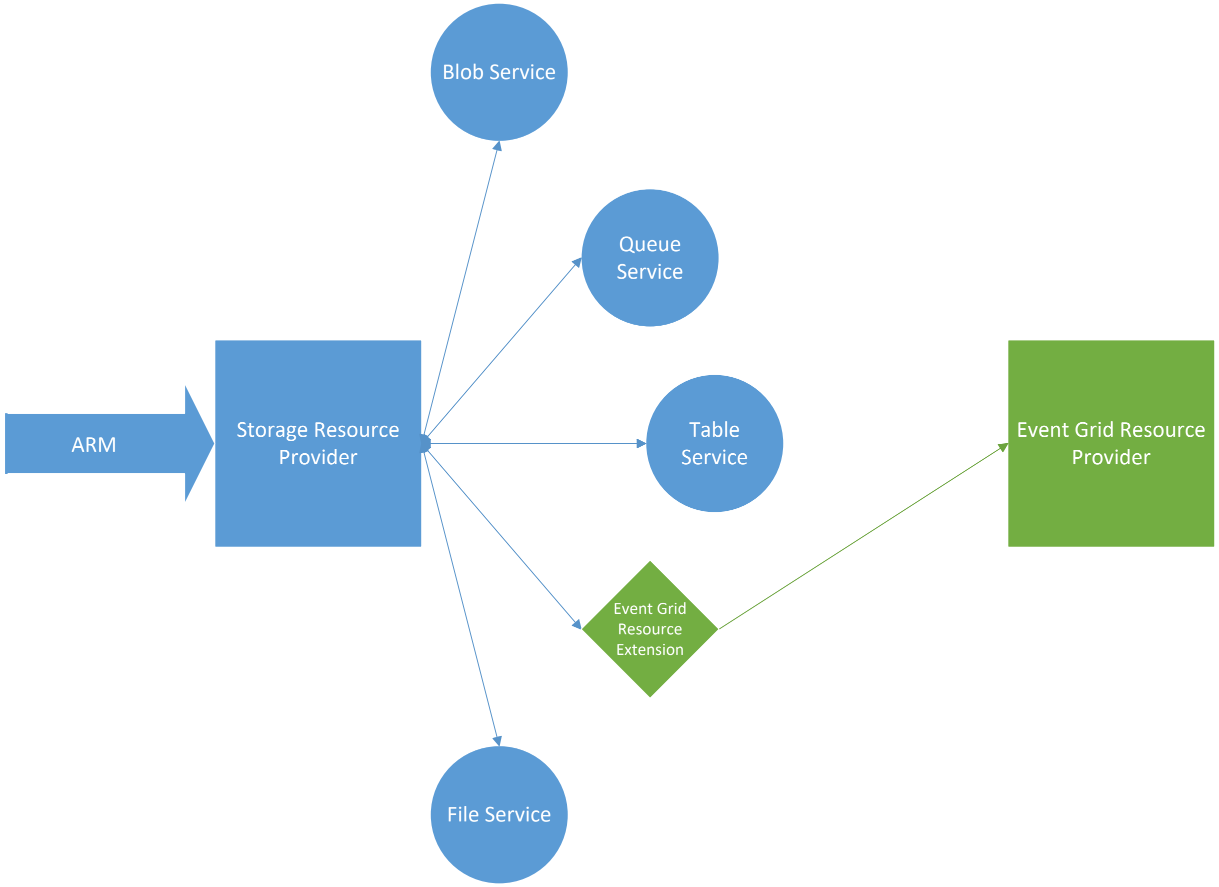


```
[
  {
    "topic": "/subscriptions/{subscription-id}/resourceGroups/Storage/providers/Microsoft.Storage/storageAccounts/xstoretestaccount",
    "subject": "/blobServices/default/containers/oc2d2817345i200097container/blobs/oc2d2817345i20002296blob",
    "eventType": "Microsoft.Storage.BlobCreated",
    "eventTime": "2017-06-26T18:41:00.9584103Z",
    "id": "831e1650-001e-001b-66ab-eeb76e069631",
    "data": {
      "api": "PutBlockList",
      "clientRequestId": "6d79dbfb-0e37-4fc4-981f-442c9ca65760",
      "requestId": "831e1650-001e-001b-66ab-eeb76e000000",
      "eTag": "0x8D4BCC2E4835CD0",
      "contentType": "application/octet-stream",
      "contentLength": 524288,
      "blobType": "BlockBlob",
      "url": "https://oc2d2817345i60006.blob.core.windows.net/oc2d2817345i200097container/oc2d2817345i20002296blob",
      "sequencer": "00000000000004420000000000028963",
      "storageDiagnostics": {
        "batchId": "b68529f3-68cd-4744-baa4-3c0498ec19f0"
      }
    }
  }
]
```

Resource model: extension resource

ARM calls are made to a parent resource

ARM reroutes all Event Grid calls to the Event Grid RP


ARM
Storage Resource Provider
Blob Service
Queue Service
Table Service
Event Grid Resource Extension
File Service
Event Grid Resource Provider

Best Practices for using Custom Vision

- Use at least 30 images for each tag
- Images should be the focus of the picture
- Use sufficiently diverse images and backgrounds (ex: cats with red background and dogs with blue background)
- Train with images that are similar in {quality, resolution, lighting, etc.} to the images that will be used in prod
- Supports Microsoft accounts (MSA) and AAD

Pricing (public preview)

- ✓ $0.30 per million operations ($0.60 per million operations upon GA)
- ✓ Operations include:
  - Ingress events
  - Advanced matches
  - Delivery attempts
  - Management calls
- ✓ 100,000 free operations per month
- ✓ Management operations throttled to 10 per second
- ✓ 1,000 event subscriptions per account

Operation definitions, number of free operations, management throttling, and number of subscriptions per account subject to change upon GA

Learn more at azure.com/EventGrid

Cloud Architecture

Micro Services

Agenda

- Introduction to Software Architecture
  - Requirements
  - Architecture
  - Design Principles
- Cloud Application Architecture
  - Architecture Attributes in Cloud Scale applications
  - High Availability, Management, Multi-Tenancy

Software Architecture

Software Architecture

- Architecture defines the system structure
  - Built from software components
  - The relationship between components
- Very young discipline
  - designing software architecture is still a mix of art and science!
- Usually architecture goes with high level design

Requirements & Constrains

- **Non-Functional** requirements
  - Mainly during architecture phase
- **Functional** requirements
  - Mainly during design phase
- **Constraints**
  - Decision helpers

ISO 9001 SGS

Non-Functional Requirements

It is hard to express these requirements in Use Cases or User Stories

- Express these requirements in a measurable way
  - Instead of: “The system has to be responsive”
  - Say: “The system has to respond to user action in less than 50 milliseconds”
- Non-Functional requirements have more influence on the architecture than functional requirements
  - Availability, scalability, security, …

ISO 9001 SGS

Architecture Attributes

- **Performance**
  - Localize operations to minimize sub-system communication
- **Security & Identity**
  - Use a layered architecture with critical assets in inner layers
  - Provide Identity flow diagram for the whole system
- **Availability**
  - Include redundant components in the architecture
- **Maintainability**
  - Use fine-grain, self-contained components
- **Scalability**
  - Include redundant components and handle state

More *-ilities

- Stability
- Backward compatibility
- Extensibility
- Reliability
- Maintainability
- Availability
- Security
- Usability
- Auditability

- Scalability
- Testability
- Composability
- Demonstrability
- Deployability
- Efficiency
- Learnability
- Manageability
- Operability

Cross Cutting Concerns

Those aspects that span across components and layers

- Identify each of the crosscutting concerns
  - Design separate components to manage these concerns
  - This approach provides better reusability and maintainability
- Avoid mixing the crosscutting code with the component code
  - You can use DI or AOP to decouple cross cutting concerns from your components
- Crosscutting Concern libraries must be deployed in such a way that all other component can use them

Cross Cutting Concerns

- Logging
- Authentication and Authorization (Identity & Security)
- Error Handling
- Communication & Hosting
  - Pub/Sub, Queues, Notifications, REST, SOAP
- Storage
- Report
- Caching
- Configuration
- Validation

Reuse:

- Platform SDK/Libraries
- Programming Language Libs
- 3rd party/open source project
- Cloud Services

Cloud Architecture

State Management

A Program is something that reads input, changes state and provides output

- The way you manage state has huge impact on your architecture and design
  - Sometime your state is transient
  - In many other cases you rely on durable storage
    - Transaction or Compensation (ACID vs BASE)
    - In-process state management vs. external state management
      - What does stateless mean?
- Bad state management can lead to poor performance, eliminate scalability or reduce correctness

Data

The sun-light that you see indicates that the sun was still there 8 minutes ago!

- When you see in a shopping site that the product is available, is it?
- For Web/Cloud Scale application:
  - Prefer read-only data (Great for caching, CDN, Scale)
  - Understand your data flow & data consistency
  - Be prepared to compensate because of stale data

Cloud Scale

High Scale, High Availability, High Maintainability – to the Extreme!

- The Cloud technology enables cloud scale applications with much less effort
  - Answers many of the non-functional requirements and the cross-cutting concerns out of the box!
- Enables elastic, theoretically endless scale
- Surfaces more concerns:
  - More geographic options
  - Promotes “design for failure” (be prepared for VM shutdown)
  - Promotes Agility (continued integrations/deployments)

High Availability

Remove any single-point-of-failure throughout your application

- Make sure you have redundant services
- Distribute services and data across geographies
- Make sure your storage is HA
- Take storage snapshots
- Extreme monitoring & automatic responses
  - Elastic scaling
  - Self healing mechanisms
- Long latency is a failure!
  - Make sure your services respond quickly even on load
    - Spawn more instances, buy better QoS resources (I/O, CPUs, Memory, Network)
    - Load balance to reduce the pressure
- Have a storage only based failover web site for web applications

Load Balancing

- Use Auto Scaling feature to add or remove compute resources
- Azure Load Balancer provides Layer 4 load balancing
  - LB across Fault Domains
  - Hash-based distribution
  - Port forwarding
- Azure Application Gateway provides Layer 7 (HTTP) load balancing
  - Cookie based distribution (session affinity)
  - Configurable distribution rules
  - SSL offloading
- Sticky Session – try to avoid it
- Azure Traffic Manager provides DNS based Geo Load-Balancing
  - Failover
  - Round Robin
  - Performance
- Preferably use the CNAME record in the DNS and not a single server IP or Virtual IP

Queues in Cloud Applications

Load Balancing over time, and:

- A Transport
  - Asynchronous data distribution
  - Reliable delivery of messages
- Decoupled Services
  - Horizontal Scaling
    - S : R ratio
      - Set the number of senders and receivers according their workload
- A Buffer
  - Time Scaling (Buffer over time)
  - Reliable mechanism for partially offline services

Using Queues

- Azure Queue Storage Service / Service Bus Queues
  - Since there is a size limit to a message
    - For large messages, put a URL to storage item in the message
  - Some queues do not guarantee FIFO
    - Build your app to handle it (Add counter to the message)
  - Idempotency is important (Handle duplicate message gracefully)
    - Azure Service Bus Queues can guarantee reliable messaging and duplicate detection
  - Understand the access control of the queue
  - Handle polling rate & batch operations wisely
  - Set visibility timeout to handle message concurrency control
  - Set the retention period (when the message will be deleted)
  - Handle poison messages

ISO 9001 SGS

Service Oriented Architecture

- An architectural pattern in which application components provide services to other components via a communications protocol
- A service is a self-contained unit of functionality
- Services can be combined to provide the functionality of a large software application
- SOA makes it easier for software components on computers connected over a network to cooperate.
- Every compute resource can run any number of services, and each service is built in a way that ensures that the service can exchange information with any other service in the network without human interaction and without the need to make changes to the underlying program itself

The 24/7 Challenge

- How do you update a system running 24/7/365?
  - How do you keep the application servers responsive?
  - How do you keep all application servers synced?
  - How do you update the data/schema?
  - How do you update all your clients' software?
    - Web, Mobile, Desktop…
  - How do you rollback on error?
    - How do you rollback data?
    - How do you know there is an error?

Microservices Architecture (MSA) - Wikipedia

- *“Microservices is a specialization of and implementation approach for service-oriented architectures (SOA) used to build flexible, independently deployable software systems”*
- *“Services are **small in size, messaging enabled, bounded by contexts**, **autonomously developed**, **independently deployable**, **decentralized** and built and released with **automated processes**”*
- *“The benefit of distributing different responsibilities of the system into different smaller services is that it **enhances the cohesion** and **decreases the coupling**”*

Server Side Patterns – Micro Services

“Microservice applications are composed of small, independently versioned, and scalable customer-focused services that communicate with each other over standard protocols with well-defined interfaces.”

- Extreme SOA
  - A class is a service
  - Each service encapsulates a simple business functionality
    - Volatile boundary
    - Versioning is backed in to the method
- Requires hosting and management system
  - Azure Service Fabric for example
  - Docker or other Container technology

Micro Service Architecture

- Some principles:
  - A micro service should be less then 100 lines of code
    - It should be easy to understand, fast to deploy, and cheap to reimplement (throw the old one)
  - A micro service should be independently developed & deployed
  - A micro service should has private data ownership
  - Eventual Consistency
  - Versioning

Microservices & SOA

- Microservices
  - Small, independent processes that communicate with each other to form complex applications which utilize language-agnostic APIs
    - Small building blocks
    - Highly decoupled
    - Focused on doing a small task
  - Facilitate a modular approach to system-building.
  - The microservices architectural style is becoming the standard for building continuously deployed systems
  - Advantages
    - Services are easy to replace
    - Services are organized around capabilities, e.g., user interface front-end, recommendation, logistics, billing, etc.
    - Services can be implemented using different programming languages, databases, hardware and software environment, depending on what fits best

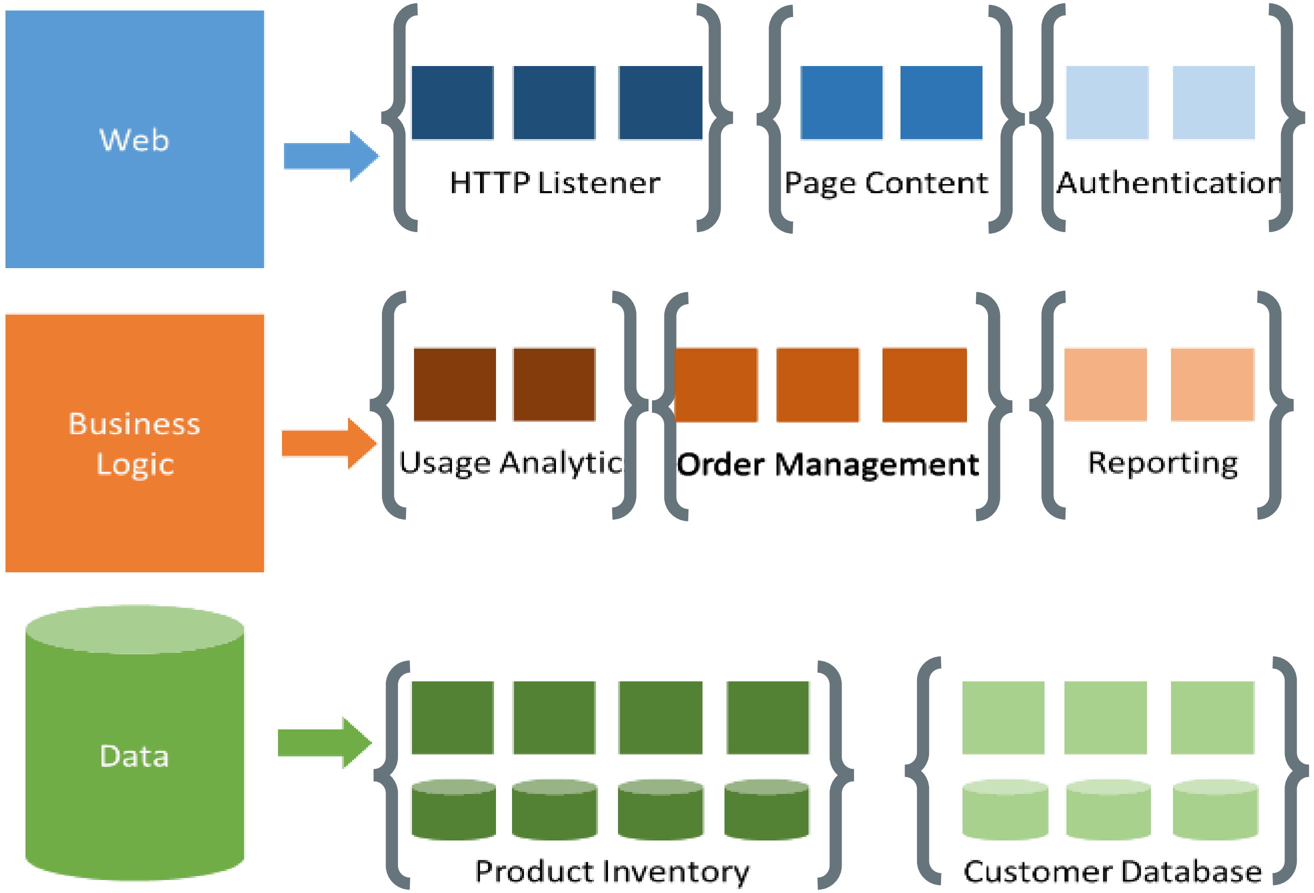
ISO 9001 SGS

Microservices & SOA

- SOA aims at integrating various (business) applications
- Microservices belong to a single application
  - Naturally enforces a modular structure
  - Architectures are usually symmetrical rather than hierarchical
- Microservices sometimes referred to as a finer-grained SOA
  - Or SOA as SOA should have been
- How small is “micro”?
  - It depends!

Modernization with Microservices

- Individually built and deployed
- Small, independently executing services
- Integrate using published API calls for overall application’s functionality
- Fine-grained, loosely coupled application



MSA - Signs that you do something wrong

- If you find the need to deploy services together, you're doing something wrong
- If you have a single codebase for all your services, you're doing something wrong
- If you find you have a service talking directly to the database of another service, you're doing something very wrong
- If you have to send out a warning before each deploy of a service, you're doing something wrong

Microservices

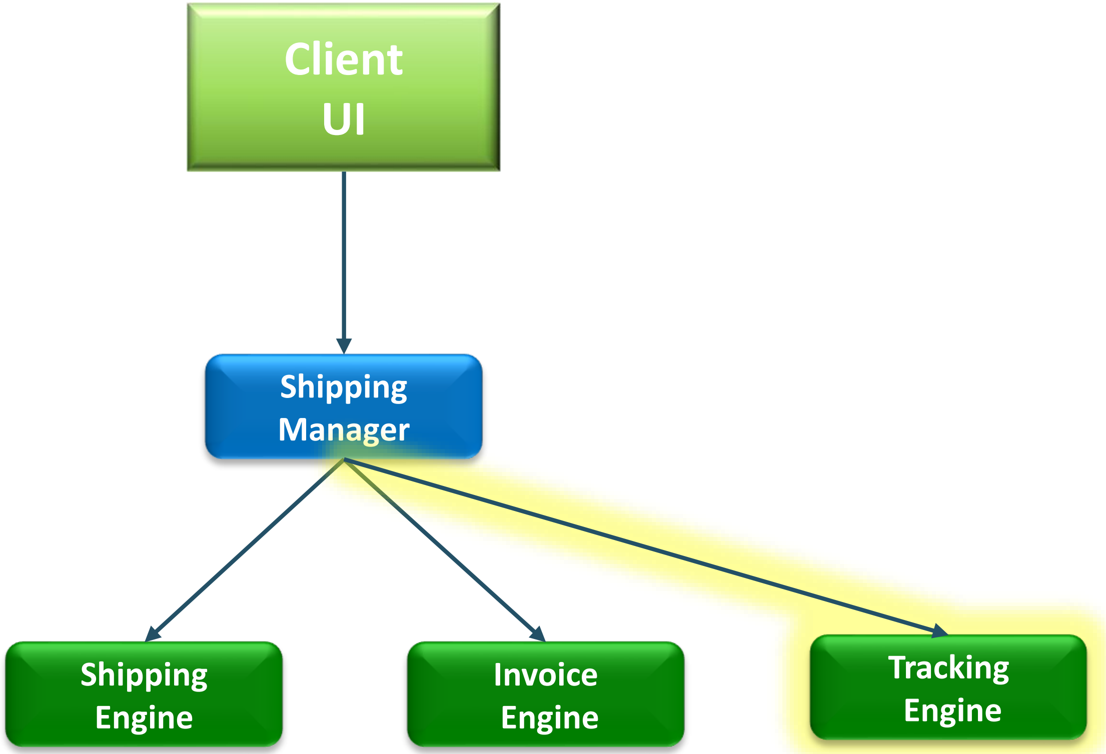
- Microservices is traditionally hard
  - Deploy & manage multiple (high) amount of distinct components
  - Ensure scalability & HA of each component separately
  - Ensure sufficient compute resources for each component separately
  - Distribution of components across a server farm
- Today we have tools to manage these concerns for us
  - Or at least help us manage them correctly
- Two popular options on Azure
  - Azure Container Service (Docker, Managed Kubernetes)
  - Azure Service Fabric

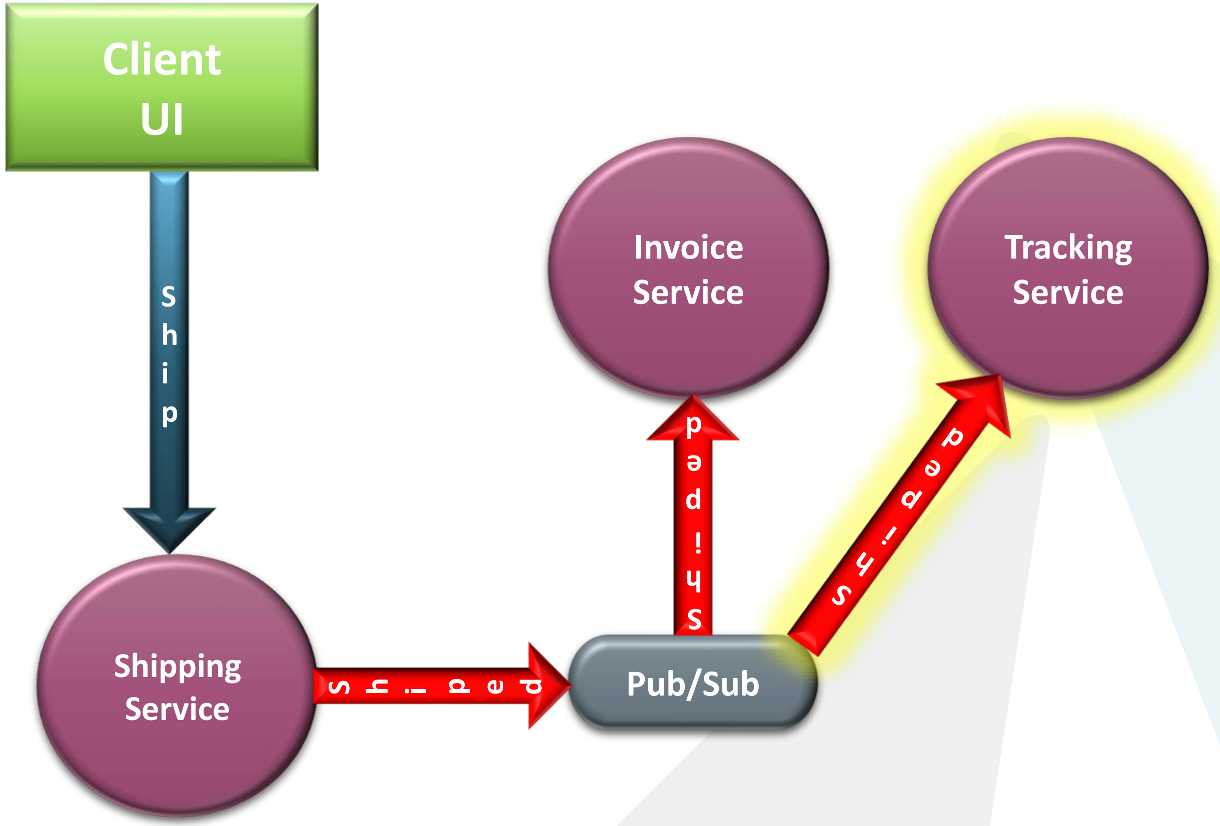
ISO 9001 SGS

Decoupling Business Workflows

- Orchestration vs Choreography
- The Manager service in IDesign method encapsulates the business process
  - A major change in the business-flow requires a change in the manager
- In a non-restrictive MSA, any service can consume any message and call any other service – this is the common approach!
  - Either by using an API, or by sending a message to a broker
    - Event Driven Messaging, Pub/Sub, Queues, etc.
  - With this approach one can change the business flow just by introducing a new micro service!!!
  - However, the complexity of the system is much higher!!!
- Does MSA flexibility worth the complexity?
- Server-less application is another form of MSA, and introduces even higher complexity

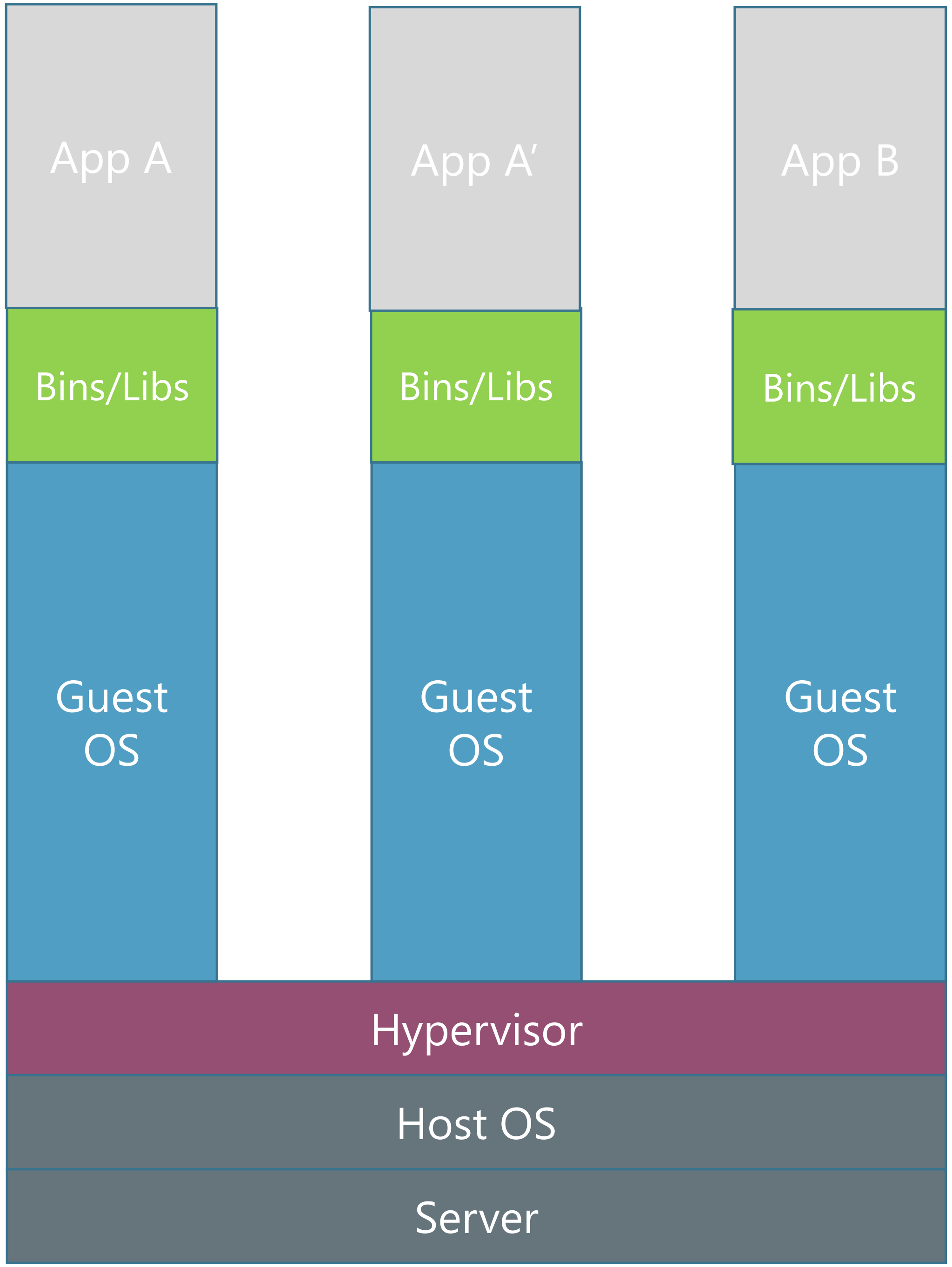
Orchestration vs Choreography



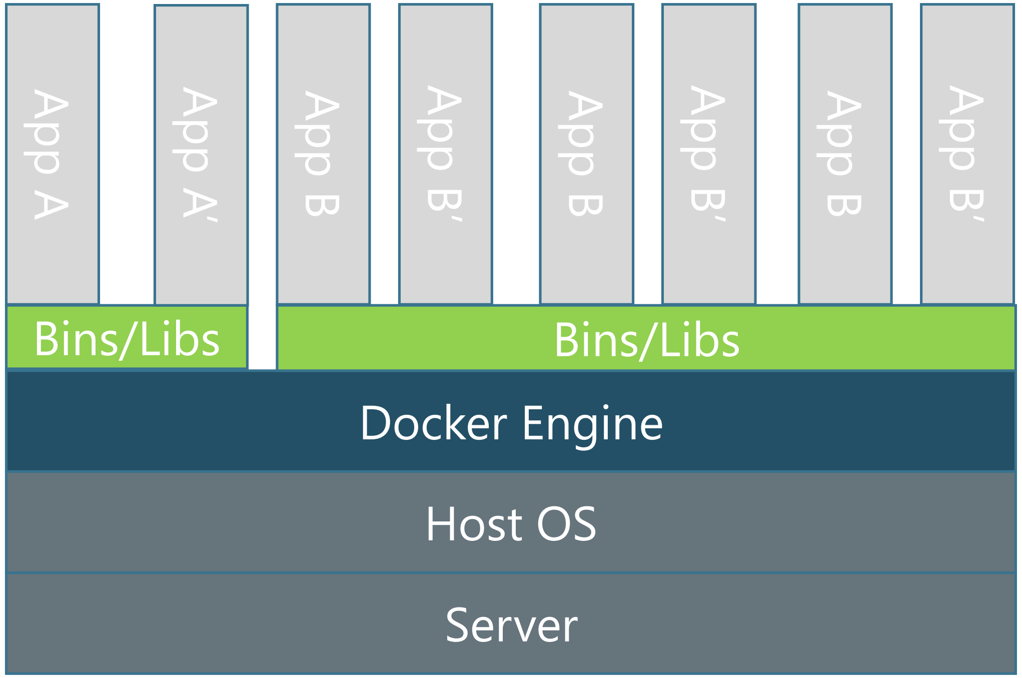


The Seek for the Perfect Host

- To deal with MSA complexity the requirements are:
  - Isolation, Security
  - Maintainability, Manageability, Traceability
  - Fast Auto Scaling, Health monitoring, Self Healing
- Famous Hosts:
  - Process
    - (Console) Application
    - Windows Service
    - Unix/Linux Demon
  - Web Servers
    - IIS, Apache, NGINEX
  - Framework and language runtimes
    - .NET AppDomain, JVM
  - Virtual Machines
    - Hyper-V, VMWare, VirtualBox
  - Containers
    - Docker Containers, Windows Containers, Windows Hyper-V Containers, Service Fabric
- Containers are the current MSA favorite hosting model



Containers are isolated, but share OS and, where appropriate, bins/libraries



Microservices

- Use Azure Container Service
  - Existing Docker enabled systems
  - Usage of existing 3rd party tools which integrate with Docker
  - Want to customize various elements of the framework
    - Registry service
    - Image repository
    - Cluster orchestration
- Use Azure Service Fabric
  - Prescribed holistic solution
  - New software taking advantage of the service fabric SDK
    - Stateful Services
    - Actor Model
  - Easier to use and get started
- Integration is possible!
  - Run Docker containers in Service Fabric
  - Run Service Fabric in a Docker container

Multitenancy

One Application to Rule Them All!

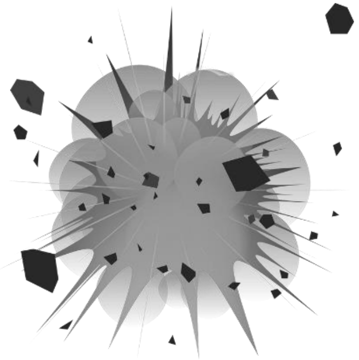
- The cloud provides many benefits
  - You develop and deploy to the cloud
    - Staging/Production cloud environments
  - Cloud management & monitoring
- You'd like to share those effort for all of your customers
- A Multi-tenant application is one application that serves many isolated application instances

Multitenancy

- Consider which resources are shared and which are separate
  - Beware of privacy!
- Multi-factor optimization question
  - Cost of resources
  - Maintenance difficulty
    - Failures
    - Updates
  - Security constraints
    - The more separation, the more security…
    - Strive to avoid potential code issues which can retrieve wrong data

Azure IoT

Traffic Light History Facts

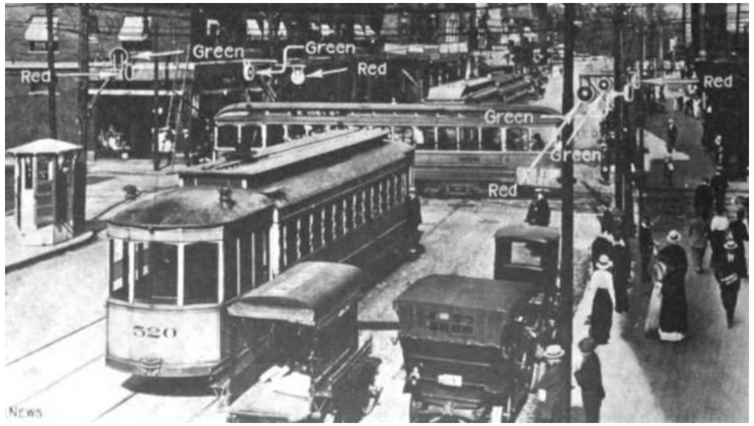


1868

London

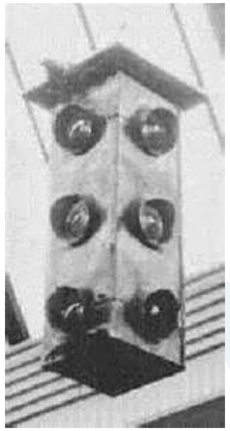
1912

SLC



1914

Cleveland



1920

Detroit

1922

Automatic timer controlled

1950s

Computer Detection

*light*TRAFFIC

Agenda

- Introduction
- The Simple System
- IoT Device Lifecycle
- The Modern IoT System
- Azure IoT PaaS & SaaS
- Azure IoT Hub
- Device Registry & Provisioning Service
- Twin, Routing and Jobs
- Smart cloud & intelligence edge
- Summary

Technology advances in “Buzzwords” steps

- It starts with the basic technology and slowly evolves
- After several years the world is ready to embrace the technology
- And here is where the big buzz begins
- For the Internet of Things:
  - The basic technology is already here for almost a decade by now
    - Amazon Web Services cloud has started in 2006
    - Azure was announced in October 2008 and released on February 1, 2010
    - Devices such as the Amtel AVR are more than 20 years old
    - The Arduino family that many IoT hobbyists are using it as a cheap IoT end device is 10 years old

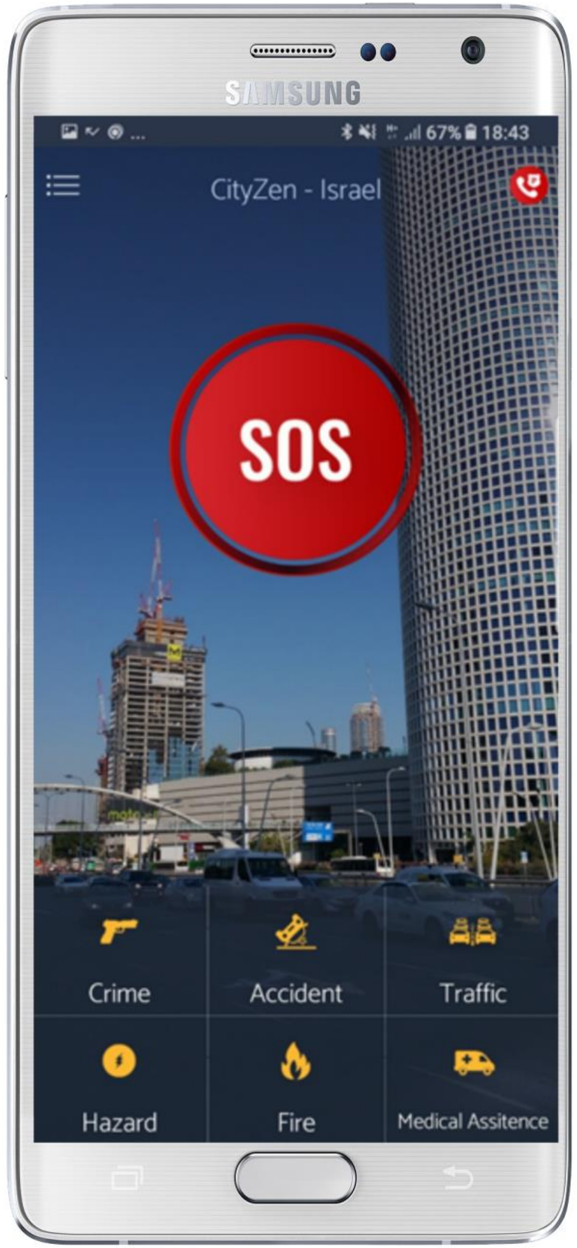
IoT - More Than The Core Technology

- It is not (just) the technology that makes IoT what it is
- It is the
  - Concepts, perception, commitment and the challenges
  - Facts that the entire industry is dealing with it nowadays
- The IoT Challenge:
  - ***vast*** *amount of* ***devices*** *using different hardware and software technologies, are* ***connected*** *between them and to the* ***cloud*** *which in turn provides many* ***services****, which handle a* ***huge stream of data*** *and* ***analyze*** *it and extract vital* ***information*** *about the current state of the system and via extended processing it can even* ***predict*** *future state*

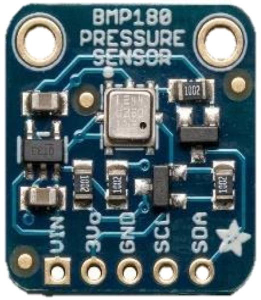
The Citizen App

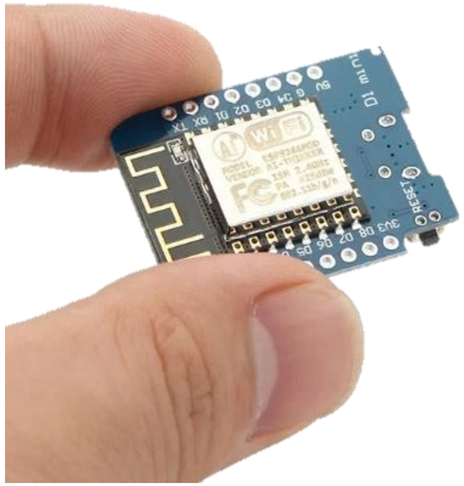
- Make any citizen a sensor(s)
  - Text, picture, video, audio, location
- Large city scale – 10s Millions
- Server-less Architecture:
  - IoT Hub per city
  - Stream Analytics
  - CosmosDB
  - Azure Functions
  - File uploads
- Small team: 2-3 developers, 6-8 months

Amazing!



IoT System Basic Components





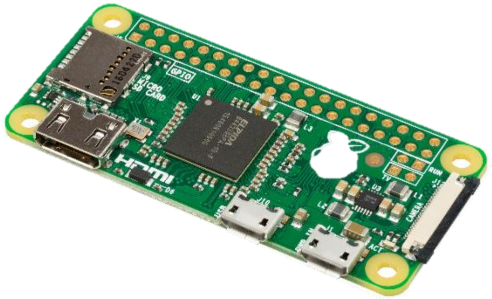
+

=

```
var sensorData = await _bmp180.GetSensorDataAsync(Bmp180.UltraHighResolution);
var messageString = JsonConvert.SerializeObject(sensorData);
var message = new
    Microsoft.Azure.Devices.Client.Message(Encoding.ASCII.GetBytes(messageString));
await deviceClient.SendEventAsync(message);
```

The Device

- There are many System on a Chip (SoC) devices to choose from
- Raspberry Pi family
- Arduino Compatible Family
  - ESP 8266 based devices
- Intel devices
- …



Raspberry Pi Kit

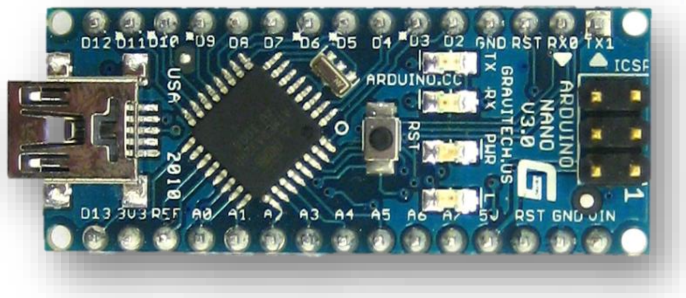
Windows 10 and Raspbian

Samples in C and C#

Intel Edison Kit

Linux Yocto

Samples in JavaScript (Node.js)



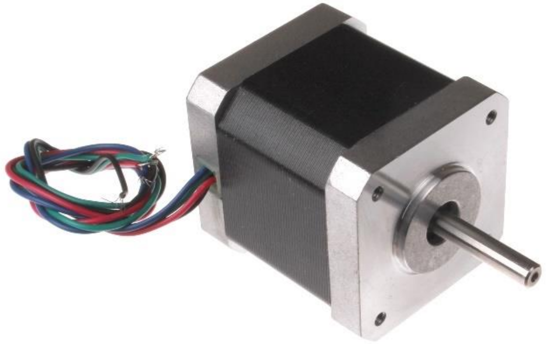
Feather Huzzah ESP8266 Kit

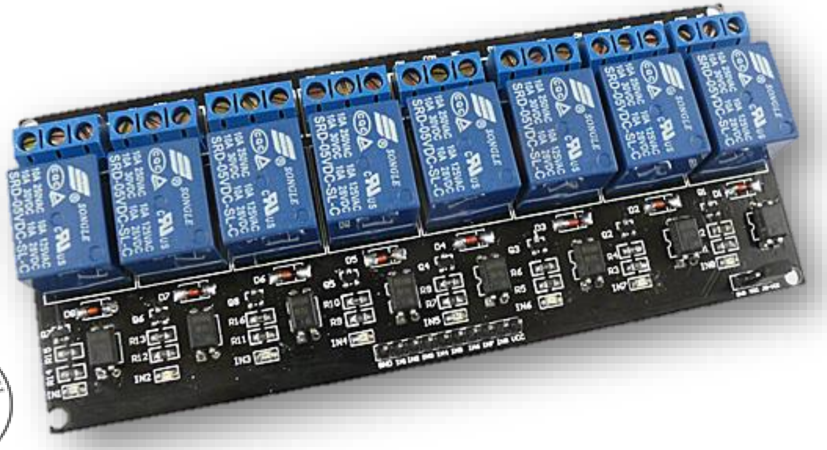
RTOS

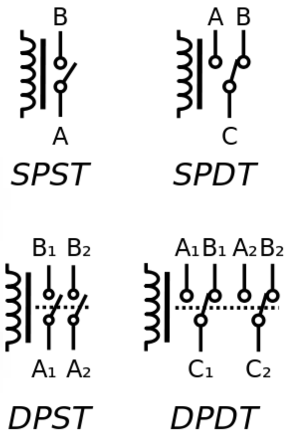
Samples in Arduino IDE and C

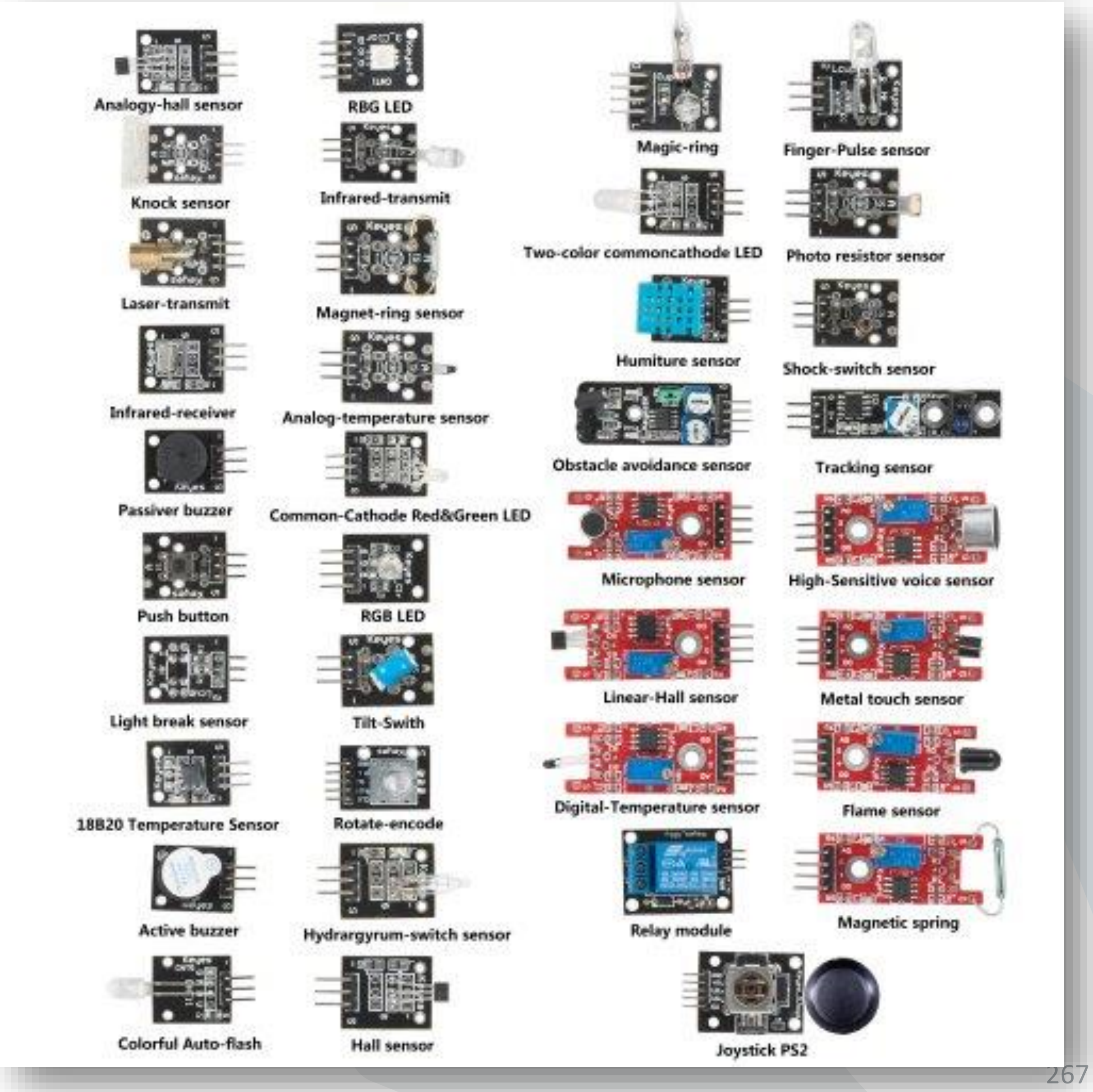
Sensors, Actuators, Motors

- There are [plenty](#) of them









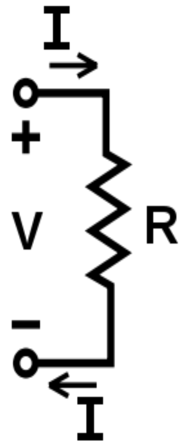
How do I play with it?

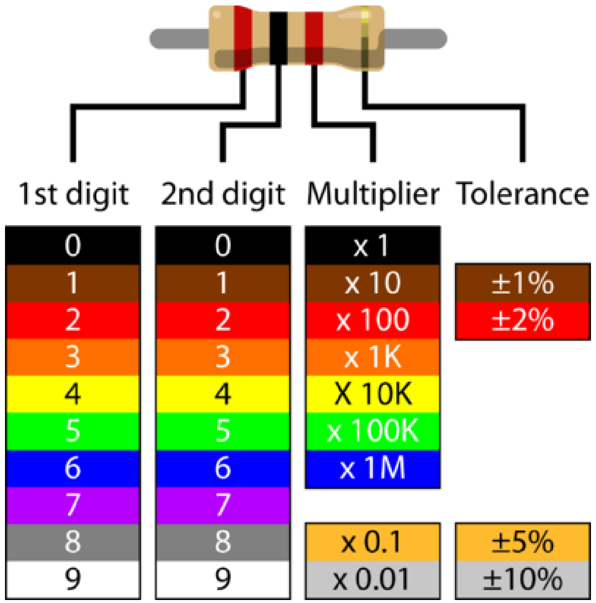
- Pick your weapon
  - A prototype board that has networking capabilities
    - Raspberry Pi, WeMos, Intel IoT, Tessel, NetDuino
    - The complete list @ Azure IoT hardware catalog
- Some Electronics & Hardware Programming
  - Understand how to connect sensors and communicate with them
- Pick your Cloud Services and technologies
  - Microsoft Azure, AWS
  - Do something with the (Big) data

ISO 9001 SGS

Electronics 101

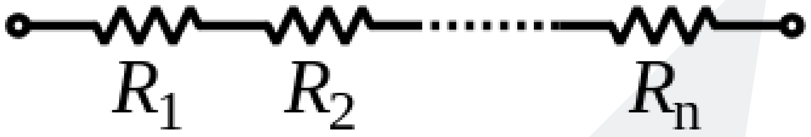
- Ohm’s law: V = R*I
  - V → Volt (V, mV)
  - I → Ampere (A, mA)
  - R → Ohm (Ω)



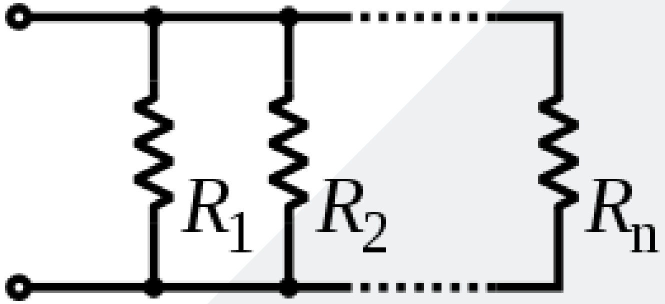


1st digit	2nd digit	Multiplier	Tolerance
0	0	x 1	
1	1	x 10	±1%
2	2	x 100	±2%
3	3	x 1K	
4	4	X 10K	
5	5	x 100K	
6	6	x 1M	
7	7		
8	8	x 0.1	±5%
9	9	x 0.01	±10%

- Series and Parallel Resistors
  - Series: $R_{EQ} = R_1 + R_2 + \ldots + R_n$

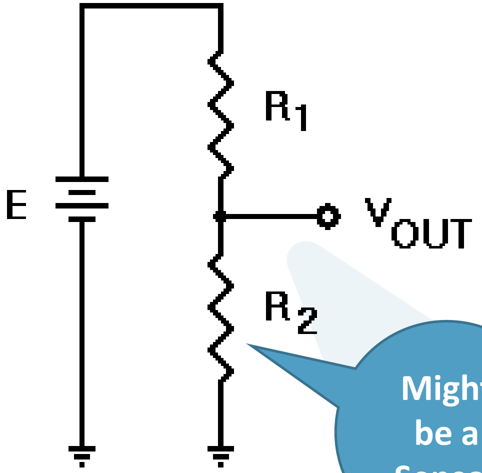


  - Parallel: $1/R_{EQ} = 1/R_1 + 1/R_2 + \ldots + 1/R_n$

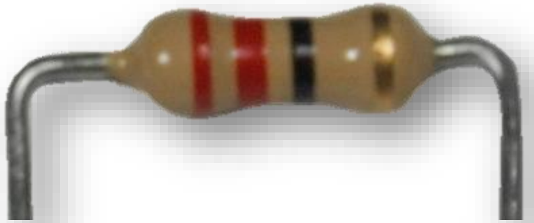


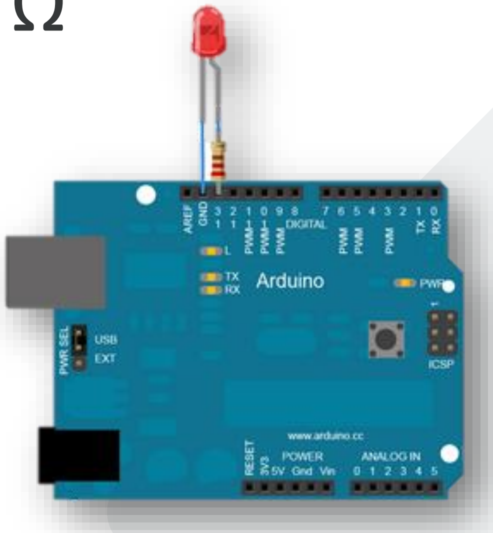
Resistive divider, LED current limiter

- Voltage divider: $V_{OUT} = V_{IN}*(R_2/(R_1 + R_2))$
  - 3.3V supply, R_1 = 5kΩ, R_2 = 10kΩ
    V_{OUT} = 3.3V*(10kΩ/(10kΩ+5kΩ)) = 2.2V
- LED resistor: $R = (V_{SUPPLY} – V_{LED})/I_{LED}$
  - 5V supply, 0.7V 20mA LED: R = (5V – 0.7V)/20mA = 215Ω
    ➔ Nearest higher rated resistor 220 Ω



Might be a Sensor



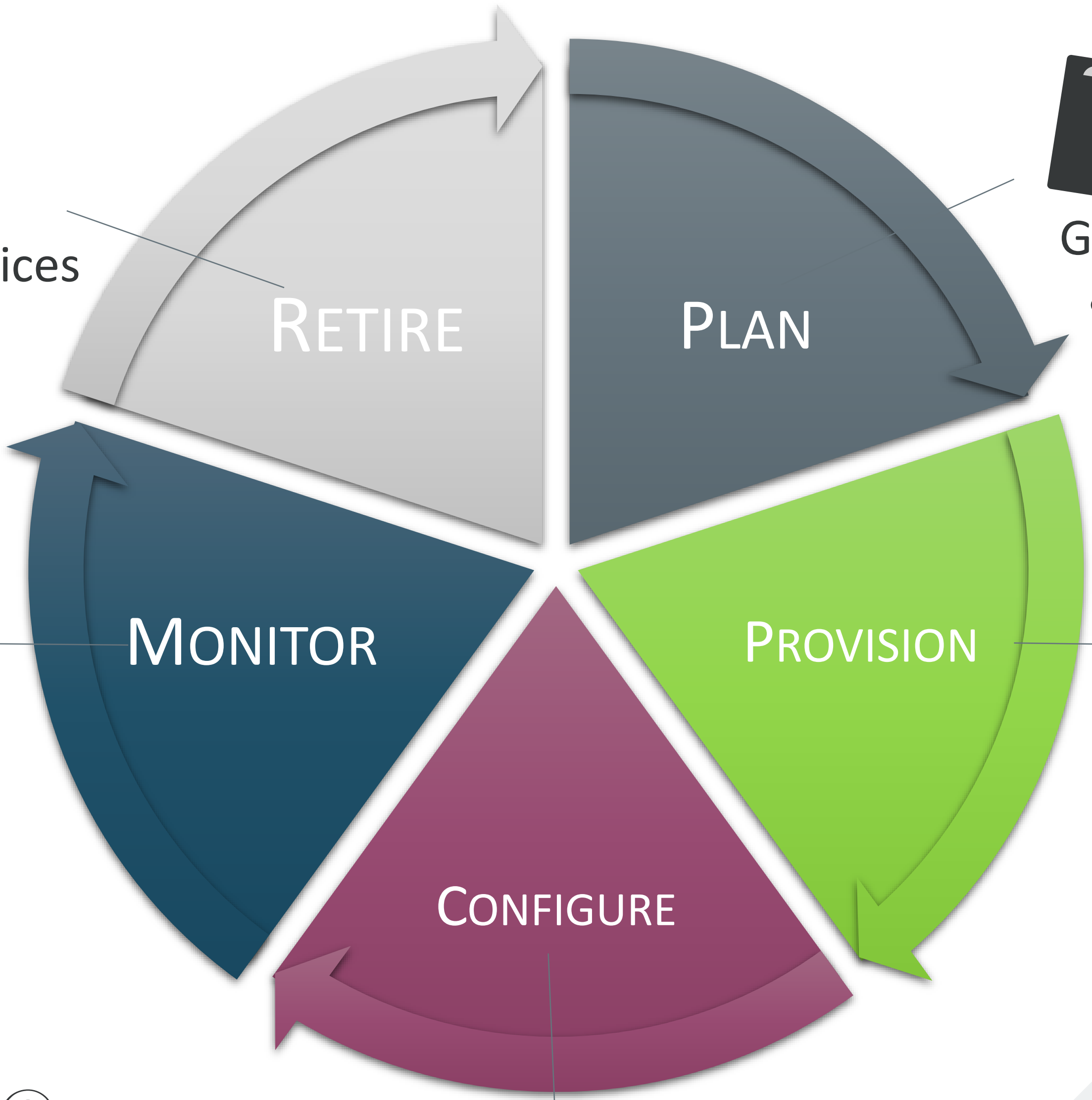


The Modern IoT System

- Most large IoT systems include one or more of the following:
  - Many different end **devices** with **sensors** and **actuators**
  - Local **gateways**
  - A collection of **cloud services** that enables:
    - **Registration** of end devices
    - **Management** of end devices
    - **Controlling** of end devices
    - Different **communication protocols** that provide reliability and security
    - The ability to **collect a vast amount** of data in a very **high rate**
    - The ability to **analyze** the **stream** of information in **close to real-time** manner
    - The ability to **analyze** the **current** and **historical** collected information
    - The ability to **show** the resulted **conclusion** and the **collected data**

The IoT Challenge - Pets Vs Cattle – Pettle?

IoT Device Lifecycle



Replace or decommission devices after failure, upgrade cycle or service lifetime

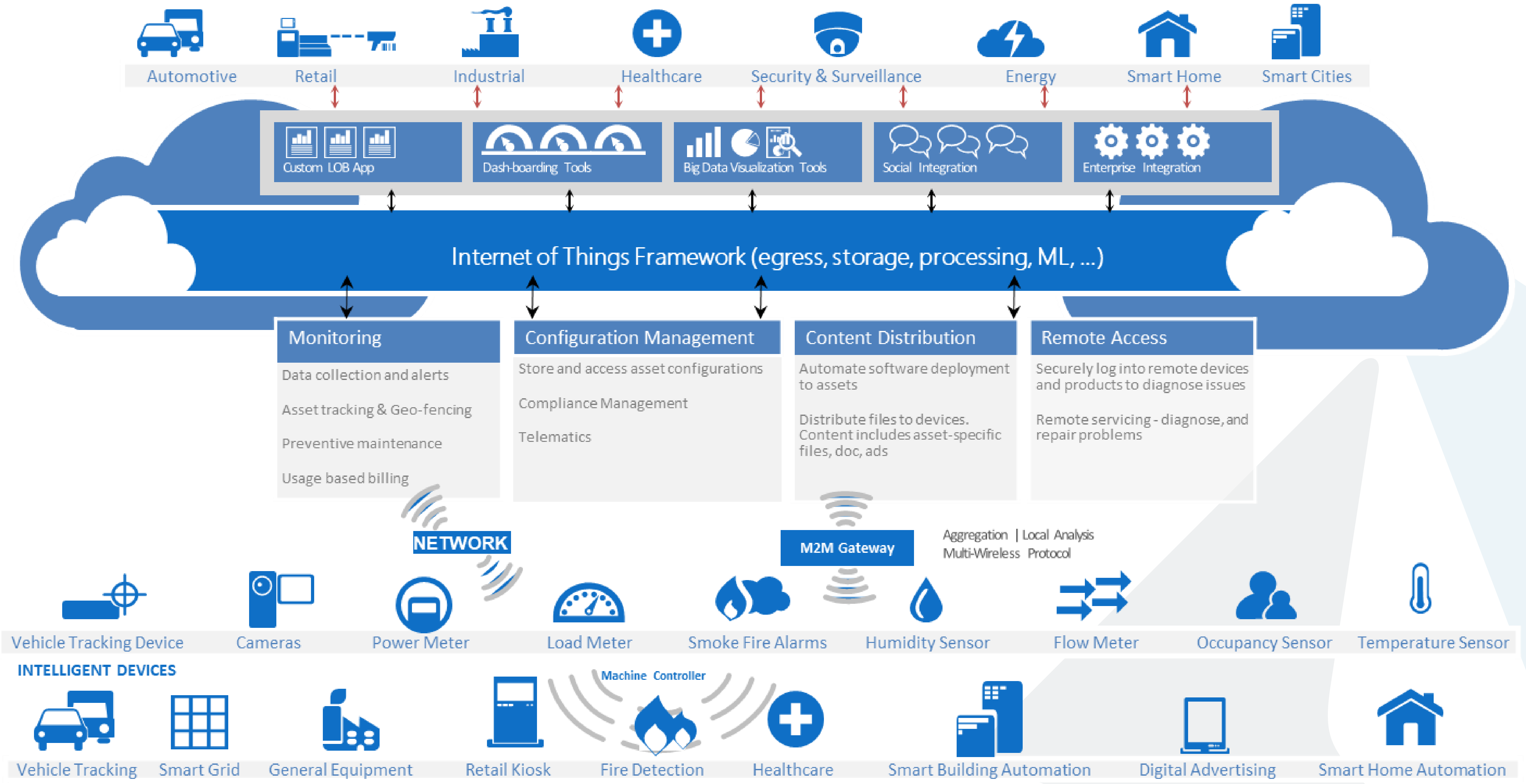
Group devices and control access according to your organization's needs

Monitor device inventory, health & security while providing proactive remediation of issues

Securely authenticate devices, on-board for management and provision for service

Provide updates, configuration & applications to assign the purpose of each device

High Level Architecture



Automotive

Retail

Industrial

Healthcare

Security & Surveillance

Energy

Smart Home

Smart Cities

Custom LOB App

Dash-boarding Tools

Big Data Visualization Tools

Social Integration

Enterprise Integration

Internet of Things Framework (egress, storage, processing, ML, ...)

Monitoring

Data collection and alerts

Asset tracking & Geo-fencing

Preventive maintenance

Usage based billing

Configuration Management

Store and access asset configurations

Compliance Management

Telematics

Content Distribution

Automate software deployment to assets

Distribute files to devices. Content includes asset-specific files, doc, ads

Remote Access

Securely log into remote devices and products to diagnose issues

Remote servicing - diagnose, and repair problems

NETWORK

M2M Gateway

Aggregation | Local Analysis
Multi-Wireless Protocol

Vehicle Tracking Device

Cameras

Power Meter

Load Meter

Smoke Fire Alarms

Humidity Sensor

Flow Meter

Occupancy Sensor

Temperature Sensor

INTELLIGENT DEVICES

Machine Controller

Vehicle Tracking

Smart Grid

General Equipment

Retail Kiosk

Fire Detection

Healthcare

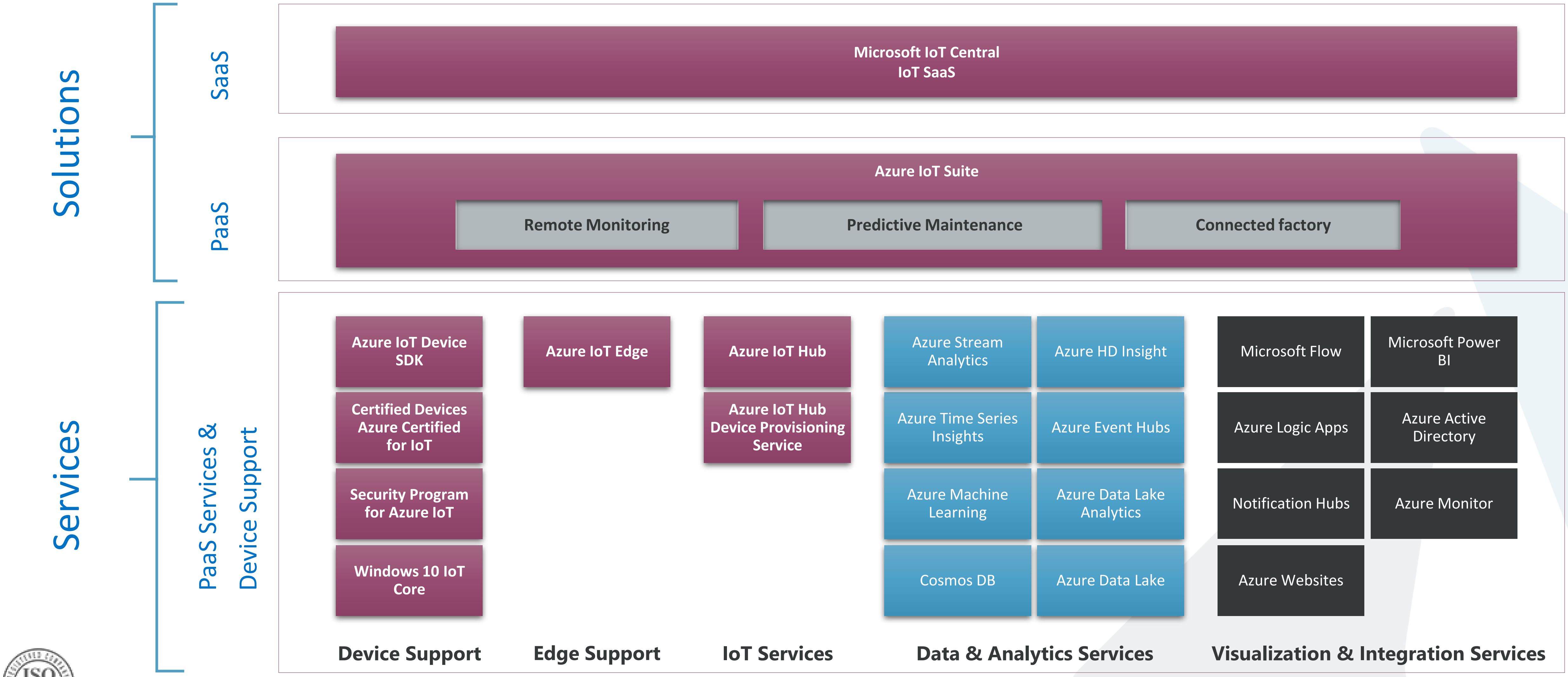
Smart Building Automation

Digital Advertising

Smart Home Automation

ISO 9001 SGS

Comprehensive set of capabilities for IoT solutions



Solutions

SaaS

Microsoft IoT Central
IoT SaaS

PaaS

Azure IoT Suite

- Remote Monitoring
- Predictive Maintenance
- Connected factory

Services

PaaS Services & Device Support

Device Support

- Azure IoT Device SDK
- Certified Devices Azure Certified for IoT
- Security Program for Azure IoT
- Windows 10 IoT Core

Edge Support

- Azure IoT Edge

IoT Services

- Azure IoT Hub
- Azure IoT Hub Device Provisioning Service

Data & Analytics Services

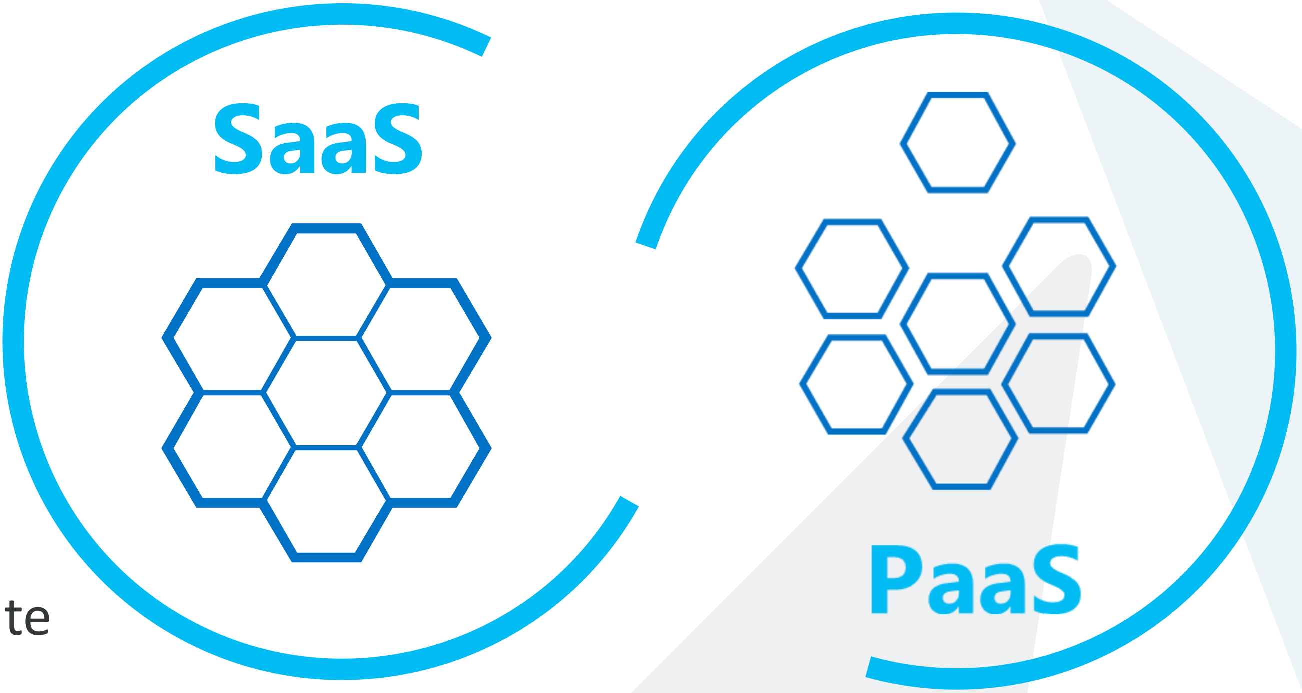
- Azure Stream Analytics
- Azure HD Insight
- Azure Time Series Insights
- Azure Event Hubs
- Azure Machine Learning
- Azure Data Lake Analytics
- Cosmos DB
- Azure Data Lake

Visualization & Integration Services

- Microsoft Flow
- Microsoft Power BI
- Azure Logic Apps
- Azure Active Directory
- Notification Hubs
- Azure Monitor
- Azure Websites

Azure IoT solutions approach

- SaaS – Microsoft IoT Central
  - Fully managed IoT SaaS
  - No cloud solution development expertise required
  - Configurable to your needs
  - Ideal for straightforward IoT needs
- PaaS – Azure IoT Suite
  - Preconfigured solutions
  - Deploy in minutes
  - Accelerate time to value
  - Ideal for solutions that require ultimate control



IoT Central Features

Connectivity Hub & Telemetry ingestion

Connects a variety of devices to the cloud through an open platform

Device management

Enables understanding, control, and optimization of investments

Analytics & dashboards

Provide simple and consumable reports and visualizations for any platform

Rules engine

Real time data processing

Time-series insights

Identify trends among millions of IoT events

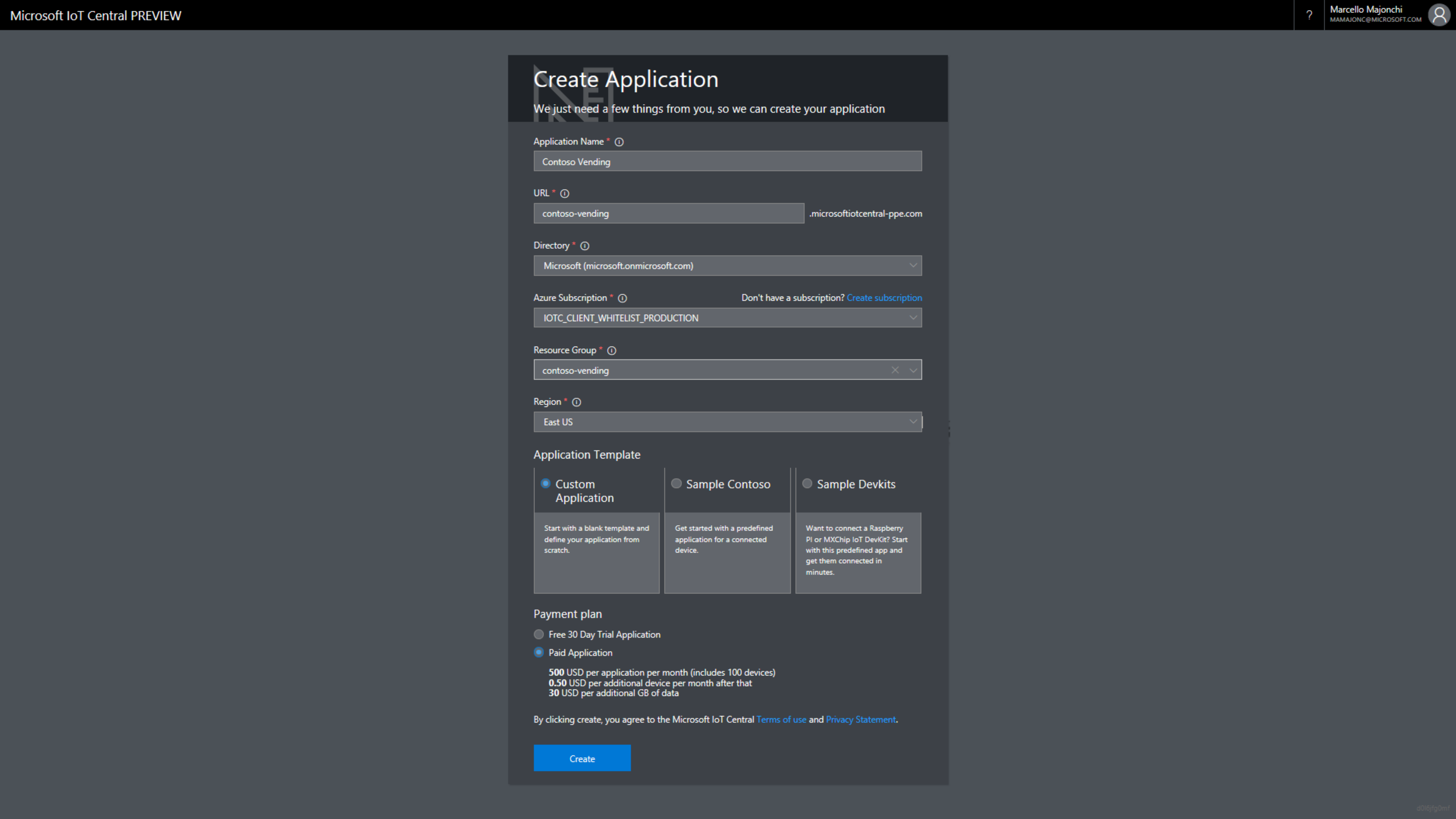
Digital twin management

Enables actionable insights through modeling and simulation

User and identity management

Delivers customized levels of permissions across users and protect from unauthorized access

ISO 9001 SGS



Microsoft IoT Central PREVIEW

?

Marcello Majonchi
MAMAJONC@MICROSOFT.COM

Create Application

We just need a few things from you, so we can create your application

Application Name *
Contoso Vending

URL *
contoso-vending .microsoftiotcentral-ppe.com

Directory *
Microsoft (microsoft.onmicrosoft.com)

Azure Subscription * Don't have a subscription? Create subscription
IOTC_CLIENT_WHITELIST_PRODUCTION

Resource Group *
contoso-vending

Region *
East US

Application Template

● Custom Application
Start with a blank template and define your application from scratch.

○ Sample Contoso
Get started with a predefined application for a connected device.

○ Sample Devkits
Want to connect a Raspberry PI or MXChip IoT DevKit? Start with this predefined app and get them connected in minutes.

Payment plan

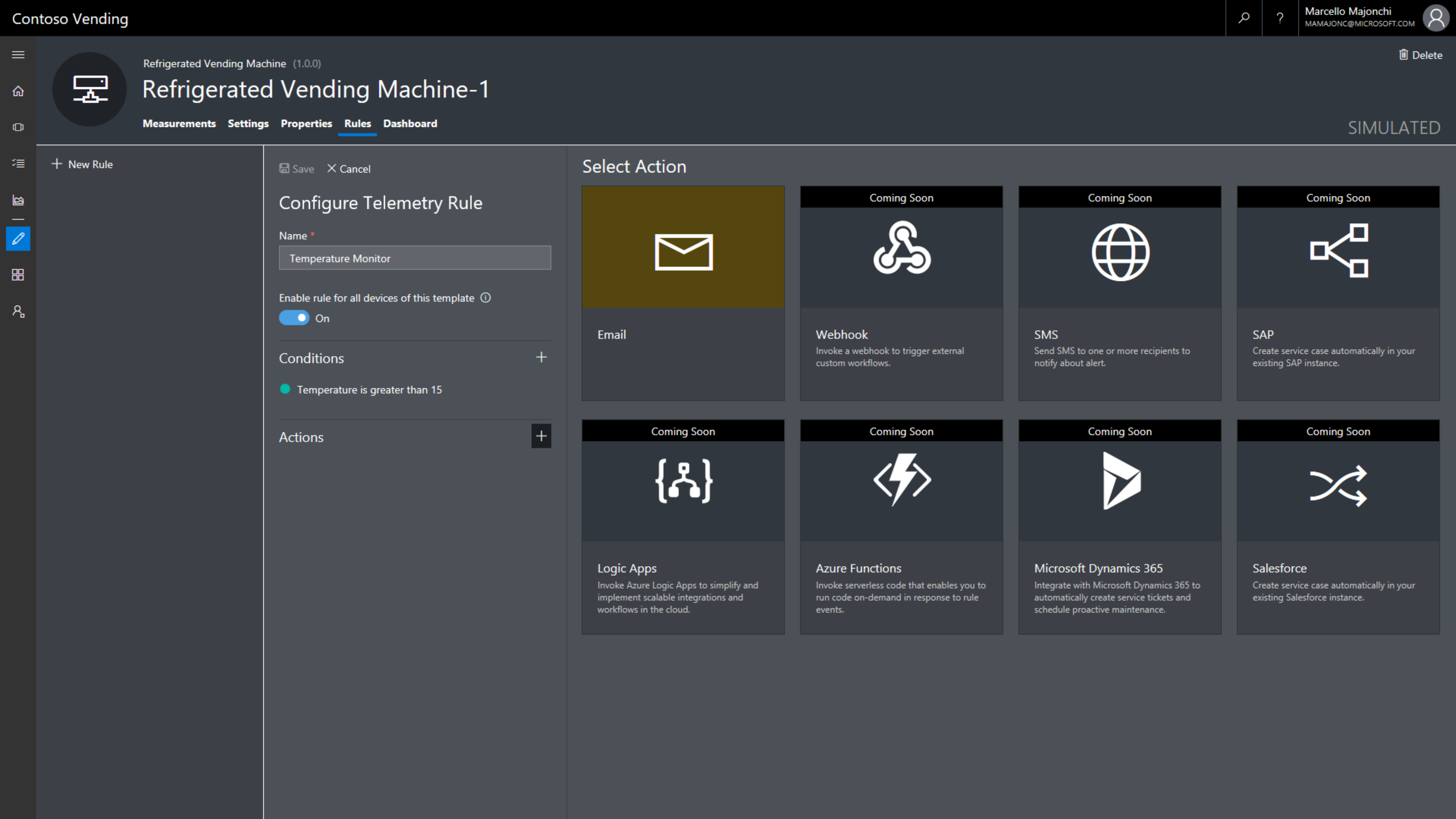
○ Free 30 Day Trial Application

● Paid Application

500 USD per application per month (includes 100 devices)
0.50 USD per additional device per month after that
30 USD per additional GB of data

By clicking create, you agree to the Microsoft IoT Central Terms of use and Privacy Statement.

Create



Refrigerated Vending Machine (1.0.0)

Refrigerated Vending Machine-1

Configure Telemetry Rule

Name *: Temperature Monitor

Enable rule for all devices of this template: On

Conditions

Temperature is greater than 15

Actions

Email

Webhook
Invoke a webhook to trigger external custom workflows.

SMS
Send SMS to one or more recipients to notify about alert.

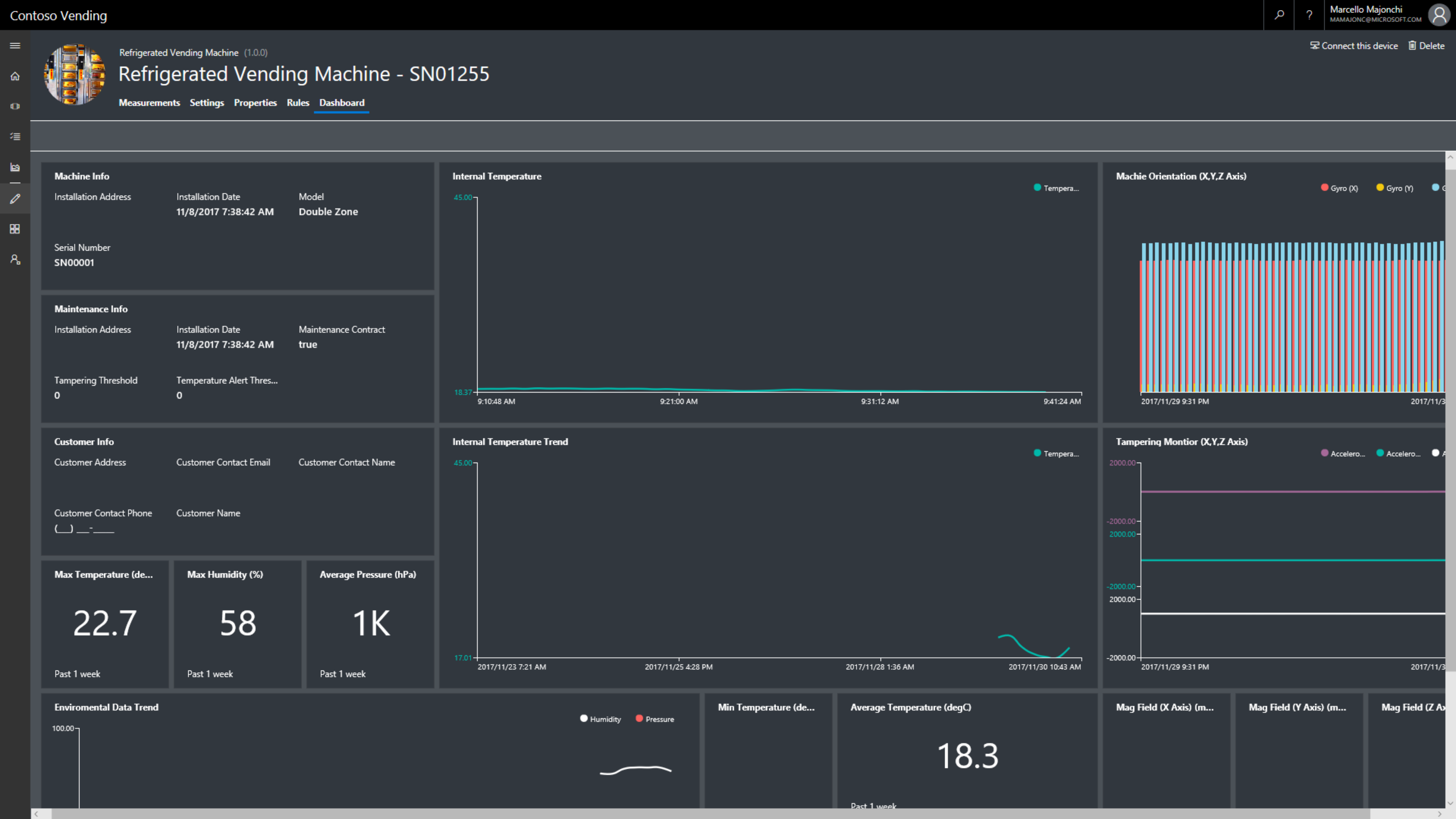
SAP
Create service case automatically in your existing SAP instance.

Logic Apps
Invoke Azure Logic Apps to simplify and implement scalable integrations and workflows in the cloud.

Azure Functions
Invoke serverless code that enables you to run code on-demand in response to rule events.

Microsoft Dynamics 365
Integrate with Microsoft Dynamics 365 to automatically create service tickets and schedule proactive maintenance.

Salesforce
Create service case automatically in your existing Salesforce instance.



Refrigerated Vending Machine (1.0.0)

Refrigerated Vending Machine - SN01255

Machine Info

Installation Address

Installation Date
11/8/2017 7:38:42 AM

Model
Double Zone

Serial Number
SN00001

Maintenance Info

Installation Address

Installation Date
11/8/2017 7:38:42 AM

Maintenance Contract
true

Tampering Threshold
0

Temperature Alert Thres...
0

Customer Info

Customer Address

Customer Contact Email

Customer Contact Name

Customer Contact Phone
(___) ___-____

Customer Name

Max Temperature (de...
22.7
Past 1 week

Max Humidity (%)
58
Past 1 week

Average Pressure (hPa)
1K
Past 1 week

Internal Temperature

Machie Orientation (X,Y,Z Axis)

Internal Temperature Trend

Tampering Montior (X,Y,Z Axis)

Enviromental Data Trend

Min Temperature (de...

Average Temperature (degC)
18.3
Past 1 week

Mag Field (X Axis) (m...

Mag Field (Y Axis) (m...

Mag Field (Z Ax

Microsoft IoT Central - Simplified predictable pricing

FREE

Trial for 30 days

Includes 10 devices
and 100MB data traffic

$0.50 USD

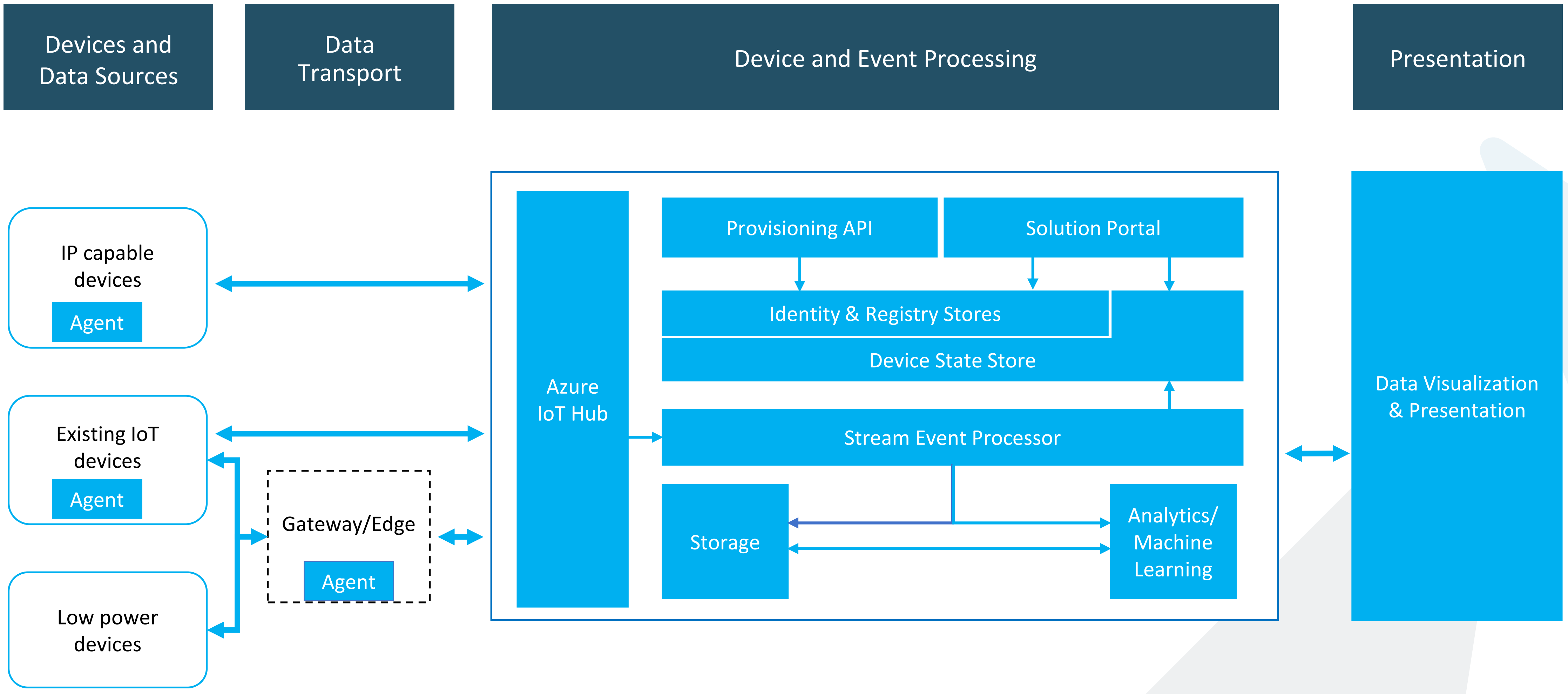
Per device, per month

$500 fixed fee per month *includes first 100 devices and 1000 MB data traffic*

Additional devices $0.50 USD includes 10 MB data traffic

Additional data traffic $30 per 1 GB

Azure IoT Services Reference Architecture



Devices and Data Sources

Data Transport

Device and Event Processing

Presentation

IP capable devices

Agent

Existing IoT devices

Agent

Low power devices

Gateway/Edge

Agent

Azure IoT Hub

Provisioning API

Solution Portal

Identity & Registry Stores

Device State Store

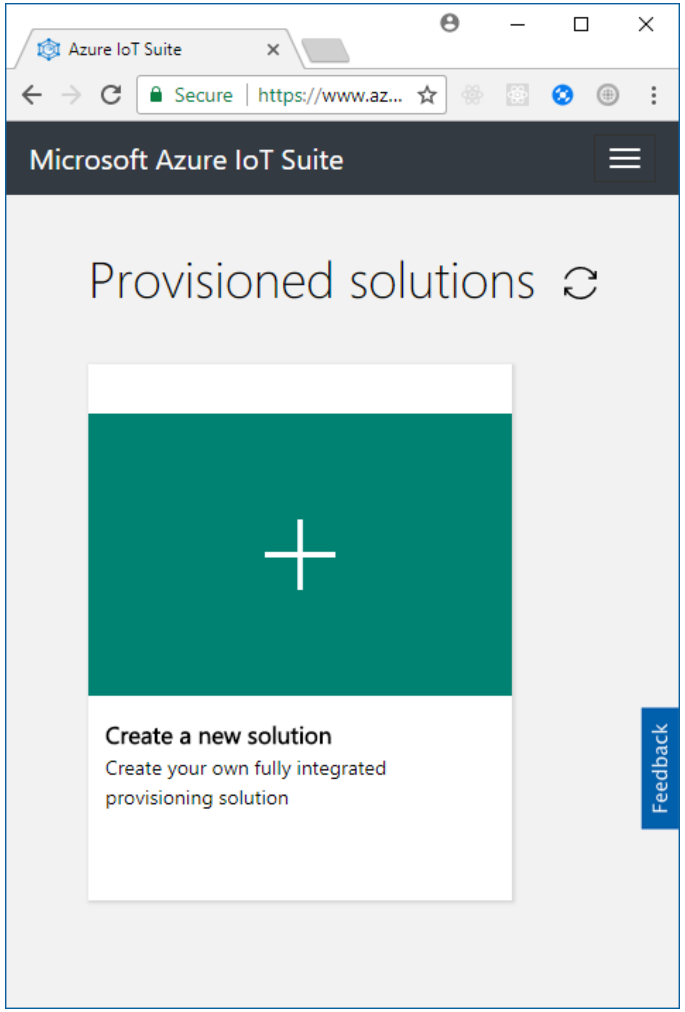
Stream Event Processor

Storage

Analytics/ Machine Learning

Data Visualization & Presentation

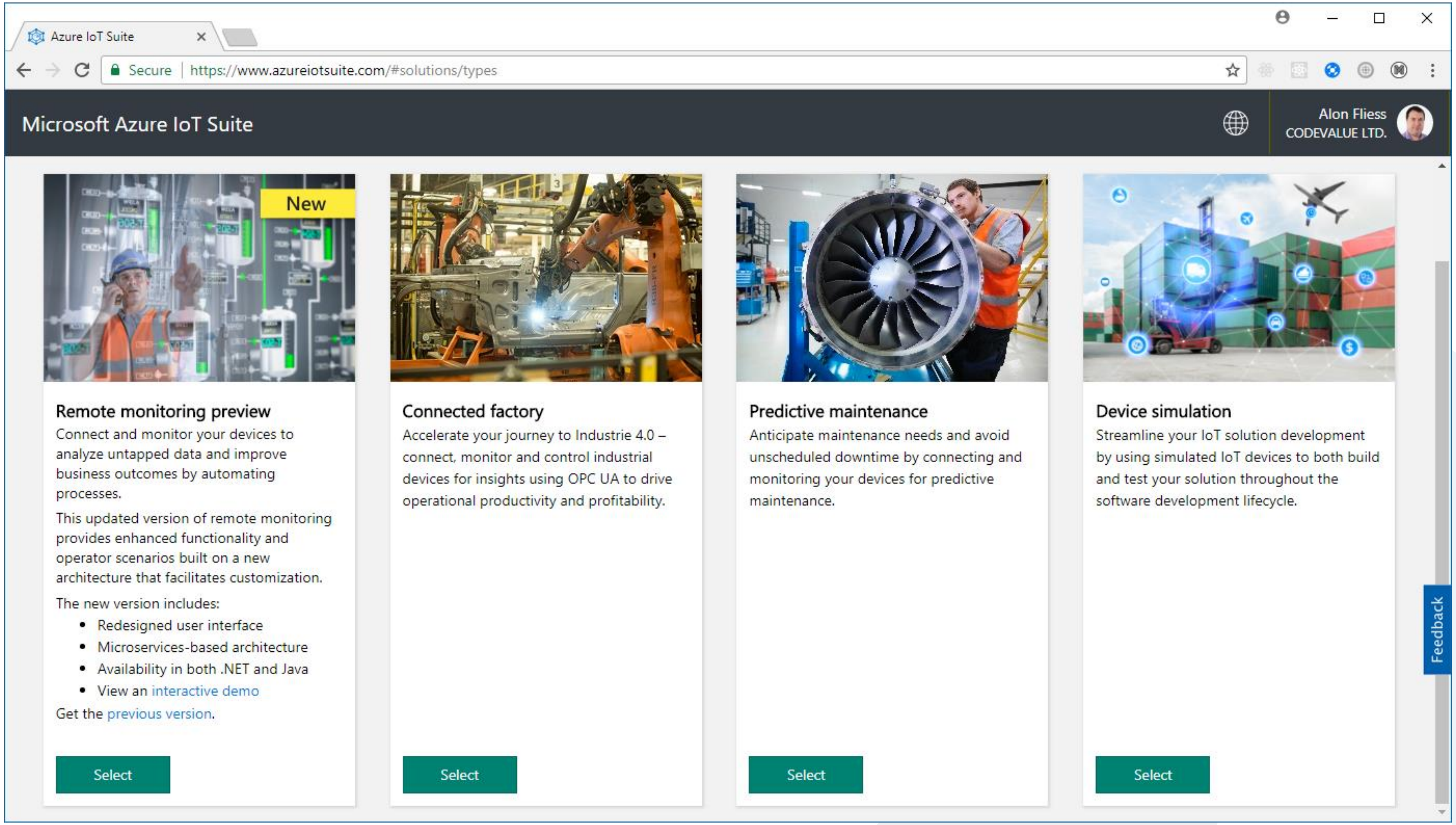
Azure IoT Suite



Provisioned solutions

Create a new solution

Create your own fully integrated provisioning solution



Remote monitoring preview

Connect and monitor your devices to analyze untapped data and improve business outcomes by automating processes.

This updated version of remote monitoring provides enhanced functionality and operator scenarios built on a new architecture that facilitates customization.

The new version includes:

- Redesigned user interface
- Microservices-based architecture
- Availability in both .NET and Java
- View an interactive demo

Get the previous version.

Connected factory

Accelerate your journey to Industrie 4.0 – connect, monitor and control industrial devices for insights using OPC UA to drive operational productivity and profitability.

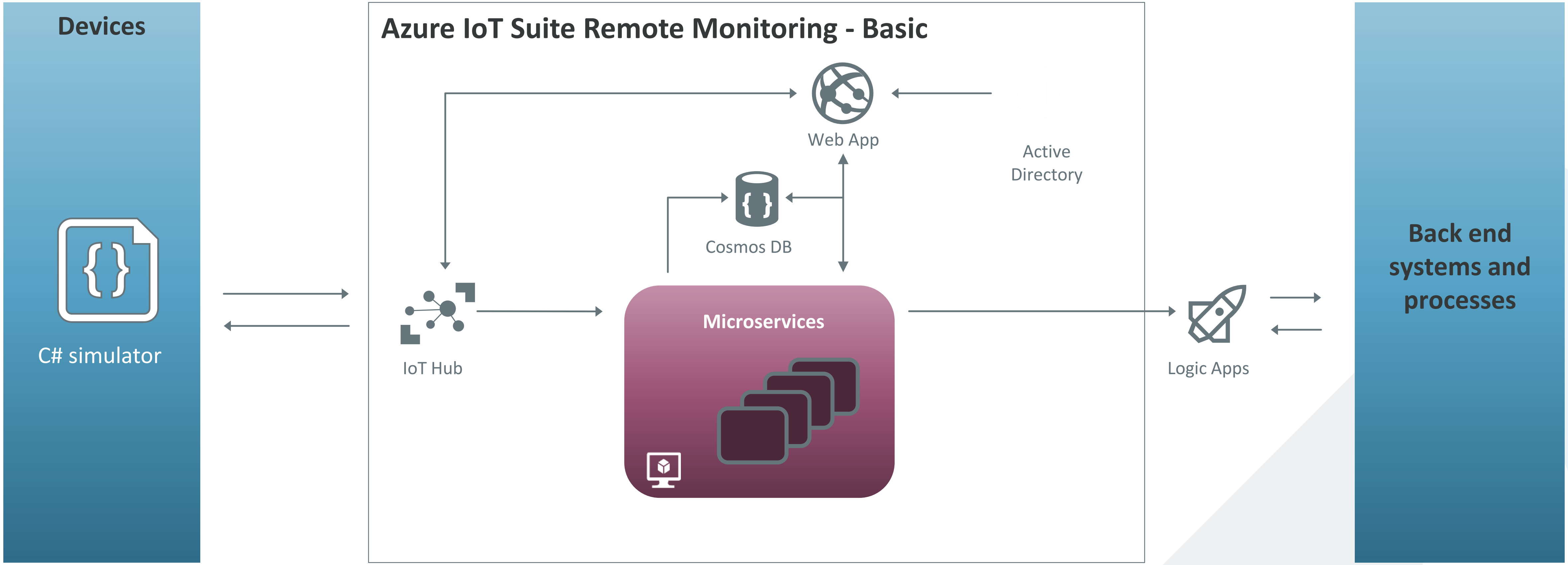
Predictive maintenance

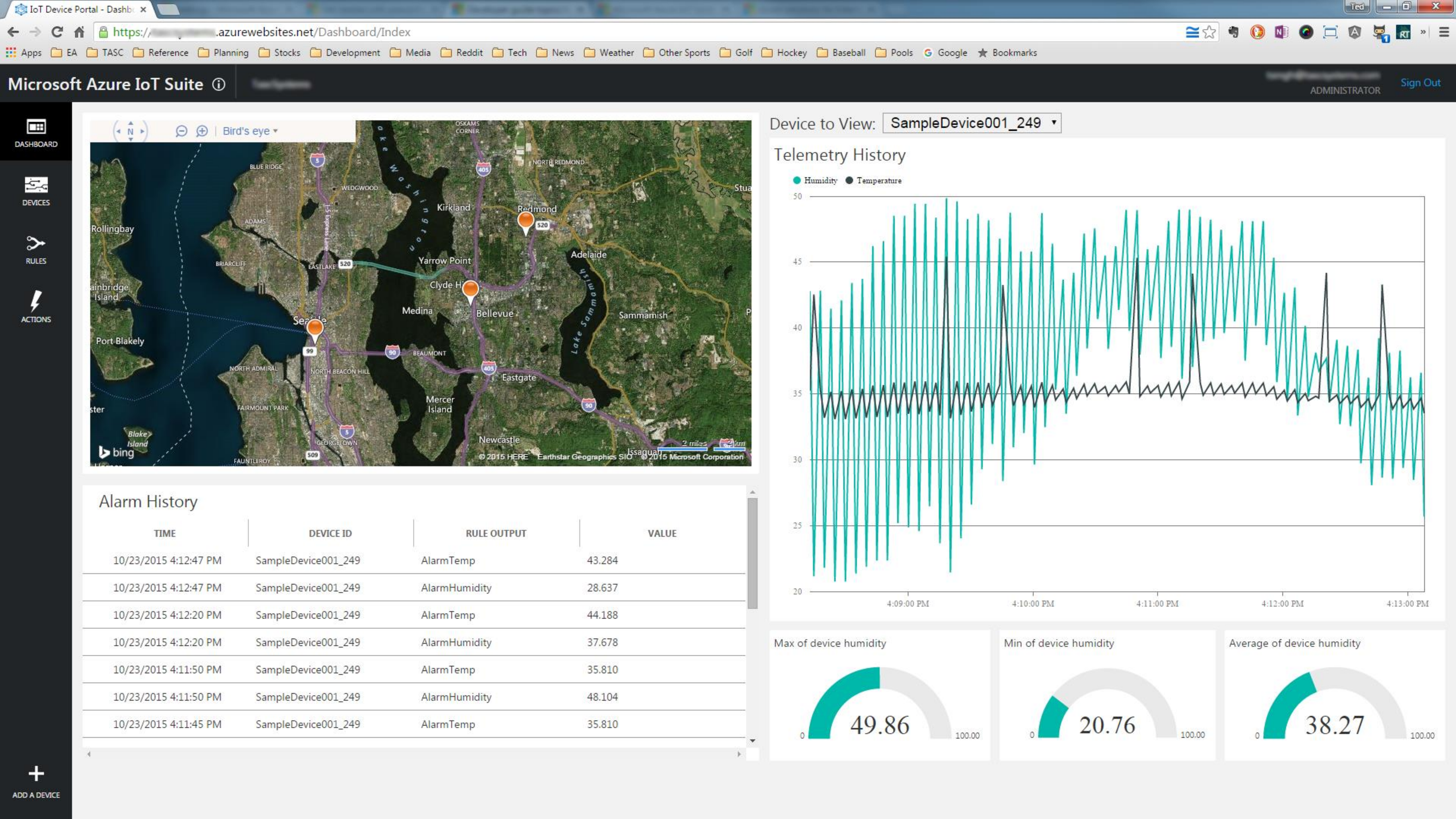
Anticipate maintenance needs and avoid unscheduled downtime by connecting and monitoring your devices for predictive maintenance.

Device simulation

Streamline your IoT solution development by using simulated IoT devices to both build and test your solution throughout the software development lifecycle.

Azure IoT Suite solution – PaaS (almost) like a SaaS





Device to View: SampleDevice001_249

Telemetry History

Alarm History

TIME	DEVICE ID	RULE OUTPUT	VALUE
10/23/2015 4:12:47 PM	SampleDevice001_249	AlarmTemp	43.284
10/23/2015 4:12:47 PM	SampleDevice001_249	AlarmHumidity	28.637
10/23/2015 4:12:20 PM	SampleDevice001_249	AlarmTemp	44.188
10/23/2015 4:12:20 PM	SampleDevice001_249	AlarmHumidity	37.678
10/23/2015 4:11:50 PM	SampleDevice001_249	AlarmTemp	35.810
10/23/2015 4:11:50 PM	SampleDevice001_249	AlarmHumidity	48.104
10/23/2015 4:11:45 PM	SampleDevice001_249	AlarmTemp	35.810

Max of device humidity 49.86

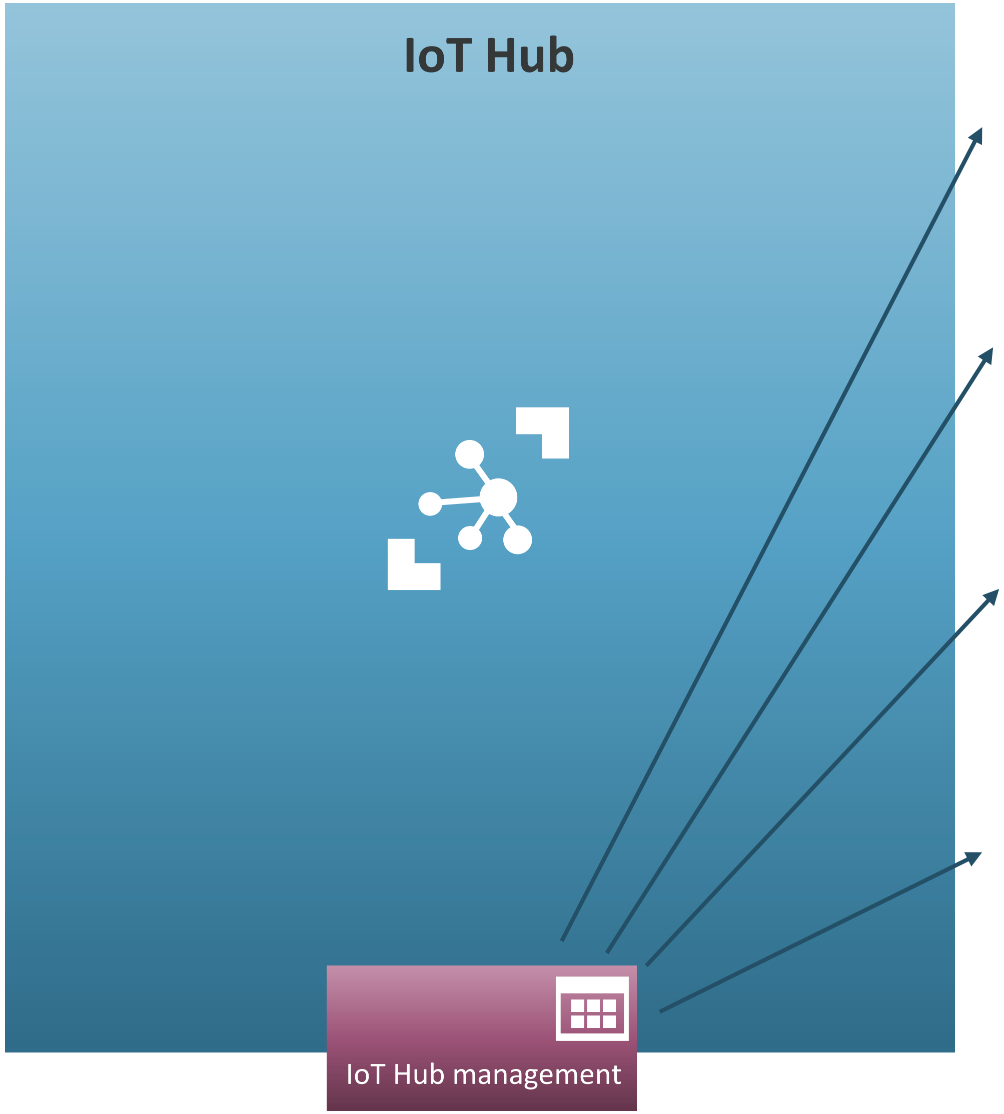
Min of device humidity 20.76

Average of device humidity 38.27

Introducing Microsoft Azure IoT Hub

- IoT Hub is available as a stand-alone service or as one of the services used in the new Azure IoT Suite
- Azure IoT Hub is designed to connect your devices to Azure. It supports:
  - Millions of simultaneously connected devices
  - Per-device authentication
  - High throughput data ingestion
  - Scale device management
  - Reliable command and control

Pick your favorite to create a hub



Azure Portal

https://portal.azure.com

ARM template

https://azure.microsoft.com/en-us/resources/templates/

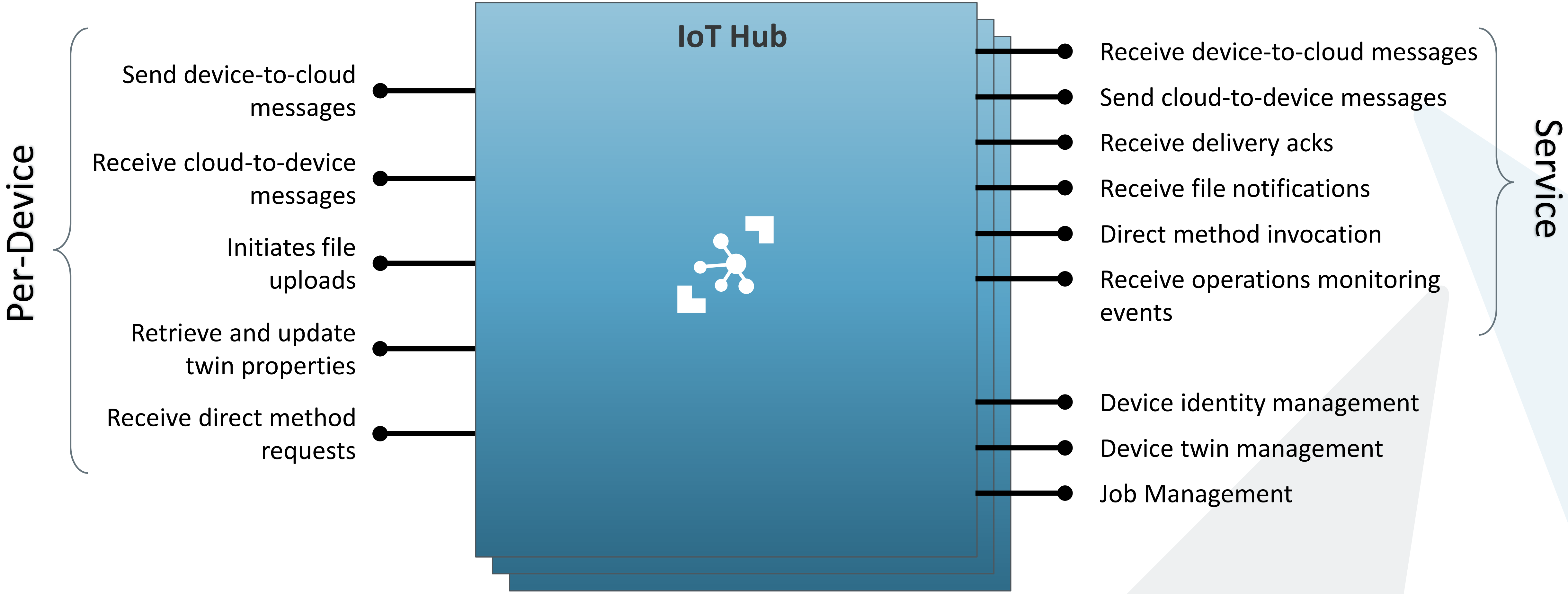
Azure CLI

https://github.com/Azure/azure-cli (v2.0 Python)

Powershell

https://docs.microsoft.com/en-us/powershell/azureps-cmdlets-docs/

Azure IoT Hub



IoT Hub

Per-Device

Send device-to-cloud messages

Receive cloud-to-device messages

Initiates file uploads

Retrieve and update twin properties

Receive direct method requests

Service

Receive device-to-cloud messages

Send cloud-to-device messages

Receive delivery acks

Receive file notifications

Direct method invocation

Receive operations monitoring events

Device identity management

Device twin management

Job Management

Azure IoT Hub SDKs

- **Device-facing**
  - For devices and field gateways
- **Platforms**
  - Many devices
  - RTOS (FreeRTOS)
  - Linux
    (Ubuntu, Debian, Fedora, Raspbian, Angstrom)
  - Windows 7/8/10
  - ARM mbed
  - Android
  - iOS
- **Device SKD by programming language**
  - For device side development
  - Azure IoT device SDK for C
  - Azure IoT device SDK for .NET
  - Azure IoT device SDK for Java
  - Azure IoT device SDK for Node.js
  - Azure IoT device SDK for Python

- **Service-facing SDK by programming language**
  - For back-ends and cloud gateways
  - Azure IoT service SDK for .NET
  - Azure IoT service SDK for Node.js
  - Azure IoT service SDK for Java
  - Azure IoT service SDK for Python
- **Azure IoT Gateway SDK**
  - Infrastructure and modules to create IoT gateway solutions
- **Azure IoT Hub REST API**
  - For all the rest…
- **Advance IoT Hub topics**
  - IoT Hub endpoints
  - IoT Hub query language for device twins and jobs
  - Quotas and throttling
  - IoT Hub MQTT support

C Language Device SDK

- Many low price, low energy, SoC can be developed only by using the C language
- The IoT team has built a full-blown C SDK to connect and communicate with the IoT Hub
  - It supports all IoT Hub Device capabilities, including:
    - Secure connection and communication using three protocols (HTTP, AMQP, MQTT)
    - Sending telemetry messages using JSON serialization and set of macros to provide message serialization
    - Receiving messages from the cloud
    - Handling device twin synchronization
    - Invoke a function with request-reply message exchange pattern when the IoT Hub calls
    - Upload files
- There are two levels of functions:
  - With *_LL_* - low level API – for device that has no threading capabilities
  - With no *_LL_* - support background message processing using threads
- Follow this intro to understand the various functions

Connecting and Defining A Model (C SDK)

```
 if ((iotHubClientHandle = IoTHubClient_LL_CreateFromConnectionString(connectionString, MQTT_Protocol)) ==
 {
     (void)printf("ERROR: iotHubClientHandle is NULL!\r\n");
 }
 else
 {


 BEGIN_NAMESPACE(WeatherStation);

 DECLARE_MODEL(ContosoAnemometer,
 WITH_DATA(ascii_char_ptr, DeviceId),
 WITH_DATA(int, WindSpeed),
 WITH_ACTION(TurnFanOn),
 WITH_ACTION(TurnFanOff),
 WITH_ACTION(SetAirResistance, int, Position)
 );

 END_NAMESPACE(WeatherStation);

ContosoAnemometer* myWeather = CREATE_MODEL_INSTANCE(WeatherStation, ContosoAnemometer);
if (myWeather == NULL)
{
    (void)printf("Failed on CREATE_MODEL_INSTANCE\r\n");
}
else
{
```

IoT Hub and IoT Device Communication Protocols

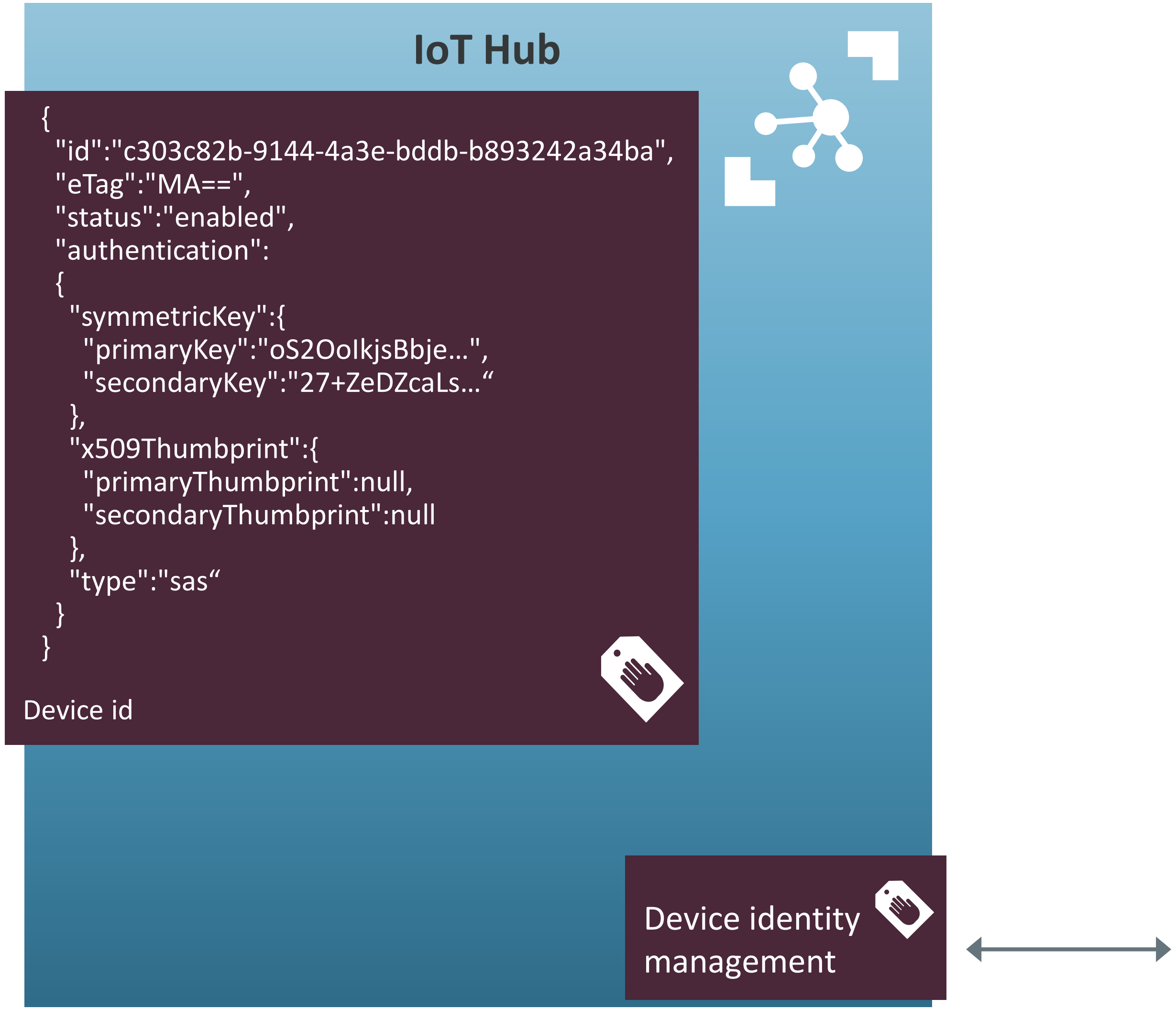
- IoT Hub supports three protocols:
  - HTTP – Use for devices that cannot support other protocols or that are rarely connected
  - AMQP – Use on field and cloud gateways to take advantage of connection multiplexing across devices
  - MQTT – Extremely lightweight, Use on all devices that do not require to connect multiple devices
- You can choose to use any protocol, however you need to take some protocol characteristics into considerations:
  - HTTP does not have an efficient way to implement server push
    - As such, when you are using HTTP, devices poll IoT Hub for cloud-to-device messages
  - AMQP returns errors for many conditions, while MQTT terminates the connection
    - As a result your exception handling logic might require some changes
  - MQTT does not support the *reject* operations when receiving cloud-to-device messages
    - If your back-end app needs to receive a response from the device app, consider using direct methods
  - The MQTT and HTTP libraries have a smaller footprint than the AMQP libraries

Supported Protocols & Port Numbers

Protocol	Port
MQTT	8883
MQTT over WebSockets	443
AMQP	5671
AMQP over WebSockets	443
HTTP	443

Device registry

- Unique id for each device
- Unique credentials for authentication
  - Private Key/SAS Token
  - X.509 Certificates
- Device Twin

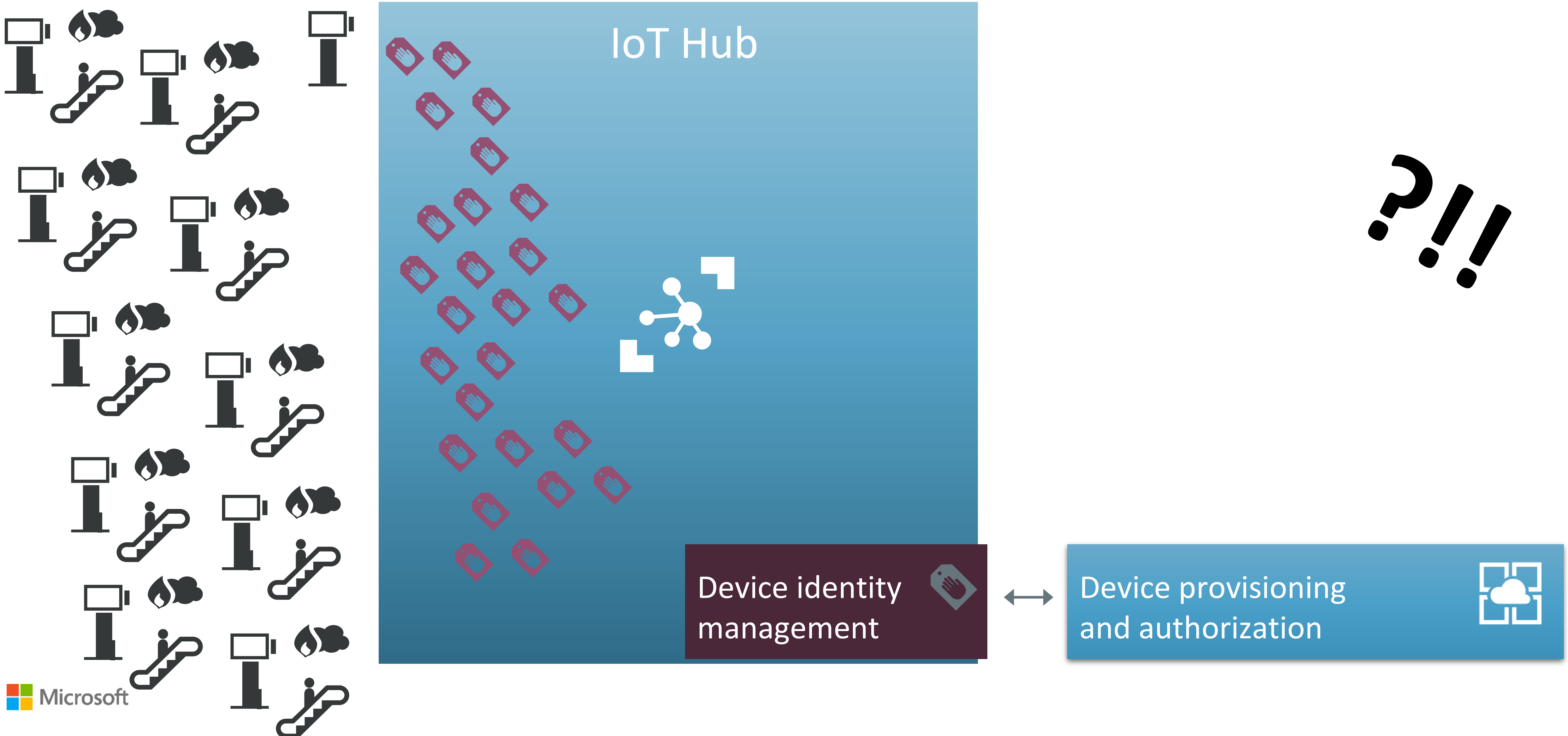


```
{
  "id":"c303c82b-9144-4a3e-bddb-b893242a34ba",
  "eTag":"MA==",
  "status":"enabled",
  "authentication":
  {
    "symmetricKey":{
      "primaryKey":"oS2OoIkjsBbje…",
      "secondaryKey":"27+ZeDZcaLs…“
    },
    "x509Thumbprint":{
      "primaryThumbprint":null,
      "secondaryThumbprint":null
    },
    "type":"sas“
  }
}
```

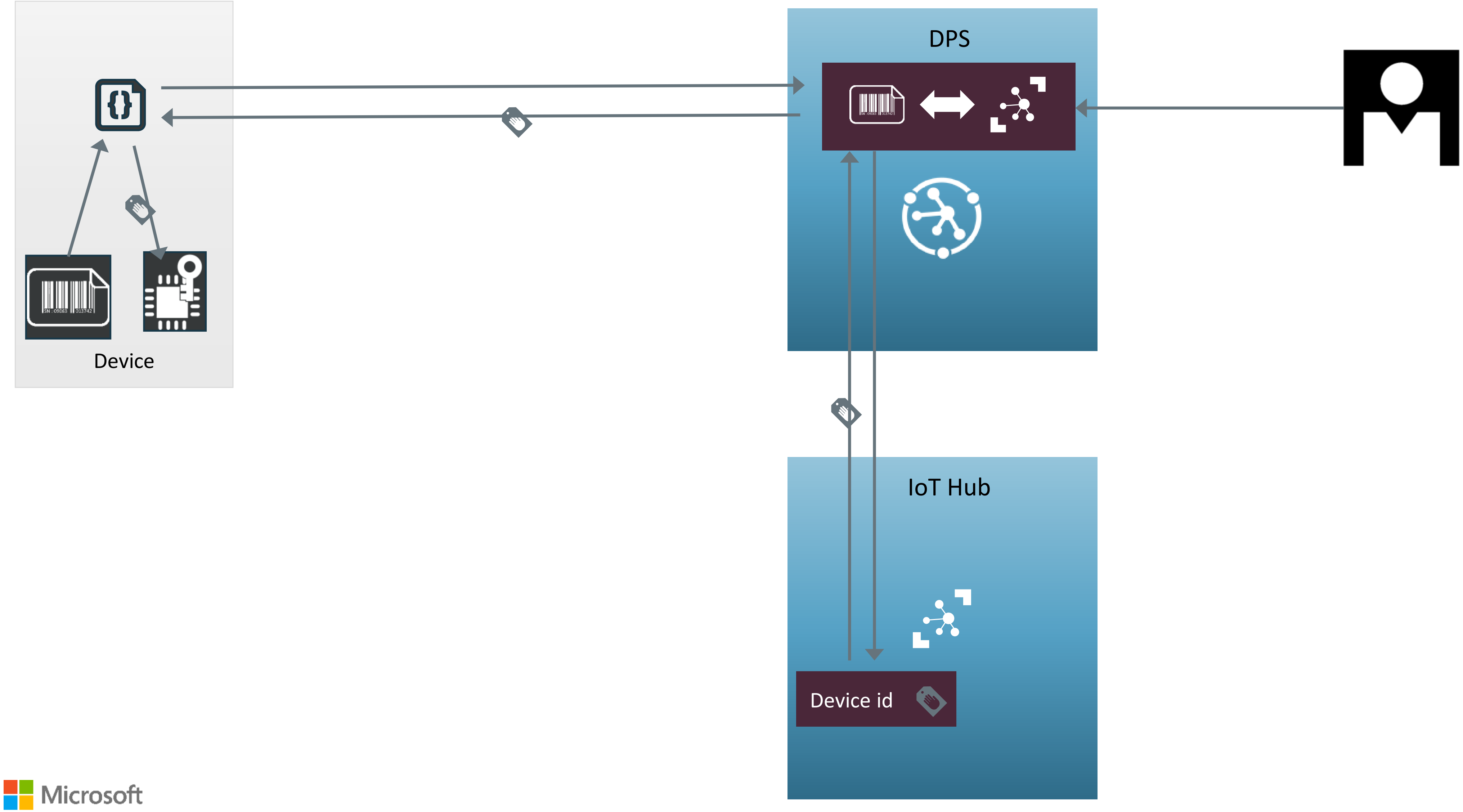
- Azure Portal
- Development tools
  - Azure CLI
  - VSCode extension
- Using a client SDK
  - .Net
  - Node
  - Java
  - Python

Device provisioning and authorization

Provisioning at scale


IoT Hub
Device identity management
Device provisioning and authorization
?!!

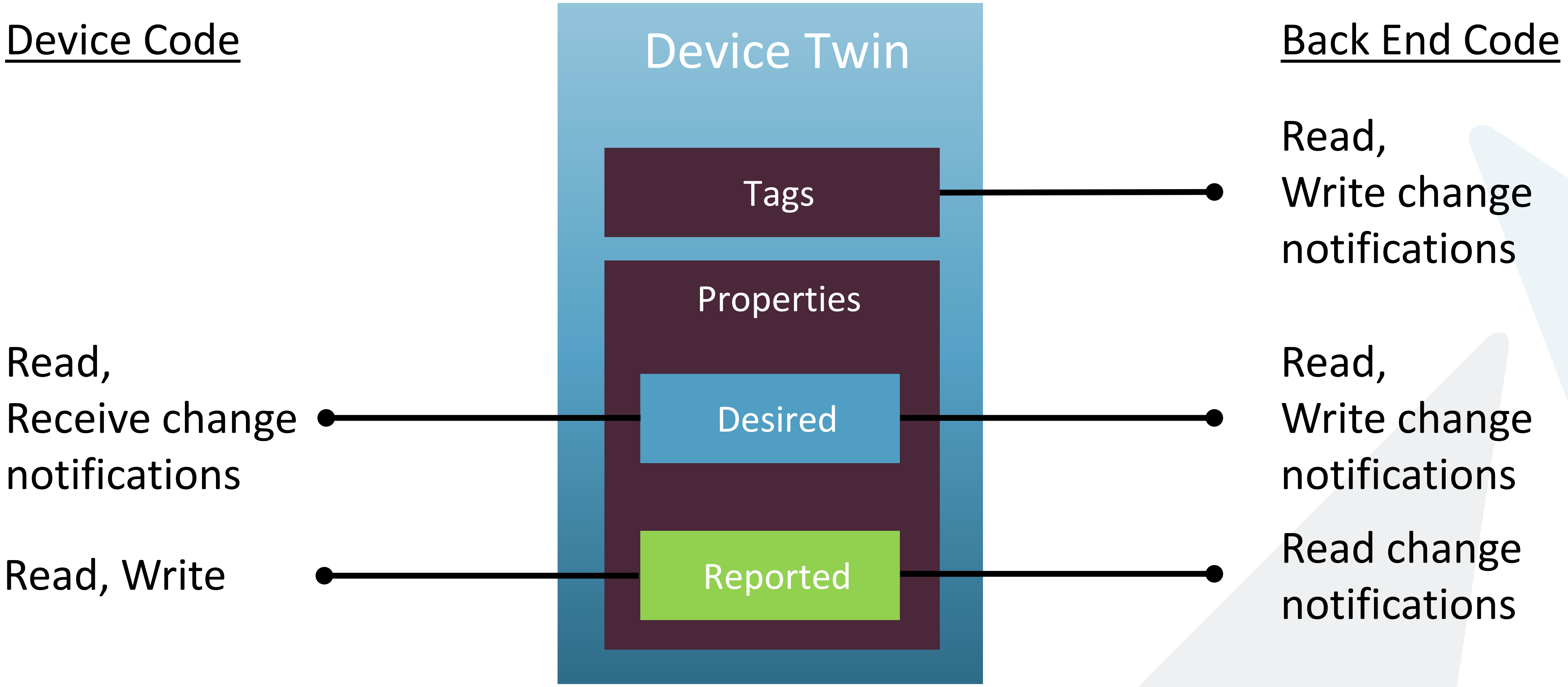
Device Provisioning Service



Device Twins

- Device twins are JSON documents that store device state information:
  - metadata, configurations, and conditions
- The IoT Hub persists a device twin for each registered device
- Use device twins to:
  - Store device-specific metadata in the cloud
  - Report current state information such as available capabilities and conditions from your device app
  - Synchronize the state of long-running workflows between device app and cloud app
  - Query your device metadata, configuration, or state
  - Get notified when a twin is modified

Device Twin



Device Code

Back End Code

Device Twin

Tags

Properties

Desired

Reported

Read,
Write change
notifications

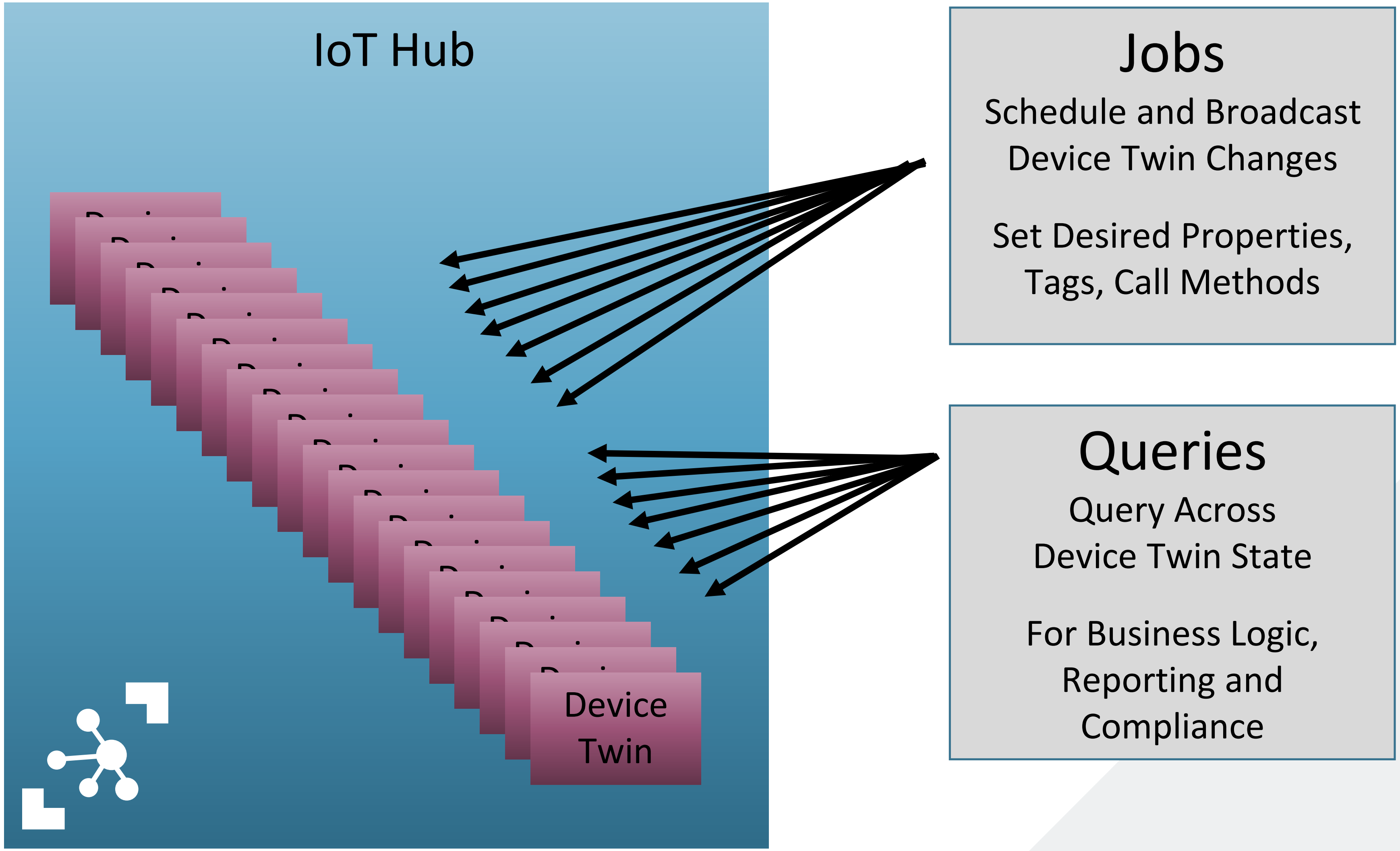
Read,
Receive change
notifications

Read,
Write change
notifications

Read, Write

Read change
notifications

Designed for IoT Scale



Jobs

Schedule and Broadcast
Device Twin Changes

Set Desired Properties,
Tags, Call Methods

Queries

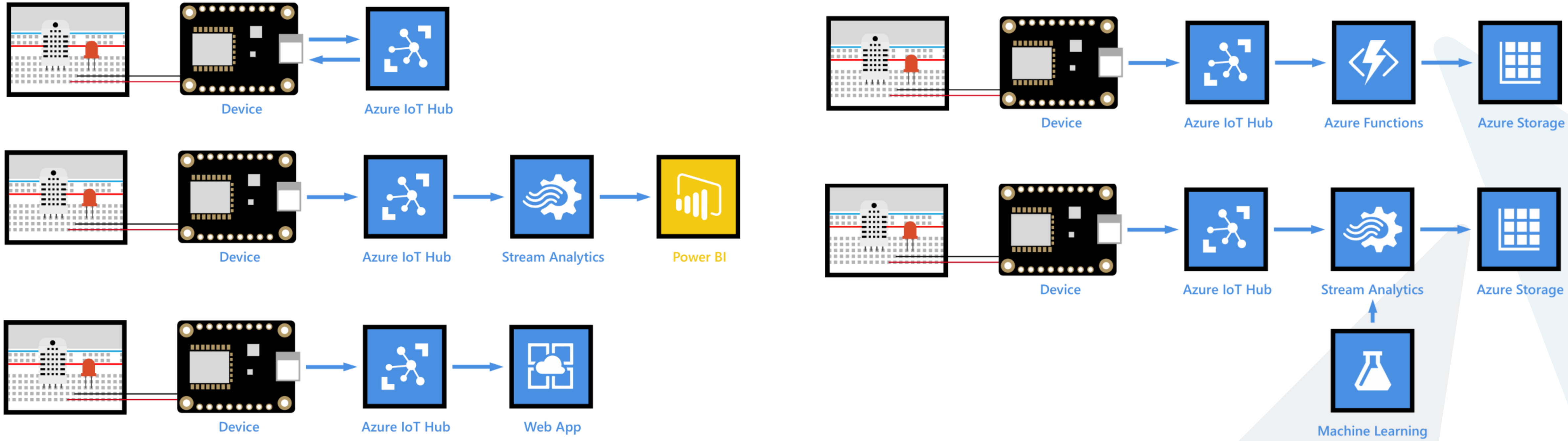
Query Across
Device Twin State

For Business Logic,
Reporting and
Compliance

Device Jobs

- To handle massive amount of devices and to communicate with offline devices, use Jobs:
  - Jobs encapsulate the execution of device twin updates and direct methods against a set of devices at a schedule time
  - The job is described as a JSON document
- Jobs are initiated by the cloud app and maintained by IoT Hub
  - Once a job is initiated, querying for jobs enables the cloud app to refresh the status of running jobs
- More information

What’s next now that I have data flowing in?



Device ⇄ Azure IoT Hub

Device → Azure IoT Hub → Stream Analytics → Power BI

Device → Azure IoT Hub → Web App

Device → Azure IoT Hub → Azure Functions → Azure Storage

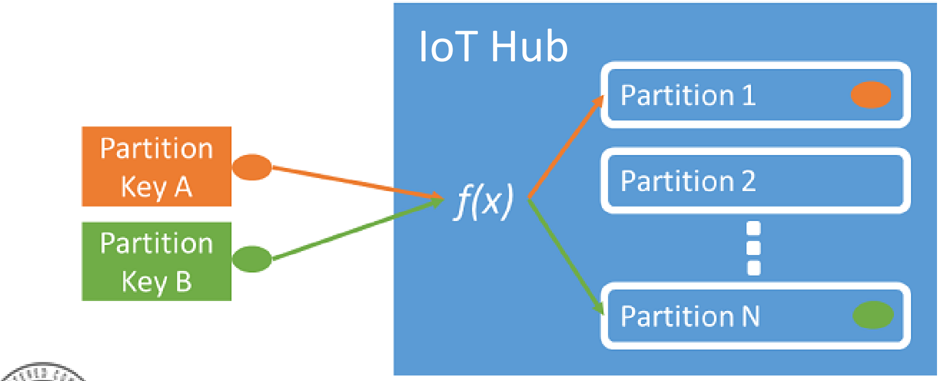
Device → Azure IoT Hub → Stream Analytics → Azure Storage

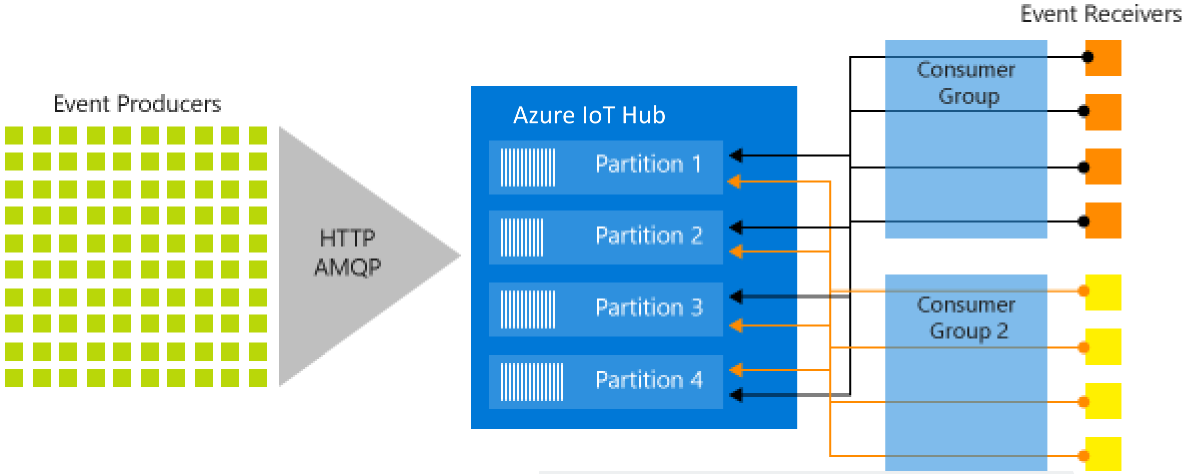
Machine Learning → Stream Analytics

Learn more: https://aka.ms/azureiotgetstarted

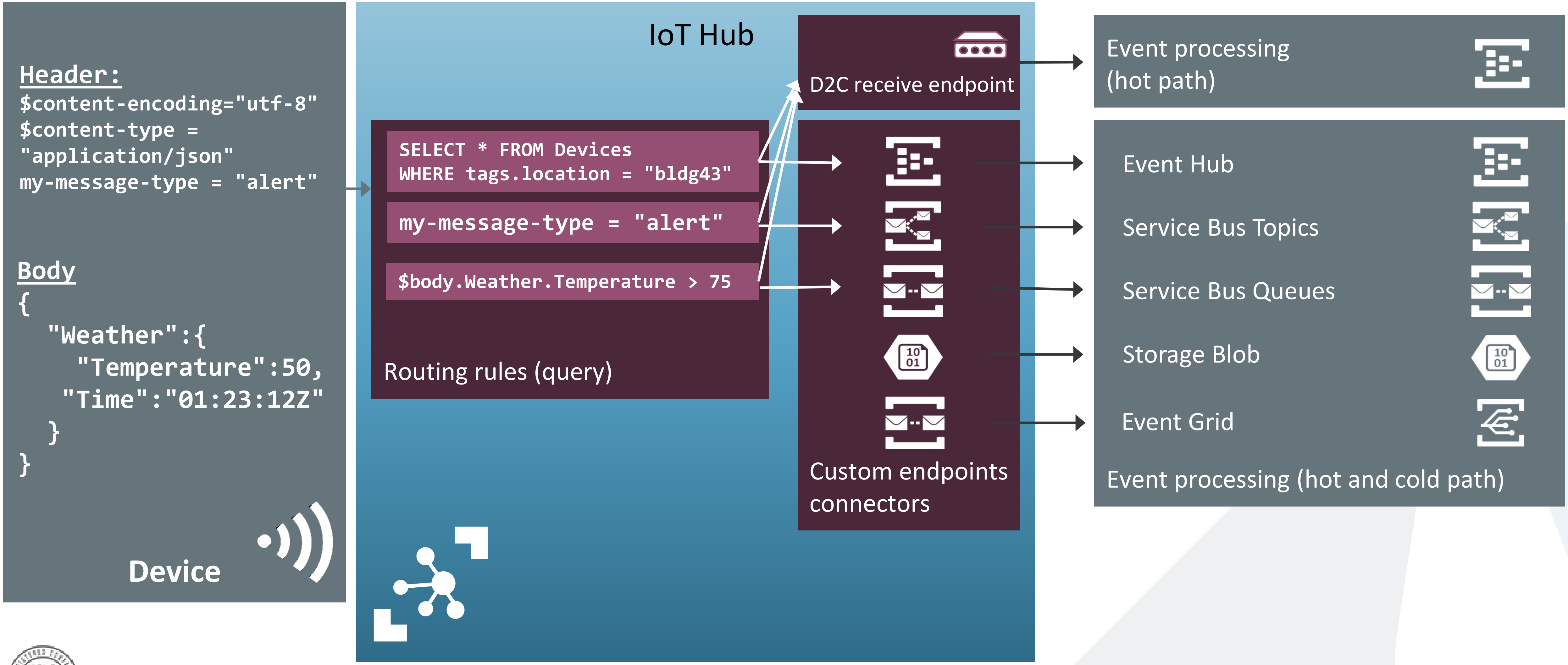
Processing IoT Hub Messages – Event Hub

- You can process IoT Hub device to cloud messages using either:
  - The built-in Event-Hub compatible endpoint
  - Rout the events to an Azure Service Bus queue
- Azure Event Hub is a very powerful telemetry ingestion service that was created by the Service Bus team
  - The key to scale for Event Hubs is the idea of **partitioned consumers**
    - **Partitioned consumers** enables very high scale by removing the contention bottleneck and facilitating end to end parallelism





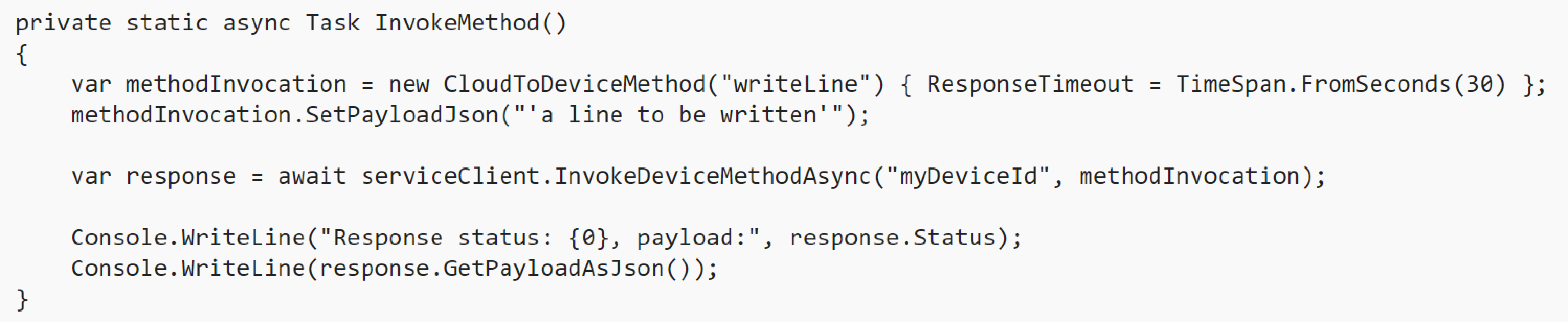
Routing telemetry data



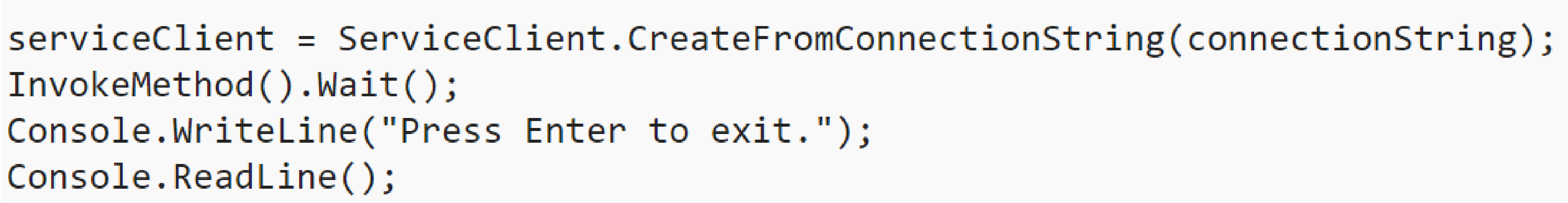
Direct Method - Calling a function in the device

- As opposed to other IoT Hub message exchange patterns that are one-way, a method call is a request-reply interaction
  - Other cloud to device communication are based on sending messages to the device, or setting desired properties
- Each device method targets a single device
  - Jobs provide a way to invoke direct methods on multiple devices, and schedule method invocation for disconnected devices
- Direct methods are synchronous and either succeed or fail
  - Failure occurs after a timeout period (default: 30 secs, settable up to 1 Hour)
- Great for interactive scenarios such as turning on a light from a phone
- Direct method are HTTP-only from the cloud side, and MQTT-only from the device side
- The payload for method requests and responses is a JSON document up to 8KB

Direct Method

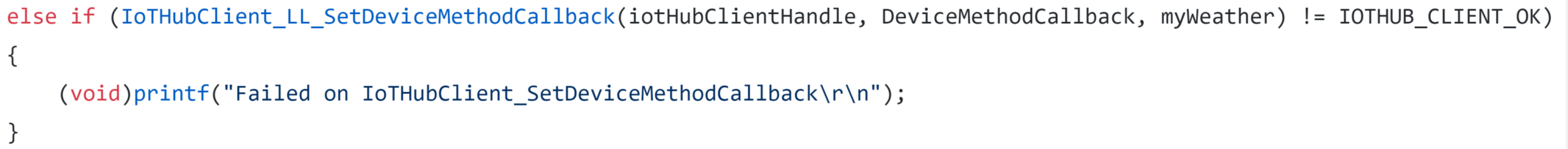


```
private static async Task InvokeMethod()
{
    var methodInvocation = new CloudToDeviceMethod("writeLine") { ResponseTimeout = TimeSpan.FromSeconds(30) };
    methodInvocation.SetPayloadJson("'a line to be written'");

    var response = await serviceClient.InvokeDeviceMethodAsync("myDeviceId", methodInvocation);

    Console.WriteLine("Response status: {0}, payload:", response.Status);
    Console.WriteLine(response.GetPayloadAsJson());
}
```



```
serviceClient = ServiceClient.CreateFromConnectionString(connectionString);
InvokeMethod().Wait();
Console.WriteLine("Press Enter to exit.");
Console.ReadLine();
```

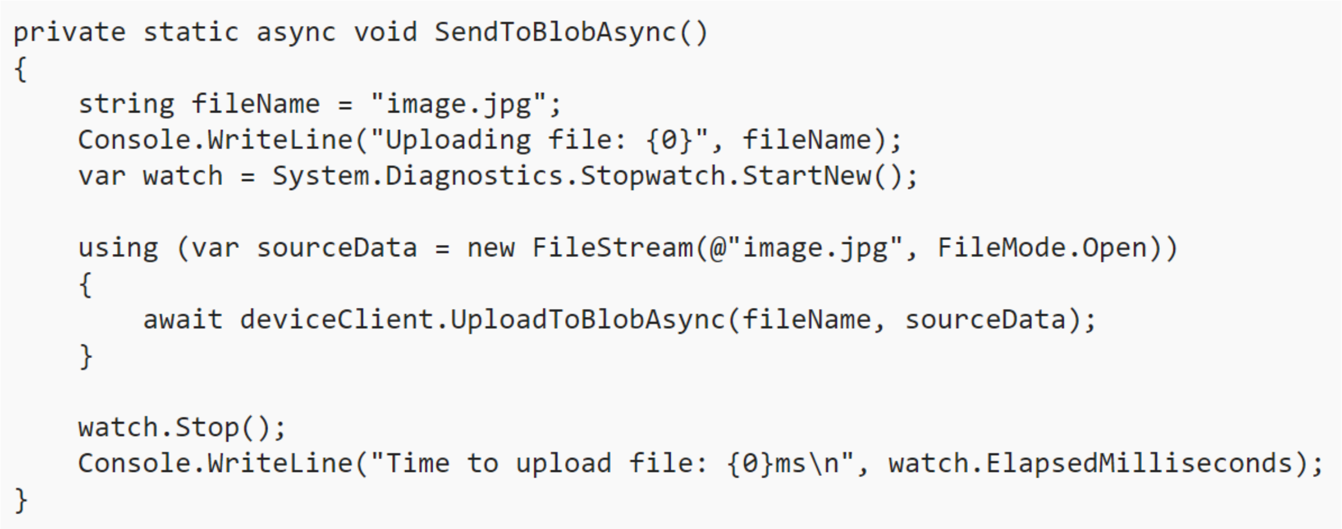
Device Side C SDK – Handling direct method



```
else if (IoTHubClient_LL_SetDeviceMethodCallback(iotHubClientHandle, DeviceMethodCallback, myWeather) != IOTHUB_CLIENT_OK)
{
    (void)printf("Failed on IoTHubClient_SetDeviceMethodCallback\r\n");
}
```

Upload Files

- Use file upload to send media files and large telemetry batches
- You must first link an Azure Storage account to the IoT Hub
  - You can do that using the portal
- The device initiates an upload
- When the upload completes, the device notifies the IoT hub
- See file upload notifications
- The SDK makes it easy: (C#)



```
private static async void SendToBlobAsync()
{
    string fileName = "image.jpg";
    Console.WriteLine("Uploading file: {0}", fileName);
    var watch = System.Diagnostics.Stopwatch.StartNew();

    using (var sourceData = new FileStream(@"image.jpg", FileMode.Open))
    {
        await deviceClient.UploadToBlobAsync(fileName, sourceData);
    }

    watch.Stop();
    Console.WriteLine("Time to upload file: {0}ms\n", watch.ElapsedMilliseconds);
}
```

Waves of Innovation

The smart cloud & Intelligent Edge

Cloud

Globally available, unlimited compute resources

IoT

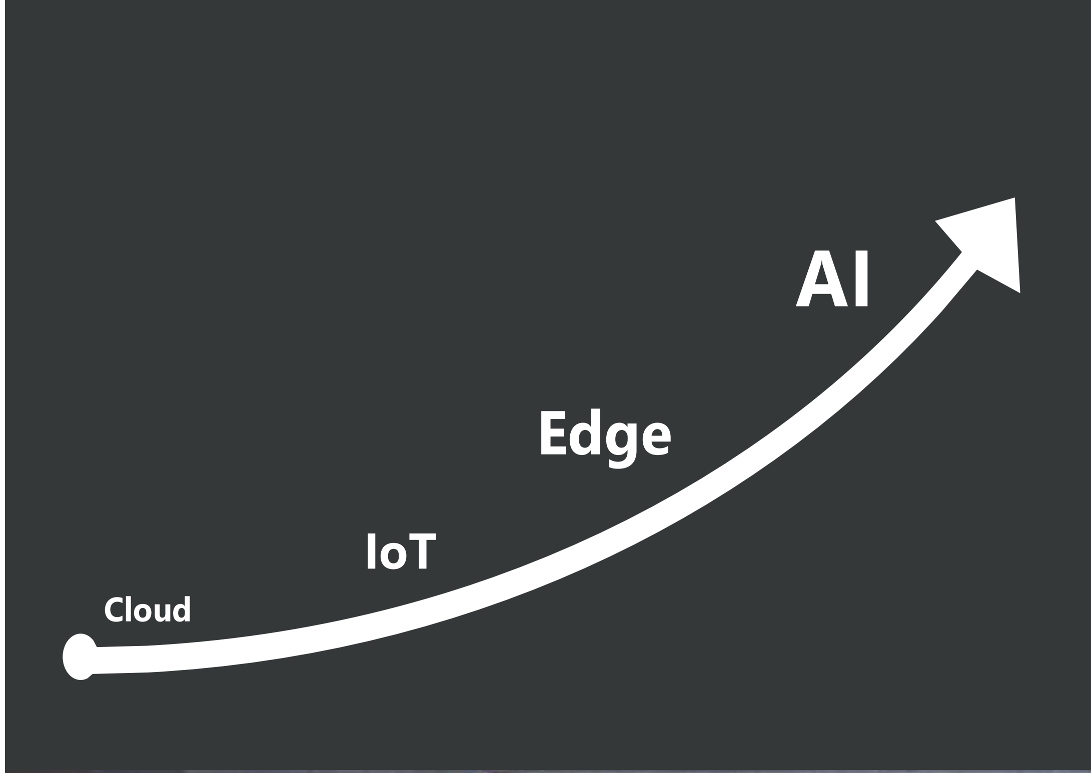
Harnessing signals from sensors and devices, managed centrally by the cloud

Edge

Intelligence offloaded from the cloud to IoT devices

AI

Breakthrough intelligence capabilities



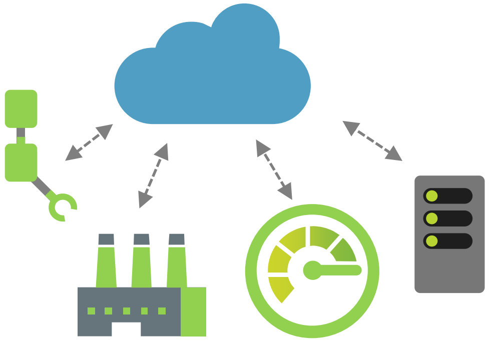
Cloud

IoT

Edge

AI

Why the edge?

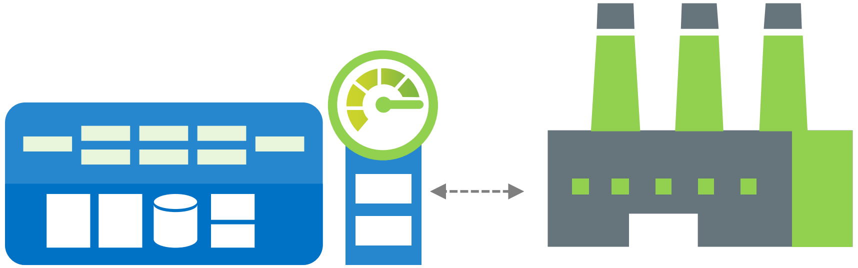


IoT in the Cloud

Remote monitoring and control

Merging remote data from across multiple IoT devices

Near infinite compute and storage to train machine learning and other advanced AI tools



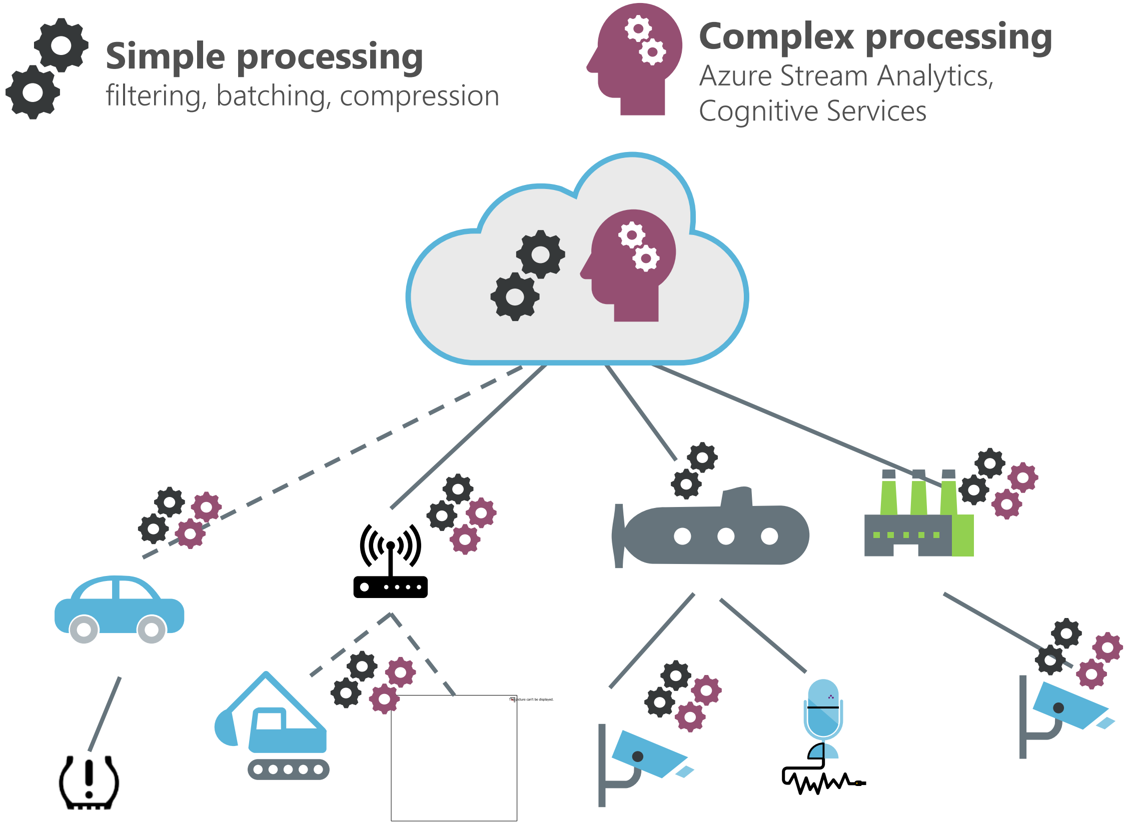
IoT on the Edge

Low latency tight control loops require near real-time response

Public internet inherently unpredictable

Privacy of data and protection of IP

Azure IoT Edge



Simple processing
filtering, batching, compression

Complex processing
Azure Stream Analytics,
Cognitive Services

Secure

- A Secure connection to the Azure IoT Edge
- Collect state and telemetry and monitor security of the device

Cloud Managed

- Enable rich management from Azure

Cross-Platform

- Enable Azure IoT Edge on both Windows and Linux

Portable

- Enable creating Docker Images that target multiple architecture

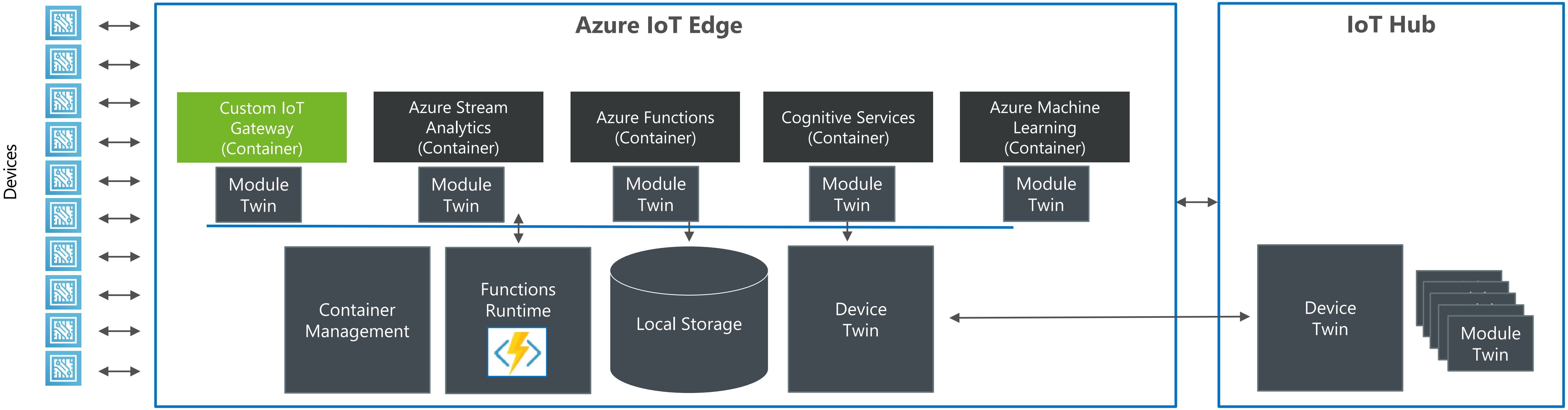
Extensible

- Enable seamless deployment of advanced capabilities modules such as **AI**, **Azure Function**, **Stream Analytics** and **3rd party**

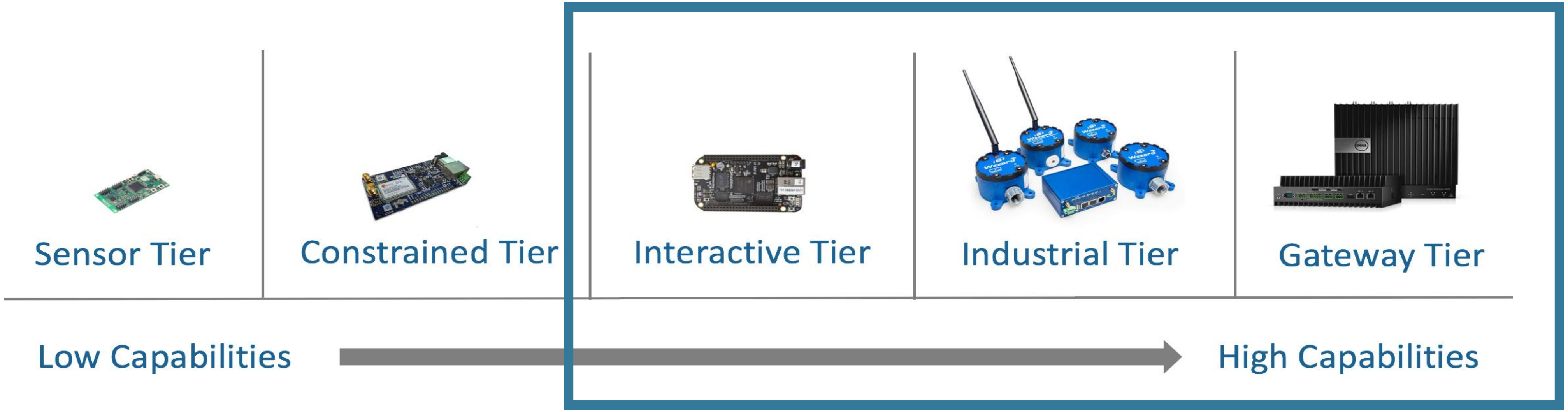
Azure IoT Edge

- Container based modules
- Azure Functions
- Azure Stream Analytics
- Azure Machine Learning
- Cognitive Services

- Offline / Synchronized Device Twins
- Local Storage
- Cloud Management & Deployment
- High Availability / Fault Tolerance
- Cloud Dev/Test Support



Hardware for Azure IoT Edge

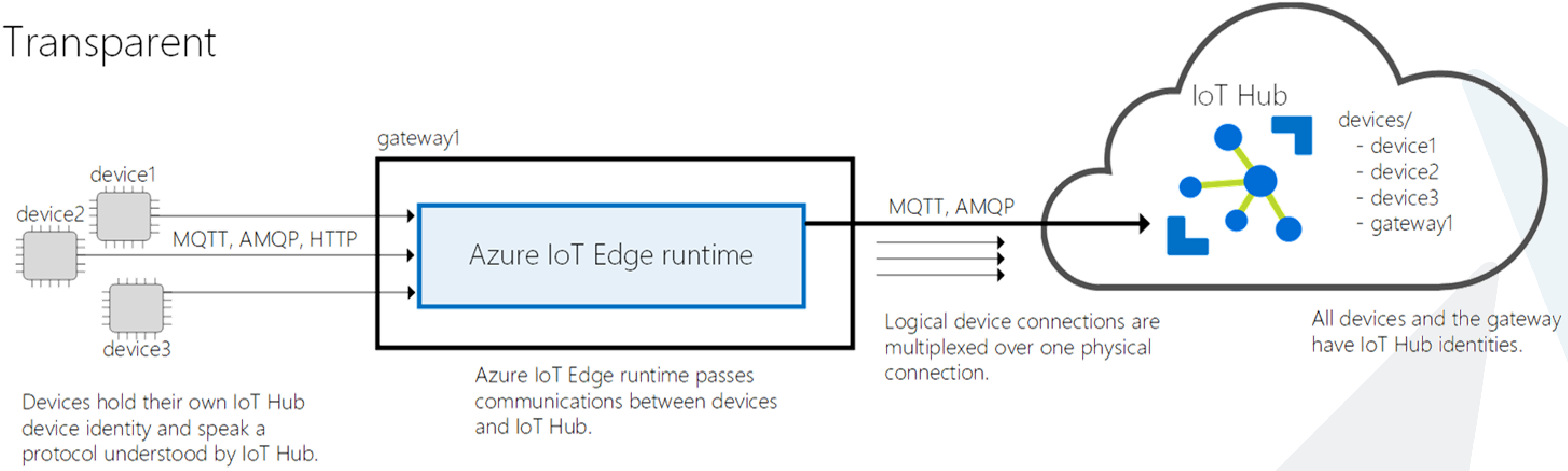


Ability to run on devices smaller than a Raspberry Pi

128MB memory

Support best in class operating systems such as Windows, and Linux

IoT Edge as a Gateway - Transparent



Devices hold their own IoT Hub device identity and speak a protocol understood by IoT Hub.

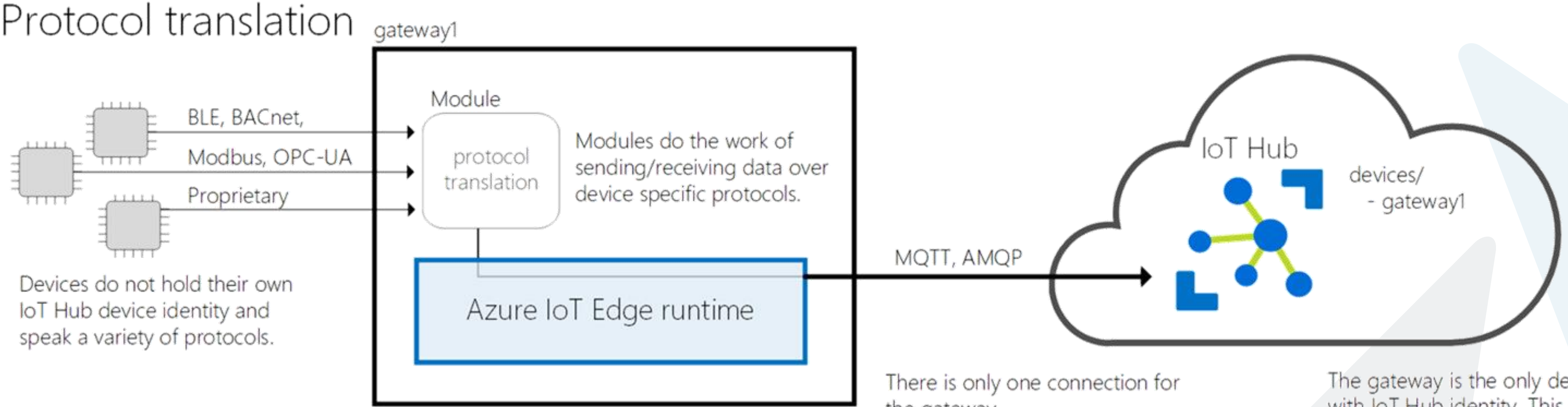
Azure IoT Edge runtime passes communications between devices and IoT Hub.

Logical device connections are multiplexed over one physical connection.

All devices and the gateway have IoT Hub identities.

IoT Edge as a Gateway - Protocol Translation

Protocol translation



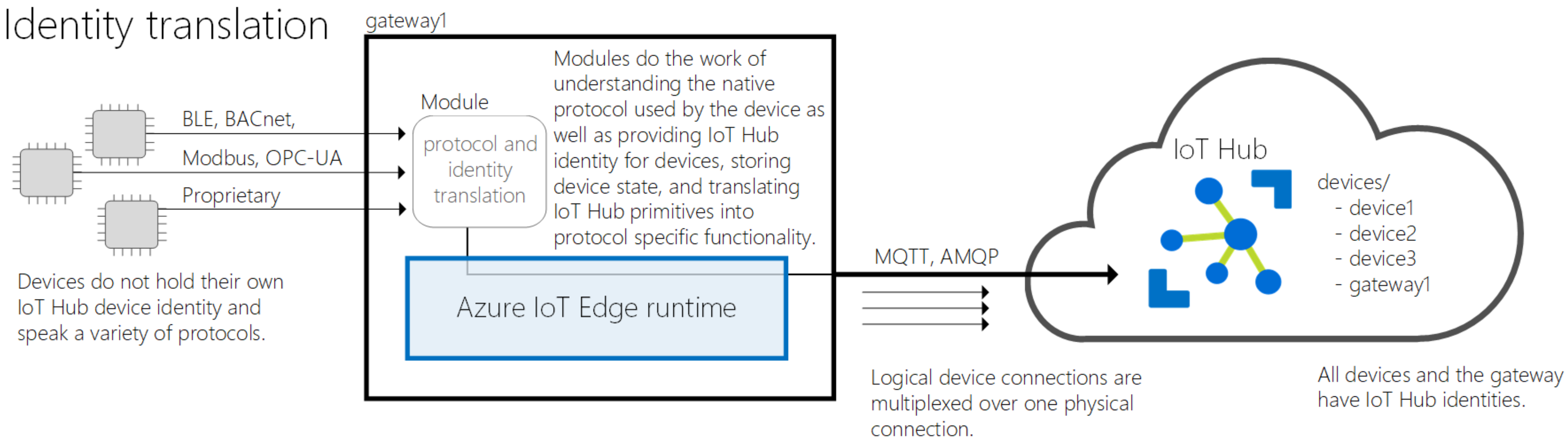
Modules do the work of sending/receiving data over device specific protocols.

Devices do not hold their own IoT Hub device identity and speak a variety of protocols.

There is only one connection for the gateway.

The gateway is the only device with IoT Hub identity. This implies it is the only device which has a twin.

IoT Edge as a Gateway – Identity Translation



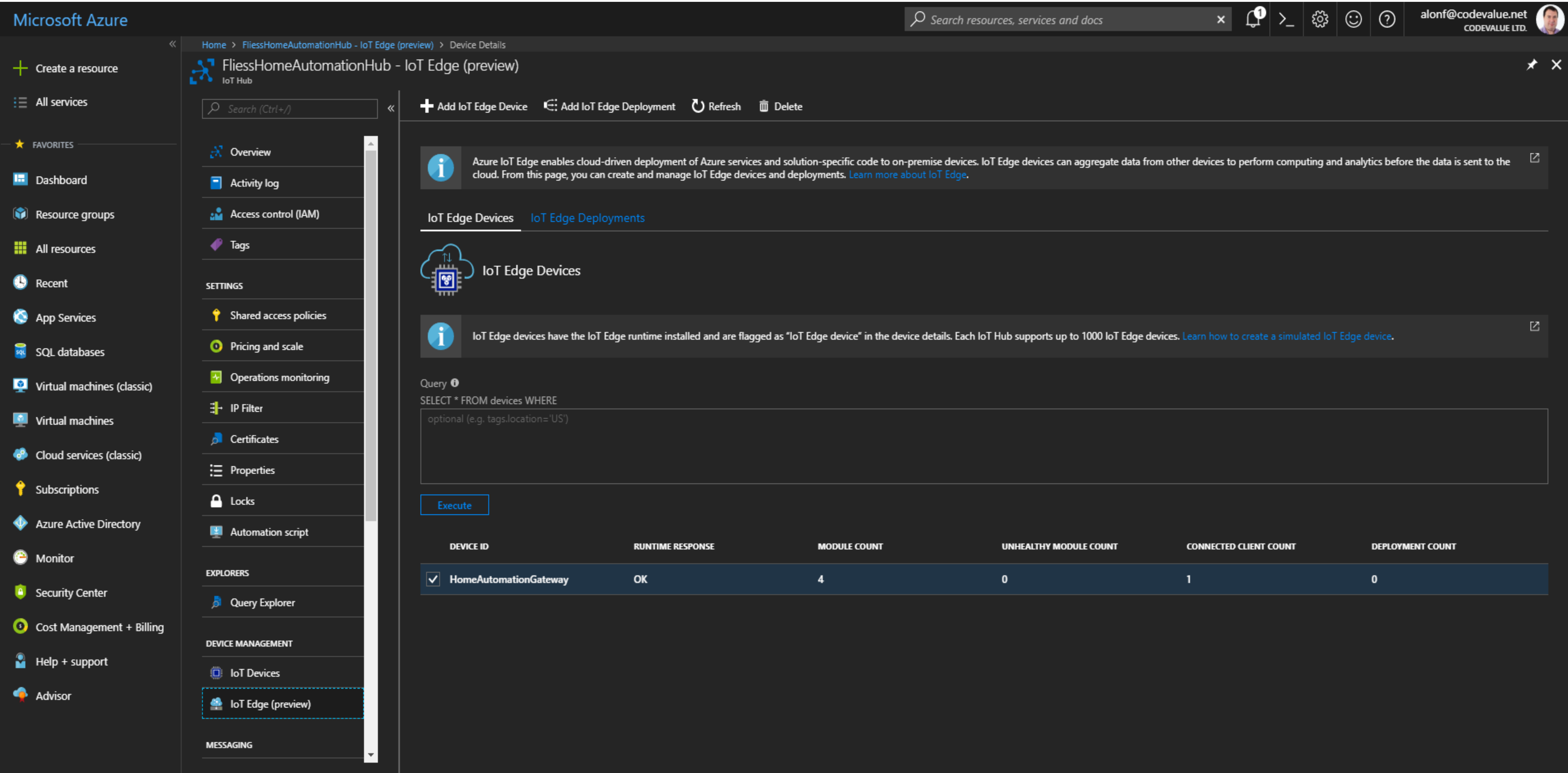
Devices do not hold their own IoT Hub device identity and speak a variety of protocols.

Modules do the work of understanding the native protocol used by the device as well as providing IoT Hub identity for devices, storing device state, and translating IoT Hub primitives into protocol specific functionality.

Logical device connections are multiplexed over one physical connection.

All devices and the gateway have IoT Hub identities.

IoT Edge Portal Support



Azure IoT Edge enables cloud-driven deployment of Azure services and solution-specific code to on-premise devices. IoT Edge devices can aggregate data from other devices to perform computing and analytics before the data is sent to the cloud. From this page, you can create and manage IoT Edge devices and deployments. Learn more about IoT Edge.

IoT Edge devices have the IoT Edge runtime installed and are flagged as "IoT Edge device" in the device details. Each IoT Hub supports up to 1000 IoT Edge devices. Learn how to create a simulated IoT Edge device.

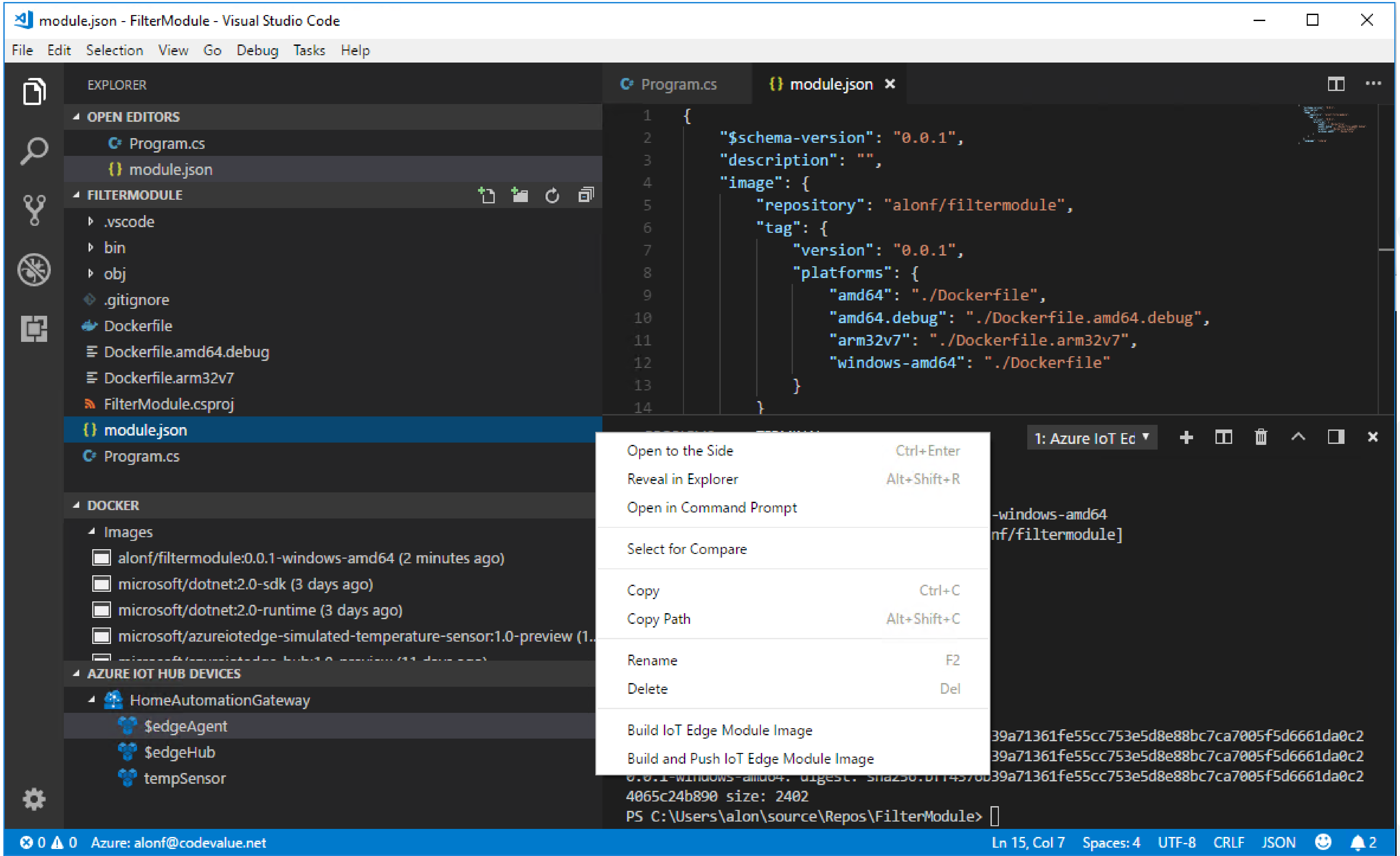
Query

SELECT * FROM devices WHERE

optional (e.g. tags.location='US')

	DEVICE ID	RUNTIME RESPONSE	MODULE COUNT	UNHEALTHY MODULE COUNT	CONNECTED CLIENT COUNT	DEPLOYMENT COUNT
☑	HomeAutomationGateway	OK	4	0	1	0

Visual Studio Code IoT Edge Extension



```json
{
    "$schema-version": "0.0.1",
    "description": "",
    "image": {
        "repository": "alonf/filtermodule",
        "tag": {
            "version": "0.0.1",
            "platforms": {
                "amd64": "./Dockerfile",
                "amd64.debug": "./Dockerfile.amd64.debug",
                "arm32v7": "./Dockerfile.arm32v7",
                "windows-amd64": "./Dockerfile"
            }
        }
```

```
-windows-amd64
nf/filtermodule]
39a71361fe55cc753e5d8e88bc7ca7005f5d6661da0c2
39a71361fe55cc753e5d8e88bc7ca7005f5d6661da0c2
0.0.1-windows-amd64: digest: sha256:b1f4376b39a71361fe55cc753e5d8e88bc7ca7005f5d6661da0c2
4065c24b890 size: 2402
PS C:\Users\alon\source\Repos\FilterModule>
```

Price & Capabilities

FEATURE	BASIC	STANDARD
Device-to-cloud telemetry	✓	✓
Per-device identity	✓	✓
Message Routing, Event Grid Integration	✓	✓
HTTP, AMQP, MQTT Protocols	✓	✓
DPS Support	✓	✓
Monitoring and diagnostics		✓
Cloud-to-device messaging		✓
Device Management, Device Twin		✓
IoT Edge		✓

Price & Capabilities

BASIC TIER	PRICE PER UNIT (PER MONTH)	TOTAL NUMBER OF MESSAGES/DAY PER UNIT	MESSAGE METER SIZE	Max # of Units
B1	$10	400,000	4 KB	200
B2	$50	6,000,000	4 KB	200
B3	$500	300,000,000	4 KB	10

STANDARD TIER	PRICE PER UNIT (PER MONTH)	TOTAL NUMBER OF MESSAGES/DAY PER UNIT	MESSAGE METER SIZE	Max # of Units
FREE	FREE	8,000	0.5 KB	1
S1	$25	400,000	4 KB	200
S2	$250	6,000,000	4 KB	200
S3	$2,500	300,000,000	4 KB	10

Operation throttles & Other Limits:

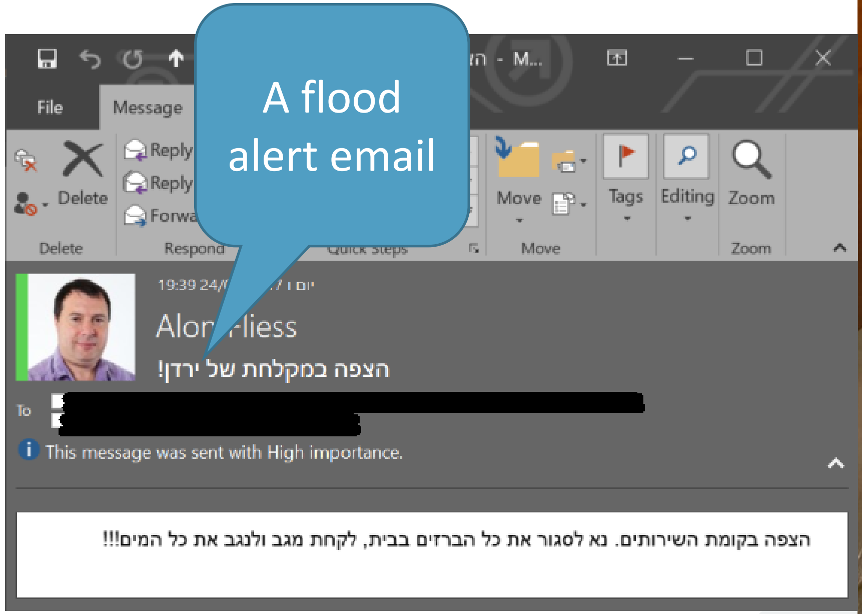
https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-devguide-quotas-throttling

Limits, Quota & Throttling

Tier	Sustained throughput	Sustained send rate
B1, S1	Up to 1111 KB/minute per unit (1.5 GB/day/unit)	Average of 278 messages/minute per unit (400,000 messages/day per unit)
B2, S2	Up to 16 MB/minute per unit (22.8 GB/day/unit)	Average of 4,167 messages/minute per unit (6 million messages/day per unit)
B3, S3	Up to 814 MB/minute per unit (1144.4 GB/day/unit)	Average of 208,333 messages/minute per unit (300 million messages/day per unit)

A True War Story

- My wife is shouting:
  - ALON!!! There’s water everywhere!!!
- By the time the plumber came to fix it, in 4 hours I had this:
- A water flood detector
  - Using small device + Azure IoT Hub + Service Bus Queue + Azure Logic App + Office 365 Email + Twilio SMS



A flood alert email

19:39 24/ ... יום ו

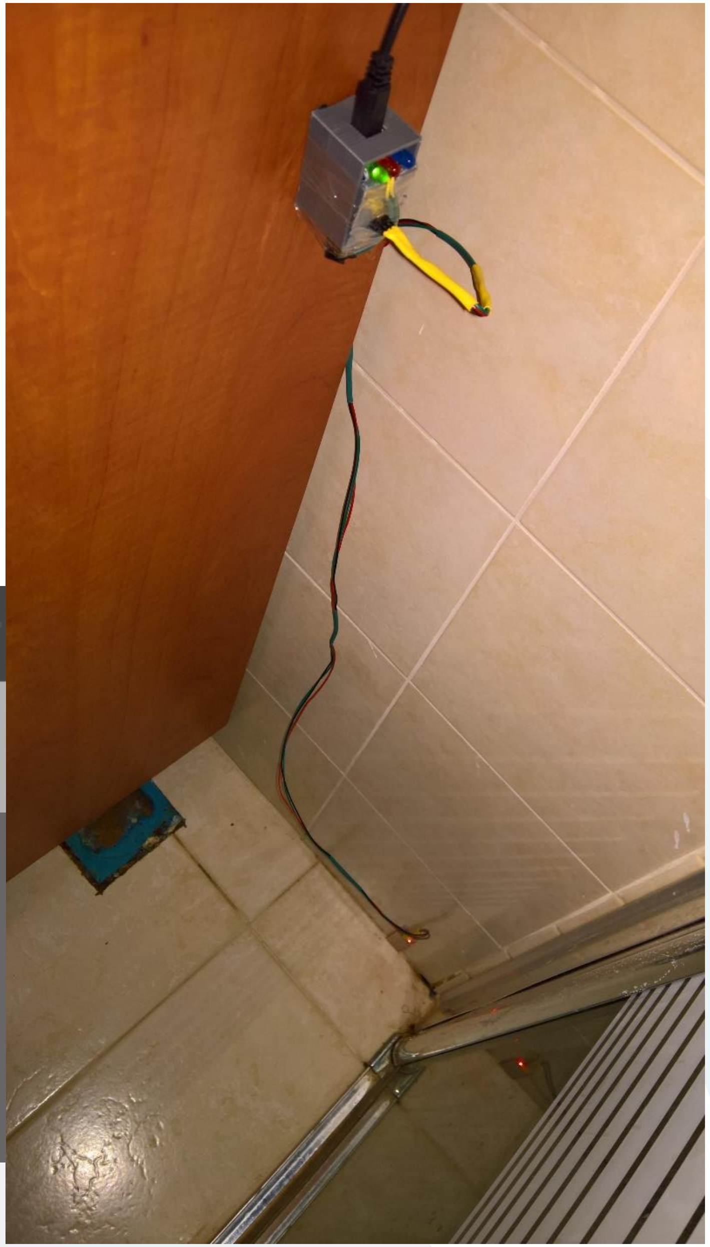
Alon Fliess

הצפה במקלחת של ירדן!

To [redacted]

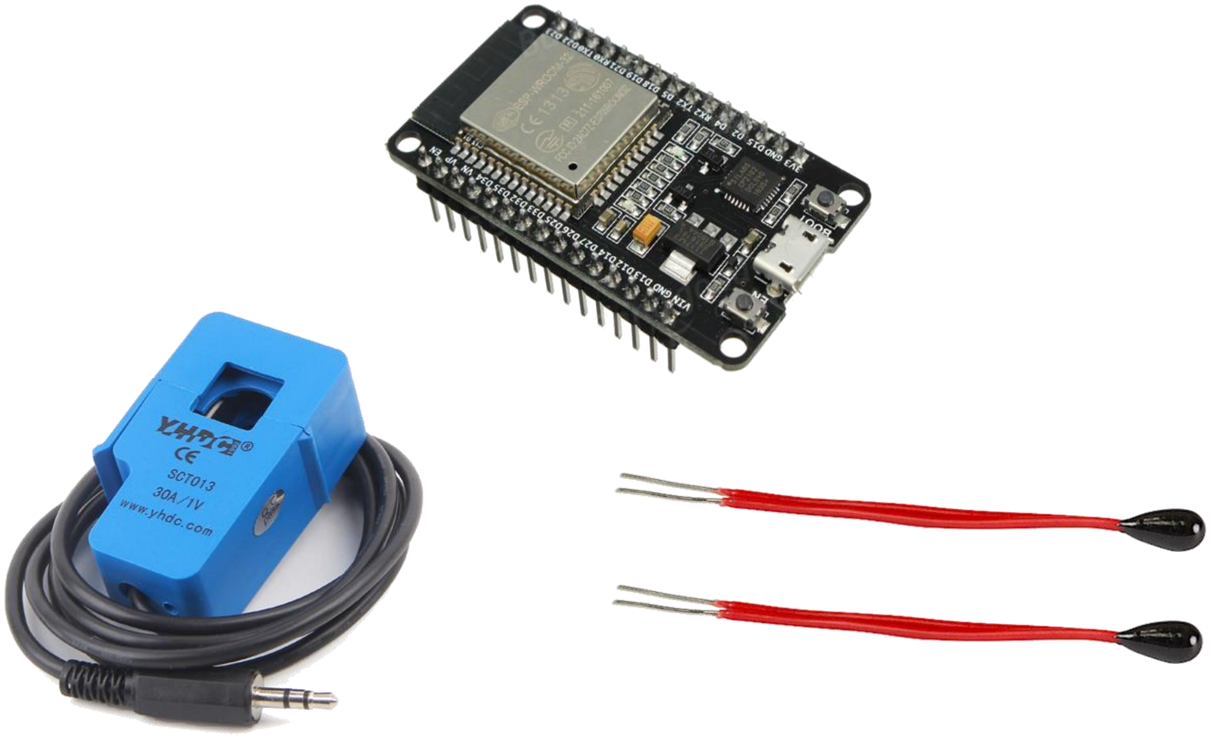
This message was sent with High importance.

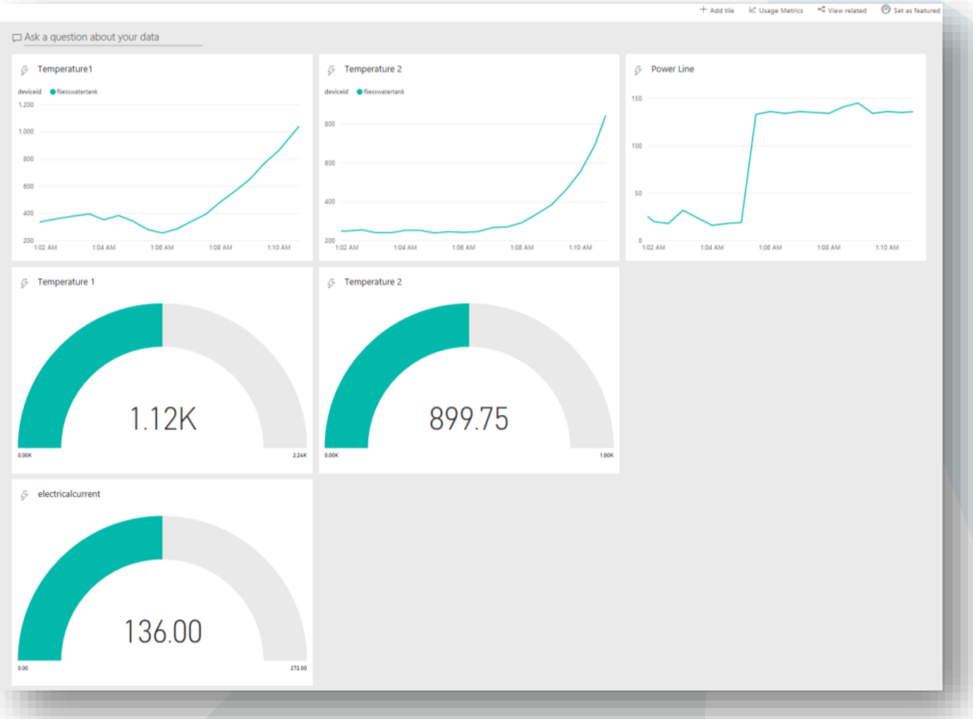
הצפה בקומת השירותים. נא לסגור את כל הברזים בבית, לקחת מגב ולנגב את כל המים!!!



The Water-Tank Boiler System

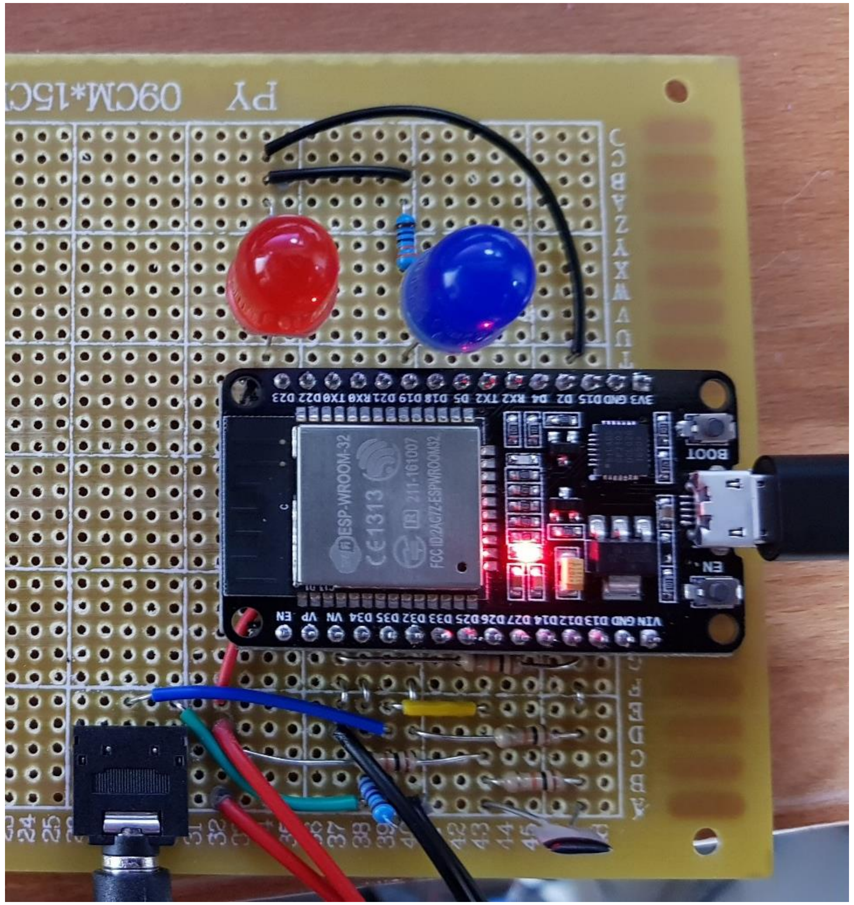
- Based on ESP32 + 2 NTC Thermistors + Current Sensor
- Hot water, when needed while saving energy!
- Over the air firmware update

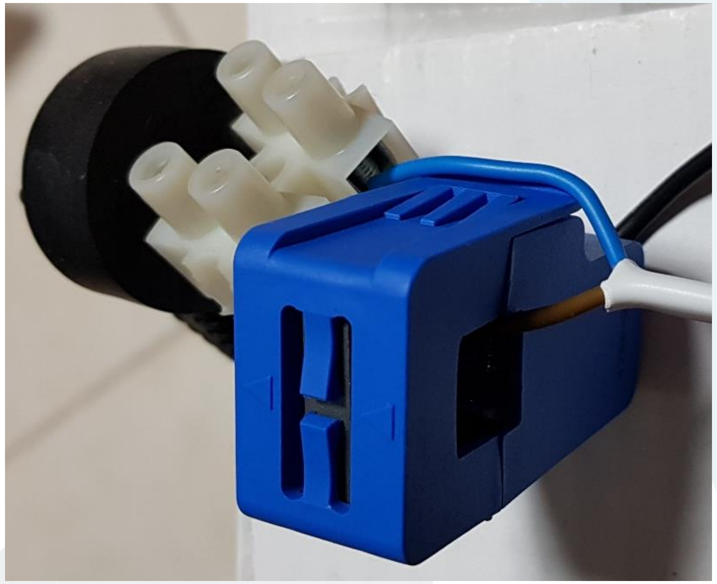




A Smart Boiler System

IoT Hub + Routing to ServiceBus Queue + Stream Analytics + Azure Function + PowerBI + Stream Insight





Demo

```
$mockQueryBuilder->shouldReceive('newQuery')->once()->andReturn($query);
$relation->expects($this->once())->method('touchIfTouching');

$this->assertTrue($relation->detach());
}


public function getRelation()
{
    list($builder, $parent) = $this->getRelationArguments();

    return new MorphToMany($builder, $parent, 'taggable', 'taggables', 'taggable_id', 'tag_id');
}

public function getRelationArguments()
{
    $parent = m::mock('Illuminate\Database\Eloquent\Model');
    $parent->shouldReceive('getMorphClass')->andReturn(get_class($parent));
    $parent->shouldReceive('getKey')->andReturn(1);
```

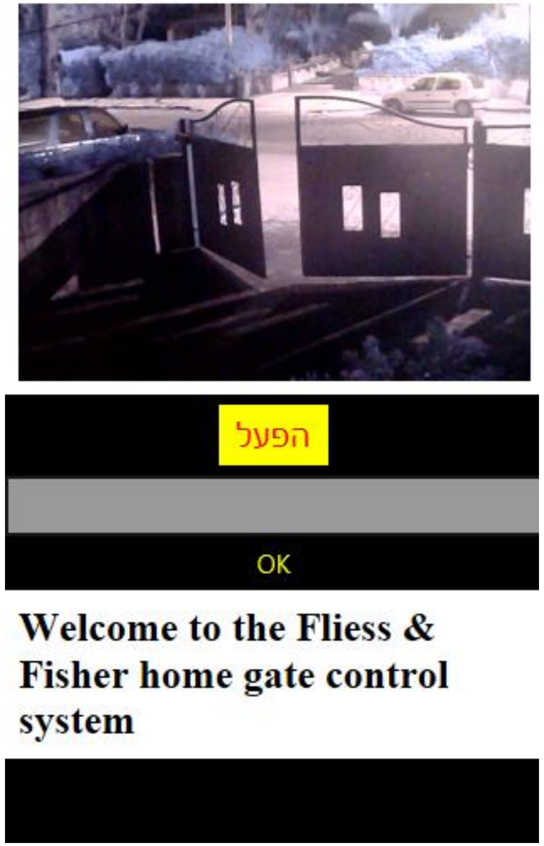
Smart Boiler - To-do...

- Use Azure IoT edge
  - Run the Azure Function locally
  - Run stream analytics locally
- Add AI module
  - Water usage patterns
  - Save even more energy
- Scale to many tenant
  - Device provisioning portal
  - Device settings and management portal

Azure IoT Summary

- IoT system architecture is a bit different then other cloud architecture
  - A “Pettle” – each device count!
- Microsoft provides SaaS and PaaS solutions
  - Azure IoT Central, Azure IoT Suite, Azure IoT Hub and cloud services
- Azure IoT Hub is designed to connect your devices to Azure. It supports:
  - SDKs, Millions of simultaneously connected devices, Per-device authentication, High throughput data ingestion, Scale device management
  - HTTP, MQTT, AMQP communication protocols
  - Cloud to Device and Device to Cloud messaging
  - State transfer with device twins
  - Query language, Job Management, File Upload
- Smart cloud & intelligent Edge

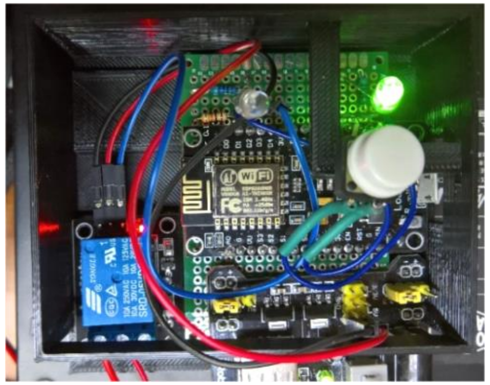
Activating a Real Gate

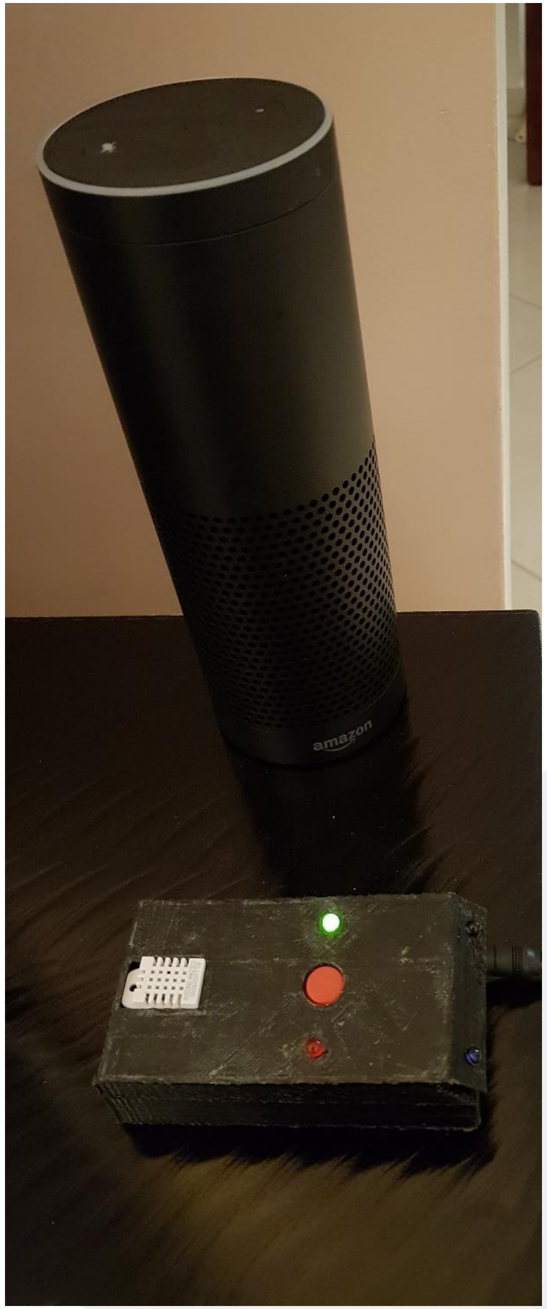


Welcome to the Fliess & Fisher home gate control system

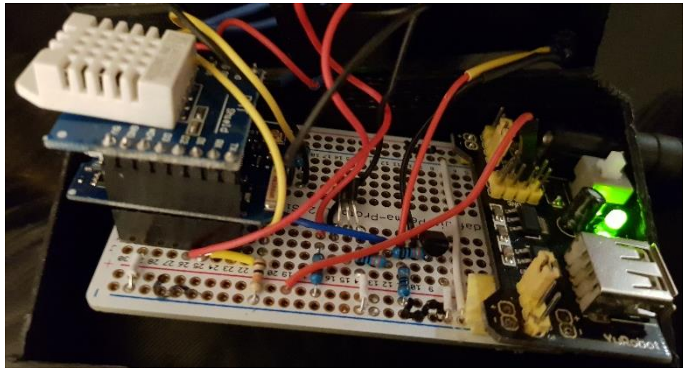
Other Usage

- My 3D Printer LED light





- Alexa: Turn on ground level AC
- Alexa set ground level AC to 20 degrees



Resources

- Demo code:
  - https://github.com/alonf/BasicGateController
  - Setup IoT Hub video: https://youtu.be/vq5AeLlsWx4
- My MSDN articles:
  - Introduction to the Internet of Things – From the Device to Microsoft Azure Cloud
    - https://blogs.msdn.microsoft.com/microsoft_press/2015/04/27/from-the-mvps-introduction-to-the-internet-of-things-from-the-device-to-microsoft-azure-cloud/
  - Efficient IoT With Azure
    - https://blogs.msdn.microsoft.com/mvpawardprogram/2016/11/15/efficient-iot-with-azure/
  - Secure Provisioning of IoT device using Azure IoT Hub device SDK
    - https://blogs.msdn.microsoft.com/mvpawardprogram/2017/03/14/provisioning-of-iot-device/
- Thingiverse
  - http://www.thingiverse.com/thing:2253418
- Azure IoT
  - IoT SDKs - https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-devguide-sdks
  - GitHub - https://github.com/Azure/azure-iot-sdks
  - Azure IoT Suite - https://azure.microsoft.com/en-us/suites/iot-suite/
  - Azure IoT Hub - https://azure.microsoft.com/en-us/services/iot-hub/

C# and Xamarin

C#

- A multi-paradigm programming language encompassing strong typing, imperative, declarative, functional, generic, object-oriented and component-oriented programming disciplines
- Developed by Microsoft and approved as a standard by Ecma (ECMA-334) and ISO (ISO/IEC 23270:2006)
- C# is intended to be a simple, modern, general-purpose, object-oriented programming language
- The most recent version is C# 6.0, which was released on July 20, 2015

C# Basics

- The platform – Compiler, Jitter, CLR, GC, Libraries (references), .NET Native
- Types, Value types and reference types (boxing), Metadata (ILDASM)
- Class and Interfaces, explicit interface, polymorphism (abstract, override, new), partial
- Methods, ref and out parameters, params, lambda expression
- Exceptions, checked and unchecked, finally
- Control Flow (if-else, while, do-while, for, foreach, switch)
- Operator overloading
- Properties
- Delegates & Events
- Attributes
- Generic, constraints
- Enumerators (yield return)
- LINQ
- Task, Async/Await

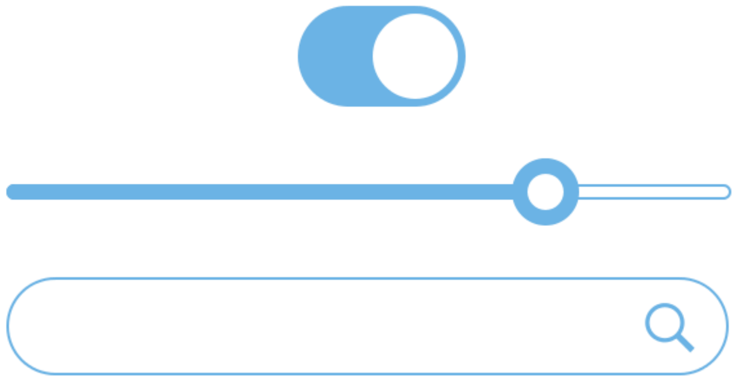
https://en.wikipedia.org/wiki/Comparison_of_C_Sharp_and_Java

ISO 9001 SGS

Xamarin (Forms)

What is native?

The Anatomy of a Native App



Native User Interfaces

Architecting

Mobile Apps

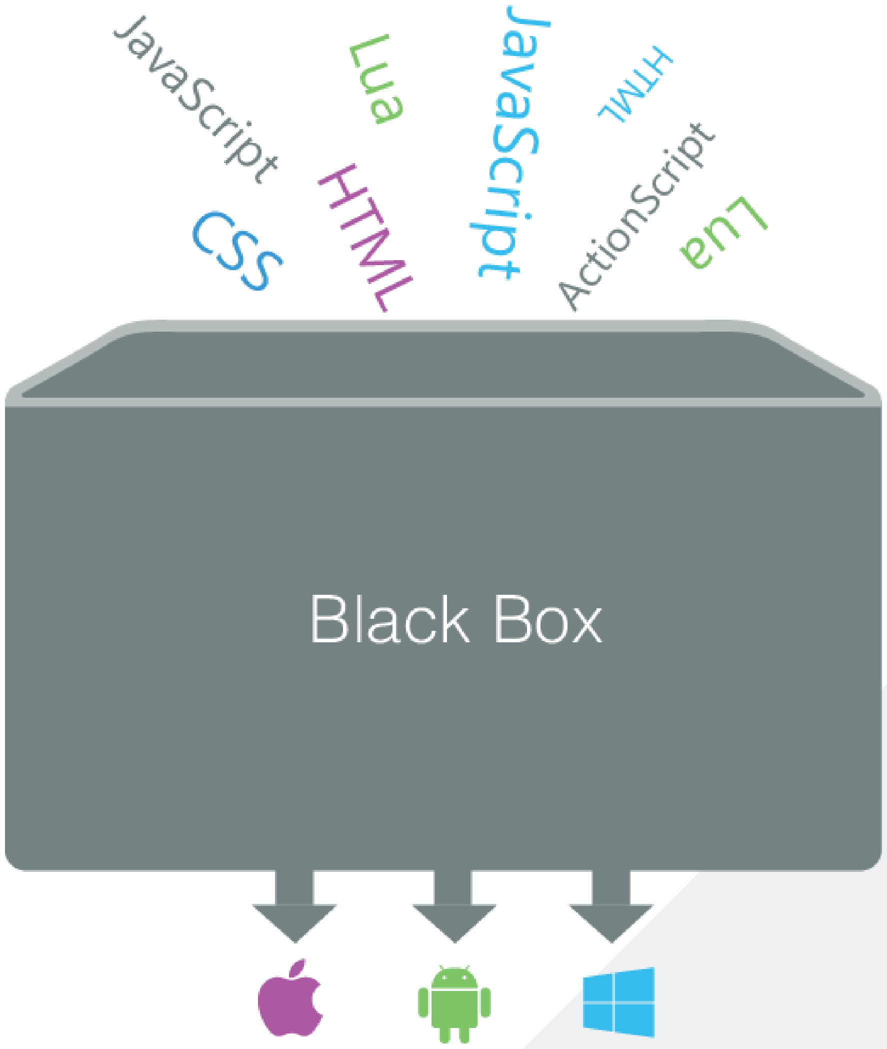
The Silo Approach

Build App
Multiple Times

iOS App	Android App	Windows App
Objective-C XCode	Java Eclipse	C# Visual Studio

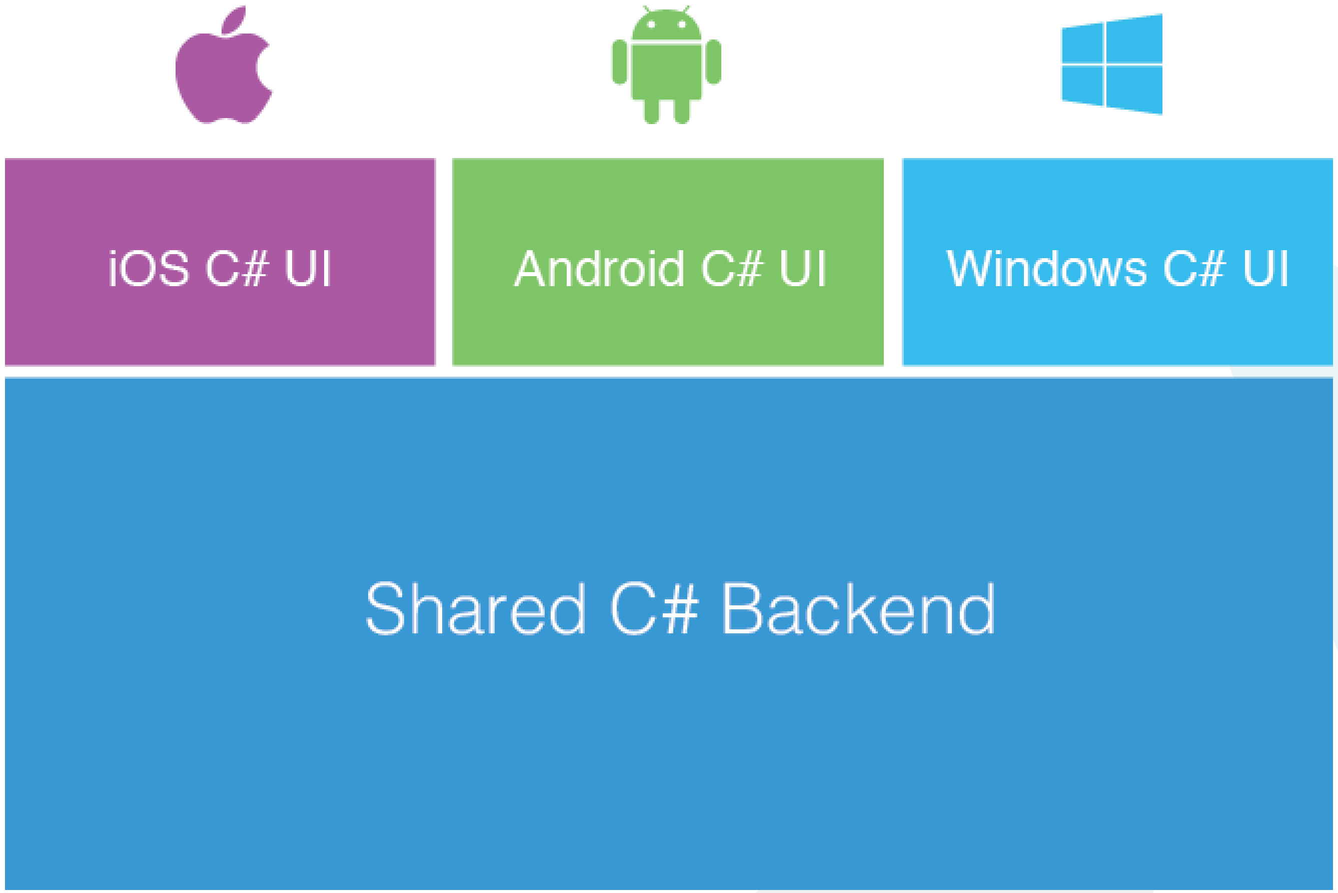
The Write-Once-Run-Anywhere Approach

Lowest Common Denominator

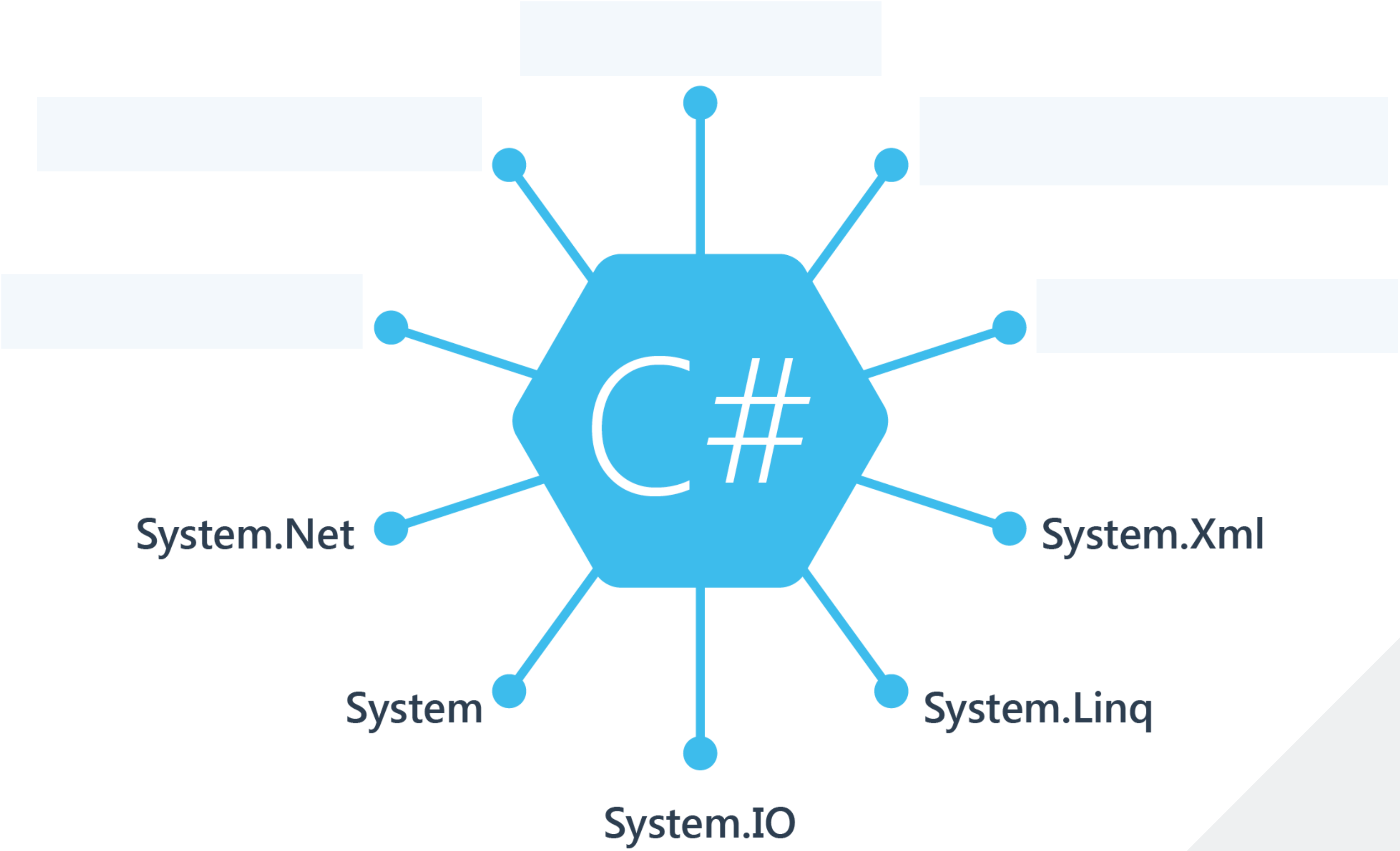


Xamarin’s Unique Approach

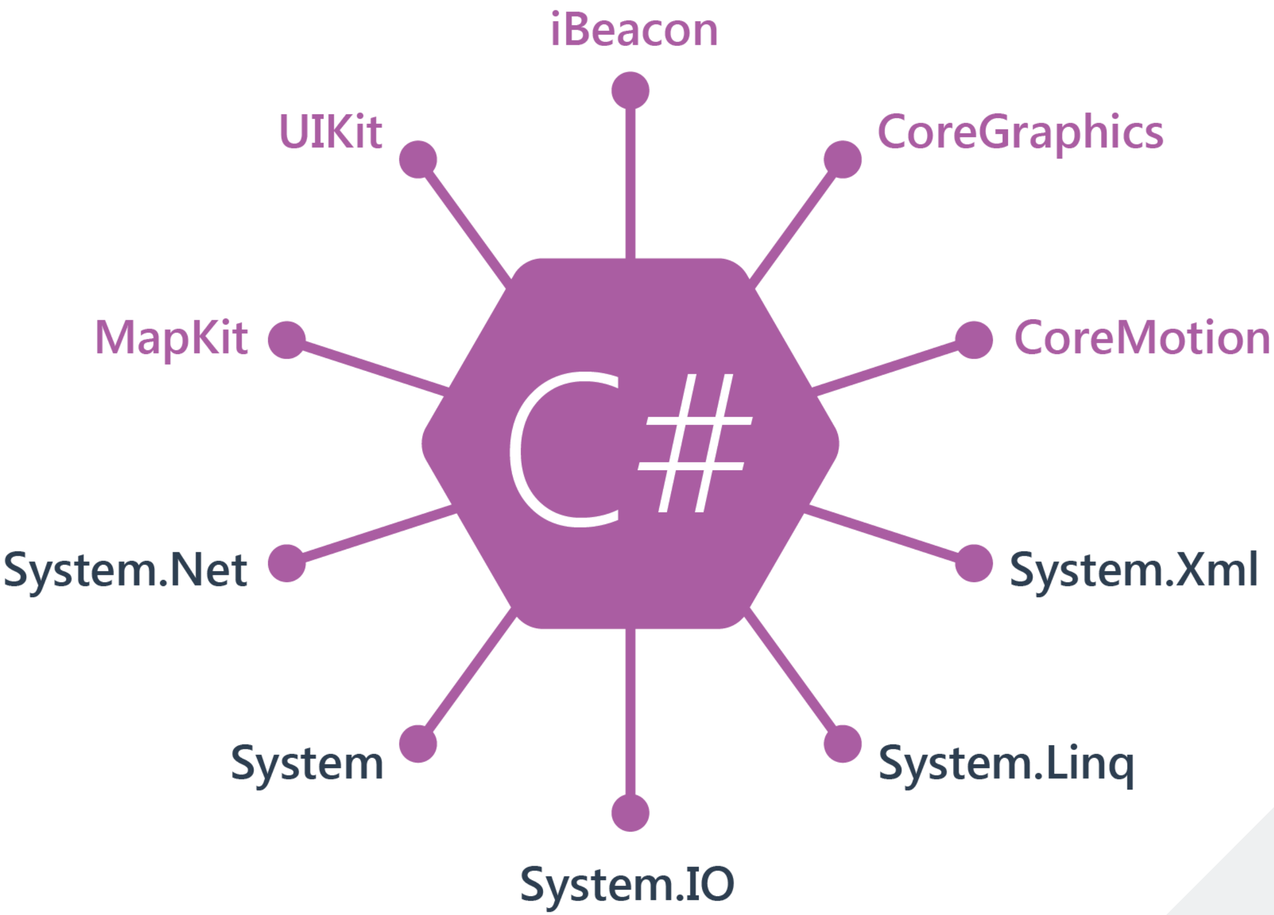
Native With
Code Sharing



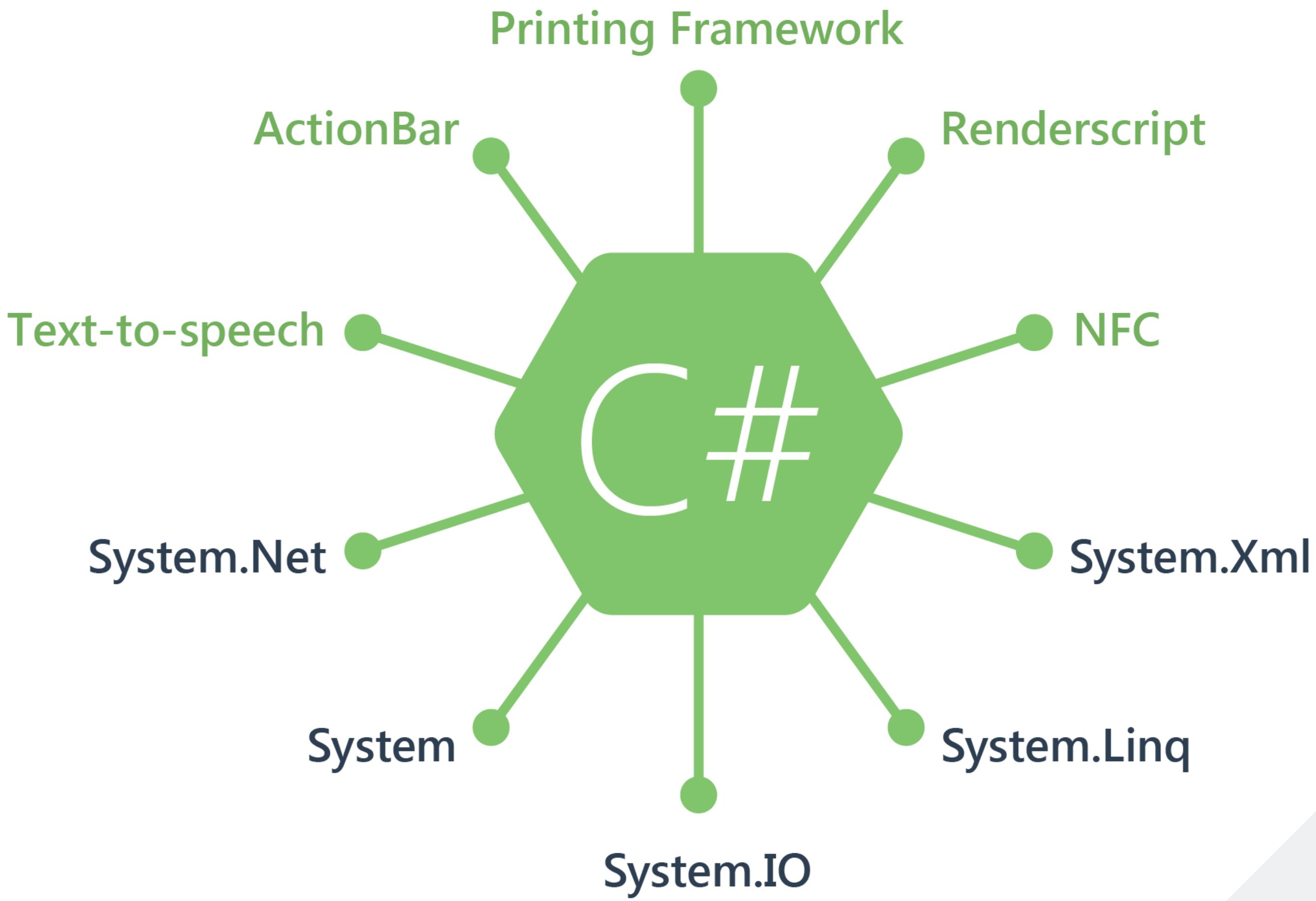
Windows APIs



iOS APIs | 100% Coverage

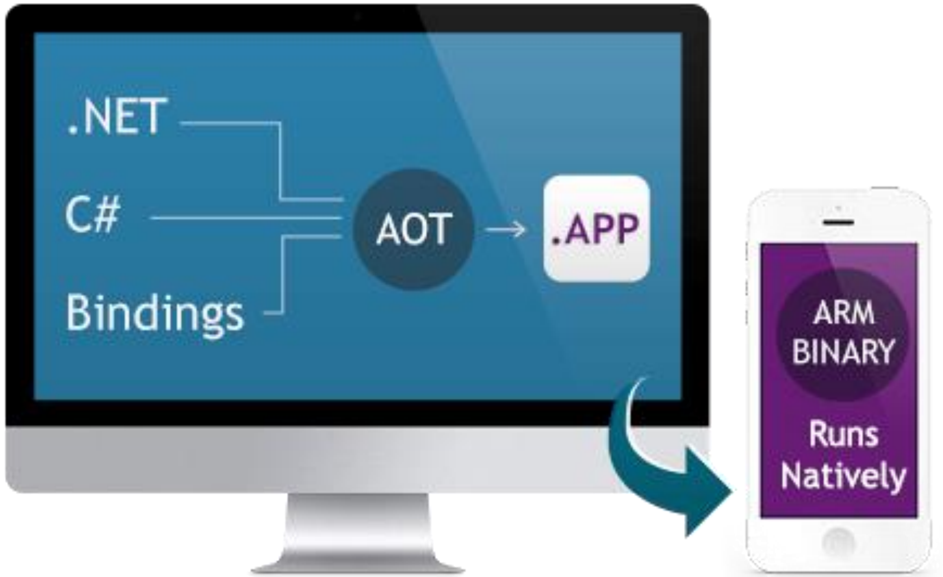


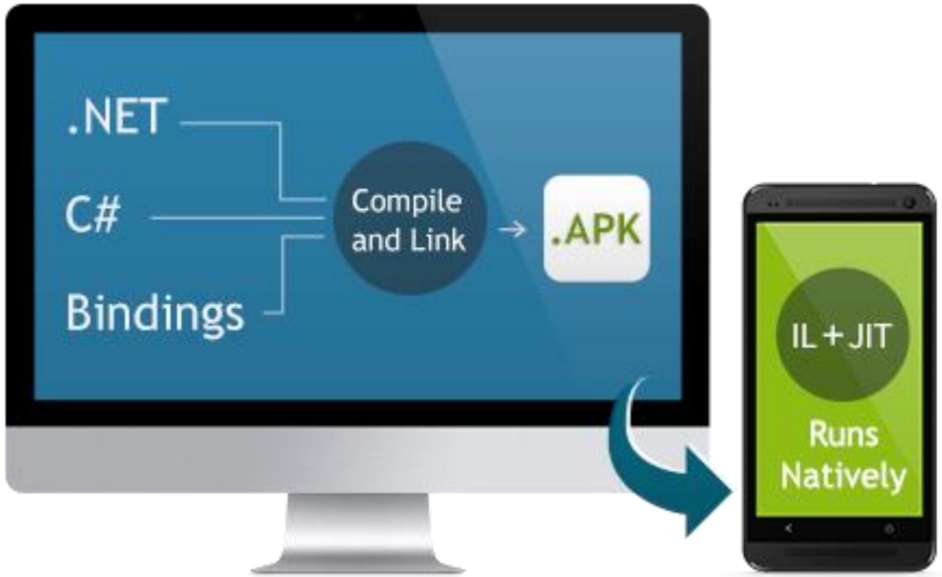
Android APIs | 100% Coverage



Anything you can do in Objective-C, Swift, or Java can be done in C# with Xamarin using Visual Studio

Native Performance

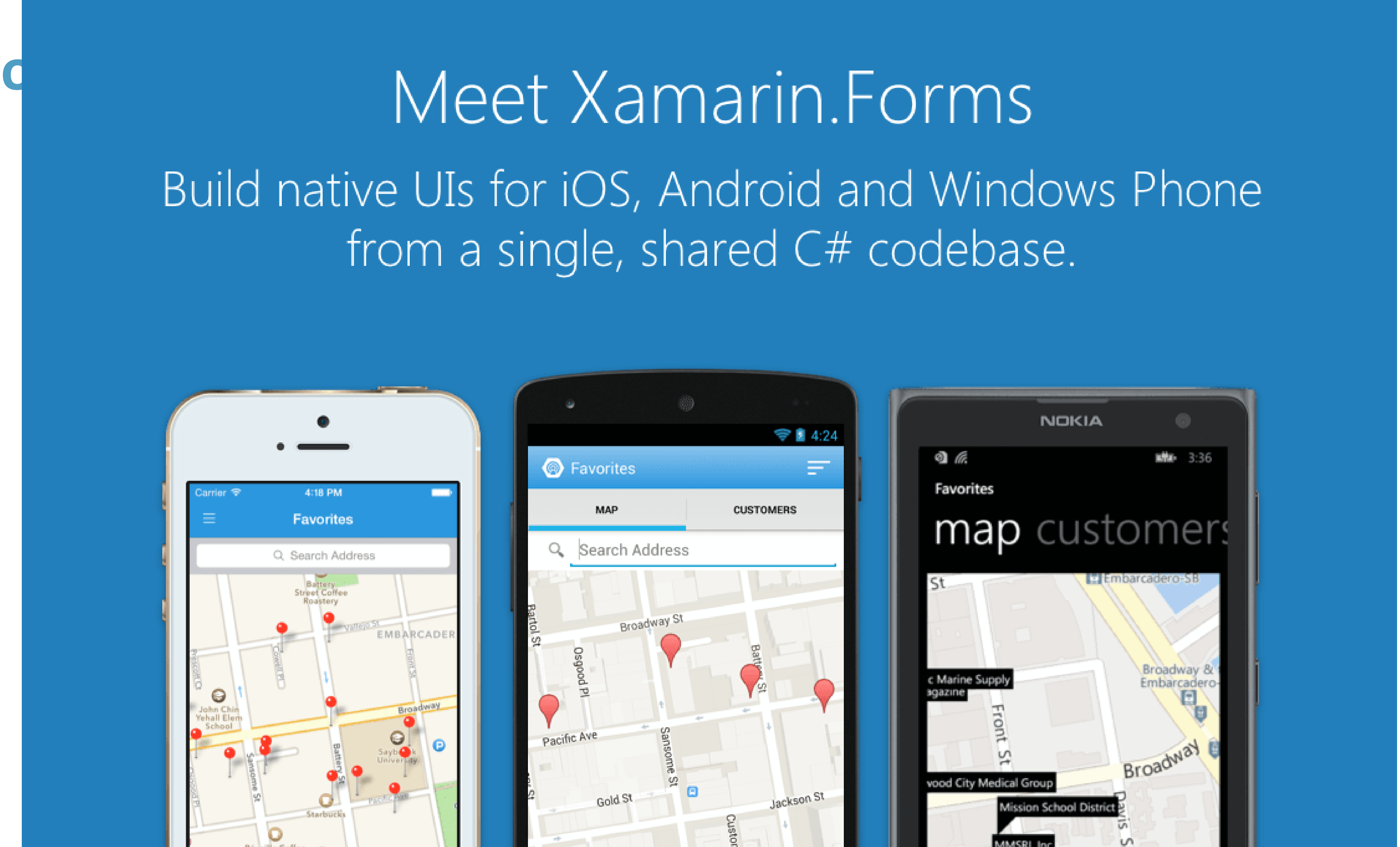




Xamarin.iOS does full Ahead Of Time (AOT) compilation to produce an ARM binary for Apple’s App Store.

Xamarin.Android takes advantage of Just In Time (JIT) compilation on the Android device.

Po

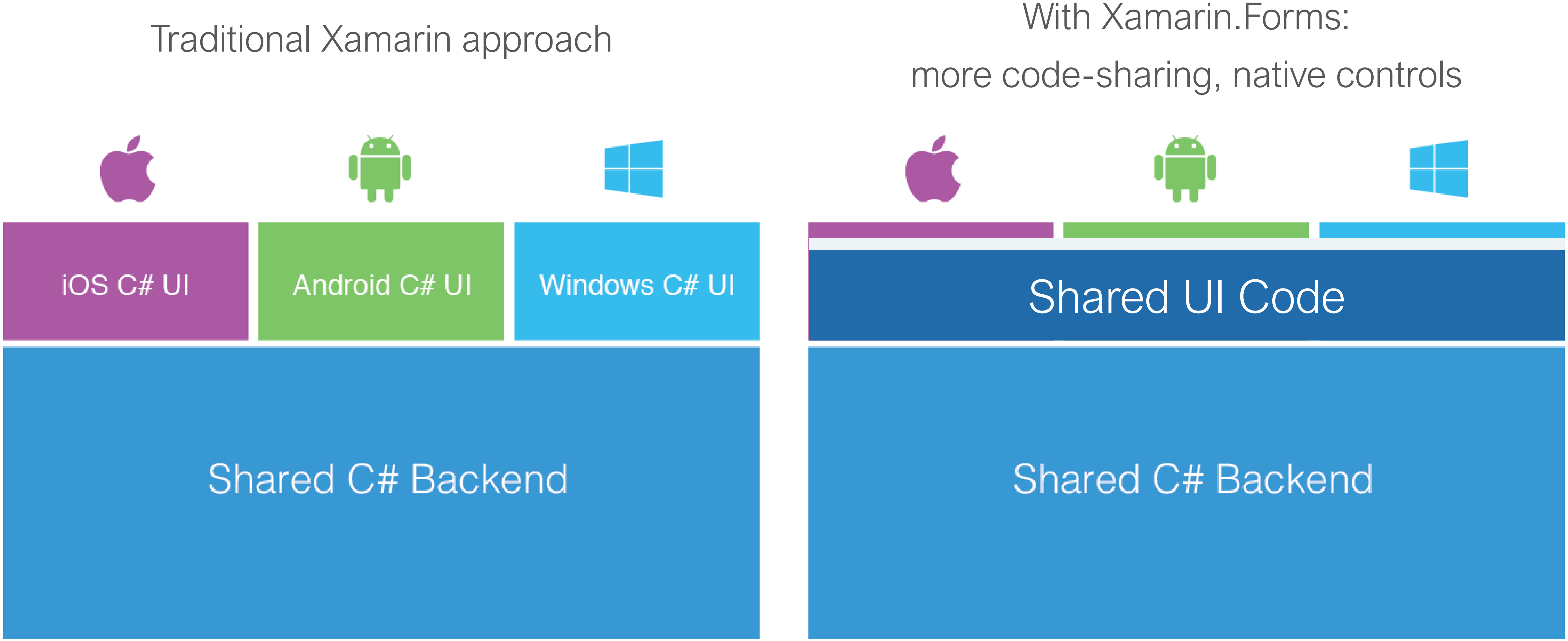


Meet Xamarin.Forms

Build native UIs for iOS, Android and Windows Phone from a single, shared C# codebase.

Xamarin + Xamarin.Forms

Xamarin



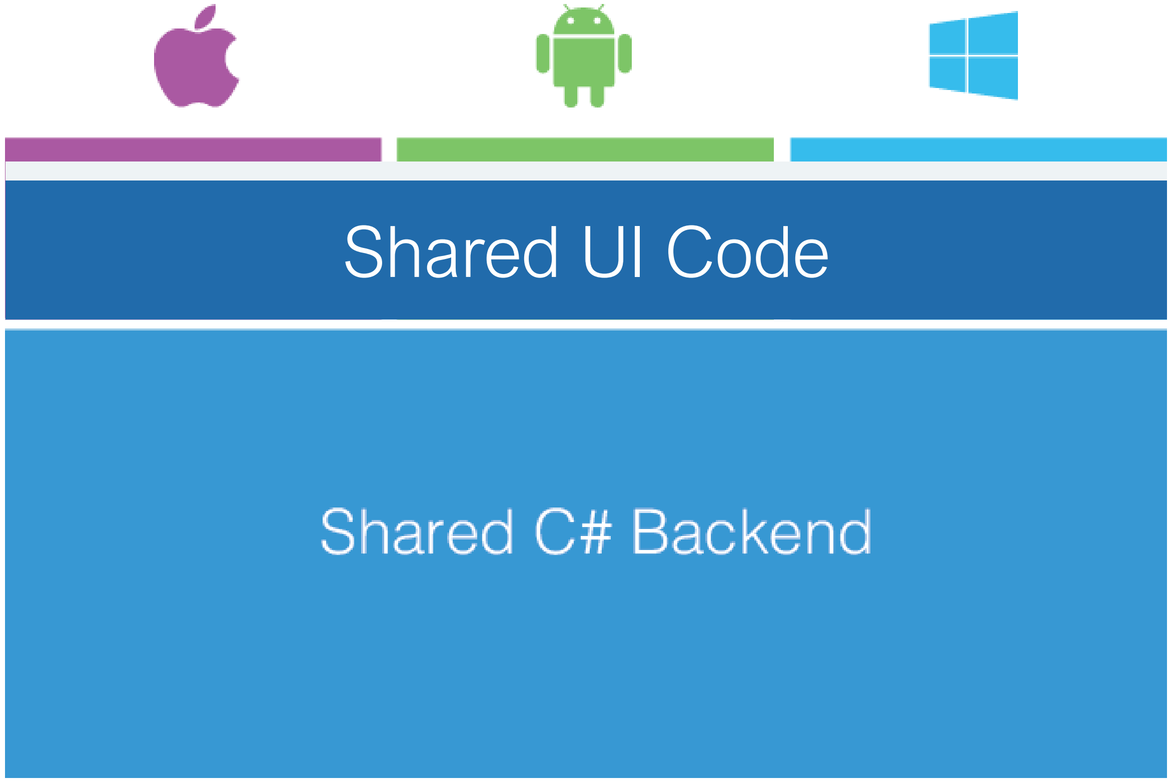
Xamarin + Xamarin.Forms

Xamarin

Quickly and easily build native user interfaces using shared code

Xamarin.Forms elements map to native controls and behaviors

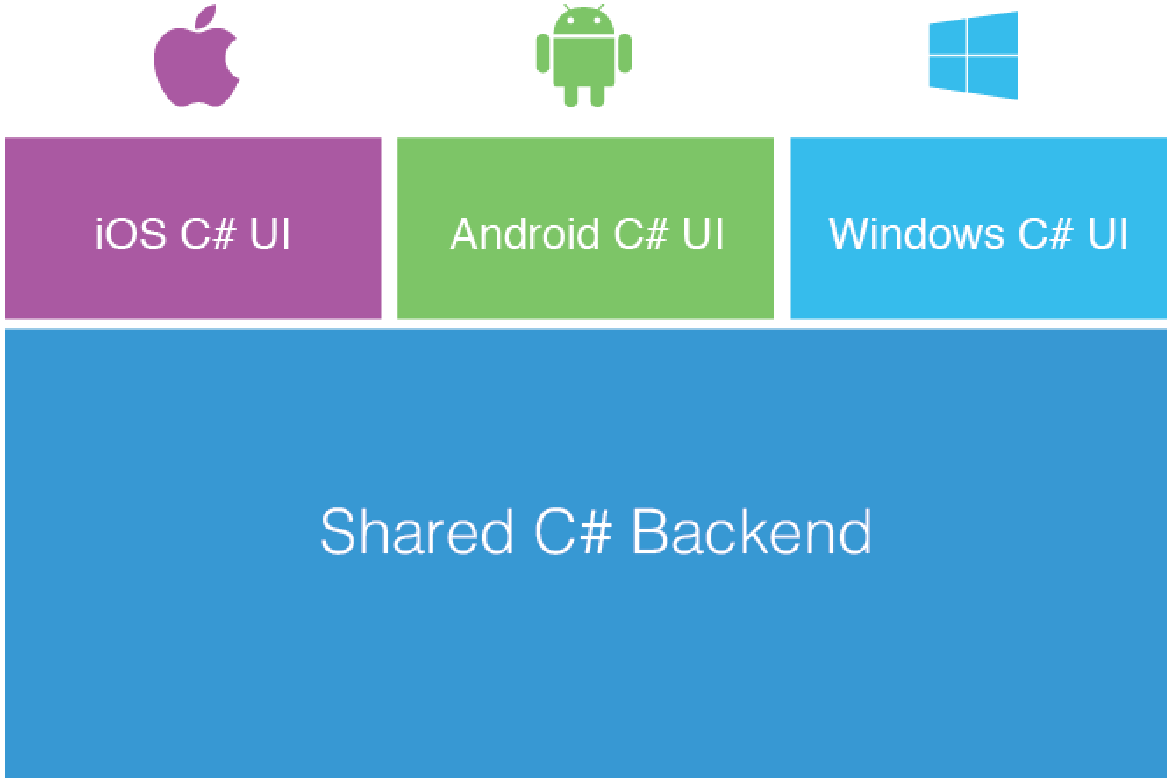
Mix-and-match Xamarin.Forms with native APIs



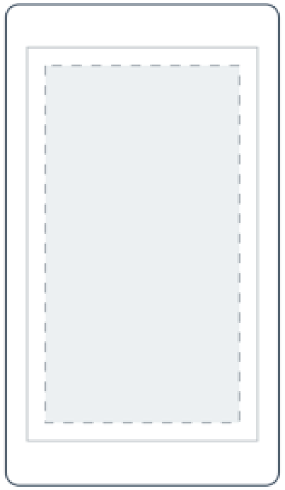
What's Included

Xamarin

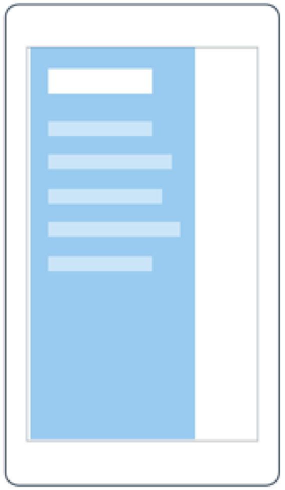
- 40+ Pages, Layouts, and Controls
  - Build from code behind or XAML
- Two-way Data Binding
- Navigation
- Animation API
- Dependency Service
- Messaging Center



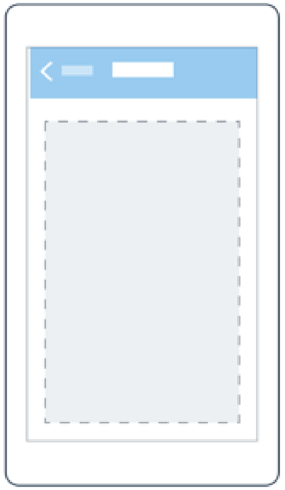
Pages



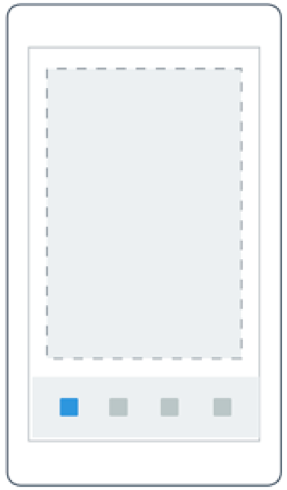
Content



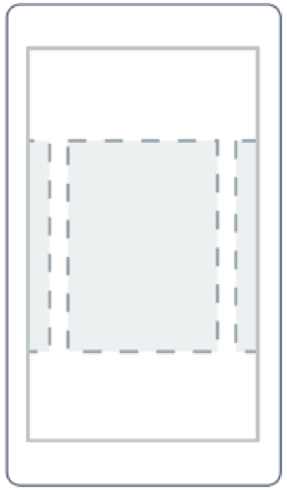
MasterDetail



Navigation

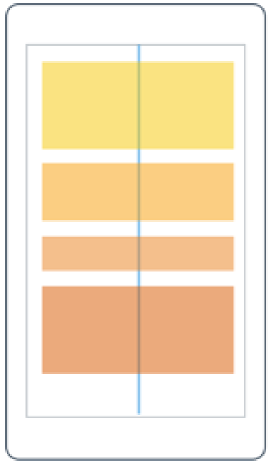


Tabbed

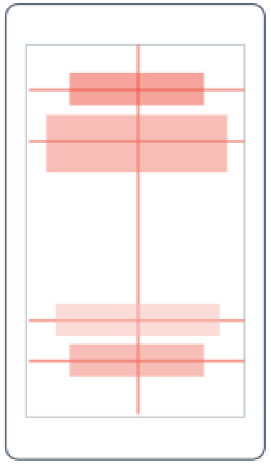


Carousel

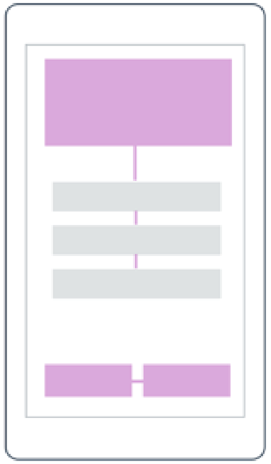
Layouts



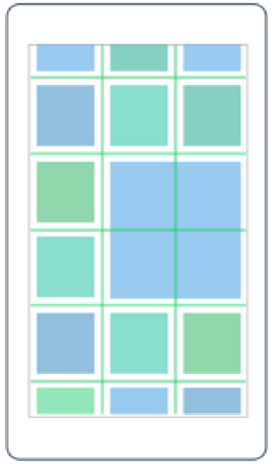
Stack



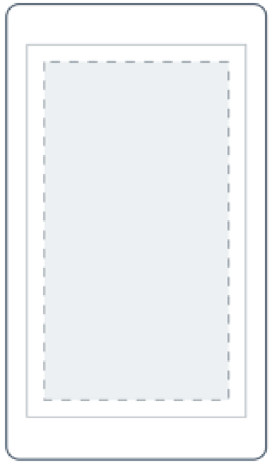
Absolute



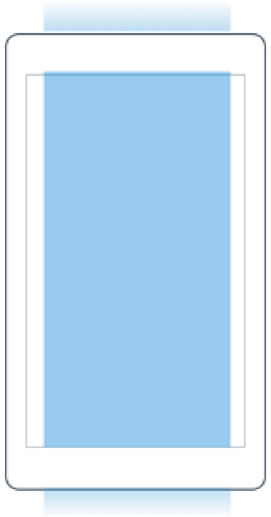
Relative



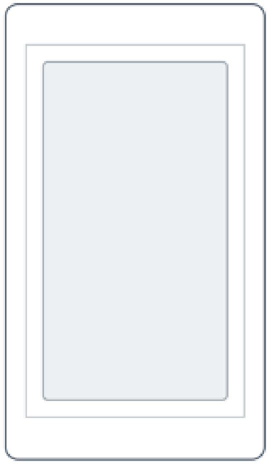
Grid



ContentView



ScrollView



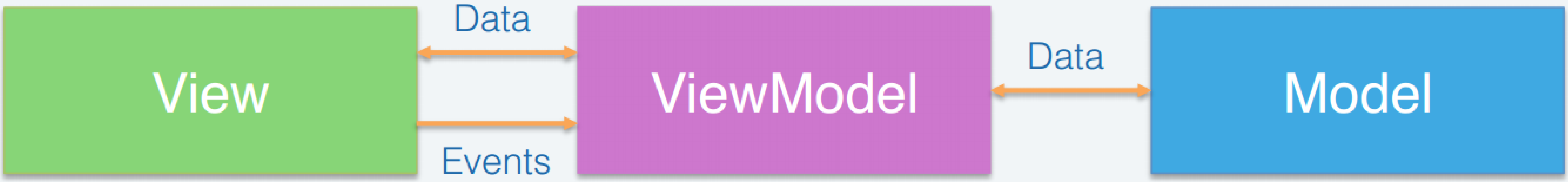
Frame

Controls

- ActivityIndicator
- BoxView
- Button
- DatePicker
- Editor
- Entry
- Image
- Label
- ListView
- Map
- OpenGLView
- Picker
- ProgressBar
- SearchBar
- Slider
- Stepper
- TableView
- TimePicker
- WebView
- EntryCell
- ImageCell
- SwitchCell
- TextCell
- ViewCell

Model-View-ViewModel

Xamarin

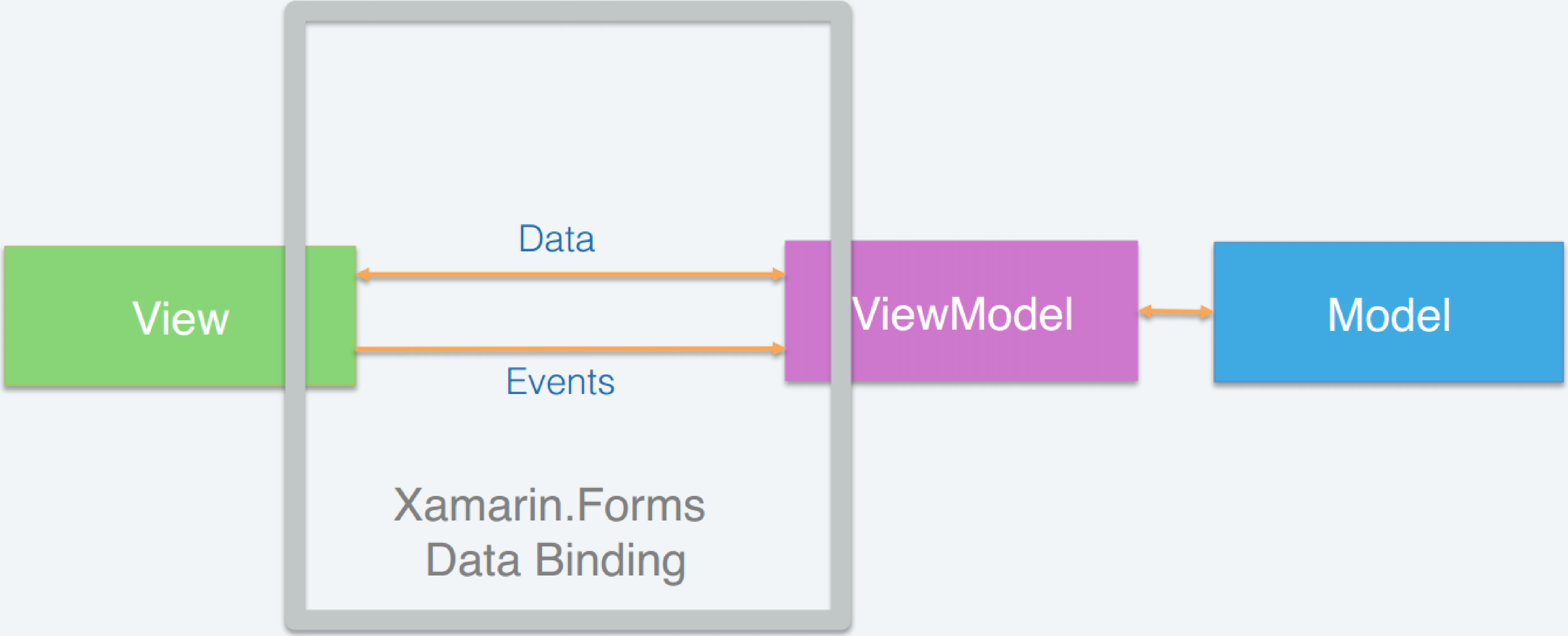


How to display information

What to display
Flow of interaction

Business Logic
Data objects

Model-View-ViewModel



DataBinding

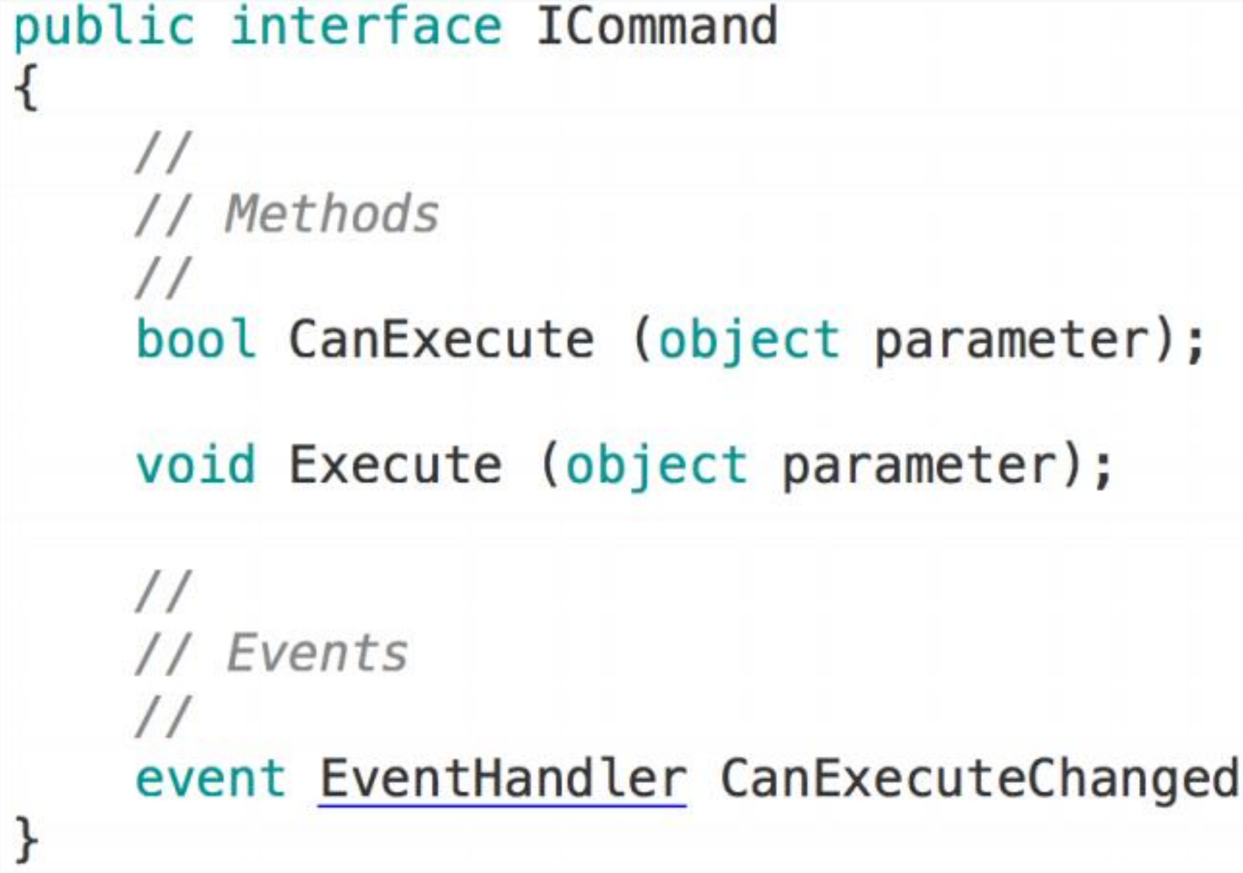
- Xamarin support rich DataBindings mechanism.
- Support for INotifyPropertyChanged notifications.
- Declare Bindings in Code/XAML

```
var label = new Label() {VerticalOptions = LayoutOptions.Center};
           label.SetBinding(Label.TextProperty, new Binding("MyName"));
```

```
<Entry Placeholder="Please input your User Name"
       Text="{Binding UserName, Mode=TwoWay}"/>
```

Commands

- Used to Execute a method when an action is performs, such as button click.
- Ability to pass parameter
- Ability to have CanExecute



```
public interface ICommand
{
    //
    // Methods
    //
    bool CanExecute (object parameter);

    void Execute (object parameter);

    //
    // Events
    //
    event EventHandler CanExecuteChanged
}
```

Commands

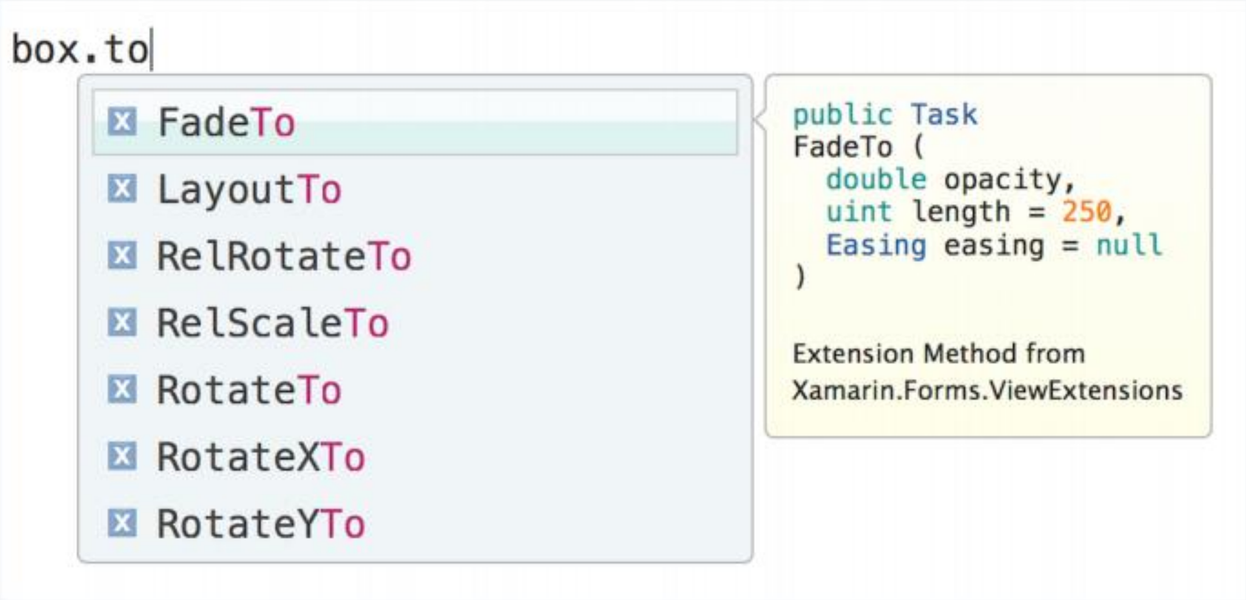
- Command type is part of the Xamarin Forms framework (no need for the 3rd Party).

```
  private Command _remindMeCommand;

public Command RemindMeCommand
{
    get
    {
        return _remindMeCommand ?? (_remindMeCommand = new Command(
            () =>
            {
                UserName = "Alon Fliess";
            }));
    }
}
```

```
<Button Text=“Remind Me…" Command="{Binding RemindMeCommand}"/>
```

Animations

- Cross-platform animations
- Platform-specific animation APIs
- Async/Await API

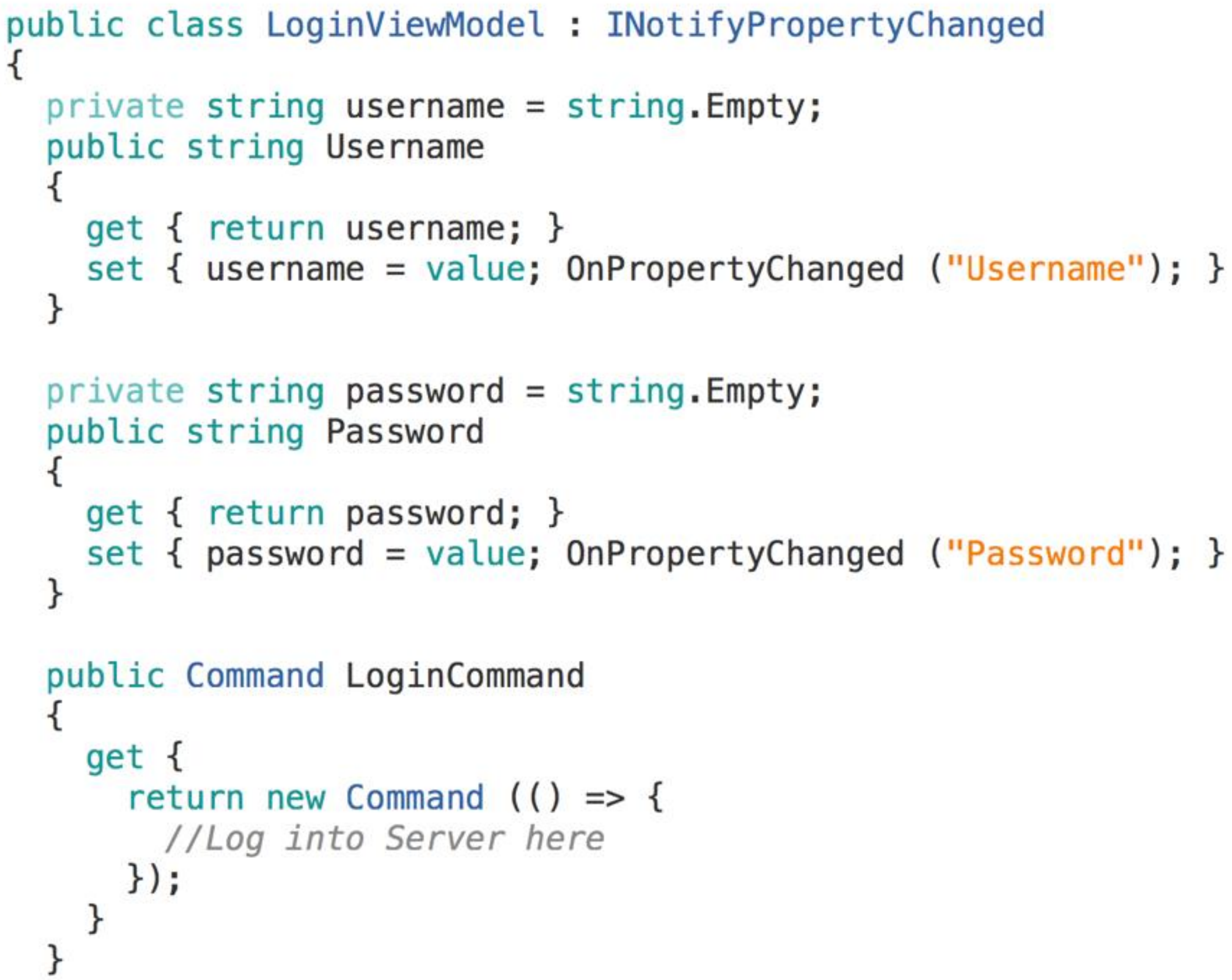


```
box.to
```

```
public Task
FadeTo (
  double opacity,
  uint length = 250,
  Easing easing = null
)
```

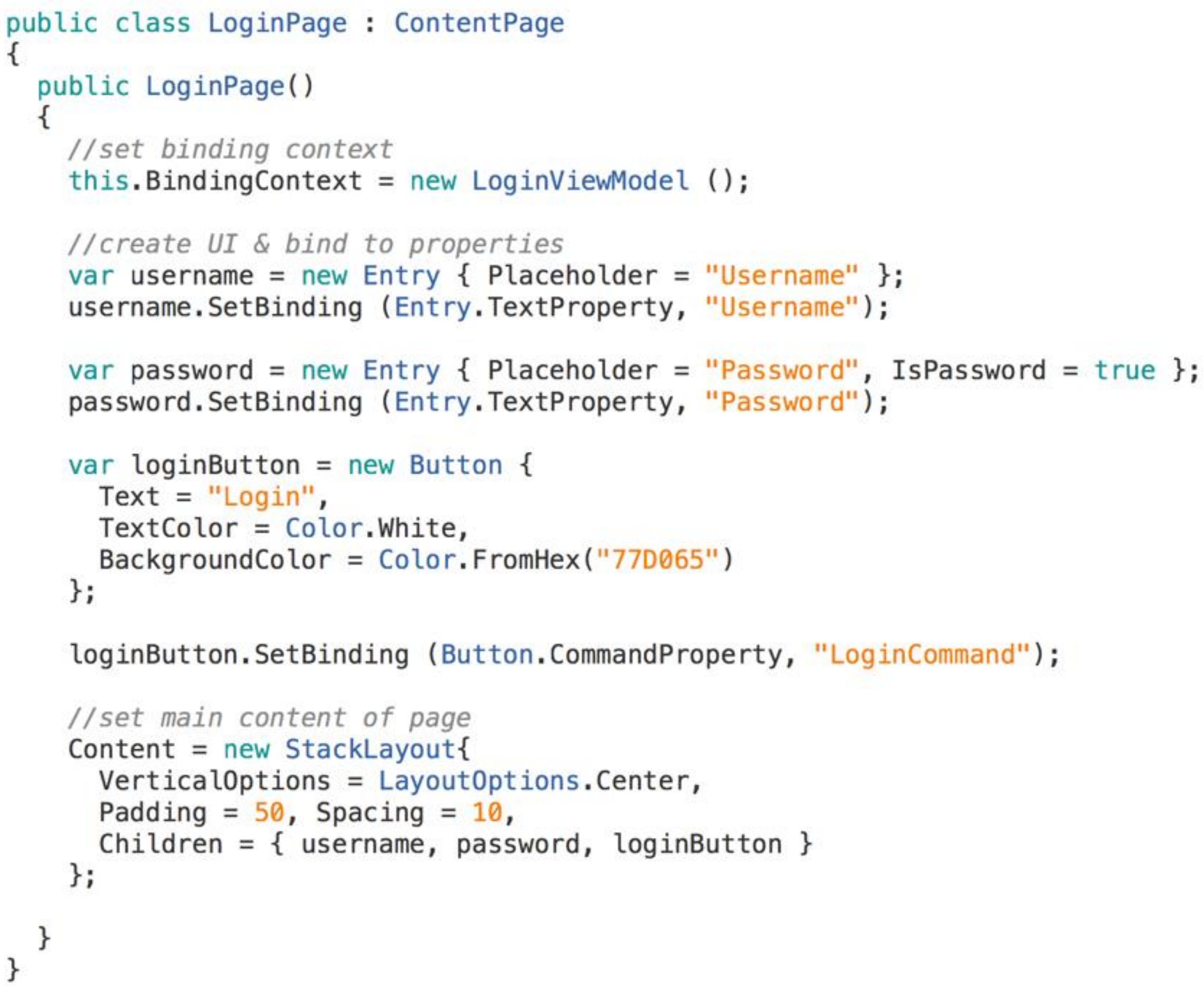
Extension Method from Xamarin.Forms.ViewExtensions

Login ViewModel



```
public class LoginViewModel : INotifyPropertyChanged
{
  private string username = string.Empty;
  public string Username
  {
    get { return username; }
    set { username = value; OnPropertyChanged ("Username"); }
  }

  private string password = string.Empty;
  public string Password
  {
    get { return password; }
    set { password = value; OnPropertyChanged ("Password"); }
  }

  public Command LoginCommand
  {
    get {
      return new Command (() => {
        //Log into Server here
      });
    }
  }
```

Login Page – Code Behind

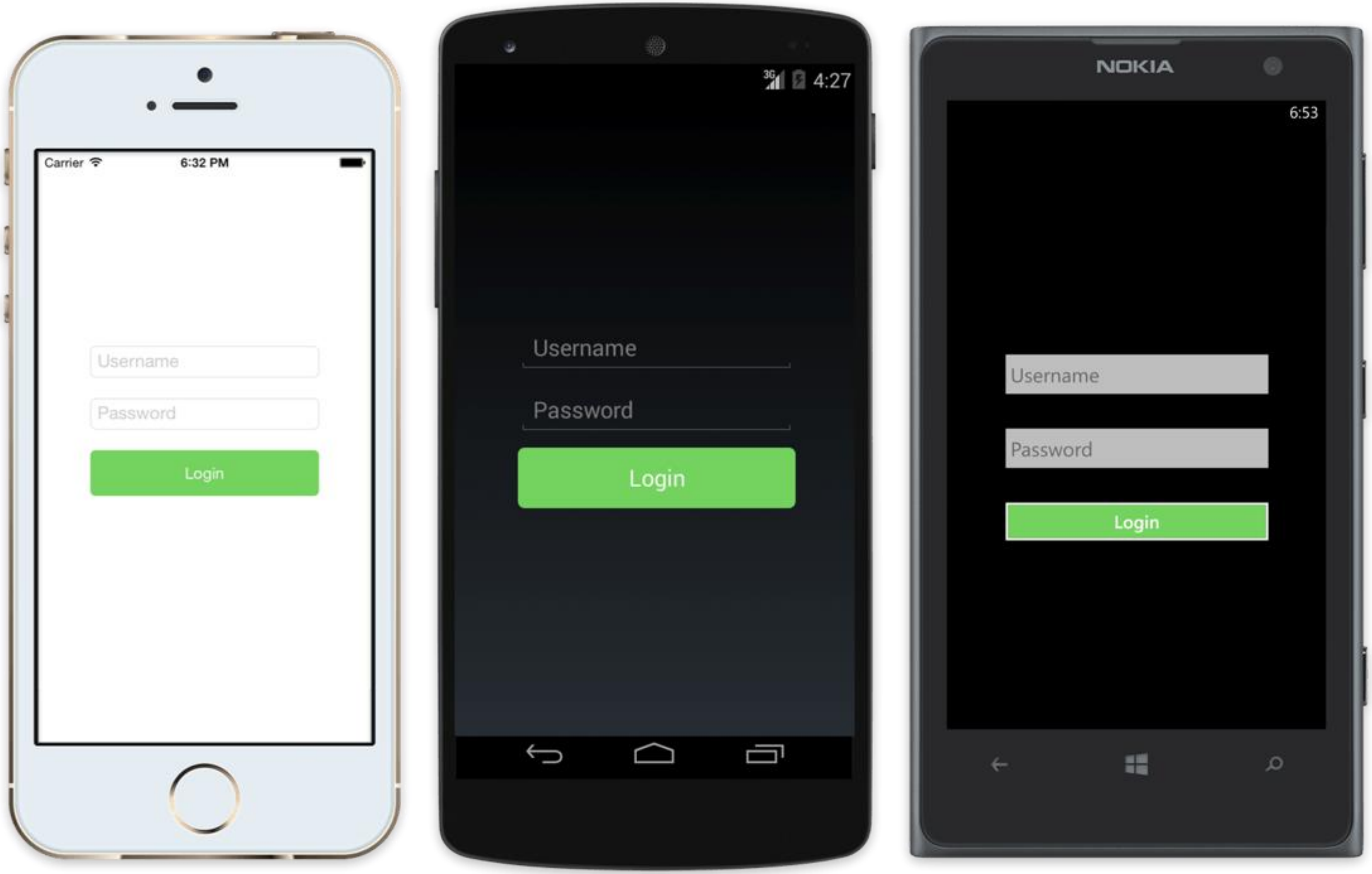


```
public class LoginPage : ContentPage
{
  public LoginPage()
  {
    //set binding context
    this.BindingContext = new LoginViewModel ();

    //create UI & bind to properties
    var username = new Entry { Placeholder = "Username" };
    username.SetBinding (Entry.TextProperty, "Username");

    var password = new Entry { Placeholder = "Password", IsPassword = true };
    password.SetBinding (Entry.TextProperty, "Password");

    var loginButton = new Button {
      Text = "Login",
      TextColor = Color.White,
      BackgroundColor = Color.FromHex("77D065")
    };

    loginButton.SetBinding (Button.CommandProperty, "LoginCommand");

    //set main content of page
    Content = new StackLayout{
      VerticalOptions = LayoutOptions.Center,
      Padding = 50, Spacing = 10,
      Children = { username, password, loginButton }
    };

  }
}
```

Login Page – XAML

```
<?xml version="1.0" encoding="UTF-8" ?>
<ContentPage
    xmlns="http://xamarin.com/schemas/2014/forms"
    xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
    x:Class="LoginExampleForms.LoginPageXAML">
    <ContentPage.Content>

    <StackLayout VerticalOptions="Center" Padding="50" Spacing="10">

        <Entry Placeholder="Username" Text="{Binding Username}"/>
        <Entry Placeholder="Password" Text="{Binding Password}"/>
        <Button Text="Login"
                TextColor="#FFFFFF"
                BackgroundColor="#77D065"
                Command="{Binding LoginCommand}"/>

    </StackLayout>

    </ContentPage.Content>
</ContentPage>
```

Login Page


Carrier
6:32 PM
Username
Password
Login
4:27
Username
Password
Login
NOKIA
6:53
Username
Password
Login

Summary

- Quickly and easily build native user interfaces using shared code
- Xamarin.Forms elements map to native controls and behaviors
- Mix-and-match Xamarin.Forms with native APIs

ISO 9001 SGS

XAML Fundamentals

Inspiring Code, Creating Value.

What is XAML?

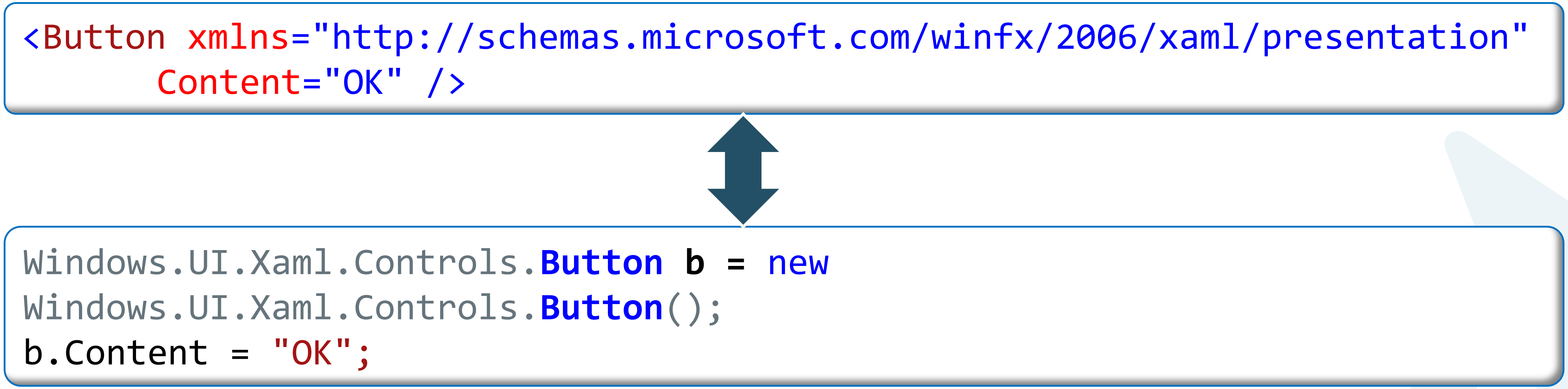
- XML based language
- Enable separation of UI and behavior (code)
- Windows Phone related tools emit XAML
- XAML allows
  - Creation of objects
  - Setting of properties
  - Connection to events
  - Custom behaviors
- XAML cannot call methods directly

XAML vs. Code

- Anything that can be done in XAML can be done in code
  - But not vice versa
- XAML is usually shorter and more concise than the equivalent code
  - Thanks to type converters and markup extensions
- XAML should be used for initial UI
- Code will handle events and change items dynamically

ISO 9001 SGS

Simple XAML Example



```
<Button xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
      Content="OK" />
```

```
Windows.UI.Xaml.Controls.Button b = new
Windows.UI.Xaml.Controls.Button();
b.Content = "OK";
```

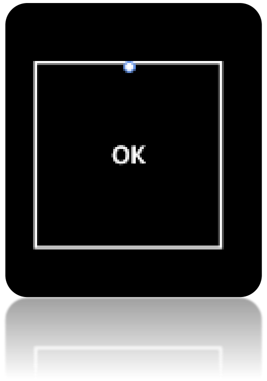
- Visual Studio UI designer generates XAML on each control picked from the toolbox
- XAML Can be visually viewed in the UI designer

XAML Namespaces

- The default XAML namespace is assigned a value that is mapped to some of the runtime namespaces contain UI elements
- Other XAML namespaces may be mapped to custom namespaces and other runtime namespaces
- The “x” namespace is mapped to a special namespace, contains XAML parser specific types
- XAML namespace can be defined on each element level

XAML Example

```
<!--ContentPanel - place additional content here-->
<Grid x:Name="ContentPanel" Grid.Row="1" Margin="12,0,12,0">
    <Button x:Name="buttonOk"
            Width="200"
            Height="200"
            Content="OK"
            Click="buttonOk_Click" />
</Grid>
```

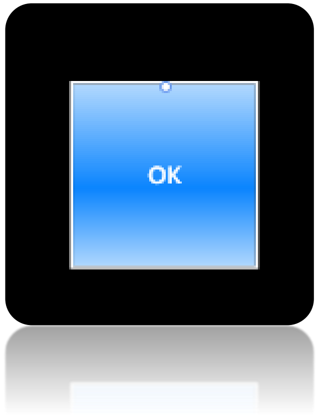


Elements and Attributes

- Elements with type names only designate object creation (via the default constructor)
- Attributes indicate property or event values
  - Event values are event handlers (methods) names
- Complex properties are designated using a `<Type.Property>` element

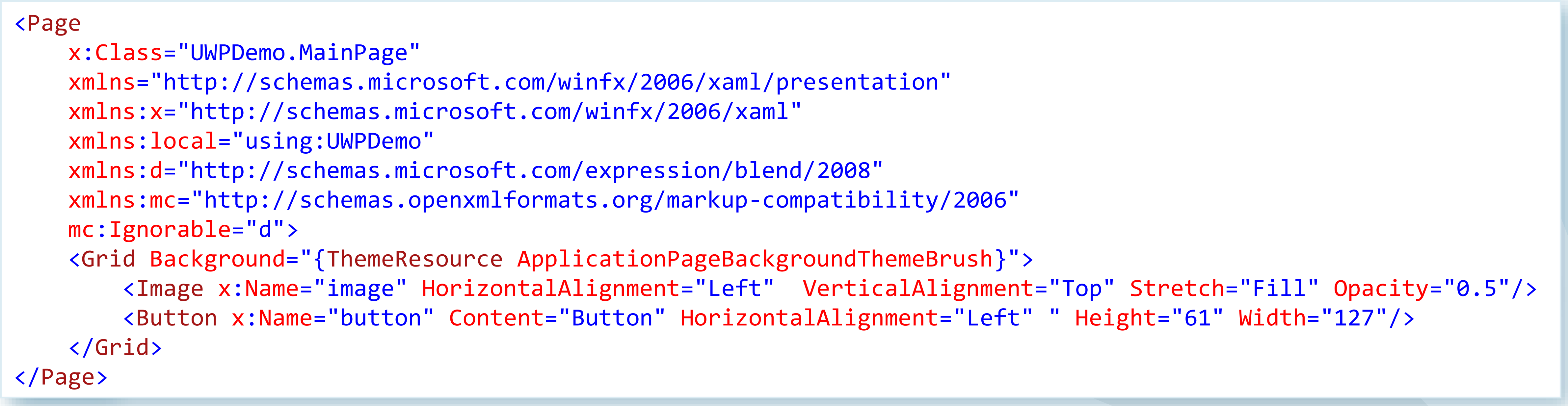
XAML Example

```
<!--ContentPanel - place additional content here-->
<Grid x:Name="ContentPanel" Grid.Row="1" Margin="12,0,12,0">
    <Button x:Name="buttonOk"
            Width="200"
            Height="200"
            Content="OK"
            Click="buttonOk_Click" >
        <Button.Background>
            <LinearGradientBrush EndPoint="0.5,1“
                                 StartPoint="0.5,0">
                <GradientStop Color="#FFB2D9FF" Offset="0.004"/>
                <GradientStop Color="#FFB0D8FF" Offset="1"/>
                <GradientStop Color="#FF0A85FF" Offset="0.571"/>
            </LinearGradientBrush>
        </Button.Background>
    </Button>
</Grid>
```



XAML And Code Behind

- A root element, usually **Page** or **UserControl** classes, can have code behind file
- The name of the code behind file is correlated to the XAML file name
- For example: MainPage.xaml and MainPage.xaml.cs
- The code behind full class name is specified from XAML using the **x:Class** directive



```
<Page
    x:Class="UWPDemo.MainPage"
    xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
    xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
    xmlns:local="using:UWPDemo"
    xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
    xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
    mc:Ignorable="d">
    <Grid Background="{ThemeResource ApplicationPageBackgroundThemeBrush}">
        <Image x:Name="image" HorizontalAlignment="Left"  VerticalAlignment="Top" Stretch="Fill" Opacity="0.5"/>
        <Button x:Name="button" Content="Button" HorizontalAlignment="Left" " Height="61" Width="127"/>
    </Grid>
</Page>
```

Child Elements

- Child elements (that are not property elements) can be one of
  - The **Content** property of the object
    - A property adorned with the attribute **Windows.UI.Xaml.Controls.ContentProperty**
  - Collection items
    - The object implements **`IList`** or **`IDictionary`**
  - A value that can be type-converted

Content Property

- A single property that is designated with the **ContentProperty** attribute on the type
- Allows shortening the markup

```
<Button Content="OK" >
</Button>
```

⟷

```
<Button>
      OK
</Button>
```

```
<Button>
      <Button.Content>
         <Rectangle Fill="Blue"/>
      </Button.Content>
</Button>
```

⟷

```
<Button>
    <Rectangle Fill="Blue"/>
</Button>
```

Collection Items

- List (IList)

```
<ListBox>
<ListBox.Items>
<ListBoxItem Content="Item 1"/>
<ListBoxItem Content="Item 2"/>
</ListBox.Items>
</ListBox>
```

- Dictionary (IDictionary)

```
<ResourceDictionary>
<SolidColorBrush x:Key="br1" Color="Aqua" />
<Rectangle x:Key="rc1" Fill="Brown" />
</ResourceDictionary>
```

Summary of XAML Rules

- XML Element – create a new instance
- XML attribute – set a property or register an event
  - Type converter may execute
- **`Type.Property`** – set a "complex" property
- **`ContentProperty`** attribute – no need to specify **`Type.Property`**
- Property of type **`IList`** or **`IDictionary`**
  - Add child elements (XAML calls appropriate **Add** method)
  - Need a **`x:Key`** in case of a dictionary

Naming Elements

- Elements can be named using the `x:Name` XAML attribute
- The code-behind file will contain a field with that name
- Elements deriving from `FrameworkElement` contain a `Name` property that can be used in code to locate elements
  - `x:Name` and `Name` cannot be set on the same element

XAML Keywords

Keyword	Valid on	Meaning
x:Class	Root element	The class that derives from the element type
x:ClassModifier	Root element, must be used with x:Class	The class visibility (public by default)
x:FieldModifier	Element, must be used with x:Name	Visibility of the field created behind the element
x:Key	Element that its parent implements IDictionary	Key in a dictionary
x:Name	Element	The element’s name, used for a field name for that element
X:Uid	Element	Identifies elements that should use localized resources

Mapping custom types to XAML namespaces

- You can define your own custom types in C# and then reference your custom types in XAML markup
- To use XAML for custom types - those that come from libraries other than the Windows Runtime core libraries:
  - You must declare and map a XAML namespace with a prefix
  - Use that prefix in element usages to reference the types that were defined in your library
  - You declare prefix mappings as xmlns attributes
- For example:
  - the attribute syntax to map a prefix myTypes to the namespace myCompany.myTypes is
  - xmlns:myTypes="using:myCompany.myTypes"
  - The representative element usage is: <myTypes:CustomButton/>

XAML Markup Extensions

- Represent some kind of "shortcut" that enables a XAML file to access a value or behavior that isn't simply declaring elements based on backing types
- In XAML attribute syntax, curly braces "{" and "}" indicate a XAML markup extension usage
- A XAML parser calls code that provides behavior for that particular markup extension
  - That code provides an alternate object or behavior result that the XAML parser needs
- Examples:
  - {x:Bind} {Binding} {StaticResource} {ThemeResource} {TemplateBinding} {RelativeSource} {CustomResource} {x:Null}

Markup Extension Example

```
<Canvas.Resources>
    <Style TargetType="Border" x:Key="PageBackground">
        <Setter Property="BorderBrush" Value="Blue"/>
        <Setter Property="BorderThickness" Value="5"/>
    </Style>
    </Canvas.Resources>
        ...
    <Border Style="{StaticResource PageBackground}">
        ...
    </Border>
```

XAML and .NET Events

- XAML has a syntax for attaching event handlers to objects in the markup
- You specify the name of the event as an attribute name on the object where the event is handled
  - For the attribute value, you specify the name of an event-handler function that you define in code
- The XAML processor uses this name to create a delegate representation in the loaded object tree, and adds the specified handler to an internal handler list

```
<Button Click="showUpdatesButton_Click">Show updates</Button>
```

Summary

- XAML is mainly used to create a Windows app user interface
- It declaratively allows object creation, property and event assignment
- A code-behind file will usually contain the procedural logic
- Sharing with designers is easier
- Tools such as Expression Blend generate XAML that is immediately usable

Event Processing

Azure Stream Analytics

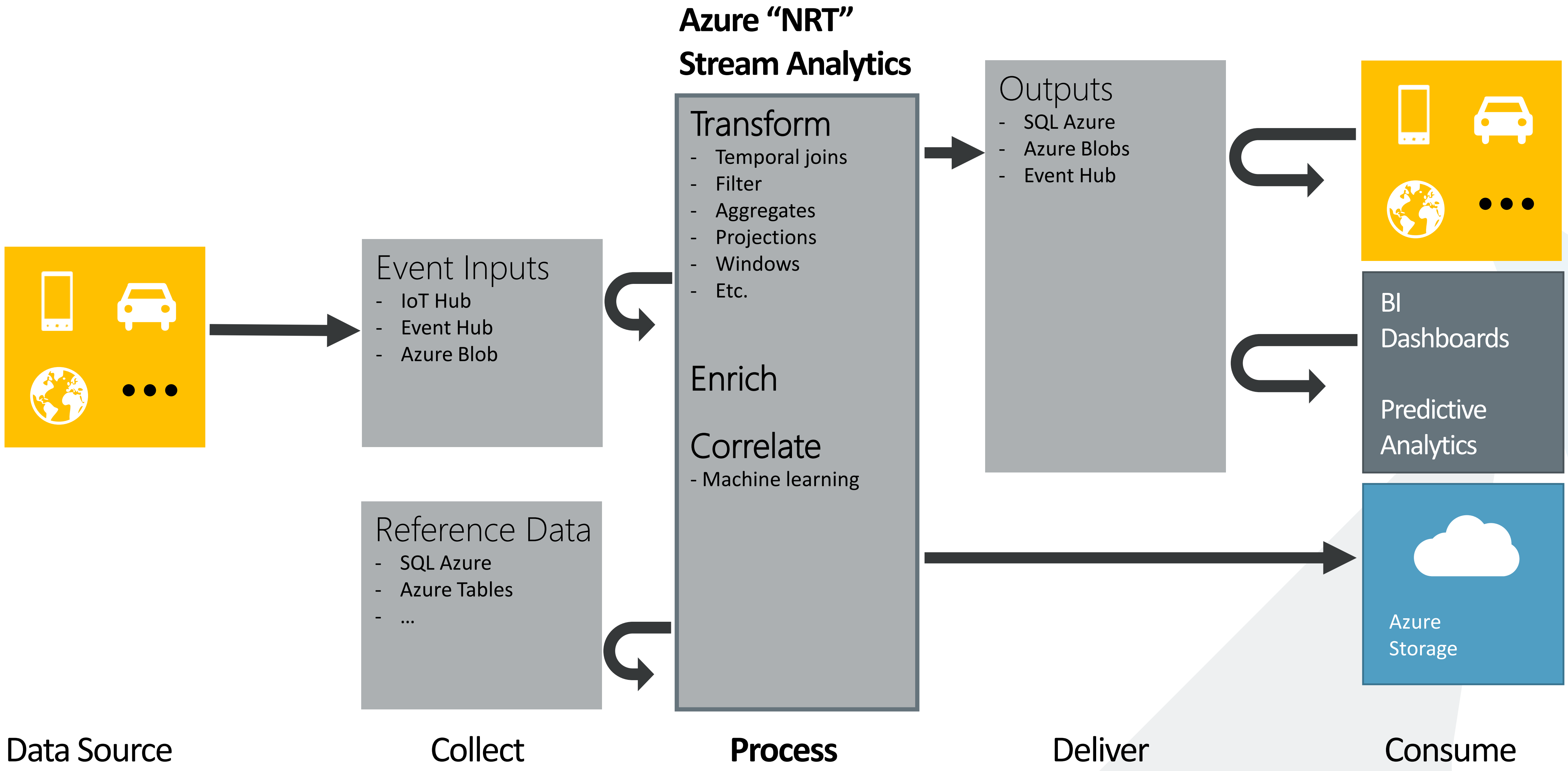
Real time event processing

- **Uncover real time insights**
  - Perform real time analytics across multiple streams
- **Rapid Deployment**
  - Use simple SQL syntax, auto distributed for scale
- **Mission critical reliability**
  - Fully managed, low latency, high throughput
- **Create real time alerts**
  - Flag alerts and alarms for attention

- **High volume**
  - Analyze millions of data points per second
- **Highly scalable**
  - Enterprise grade, predictable solution.

ISO 9001 SGS

Streaming Architecture



Azure "NRT" Stream Analytics

Event Inputs
- IoT Hub
- Event Hub
- Azure Blob

Reference Data
- SQL Azure
- Azure Tables
- …

Transform
- Temporal joins
- Filter
- Aggregates
- Projections
- Windows
- Etc.

Enrich

Correlate
- Machine learning

Outputs
- SQL Azure
- Azure Blobs
- Event Hub

BI Dashboards

Predictive Analytics

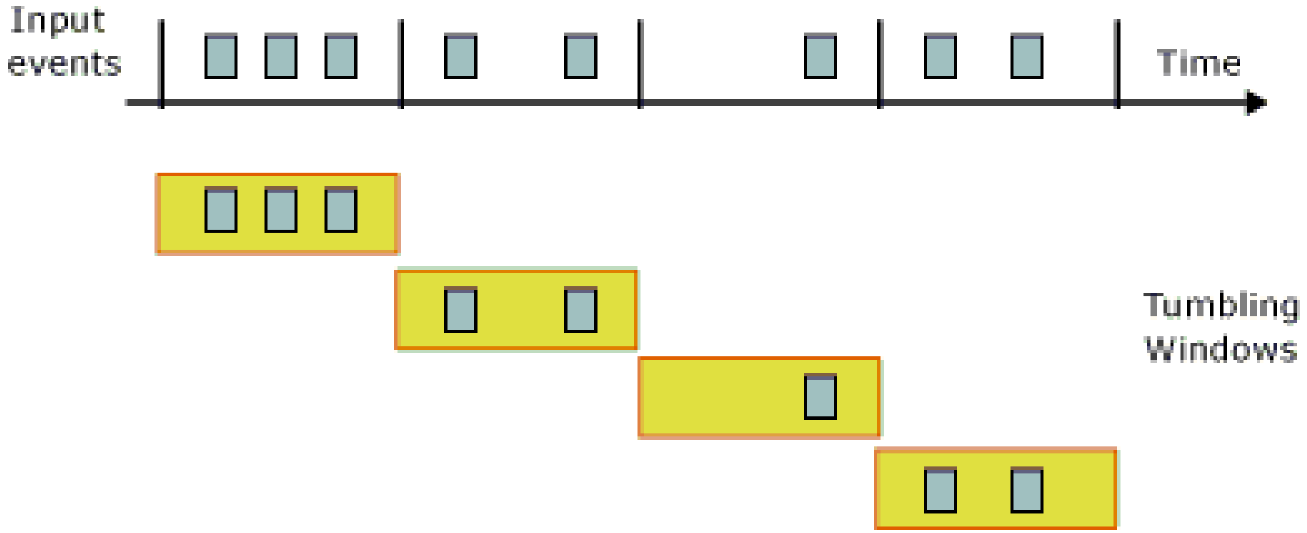
Azure Storage

Data Source | Collect | **Process** | Deliver | Consume

ISO 9001 SGS

Stream Analytics - Tumbling Windows

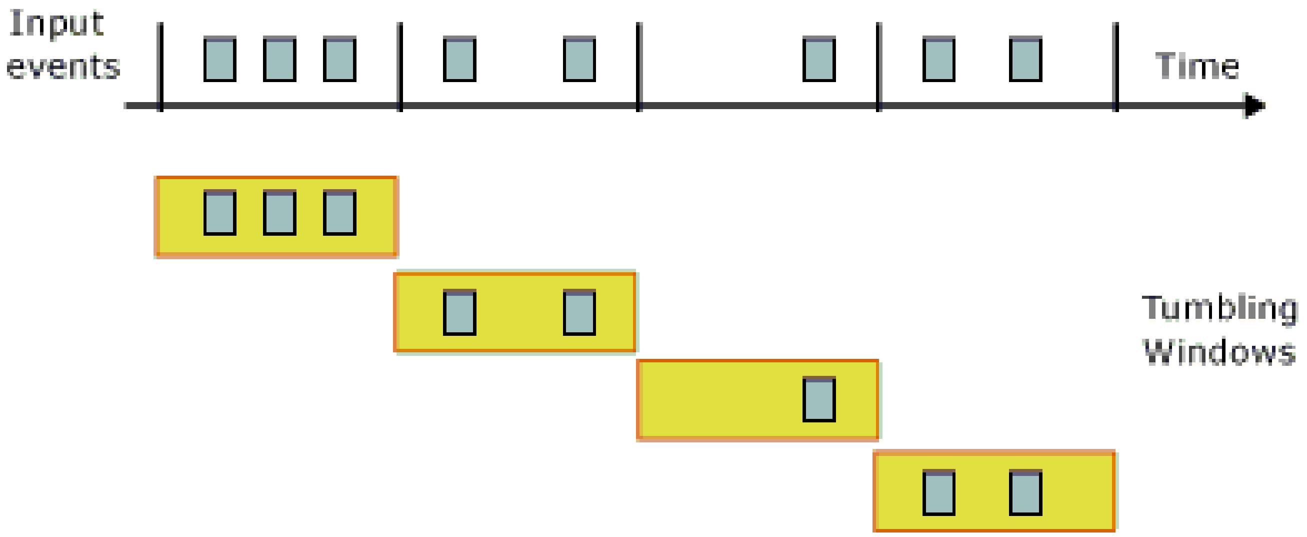
- How many vehicles entered each toll booth every 5 minutes?



```
SELECT TollId, COUNT(*) FROM EntryStream
GROUP BY TollId, TumblingWindow(minute,5)
```

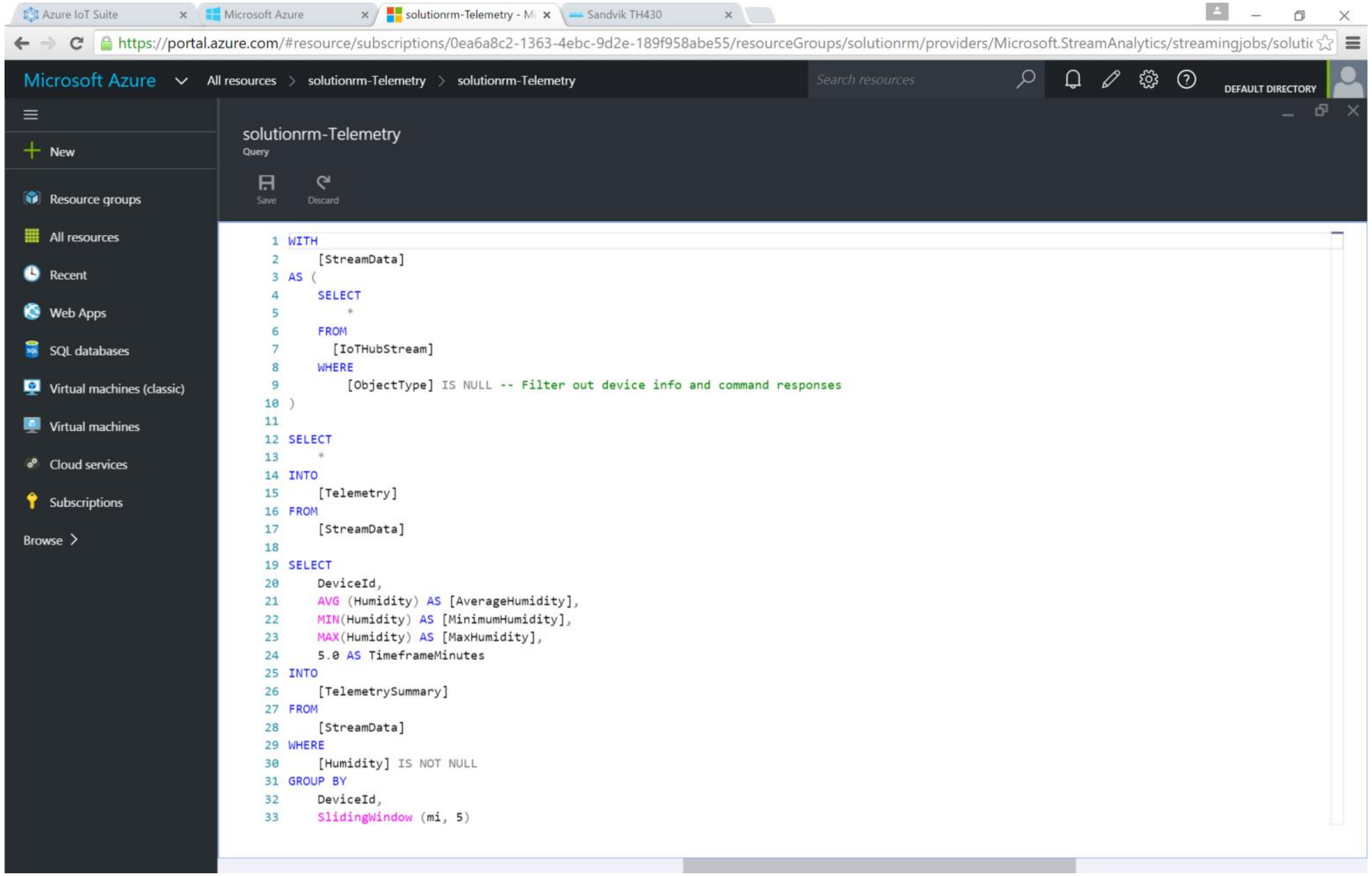
Stream Analytics - Tumbling Windows

How many vehicles entered each toll booth every 5 minutes?



```
SELECT TollId, COUNT(*) FROM EntryStream
GROUP BY TollId, TumblingWindow(minute,5)
```

Add or edit jobs using simple interface



```
WITH
    [StreamData]
AS (
    SELECT
        *
    FROM
      [IoTHubStream]
    WHERE
        [ObjectType] IS NULL -- Filter out device info and command responses
)

SELECT
    *
INTO
    [Telemetry]
FROM
    [StreamData]

SELECT
    DeviceId,
    AVG (Humidity) AS [AverageHumidity],
    MIN(Humidity) AS [MinimumHumidity],
    MAX(Humidity) AS [MaxHumidity],
    5.0 AS TimeframeMinutes
INTO
    [TelemetrySummary]
FROM
    [StreamData]
WHERE
    [Humidity] IS NOT NULL
GROUP BY
    DeviceId,
    SlidingWindow (mi, 5)
```

Rule based interface

Simple implementation and rule development using ASA UI.

Multi-channel

Analyze multiple channels of information simultaneously, in real time.

Data Visualization

Power BI

Data visualization with PowerBI

Highly accessible analytics

Cloud based dashboard and analytics tool,

360º view of business KPI's

Customize dashboards to address concerns and performance metrics.

Cross platform support

View data via web platform, on any device

Pre-built dashboards

Utilize standard dashboards for rapid deployment, based on popular solution demands.

Real time capabilities

Ingest, analyze and display data as it happens

Secure access

Secure, live communication with data source

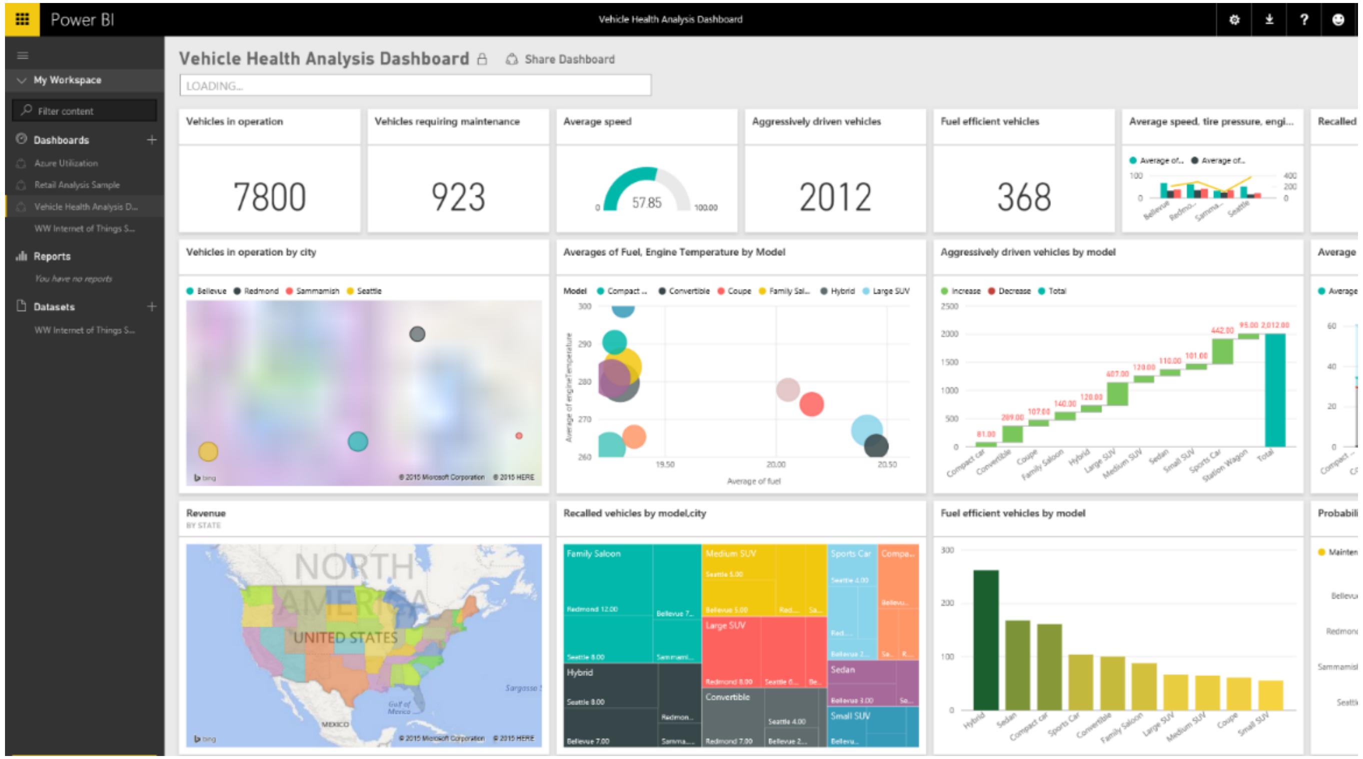
Query data

Intuitive, natural language query tool

Integrated systems

Integrate with other business systems and enrich device data with intelligence from other business systems, eg: CRM, ERP

Data visualization with PowerBI



Rich visuals

Standard and custom graphing options

Custom dashboards

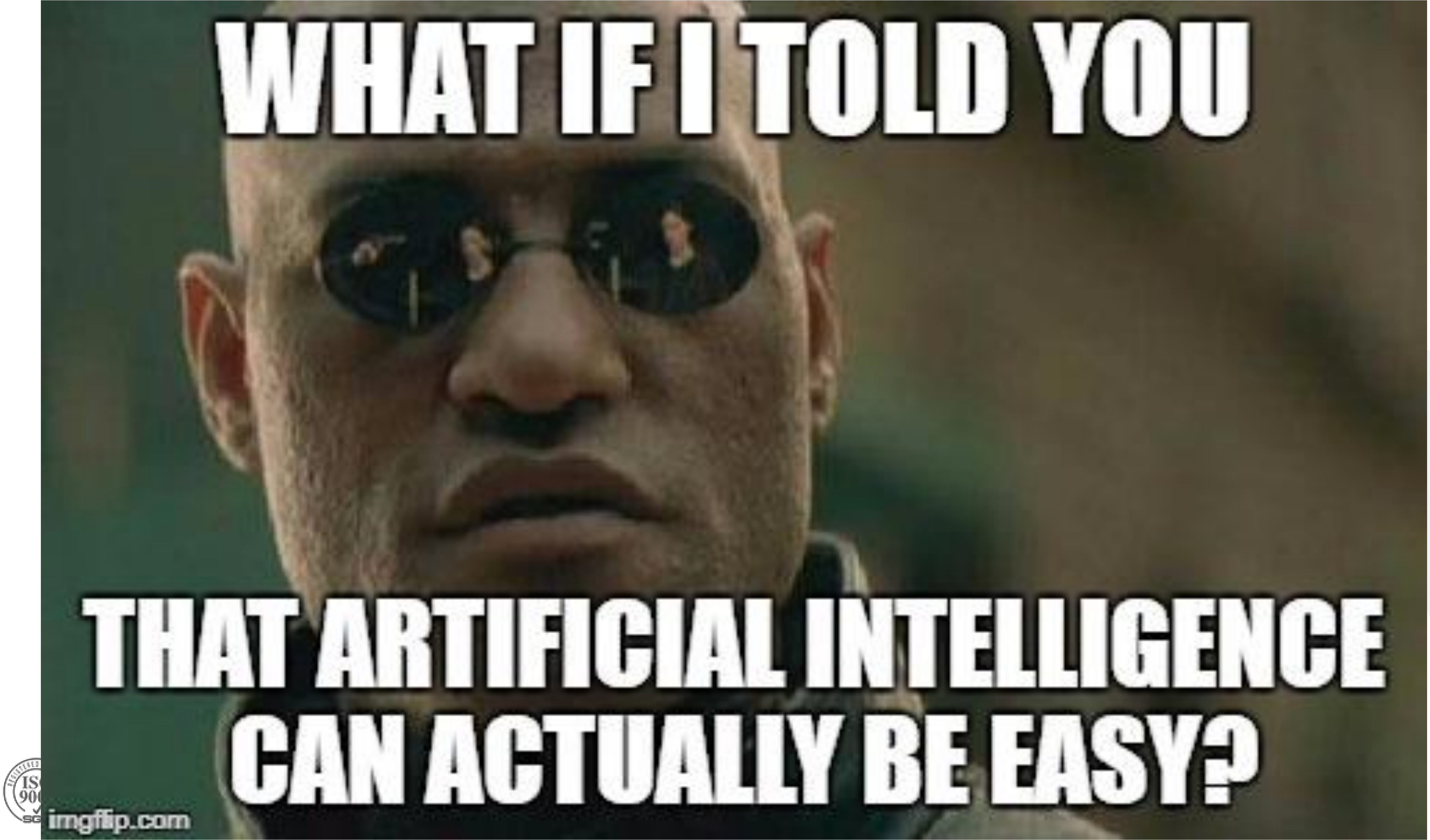
Build heat maps and visually track data

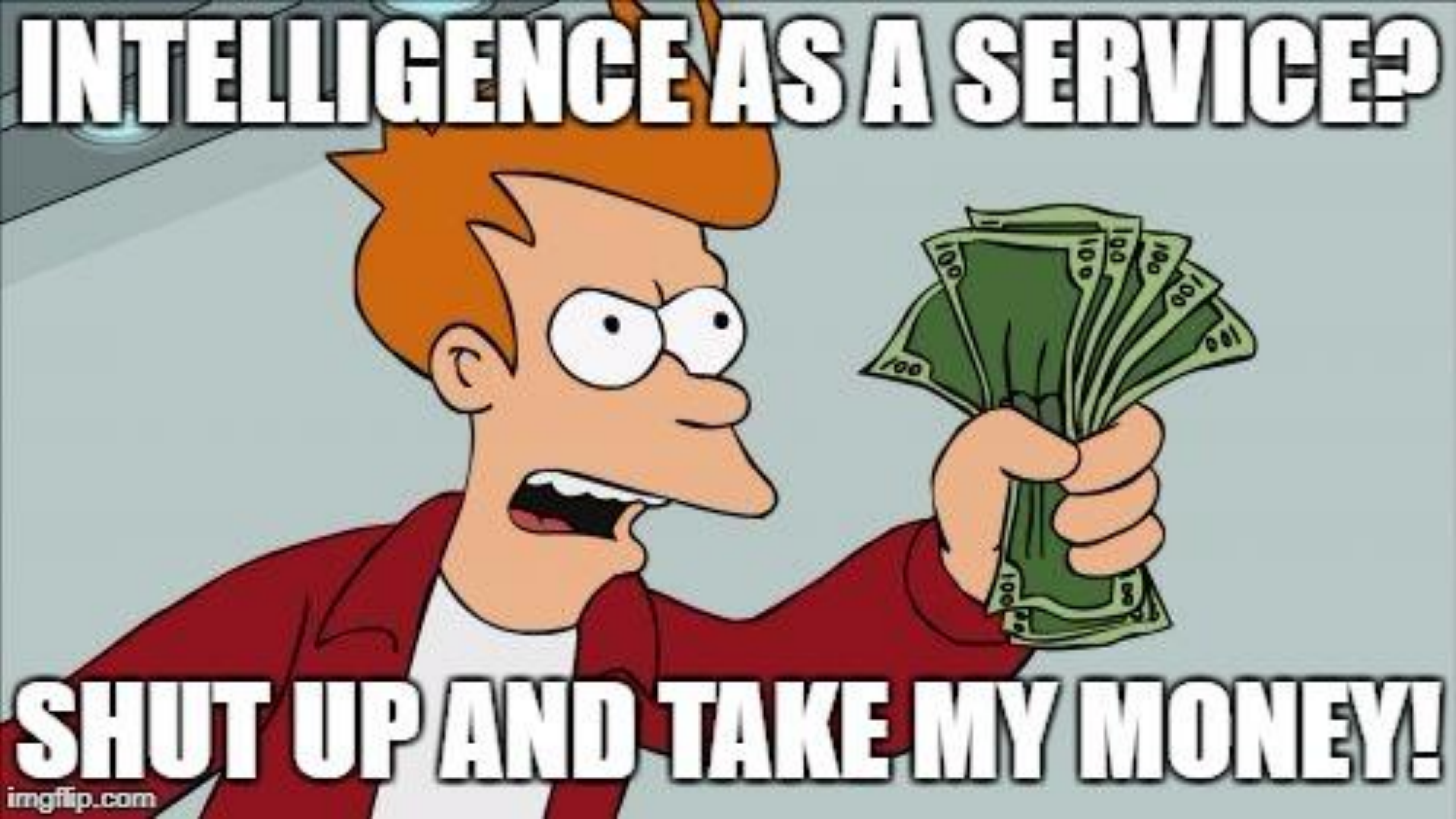
External data

Integrate external data feeds to add value to device data, or pull in external information such as weather or market information.

Artificial Intelligence

ALRIGHT TEAM, WE NEED TO ADD NEW FEATURES TO OUR APP AND WE NEED TO DO IT FAST. ANY IDEAS?
NEW THEME!
IPAD APP!
ARTIFICIAL INTELLIGENCE?
imgflip.com

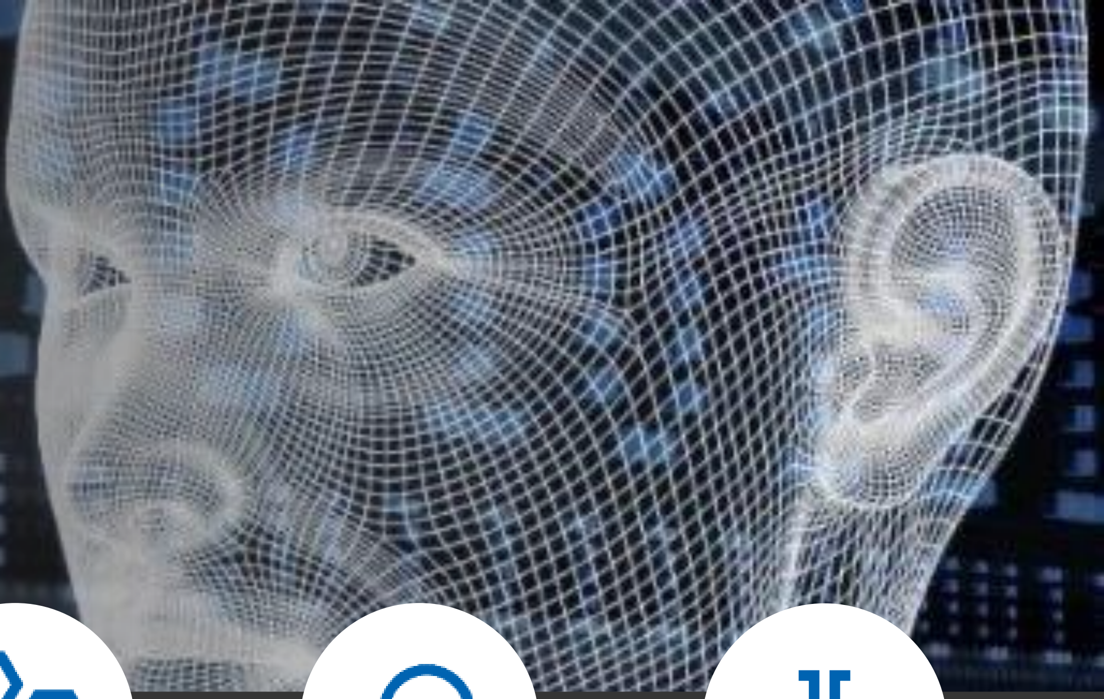

WHAT IF I TOLD YOU
THAT ARTIFICIAL INTELLIGENCE
CAN ACTUALLY BE EASY?
imgflip.com


INTELLIGENCE AS A SERVICE?
SHUT UP AND TAKE MY MONEY!
imgflip.com

Microsoft Cognitive Services

Microsoft Cognitive Services

Give your apps a human side



Vision

From faces to feelings, allow your apps to understand images and video

Speech

Hear and speak to your users by filtering noise, identifying speakers, & understanding intent

Language

Process text and learn how to recognize what users want

Knowledge

Tap into rich knowledge amassed from the web, academia, or your own data

Search

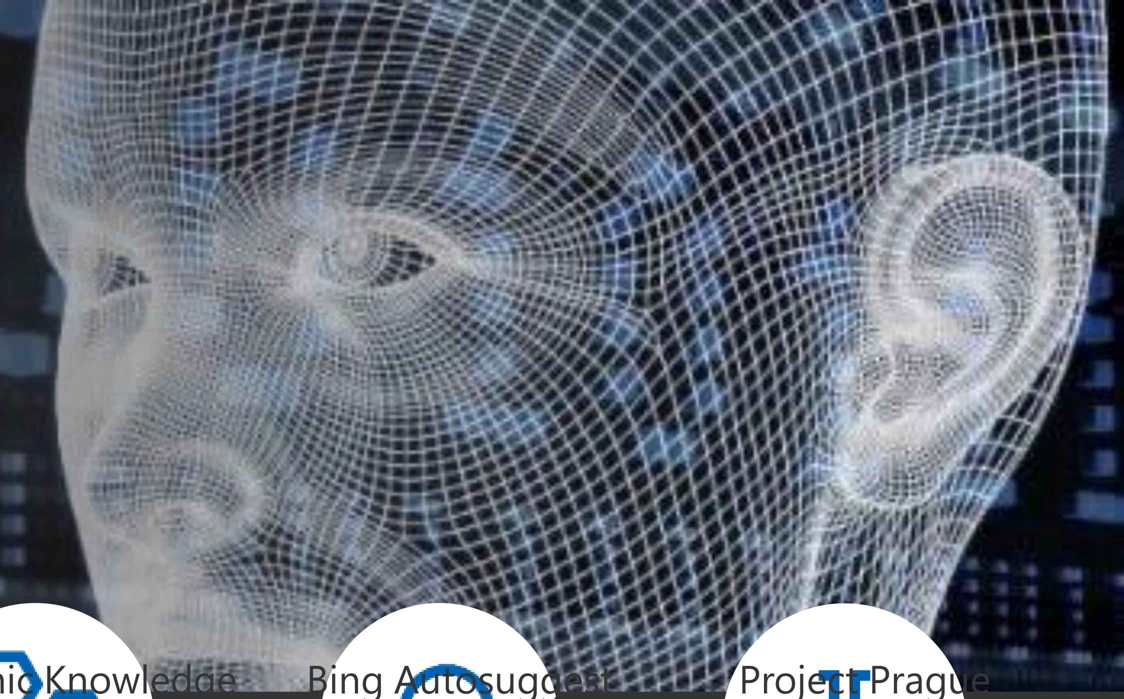
Access billions of web pages, images, videos, and news with the power of Bing APIs

Labs

An early look at emerging Cognitive Services technologies: discover, try & give feedback on new technologies before general availability

Microsoft Cognitive Services

Give your apps a human side



Vision

omputer Visio

Content Moderator

Emotion

Face

Video

Video Indexer

Custom Vision Service

Speech

Bing Speech

Speaker Recognition

Translator Speech

Custom Speech Service

Language

Bing Spell Check

Linguistic Analysis

Text Analytics

Translator Text

Web Language Model

Language Understanding

Knowledge

Academic Knowledge

Entity Linking

Knowledge Exploration

Recommendations

QnA Maker

Custom Decision Service

Search

Bing Autosuggest

Bing Image Search

Bing News Search

Bing Video Search

Bing Web Search

Bing Entity Search

Bing Custom Search

Labs

Project Prague (gesture)

Project Cuzco (events)

Project Johannesburg (routing)

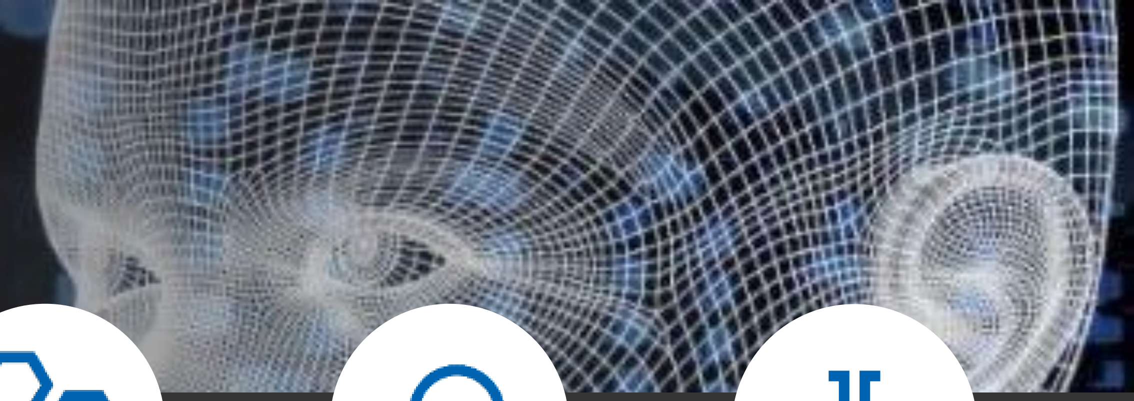
Project Nanjing (isochrones)

Project Abu Dhabi (distance matrix)

Project Wollongong (location)

Microsoft Cognitive Services

Give your apps a human side



Vision

Computer Vision

Content Moderator

Emotion

Face

Video

Video Indexer

Speech

Bing Speech

Speaker Recognition

Translator Speech

Language

Bing Spell Check

Linguistic Analysis

Text Analytics

Translator Text

Web Language Model

Knowledge

Academic Knowledge

Entity Linking

Knowledge Exploration

Recommendations

QnA Maker

Search

Bing Autosuggest

Bing Image Search

Bing News Search

Bing Video Search

Bing Web Search

Bing Entity Search

Labs

Project Prague (gesture)

Project Cuzco (events)

Project Johannesburg (routing)

Project Nanjing (isochrones)

Project Abu Dhabi (distance matrix)

Project Wollongong (location)

CUSTOMIZATION

Custom Vision Service

Custom Speech Service

Language Understanding

Custom Decision Service

Bing Custom Search

Why Microsoft Cognitive Services?

Easy

Roll your own with REST APIs

Simple to add: just a few lines of code required

GET A KEY

BUILD

Flexible

Integrate into the language and platform of your choice

Breadth of offerings helps you find the right API for your app

Bring your own data for your custom experience

node JS

python

Tested

Built by experts in their field from Microsoft Research, Bing, and Azure Machine Learning

Quality documentation, sample code, and community support

GitHub

stackoverflow

msdn

uservoice

A variety of real-world applications

Vision

What is in the image?

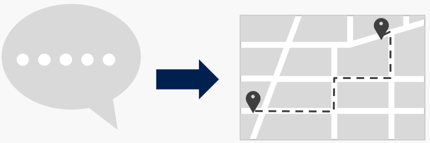
Computer Vision

Category	People; 5 faces
Adult/Racy?	False/False
Dominant colors	□ ■ ■ ■
Accent color	

Speech

Give me directions to the nearest local branch.

Bing Speech



Convert spoken audio to text

Convert text to spoken audio

Extract intent of user

Language

Play today's customer call recording.

Language Understanding

Natural Language Processing

```
Intent: PlayCall
Content: Customer#
DateTime.date: today
```

Now Playing

11/29/2016 Customer Call

Knowledge

Top publications in customer lifecycle trends?

Knowledge Exploration

Here are the top results:

Customer Relationship Management – 5 Key Trends for 2014 CRM

Oct 28, 2015 – Here are FIVE key **trends** in 2014 that would help marketers in rolling ... Of late, marketers are looking at **customer lifecycle** management (CLM)

Predictive Customer Lifecycle Management (CLM)

The purpose of **Customer Life-cycle** Management (CLM) is to maximize both customer retention and .... Predictive trend analysis provides business visibility.

Trends 2016: The Future of Customer Service

Jan 5, 2016 – The top 10 **customer** service **trends** for 2016 that .... North American Consumer

Language Around Customer Lifecycles in the Banking Industry

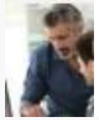
View PDF

Search

Search for 'fraud prevention'

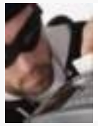
Bing News Search

Here is what I found:



Information Communications Media Market News

It also investigates the top three expected **Fraud** Detection and **Prevention** programs, in terms of demand in key markets...



The Big Question: In-House or Outsourced Fraud Protection?

First, let's point out that there is not one absolute answer—there are "pros" and "cons" to each. Those who favor in-house...

How to Protect Your Business from Online Fraud this Holiday Season

Michael heads fraud prevention tool. Online and mobile shopping are expected to continue growing apace...

How do I use them?

```
using Microsoft.ProjectOxford.Vision;
using Microsoft.ProjectOxford.Vision.Contract;

AnalysisResult analysisResult;
var features = new VisualFeature[] { VisualFeature.Ta
VisualFeature.Description };

using (var fs = new FileStream(@"C:\Vision\Sample.jpg
{
  analysisResult = await visionClient.AnalyzeImageAsy
}
```

```
POST https://api.projectoxford.ai/vision/v1.0/analyze
&subscription-key=<Your subscription key>
```

```
{
  "tags": [
    { "name": "outdoor",
      "score": 0.976 },
    { "name": "bird",
      "score": 0.95 } ],
  "description":
    { "tags":
      [ "outdoor", "bird" ],
      "captions": [
      { "text”: “partridge
          in a pear tree”,
        “confidence”: 0.96 }
      ]
    }
}
```

Vision

Vision

Computer Vision API
Distill actionable information from images

Face API
Detect, identify, analyze, organize, and tag faces in photos

Emotion API
Personalize experiences with emotion recognition

Video API
Analyze, edit, and process videos within your app

Content Moderator
Machine-assisted moderation of text and images, augmented with human review tools

Custom Vision Service
Customizable web service that learns to recognize specific content in imagery

Video Indexer
Process and extract smart insights from videos

Computer Vision API

Analyze an image

Understand content within an image

OCR

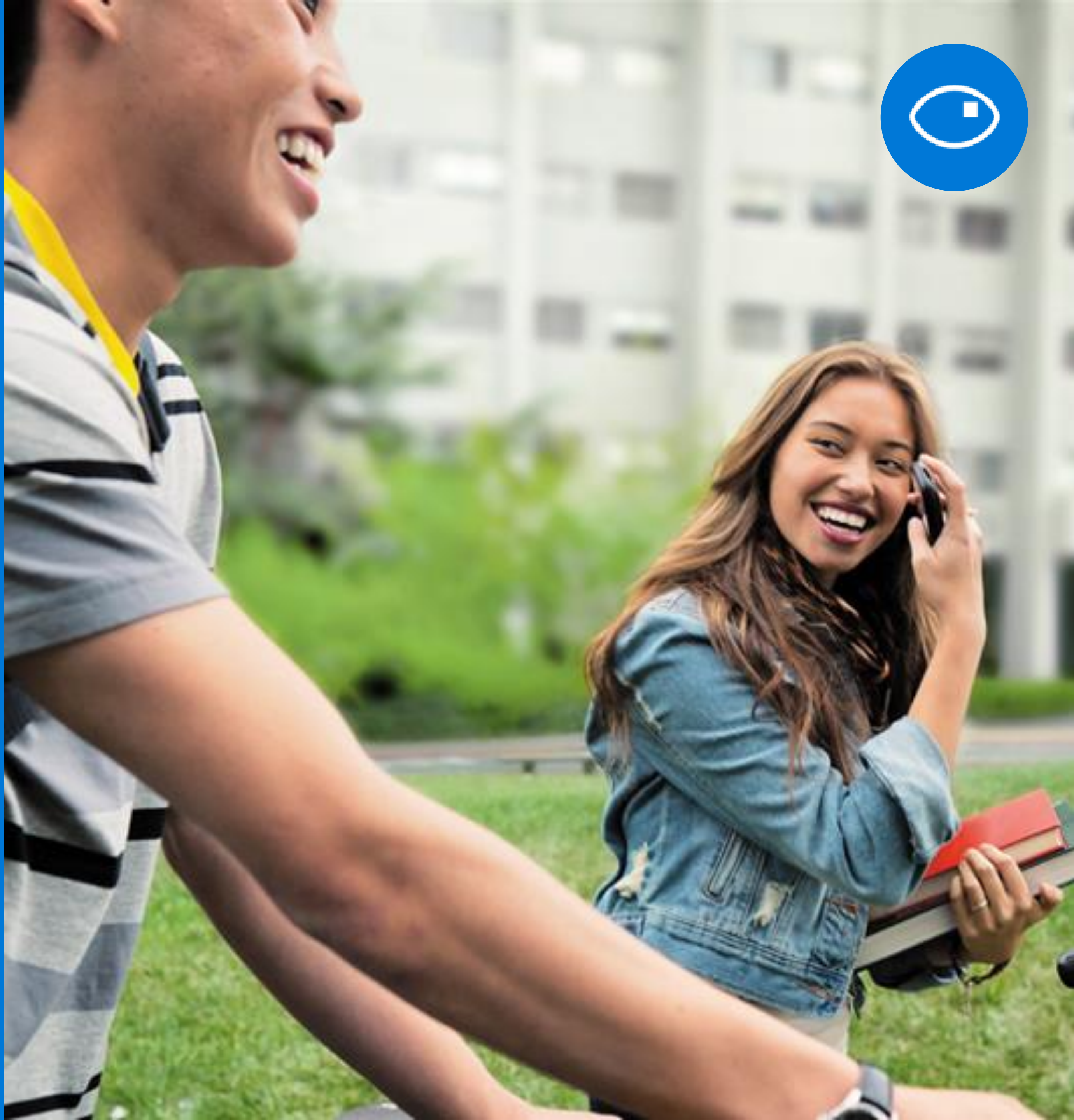
Detect and recognize words within an image

Generate thumbnail

Scale and crop images, while retaining key content

Recognize celebrities

Thanks to domain specific models, ability to recognize 200K celebrities from business, politics, sports, and entertainment around the world



Analyze image

Type of image

Clip Art Type	0 Non-clipart
Line Drawing Type	0 Non-Line Drawing
Black & White Image	False

Content of image

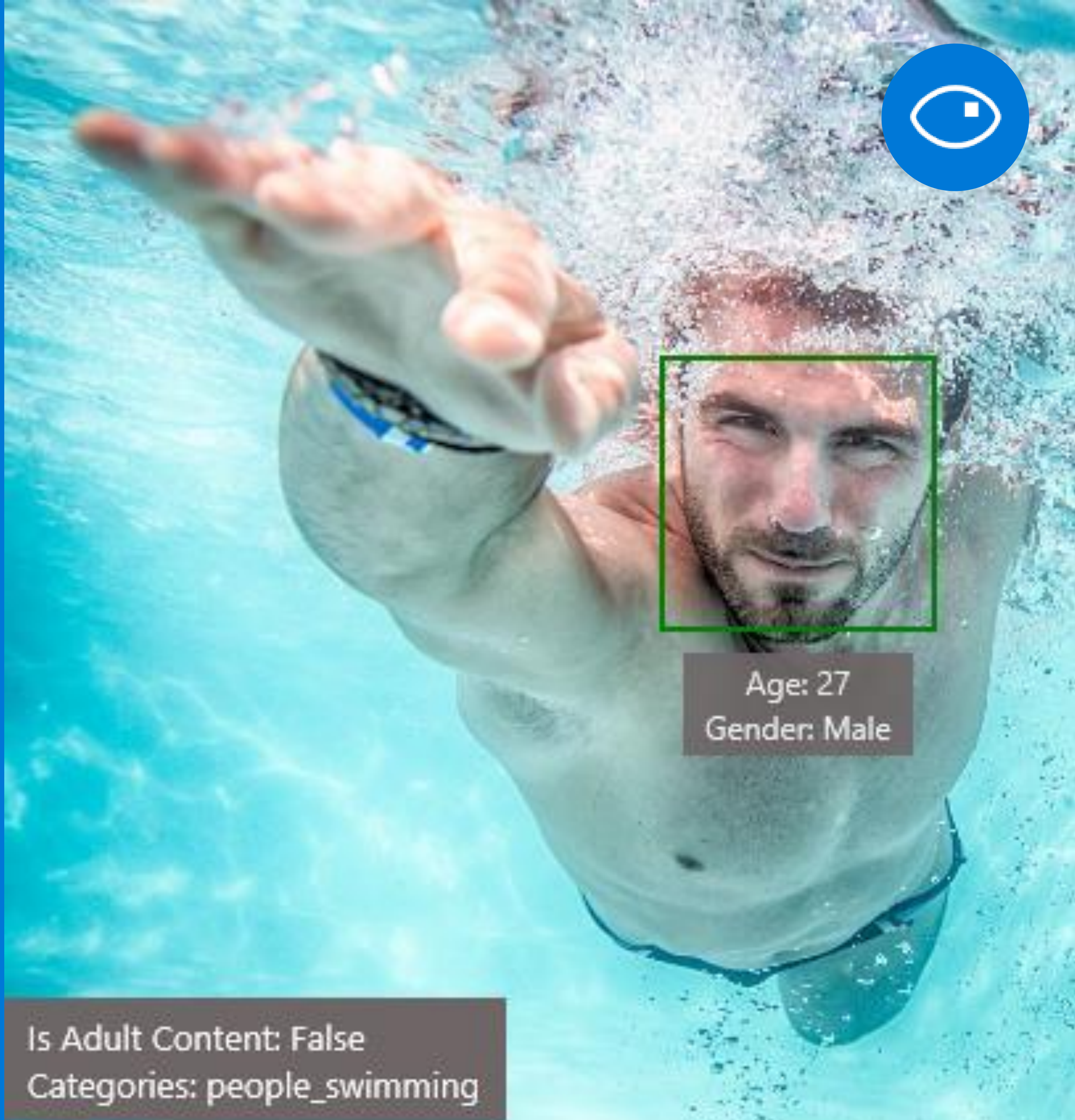
Categories	`[{ “name”: “people_swimming”,`
	`“score”: 0.099609375 }]`

Adult Content	False
Adult Score	0.18533889949321747

Faces	`[{ “age”: 27, “gender”: “Male”,`
	`“faceRectangle”:`
	`{“left”: 472, “top”: 258, “width”: 199,`
	`“height”: 199}}]`

Image colors

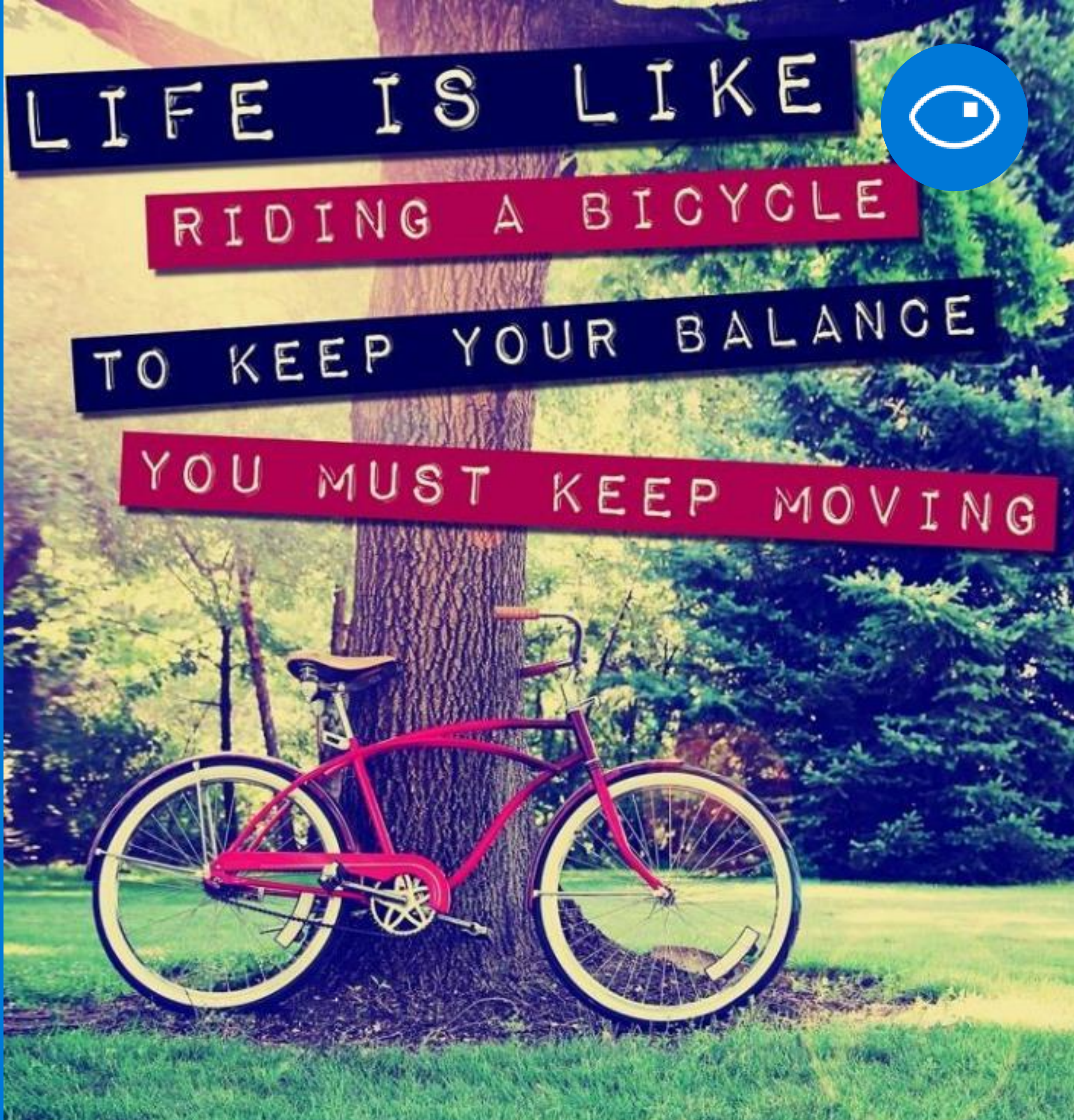
Dominant Color Background	White
Dominant Color Foreground	Grey
Dominant Colors	White
Accent Color	



OCR

JSON:

```
{
 "language": "en",
 "orientation": "Up",
 "regions": [
  {
   "boundingBox": "41,77,918,440",
   "lines": [
    {
     "boundingBox": "41,77,723,89",
     "words": [
      {
       "boundingBox": "41,102,225,64",
       "text": "LIFE"
      },
      {
       "boundingBox": "356,89,94,62",
       "text": "IS"
      },
      {
       "boundingBox": "539,77,225,64",
       "text": "LIKE"
      }
. . .
```



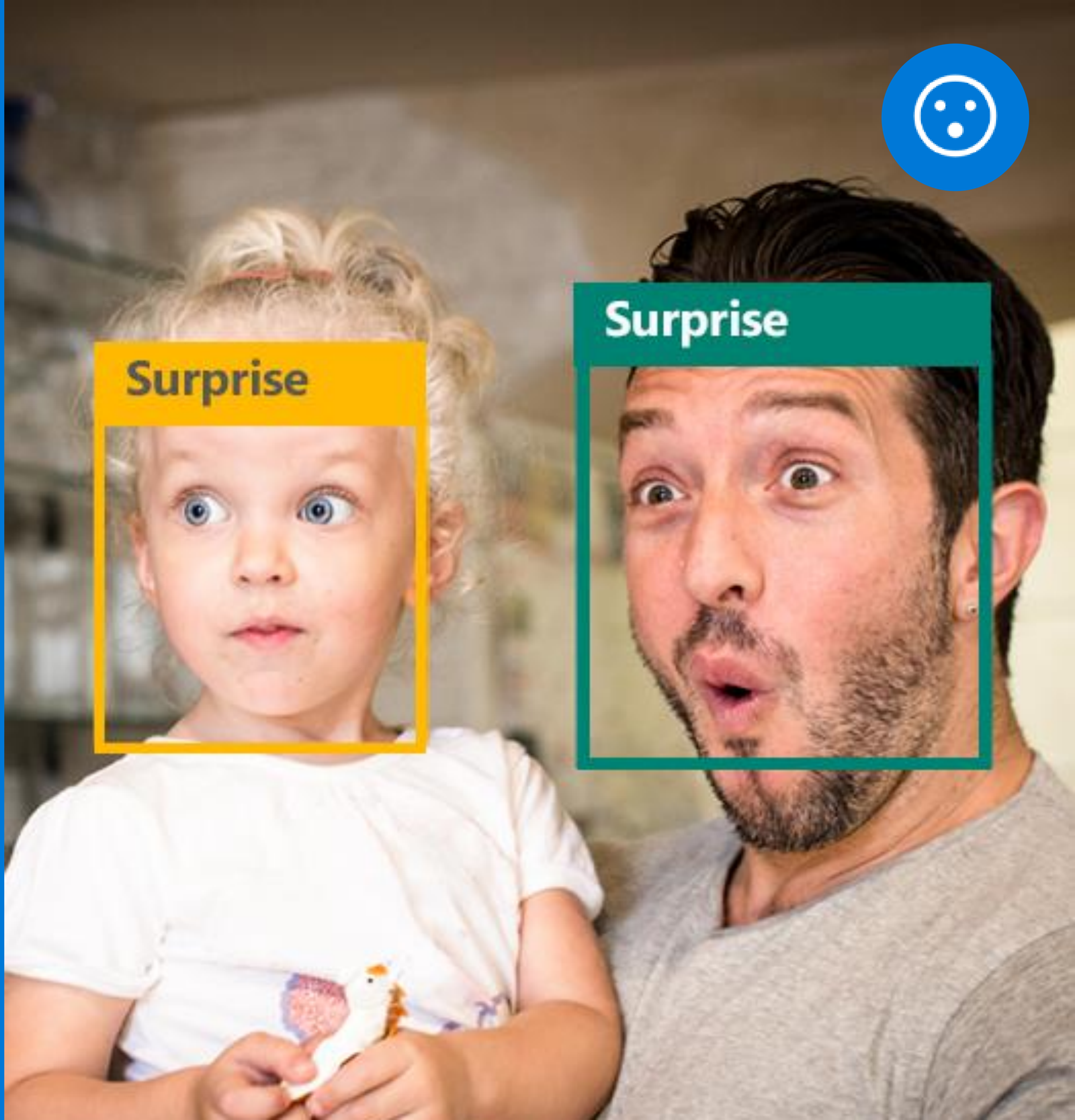
Emotion API

Face detection

```
"faceRectangle": {"width": 193,
                  "height": 193,
                  "left": 326,
                  "top": 204} …
```

Emotion scores

```
"scores": { "anger": 5.182241e-8,
            "contempt": 0.0000242813,
            "disgust": 5.621025e-7,
            "fear": 0.00115027453,
            "happiness": 1.06114619e-8,
            "neutral": 0.003540177,
            "sadness": 9.30888746e-7,
            "surprise": 0.9952837}
```



Face API

Face detection

Detect faces and their attributes within an image

Face verification

Check if two faces belong to the same person

Similar face searching

Find similar faces within a set of images

Face grouping

Organize many faces into groups

Face identification

Search which person a face belongs to

Face API

Detection

```
"faceRectangle": {"width": 193, "height": 193,
"left": 326, "top": 204}
                    …
```

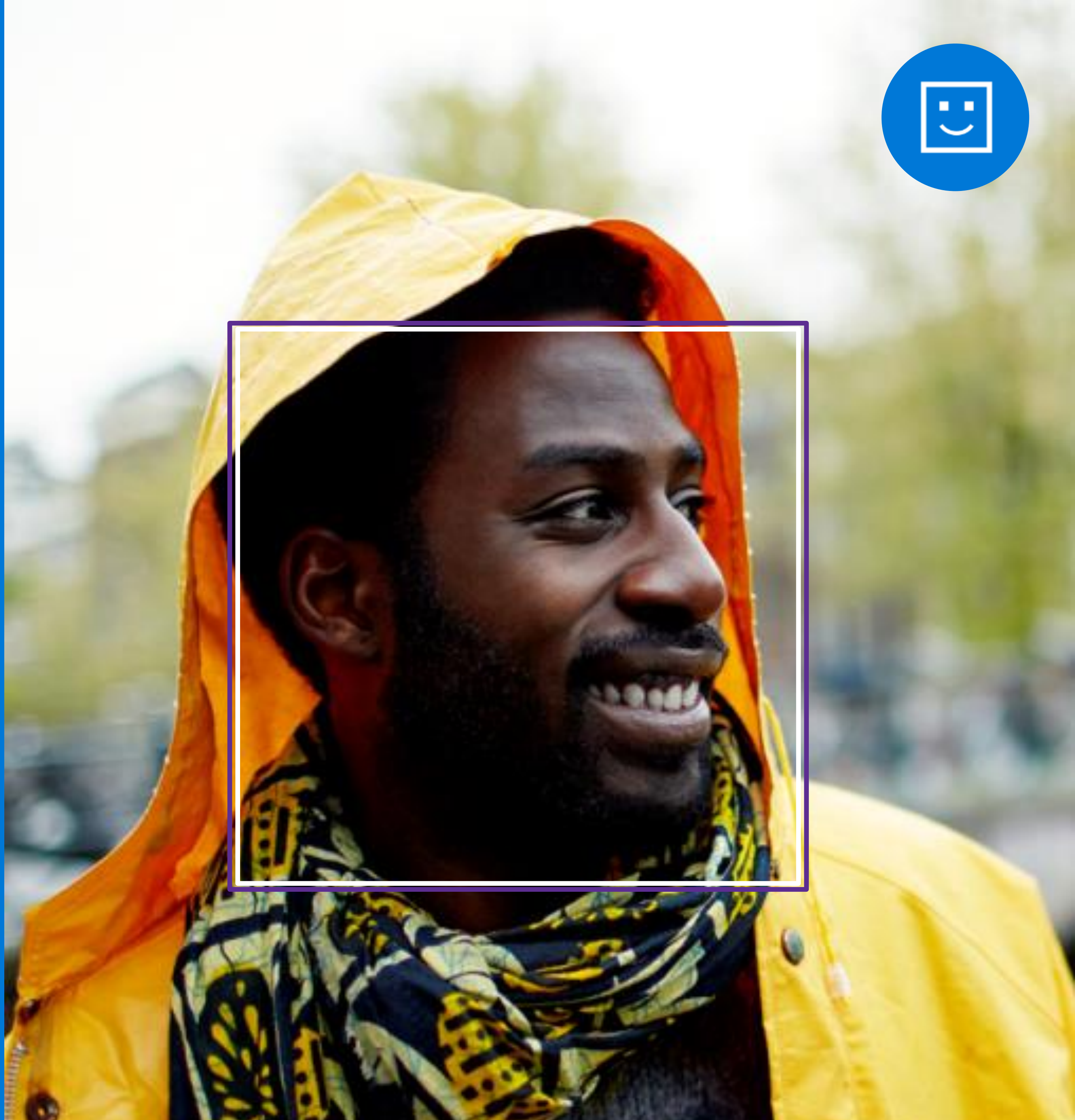
Feature attributes

```
"attributes": { "age": 42, "gender": "male",
"headPose": { "roll": "8.2", "yaw": "-37.8",
"pitch": "0.0" }}
```

Grouping

Identification

Jasper Williams



Demo

Face and Emotion: http://microsoft.com/cognitive
Intelligent Kiosk: http://aka.ms/kioskapp

Video Indexer

Unlock video insights

Upload your video and go

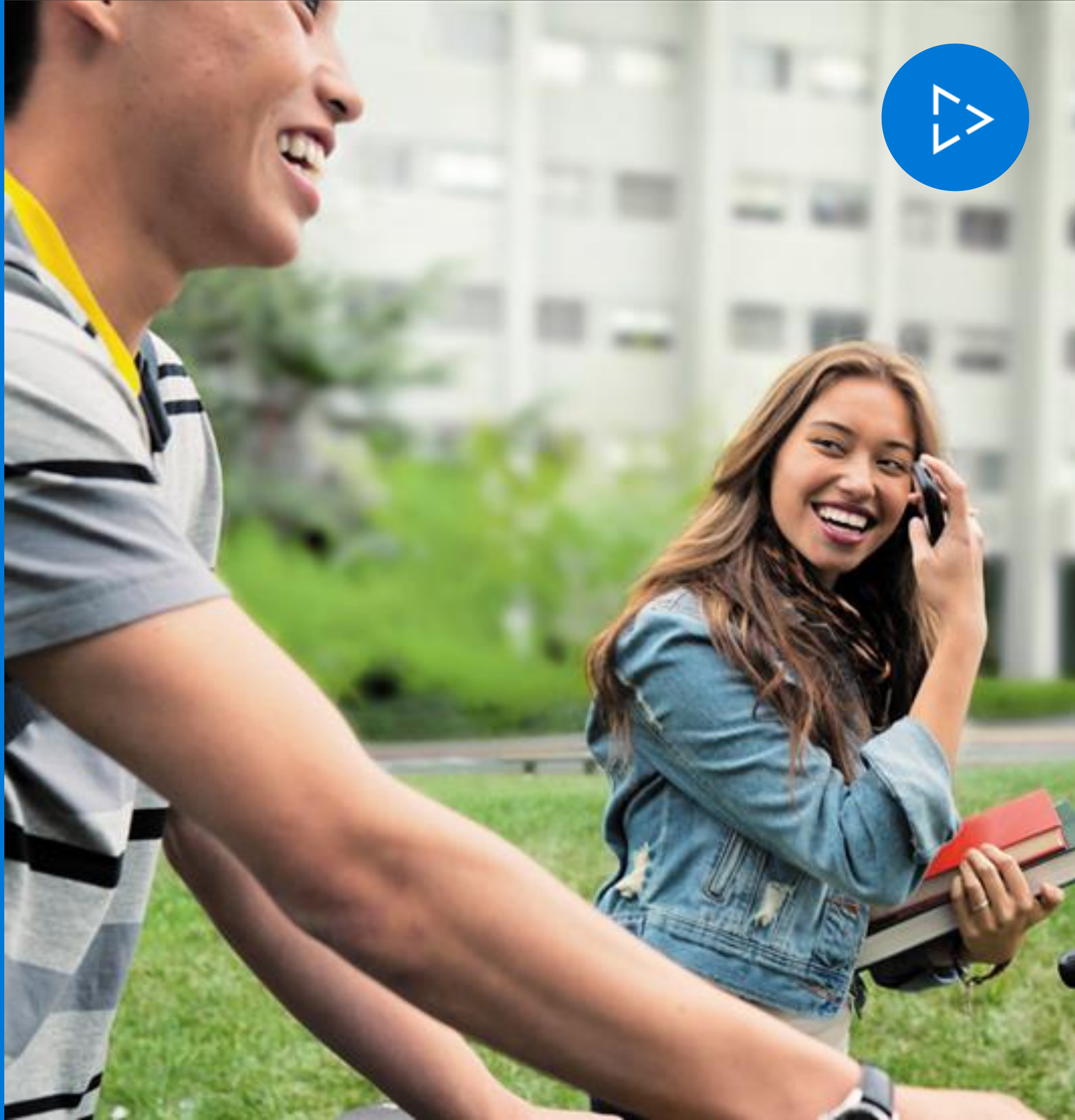
Start turning your video into insights right away.

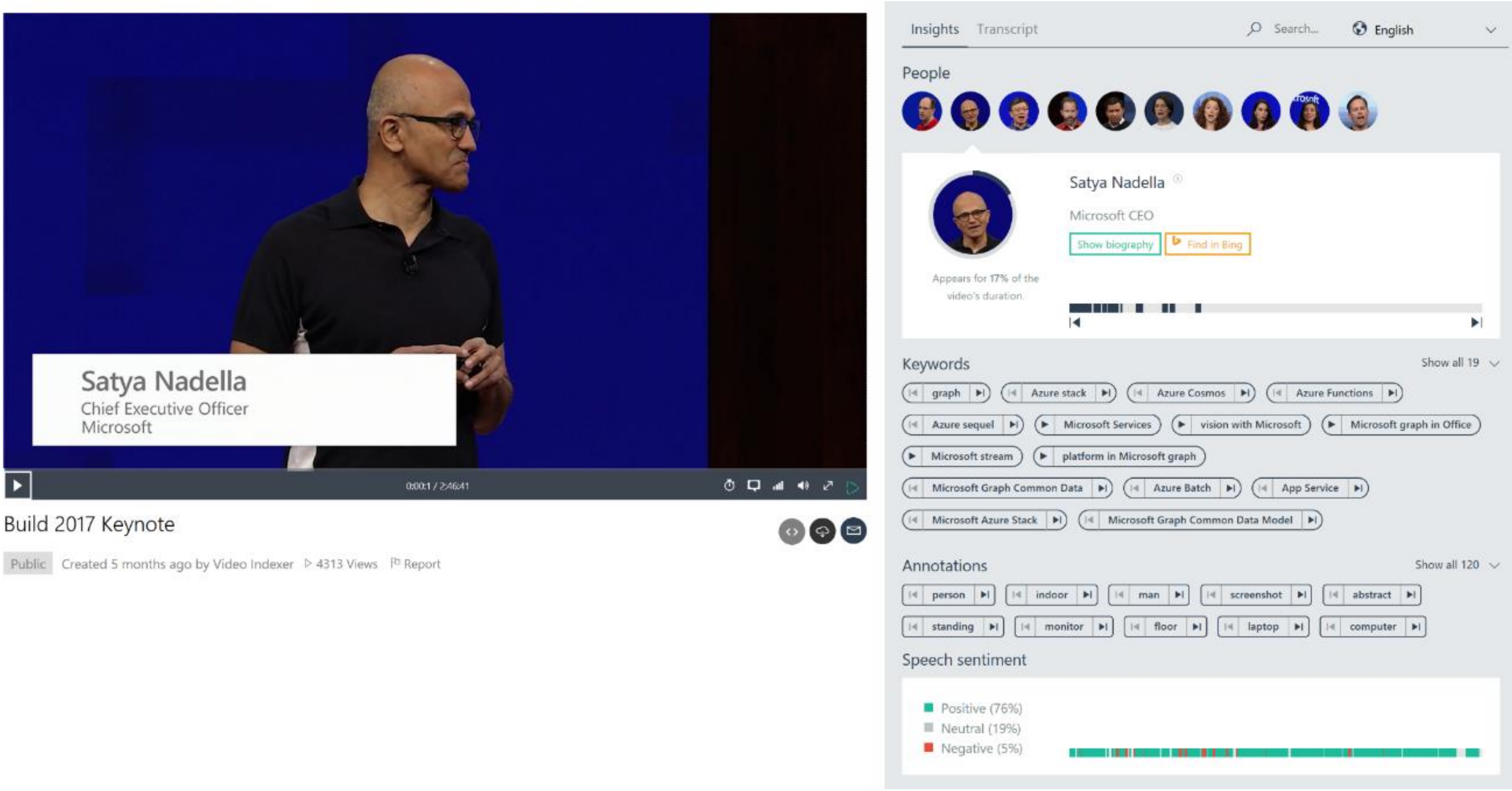
Make your content more discoverable

Quickly and easily extract insights such as spoken words, faces, characters, and emotions

Improve engagement with your video

Metadata extracted by Video Indexer can be used to build powerful engagement experiences with recommendations, highlight clips, and interactive videos





Build 2017 Keynote

Public Created 5 months ago by Video Indexer ▷ 4313 Views Report

People

Satya Nadella

Microsoft CEO

Show biography Find in Bing

Appears for 17% of the video's duration.

Keywords Show all 19

graph · Azure stack · Azure Cosmos · Azure Functions · Azure sequel · Microsoft Services · vision with Microsoft · Microsoft graph in Office · Microsoft stream · platform in Microsoft graph · Microsoft Graph Common Data · Azure Batch · App Service · Microsoft Azure Stack · Microsoft Graph Common Data Model

Annotations Show all 120

person · indoor · man · screenshot · abstract · standing · monitor · floor · laptop · computer

Speech sentiment

🟩 Positive (76%)
▨ Neutral (19%)
🟥 Negative (5%)

Demo

Video Indexer: https://www.videoindexer.ai/

Best practices for Devs

Samples and SDKs exist

For ObjectiveC/Swift/iOS, Java/Android, C#/Windows, and Python (Jupyter notebook)

https://www.microsoft.com/cognitive-services/en-us/SDK-Sample?api=computer%20vision

Limitations

Computer Vision API describes images in English only

Face API detects up to 64 human faces in one image

Facial detection: JPEG, PNG, GIF (first frame), and BMP supported, image file size of 1KB-4MB, detectable face size 36x36-4096x4096 pixels, returned faces ordered by face rect size desc

Fun random details

FindSimilarFace has 2 modes: matchPerson (default, same person) and matchFace (similar faces)

FaceGroup API takes between 2-1000 candidate faces

Documentation: https://www.microsoft.com/cognitive-services/en-us/documentation

Data

Computer Vision

Description, tags, clip art, line drawing, black & white, IsAdultContent/Score, IsRacy/Score, categories, faces, dominant colors, accent color

https://www.microsoft.com/cognitive-services/en-us/computer-vision-api

Emotions

Anger, contempt, disgust, fear, happiness, sadness, surprise, and neutral

https://www.microsoft.com/cognitive-services/en-us/emotion-api

Face

Bounding box, 27 facial landmarks, age, gender, head pose, smile, facial hair, glasses

https://www.microsoft.com/cognitive-services/en-us/face-api

UBER

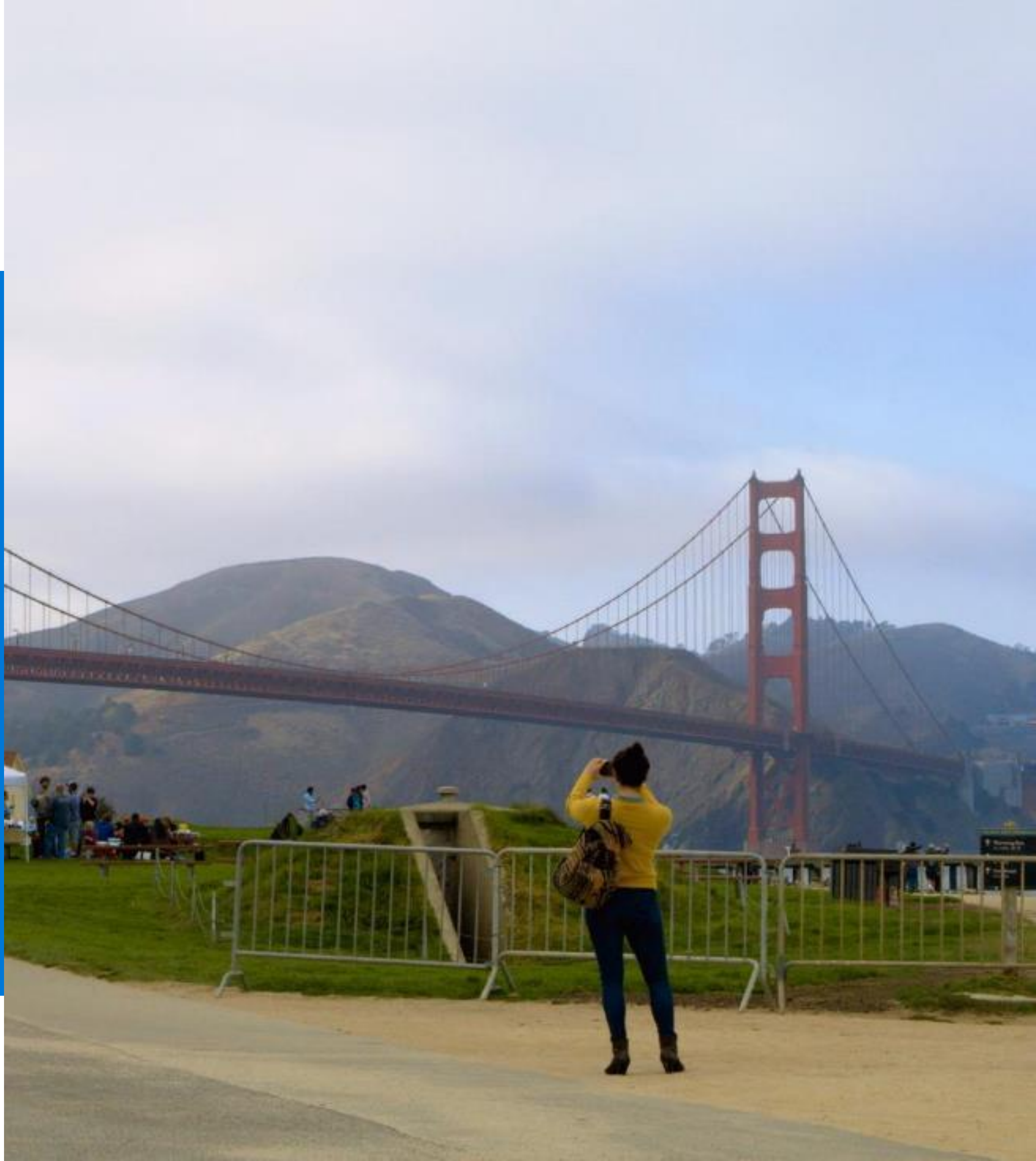
"Thousands of partners sign in to our platform every hour. The response time from the Face API is incredible, enabling us to verify our drivers without slowing them down."

Dima Kovalev, Product Manager, Uber

Face API

Read case study here

See video here



Custom Vision Service

Custom Vision Service

A customizable web service that learns to recognize specific content in imagery

Upload Images

Upload your own labeled images, or use Custom Vision Service to quickly tag any unlabeled images.

Train

Use your labeled images to teach Custom Vision Service the concepts you want it to learn.

Evaluate

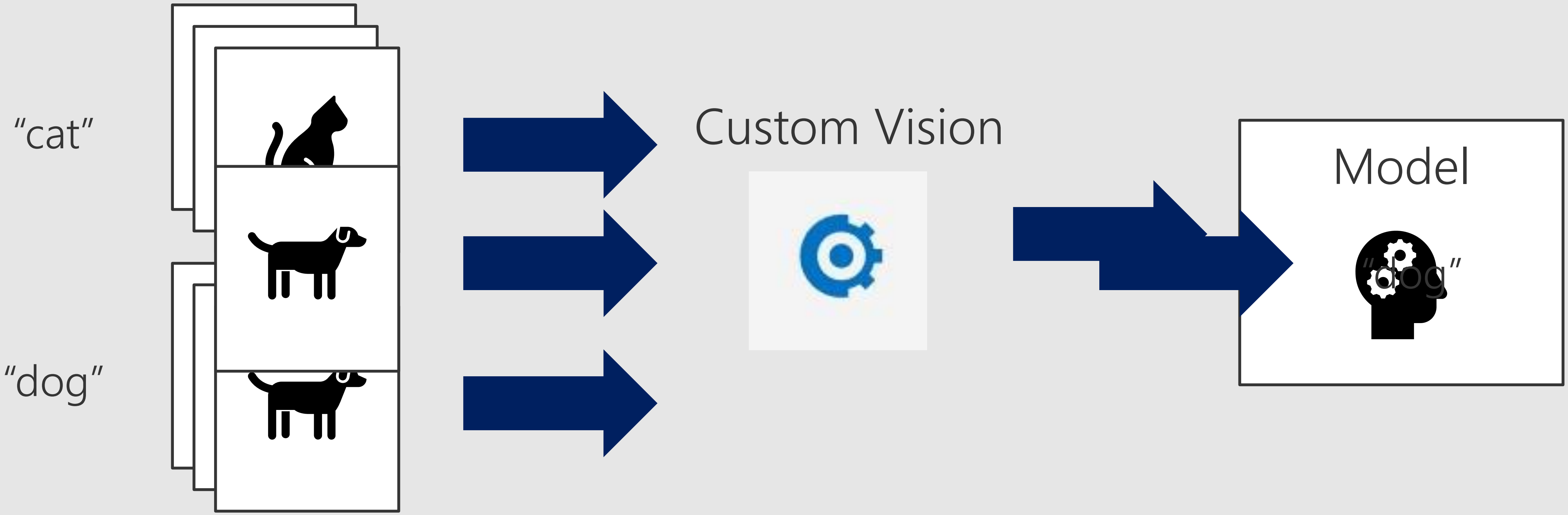
Use simple REST API calls to quickly tag images with your new custom computer vision model.

Active learning

Images evaluated through your custom vision model become part of a feedback loop you can use to keep improving your classifier.

What is it?

Custom Vision Service is an easy-to-use tool for prototyping, improving, and deploying a custom image classifier to a cloud service, without any background in computer vision or deep learning required.



Export models to mobile!

- Announcement: https://aka.ms/cvsexport
- Sample: https://github.com/Azure-Samples/cognitive-services-ios-customvision-sample
- Xamarin port: https://github.com/Xamarin/ios-samples/tree/master/ios11/CoreMLAzureModel

Demo

Dog vs. cat classifier: http://customvision.ai

Intelligent Kiosk: https://aka.ms/kioskapp

Export to CoreML on iOS11: https://aka.ms/cvsexport

Best Practices for using Custom Vision

- Use at least 30 images for each tag
- Images should be the focus of the picture
- Use sufficiently diverse images and backgrounds (ex: cats with red background and dogs with blue background)
- Train with images that are similar in {quality, resolution, lighting, etc.} to the images that will be used in prod
- Supports Microsoft accounts (MSA) and AAD

Gotchas to watch for

- V1 doesn’t currently do object detection with bounding boxes within an image
- Intended to be robust to subtle differences, so V1 is not well suited to tasks like defect detection/quality assurance
- Current project limitations while in preview: 1000 images, 50 tags, 20 iterations saved
- Current account limitations while in preview: 20 projects, 1000 predictions per day

Example Customer Scenarios

Customer Support

- Enable a customer to identify a product for support by taking a photo. No finding the manual or pulling the appliance out to identify it!

Service Engineers

- Identify parts for ordering

Manufacturing

- Fault detection on assembly lines to avoid machine downtime and drop in production rates (provided differences are obvious)

Data Scientists

- Automatic tagging instead of manual, to create features or labels

Resources: Custom Vision Service

Get started at http://customvision.ai

Build 2017 Talk:
https://channel9.msdn.com/Events/Build/2017/T6022

Programmatic API access using C# (Python and Node SDKs coming soon): https://github.com/Microsoft/Cognitive-CustomVision-Windows

Speech

Speech

Bing Speech API

Convert speech to text and back
again, and understand its intent

Speaker Recognition API

Give your app the ability
to know who's talking

Custom Speech Service

Fine-tune speech recognition
for anyone, anywhere

Demo

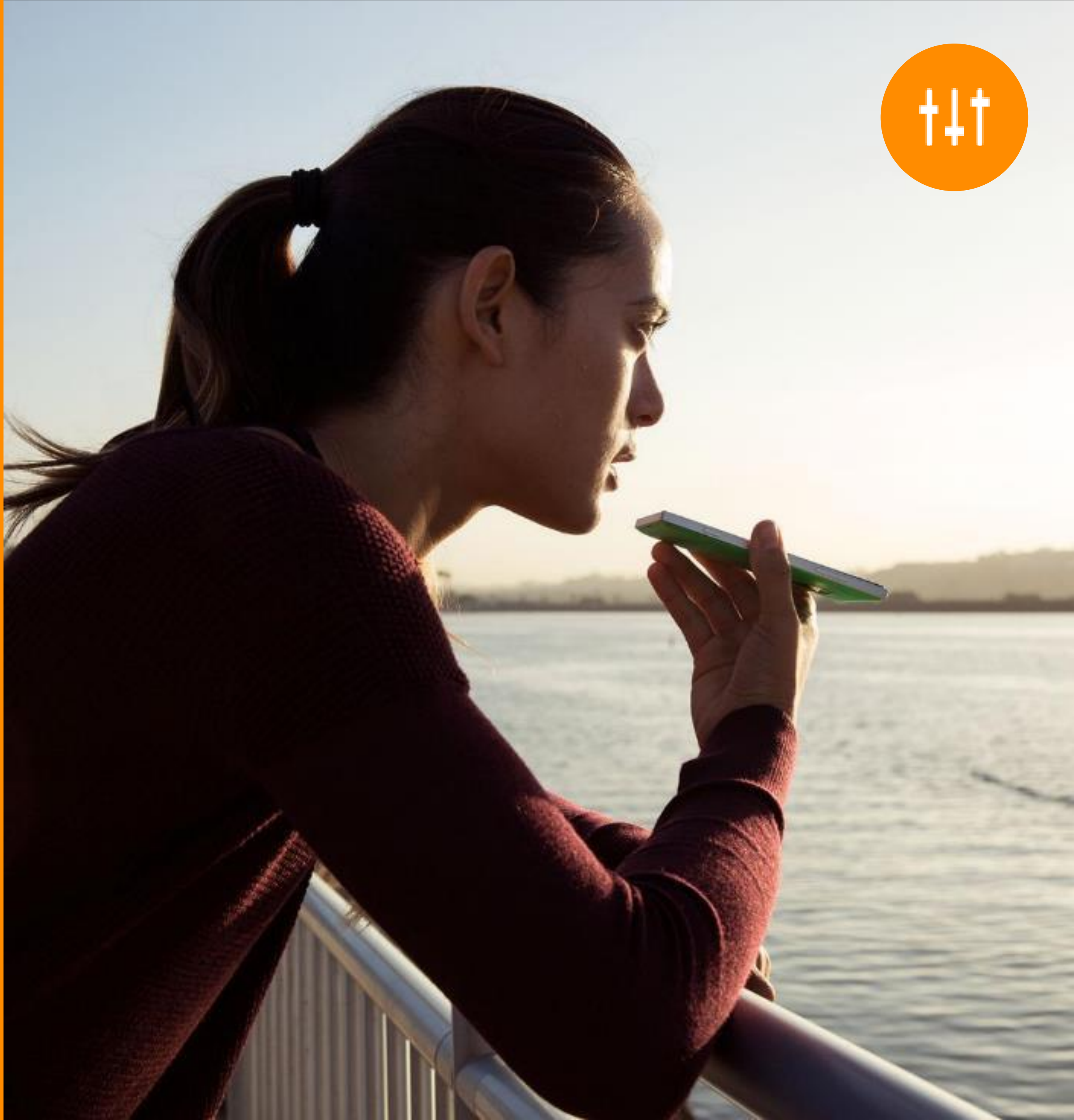
Speech: http://dictate.ms in Office
Speaker Recognition: http://microsoft.com/cognitive

Custom Speech Service

Custom Speech Service

Customize both language and acoustic models

Tailor speech recognition to your app and environment

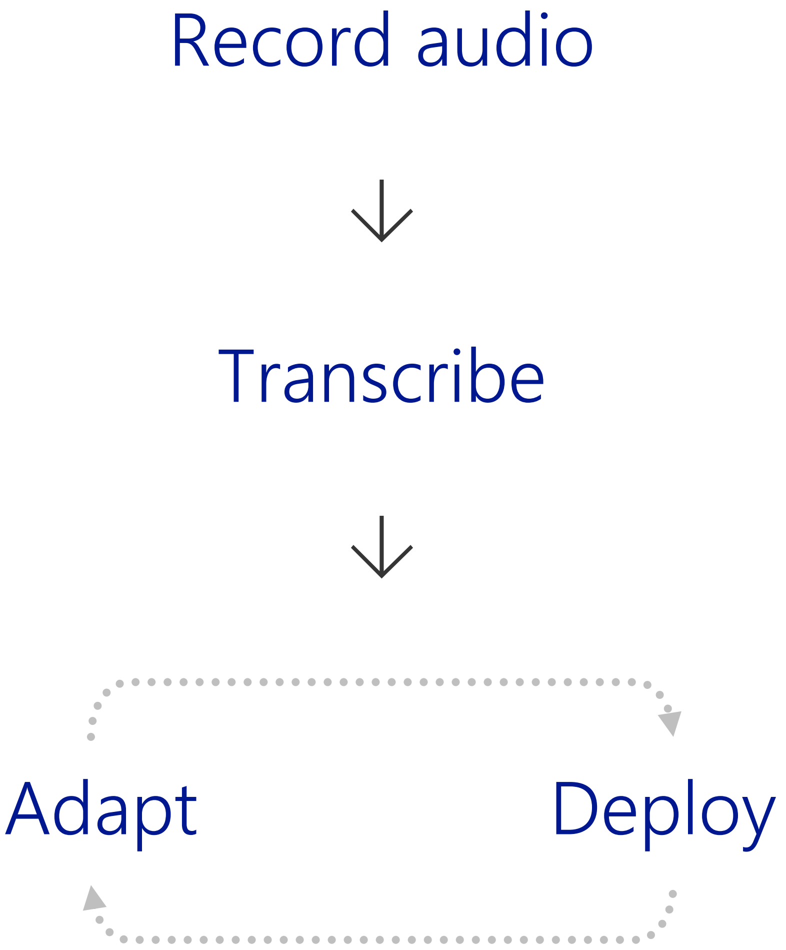


Custom Speech Service

Create custom language models for the vocabulary of the application

Adapt acoustic models to better match the expected environment of the application’s users

Deploy to a custom endpoint and access from any device



Record audio

Transcribe

Adapt

Deploy

STARSHIP COMMANDER

> “The only reason we can build a product like this is because we are building on the deep learning and speech recognition expertise at Microsoft to deliver an entertainment experience that will be revolutionary.”

Alexander Mejia, Owner and Creative Director,
Human Interact

Custom Speech Service, LUIS

Read case study here

See video here

Example Customer Scenarios

- “Hands Free” or Kiosk scenarios across Manufacturing, Retail, etc.
- Drive Through
- Call Center

Code Sample

Biology sample data: https://github.com/Microsoft/Cognitive-Custom-Speech-Service

Walkthrough: https://docs.microsoft.com/en-us/azure/cognitive-services/custom-speech-service/cognitive-services-custom-speech-get-started

Service creation: http://cris.ai

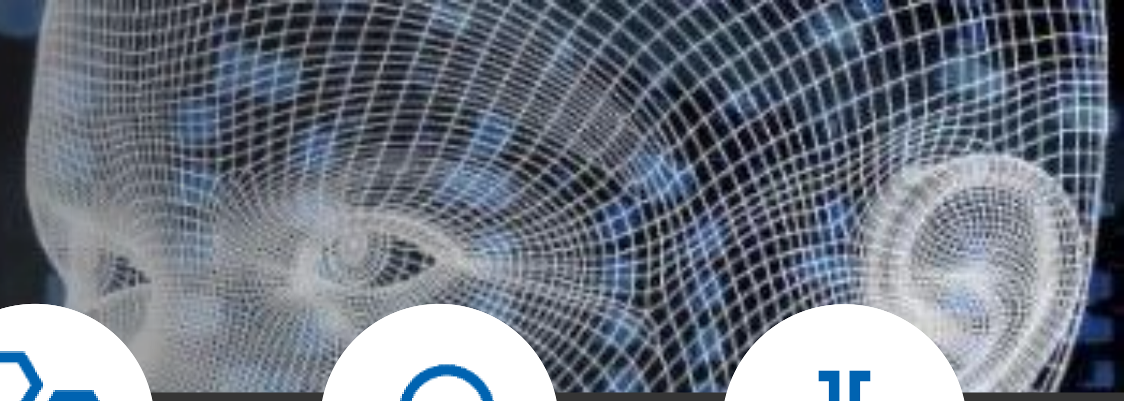
Demo

Biology data: https://cris.ai/AccuracyTests
Airport Kiosk

Summary

Microsoft Cognitive Services

Give your apps a human side



Vision

- Computer Vision
- Content Moderator
- Emotion
- Face
- Video
- Video Indexer

Speech

- Bing Speech
- Speaker Recognition
- Translator Speech

Language

- Bing Spell Check
- Linguistic Analysis
- Text Analytics
- Translator Text
- Web Language Model

Knowledge

- Academic Knowledge
- Entity Linking
- Knowledge Exploration
- Recommendations
- QnA Maker

Search

- Bing Autosuggest
- Bing Image Search
- Bing News Search
- Bing Video Search
- Bing Web Search
- Bing Entity Search

Labs

- Project Prague (gesture)
- Project Cuzco (events)
- Project Johannesburg (routing)
- Project Nanjing (isochrones)
- Project Abu Dhabi (distance matrix)
- Project Wollongong (location)

CUSTOMIZATION

- Custom Vision Service
- Custom Speech Service
- Language Understanding
- Custom Decision Service
- Bing Custom Search

Resources

- Website: http://microsoft.com/cognitive

- UserVoice: https://cognitive.uservoice.com
- StackOverflow tag: microsoft-cognitive

- Microsoft Ignite content can be found at https://channel9.msdn.com/Events/Ignite

Resources: Cognitive Services

Microsoft Cognitive Services Developer Code of Conduct: https://azure.microsoft.com/en-us/support/legal/developer-code-of-conduct

Microsoft Cognitive Services Terms - applicable to free previews, excluding generally available Microsoft Azure and Volume Licensing

https://go.microsoft.com/fwlink/?LinkId=533207

Online Services Terms – applicable to generally available Cognitive Services use via Microsoft Azure or Volume Licensing (once available)

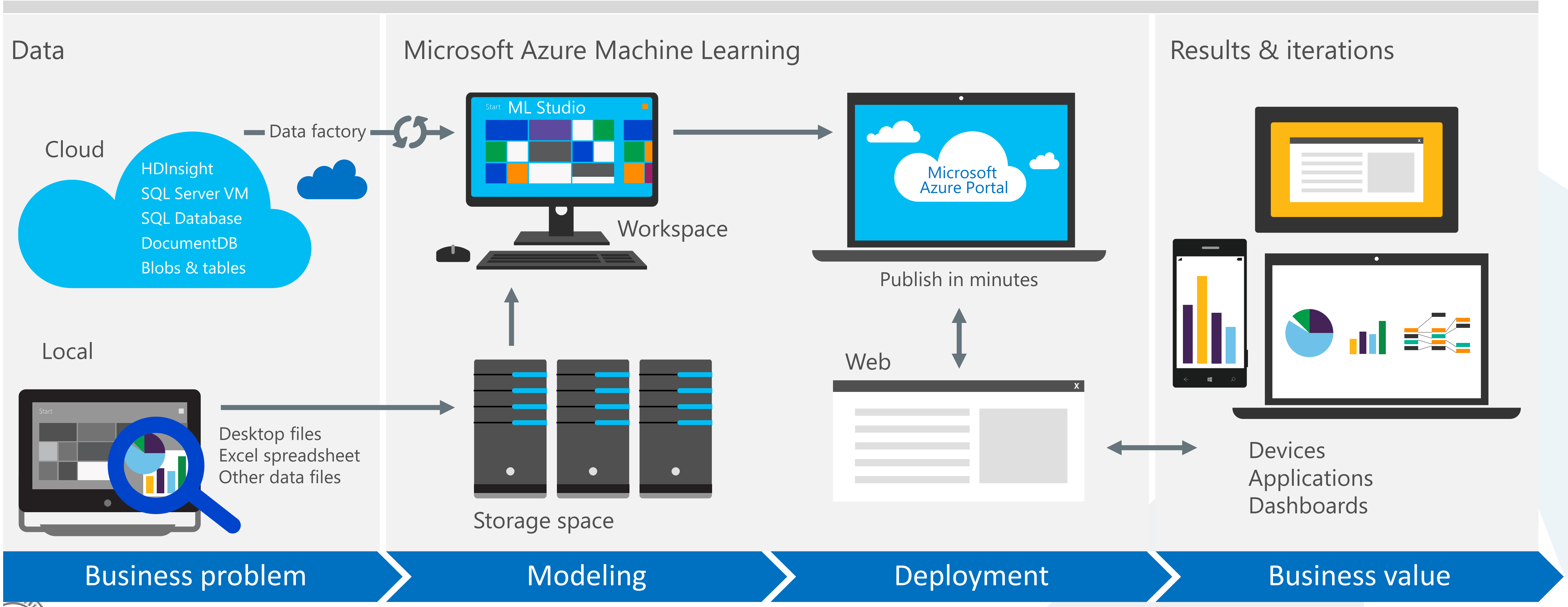
http://www.microsoftvolumelicensing.com/DocumentSearch.aspx?Mode=3&DocumentTypeId=31

Predictive Analytics

Azure Machine Learning

Azure Machine Learning

The power of machine learning to predict future trends or behaviors



Data

Cloud

HDInsight
SQL Server VM
SQL Database
DocumentDB
Blobs & tables

Data factory

Local

Desktop files
Excel spreadsheet
Other data files

Microsoft Azure Machine Learning

Workspace

Storage space

Publish in minutes

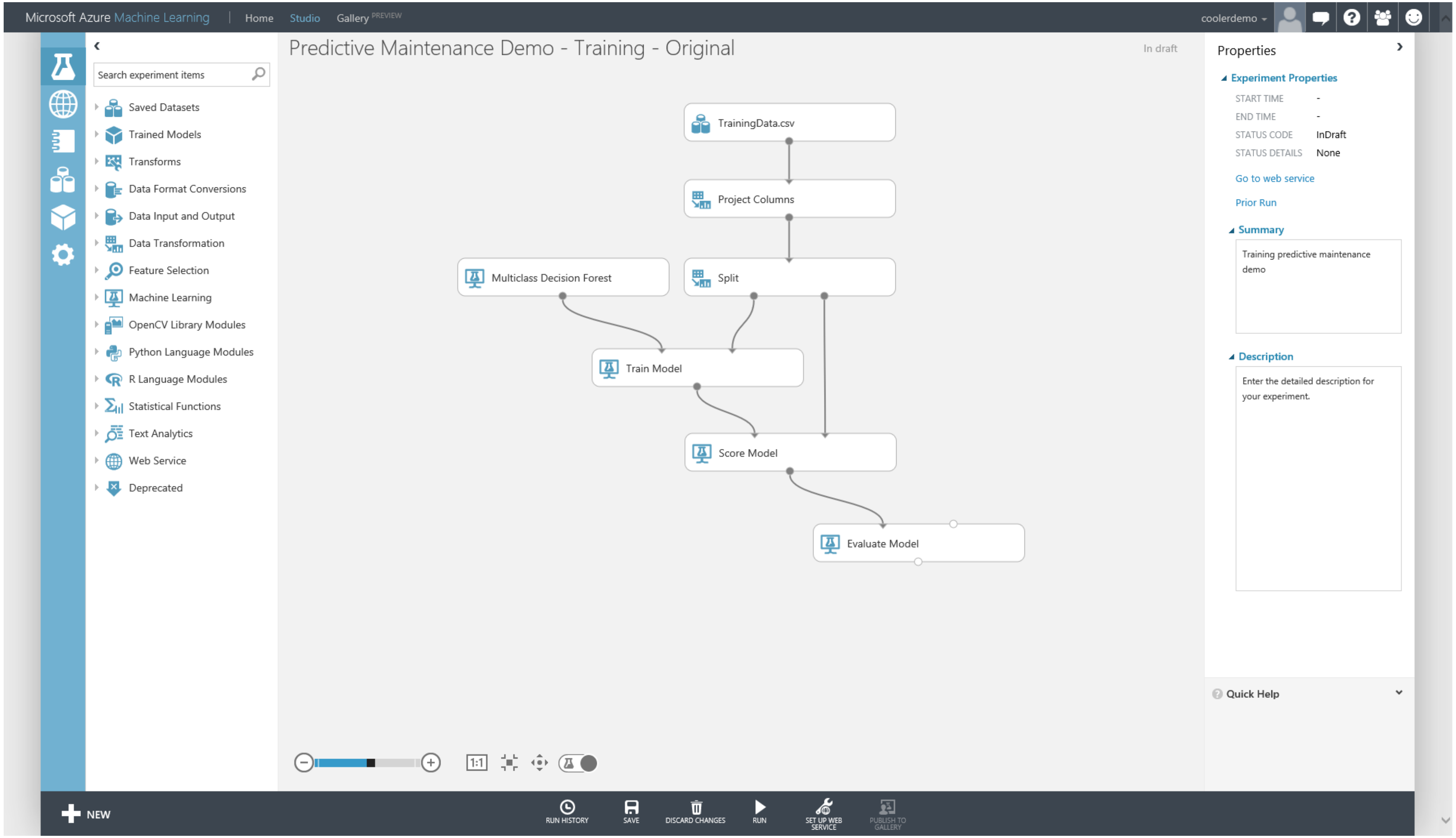
Web

Results & iterations

Devices
Applications
Dashboards

Business problem → Modeling → Deployment → Business value

AML - Drag & Drop + Best in Class Algorithms



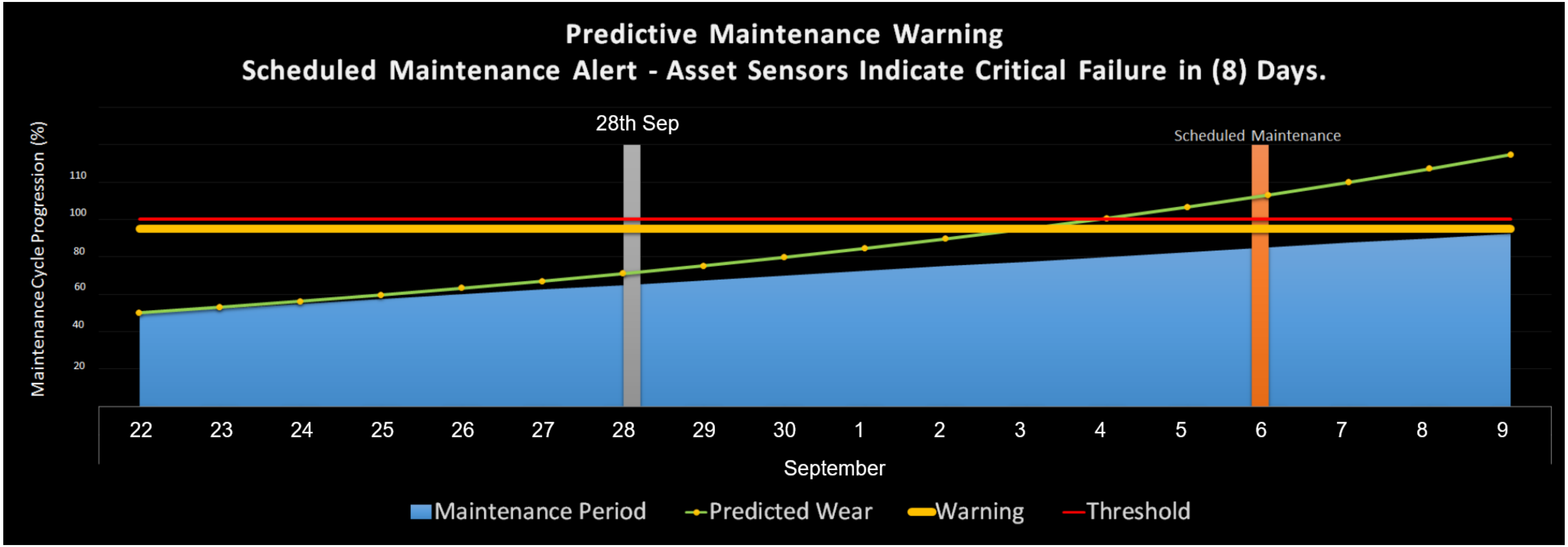
Predictive Maintenance Demo - Training - Original

START TIME -
END TIME -
STATUS CODE InDraft
STATUS DETAILS None

Summary: Training predictive maintenance demo

Description: Enter the detailed description for your experiment.

Integrated predictive analytics



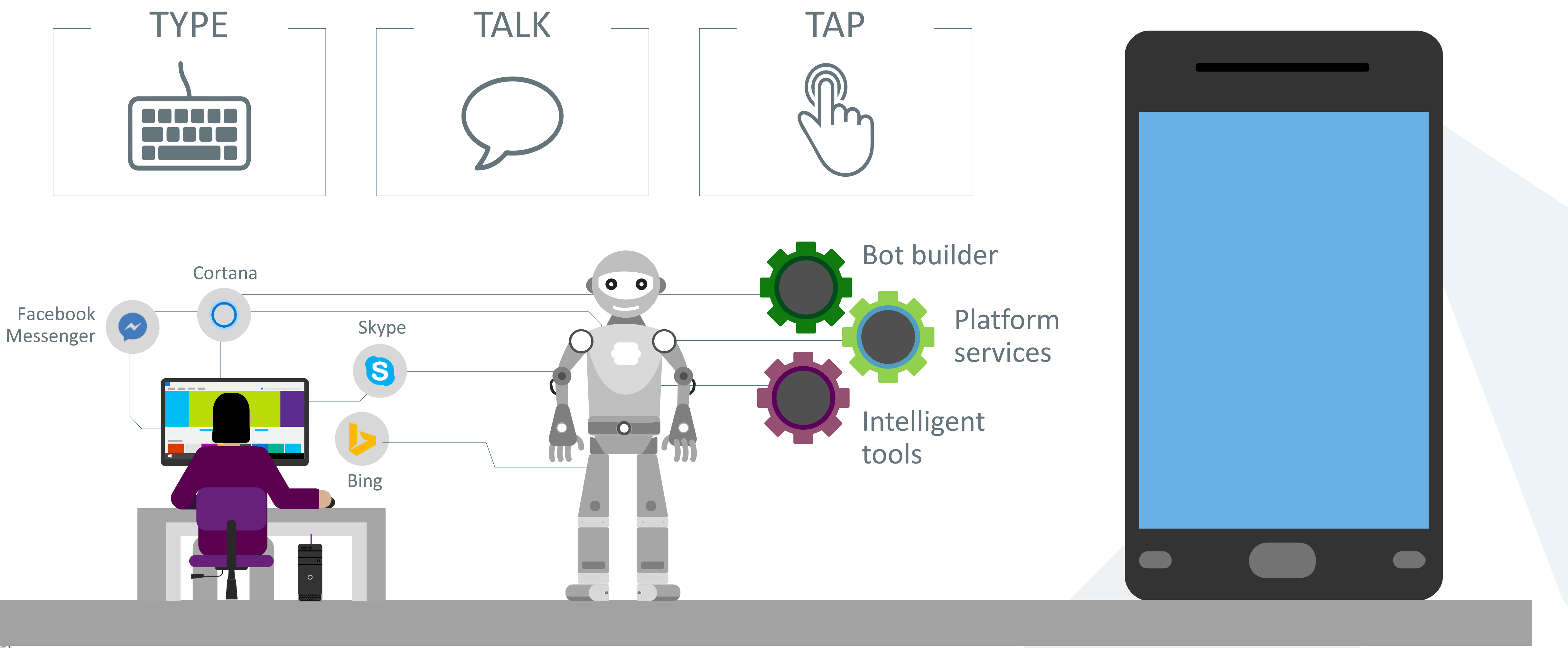
Empower with proactive analysis

Machine learning solutions allow for powerful predictive analytics solutions, leveraging historical data and real time device ingestion input.

Azure Bot Service

About

Why a bot?

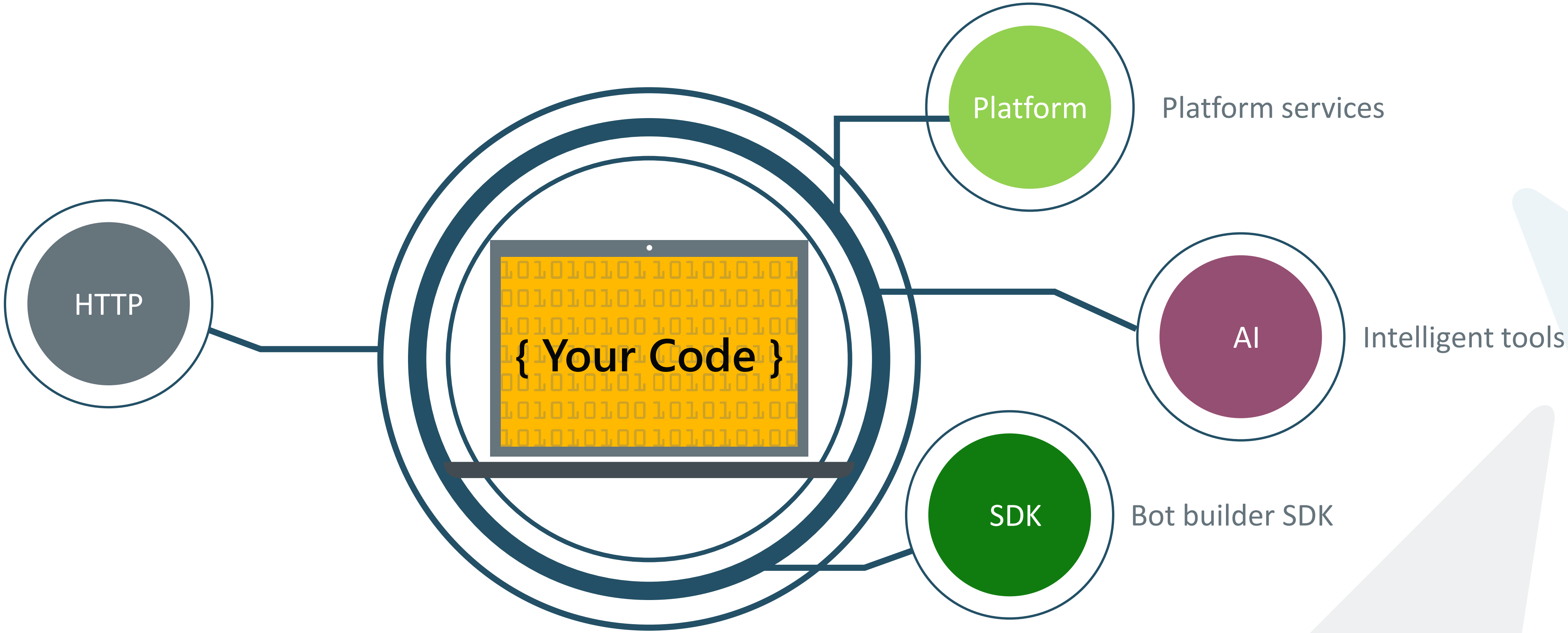


ISO 9001 SGS

Kinds of bots

Scenario	Retail	Finance	Insurance	Telecoms	Government	Automotive	Manufacturing	Healthcare	Media	Events
Customer service	✓	✓	✓	✓	✓	✓	✓	✓		✓
Customer retail	✓	✓	✓	✓				✓		
Audio/speech analysis	✓	✓	✓	✓	✓				✓	
Translation		✓	✓							
Surveillance		✓			✓					
Knowledge extraction		✓	✓	✓			✓			
Video/photo analysis		✓			✓				✓	
Product identification	✓						✓	✓		
Digital assistant						✓				
Footfall analysis	✓									✓
HD maps and object detection						✓				

What is a bot?



HTTP

{ Your Code }

Platform

Platform services

AI

Intelligent tools

SDK

Bot builder SDK

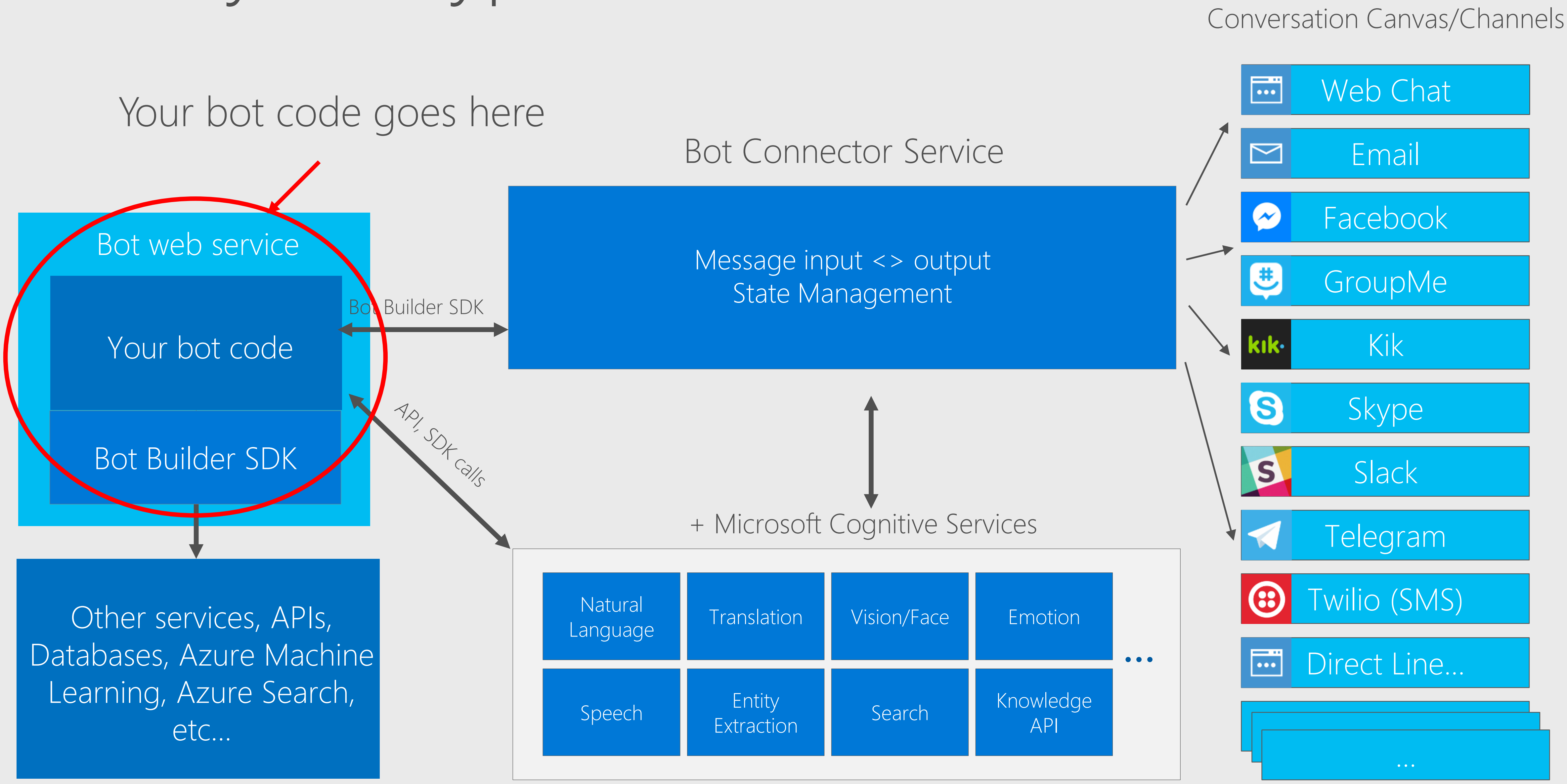
REST endpoint
[Direct Line Protocol]

Conversational and
business logic

Canvas aware

Context
sensitive

An x-ray of a typical bot



Your bot code goes here

Bot web service

Your bot code

Bot Builder SDK

Other services, APIs, Databases, Azure Machine Learning, Azure Search, etc…

Bot Builder SDK

API, SDK calls

Bot Connector Service

Message input <> output
State Management

+ Microsoft Cognitive Services

Natural Language	Translation	Vision/Face	Emotion
Speech	Entity Extraction	Search	Knowledge API

…

Conversation Canvas/Channels

- Web Chat
- Email
- Facebook
- GroupMe
- Kik
- Skype
- Slack
- Telegram
- Twilio (SMS)
- Direct Line…
- …

Building Bots

- The hard way – DIY
  - Implement a REST API
- The easy(-ier) way – Bot Builder SDK
  - .NET
  - Node.js

Getting Started with Predefined Templates

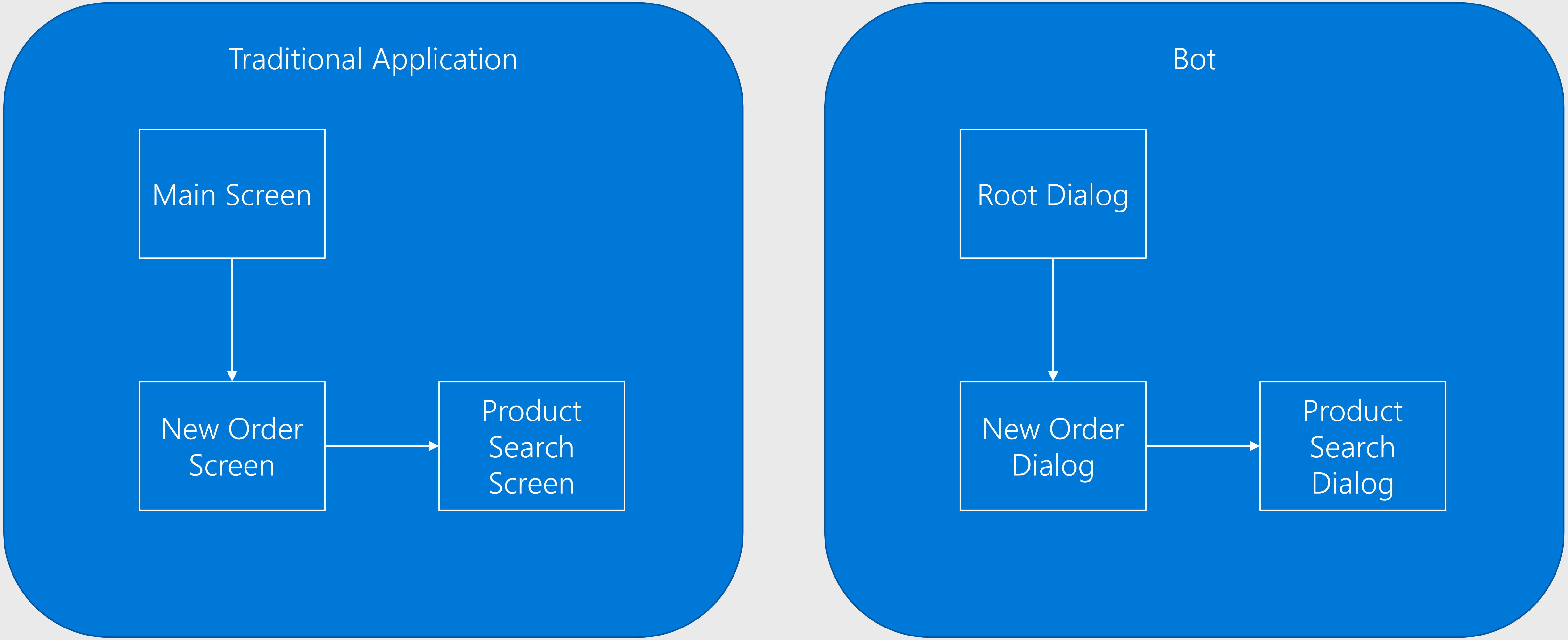
- Bot Service includes five templates to help you get started with building bots (C# and Node.js)
  - Basic bot
    - A bot that uses dialogs to respond to user input
    - The bot echoes back to the user whatever they type in
    - You can use this template to get started building conversation flow in your bot
  - Form bot
    - A bot that collects input from a user via a guided conversation
    - For example, a bot that is designed to obtain a user's sandwich order
  - Language Understanding bot
    - A bot that uses natural language models to understand user intent
    - This template leverages Language Understanding (LUIS) to provide natural language understanding
  - Question and Answer bot
    - A bot that distills semi-structured data like question and answer pairs into distinct, helpful answers,
    - This template leverages the QnA Maker service to parse questions and provide answers
  - Proactive bot
    - A bot that can send proactive messages to the user
    - For example:
      - if a bot sets a timer or reminder, it may need to notify the user when the time arrives
      - if a bot receives a notification about an external event, it may need to communicate that information to the user

Deployment Options

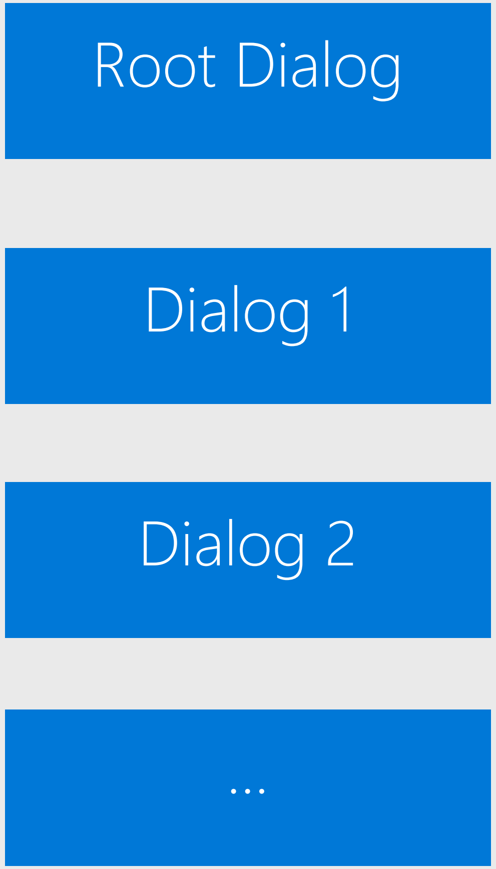
- With the App Service plan
  - a bot is a standard Azure web app you can set to allocate a predefined capacity with predictable costs and scaling.
- With a Consumption plan
  - a bot is a serverless bot that runs on Azure Functions and uses the pay-per-run Azure Functions pricing.

Dialogs are for bots like screens are for apps

They separate concerns and organize flows, exactly the same way:



Dialogs are serialized into stacks



Root Dialog

Dialog 1

Dialog 2

…

This is how we know how to navigate “back”
(Users won’t necessarily think that way)

So careful with too much “modality”: Things can get complex quickly

Read more

Implementing Dialogs

- A dialog is a class implementing a piece of the bot’s business logic
- One method to implement:

```
public interface IDialog<out TResult>
{
    Task StartAsync(IDialogContext context);
}
```

Conversation Flow

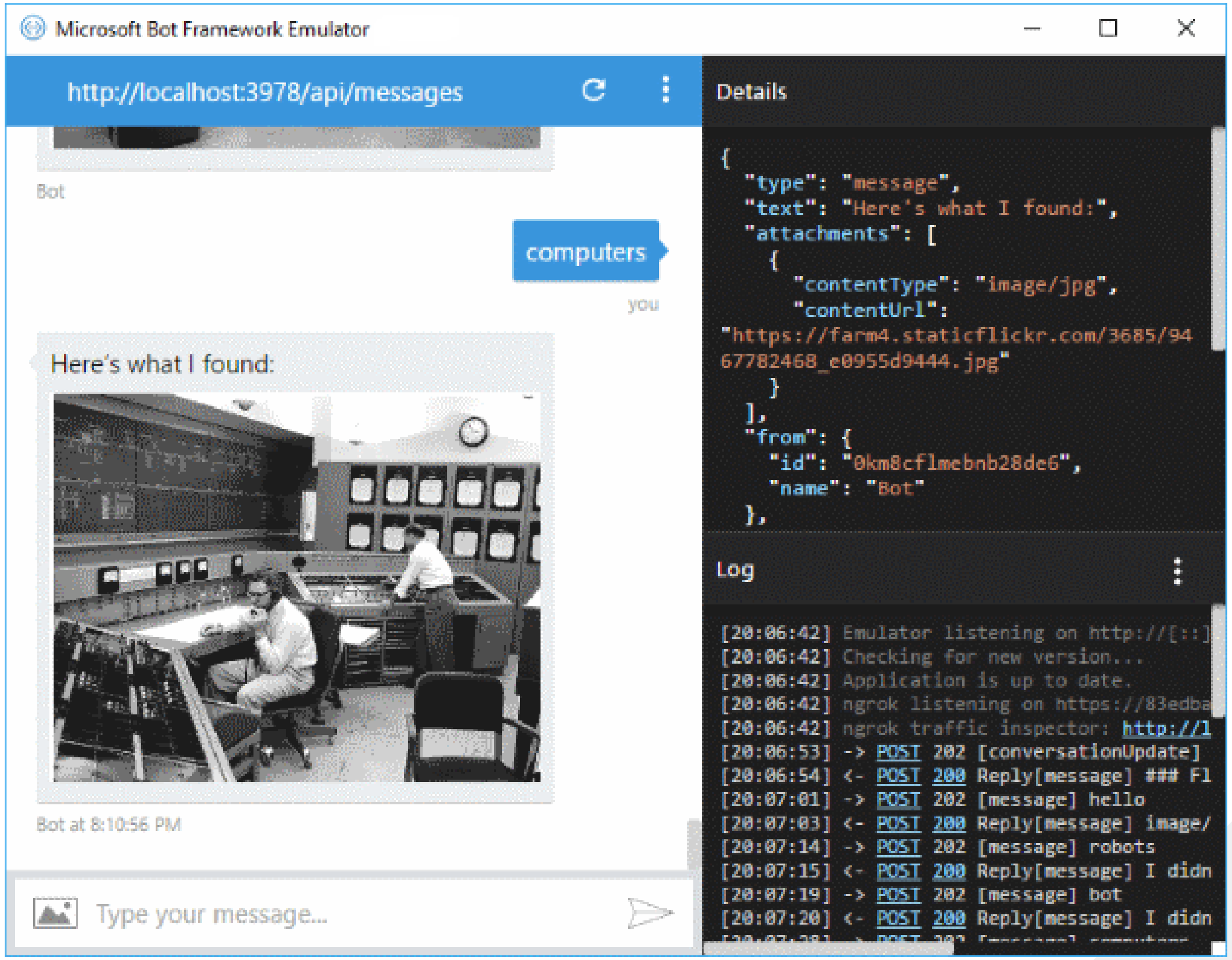
- The active dialog takes control of conversation flow
- Using the SDK you can:
  - *Context.**Wait**()*
  - *Context.**Done**()*
  - *Context.**Fail**()*
  - *Context.**Foreward**()*
  - *Context.**Call**()*

Dialog State

- Code continuations allow modeling a state machine

```
if (...)
{
    context.Call(OrderDialog.CreateDialog(), BeerOrderedAsync);
    return;
}
```

Bot Framework Emulator



```
{
  "type": "message",
  "text": "Here's what I found:",
  "attachments": [
    {
      "contentType": "image/jpg",
      "contentUrl":
"https://farm4.staticflickr.com/3685/94
67782468_e0955d9444.jpg"
    }
  ],
  "from": {
    "id": "0km8cflmebnb28de6",
    "name": "Bot"
  },
```

```
[20:06:42] Emulator listening on http://[::]
[20:06:42] Checking for new version...
[20:06:42] Application is up to date.
[20:06:42] ngrok listening on https://83edba
[20:06:42] ngrok traffic inspector: http://l
[20:06:53] -> POST 202 [conversationUpdate]
[20:06:54] <- POST 200 Reply[message] ### Fl
[20:07:01] -> POST 202 [message] hello
[20:07:03] <- POST 200 Reply[message] image/
[20:07:14] -> POST 202 [message] robots
[20:07:15] <- POST 200 Reply[message] I didn
[20:07:19] -> POST 202 [message] bot
[20:07:20] <- POST 200 Reply[message] I didn
```

Chain API

- A Fluent API for building dialogs
- Available using the ***Chain*** class

```
public static readonly IDialog<Beer> Dialog = Chain
            .From(() => new PromptDialog.PromptChoice<RecommendationOptions>(
                new[] { RecommendationOptions.Category, RecommendationOptions.Origin, RecommendationOptions.Name },
                "How would you like me to recommend your beer?",
                "Not sure I got it. Could you try again?",
                3, descriptions: new [] { "By Beer Category" , "By Beer Origin", "By Beer Name" }))
            .Switch(
                Chain.Case<...>(option => option == RecommendationOptions.Category, (context, option) => CategoryRecommendation),
                Chain.Case<...> (option => option == RecommendationOptions.Origin, (context, option) => CountryRecommendation),
                Chain.Case<...> (option => option == RecommendationOptions.Name, (context, option) => NameRecommendation)
            )
            .Unwrap();
```

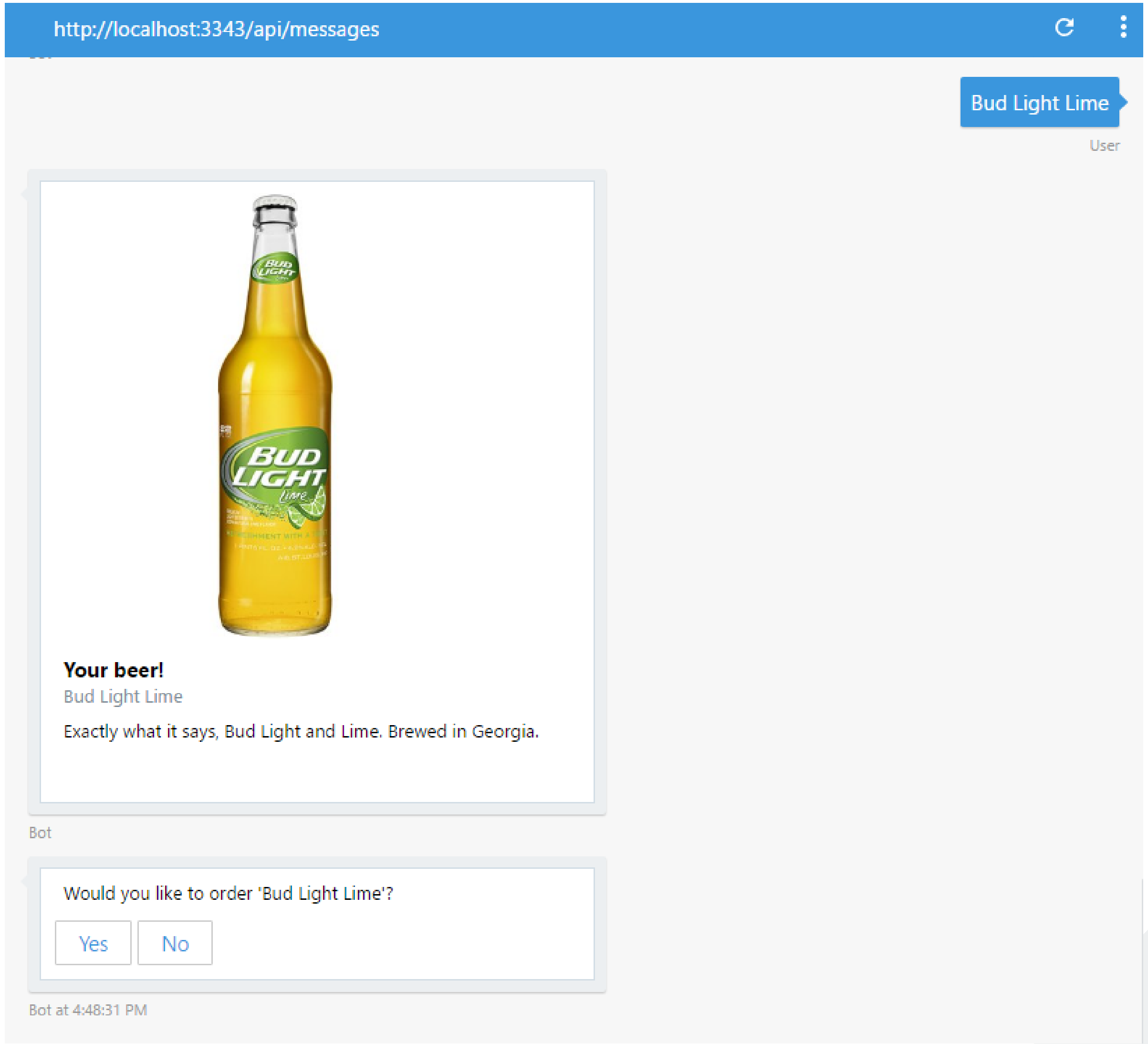
Forms

- Sometimes we need the user to input some predefined fields

Form Flow

```
public class BeerOrder
{
    [Prompt("What beer would you like?")]
    public string BeerName { get; set; }
    [Prompt("Which chaser would you like next to your beer? {||}")]
    public Chaser Chaser { get; set; }
    [Prompt("How about something to eat? {||}")]
    public SideDish Side { get; set; }
}

public static IDialog<BeerOrder> CreateDialog(string beerName = null)
{
    return new FormDialog<BeerOrder>(
        new BeerOrder { BeerName = beerName }, ...);
}
```

Rich Cards



http://localhost:3343/api/messages

Bud Light Lime

User

Your beer!

Bud Light Lime

Exactly what it says, Bud Light and Lime. Brewed in Georgia.

Bot

Would you like to order 'Bud Light Lime'?

Bot at 4:48:31 PM

Creating a Card

```
.ContinueWith(async (context, beerAwaitable) =>
{
        var chosenBeer = await beerAwaitable;
        Uri imageUrl = await ImageSearchService.SearchImage($"{chosenBeer.Name} beer");
        var card = new HeroCard("Your beer!", chosenBeer.Name, chosenBeer.Description,
                new List<CardImage> { new CardImage(imageUrl.ToString()) });

        var message = context.MakeMessage();
        message.AttachmentLayout = AttachmentLayoutTypes.Carousel;
        message.Attachments = new List<Attachment> {card.ToAttachment()};
        await context.PostAsync(message);
        return Chain.Return(chosenBeer);
});
```

“Typing” Indicator

ISO 9001 SGS

Sending a Notification

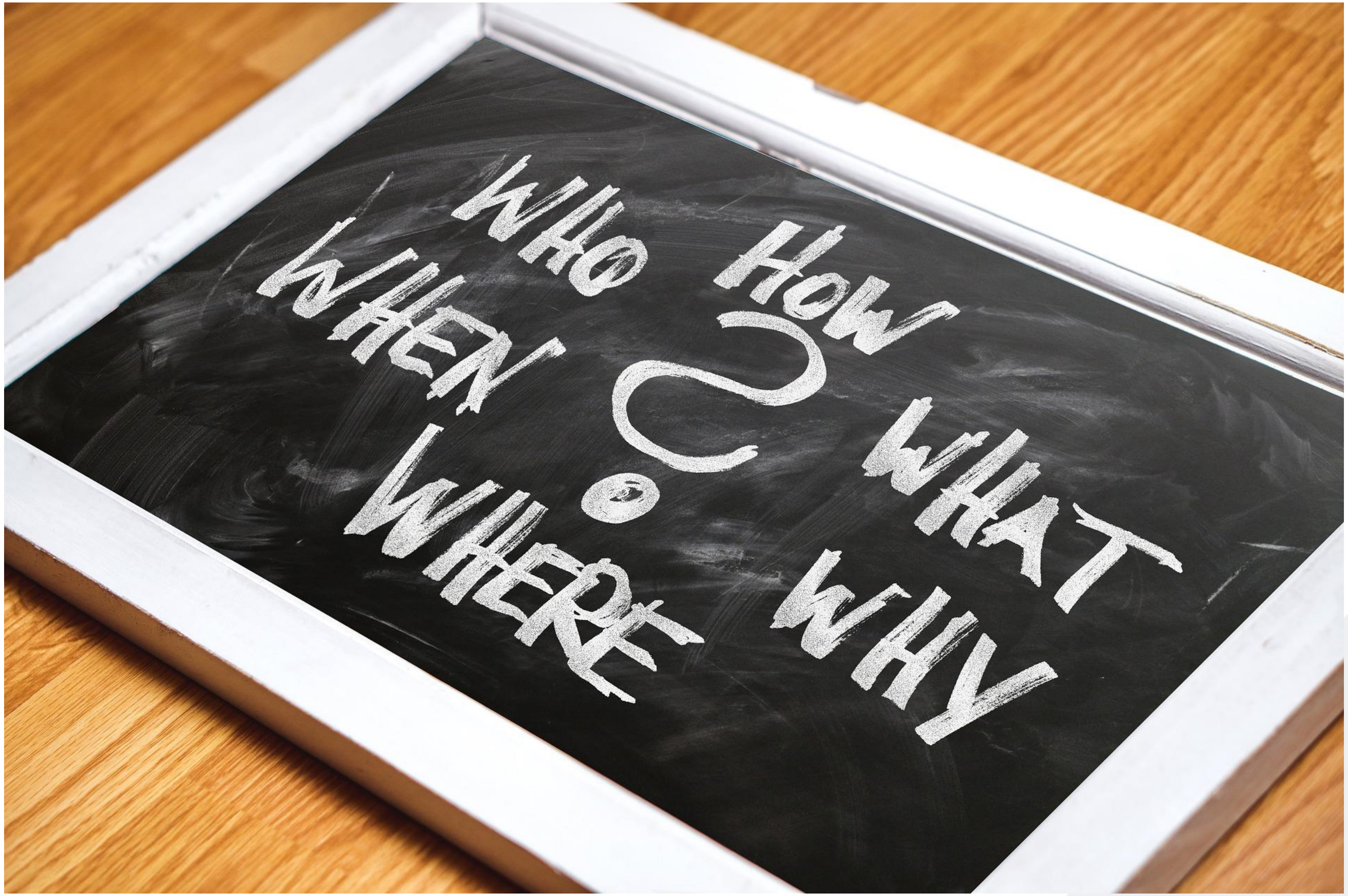
- Sending a Typing notification is easy

```
var typingMessage = context.MakeMessage();
typingMessage.Type = ActivityTypes.Typing;
await context.PostAsync(typingMessage);
await Task.Delay(1000);    // Do Work
```

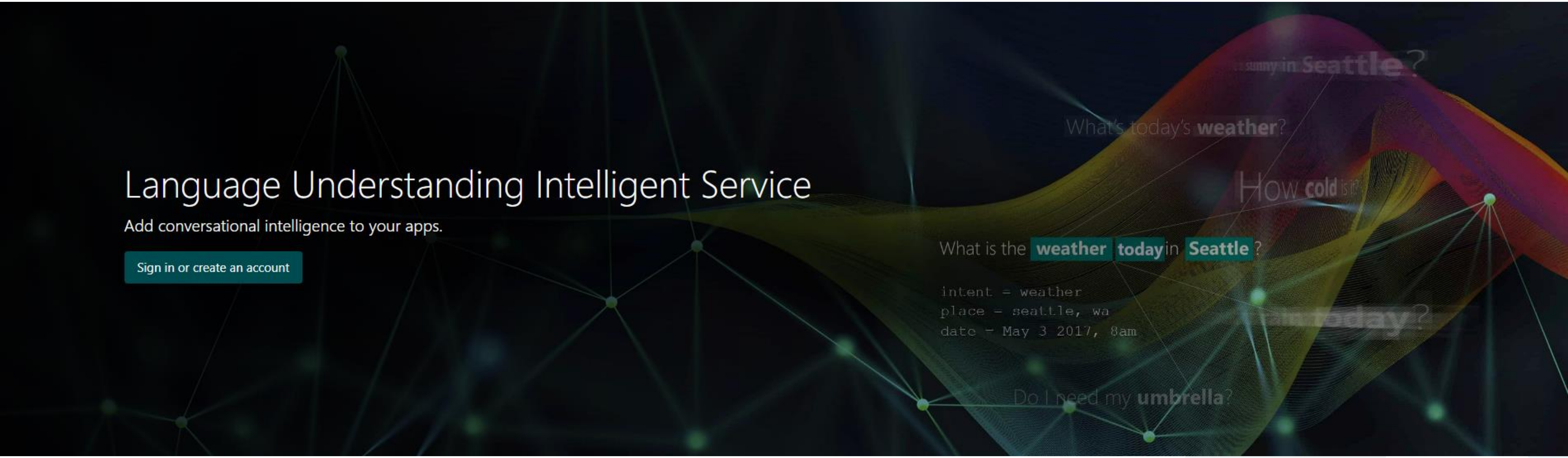
Bot State

- Use the session object
  - User Data
  - Conversation Data
  - Private Conversation Data
  - Dialog Data
- Persisted and managed by the Bot Framework State Service
  - IBotStorage
- Various providers exist
  - In-Memory
  - Table storage
  - CosmosDB
  - DIY…

Natural Language



LUIS



Language Understanding Intelligent Service

Add conversational intelligence to your apps.

Sign in or create an account

What is the weather today in Seattle?

```
intent = weather
place = seattle, wa
date = May 3 2017, 8am
```

LUIS & Bot Framework

```
[LuisIntent("Bye")]
public async Task OnByeAsync(...)
{
    await context.PostAsync("Bye bye. See you soon!");
    context.Done((object) null);
}

[LuisIntent("RecommendBeer")]
public Task OnRecommendBeerAsync(...)
{
    var beerName = GetEntity(luisResult, BeerNameEntityName);
    var brewery = GetEntity(luisResult, BreweryEntityName);
    var category = GetEntity(luisResult, CategoryEntityName);
    var country = GetEntity(luisResult, CountryEntityName);

    context.Call(RecommendationDialog.CreateDialog(beerName, brewery, category, country), BeerRecommendedAsync);
    return Task.FromResult((object) null);
}
```

Demo

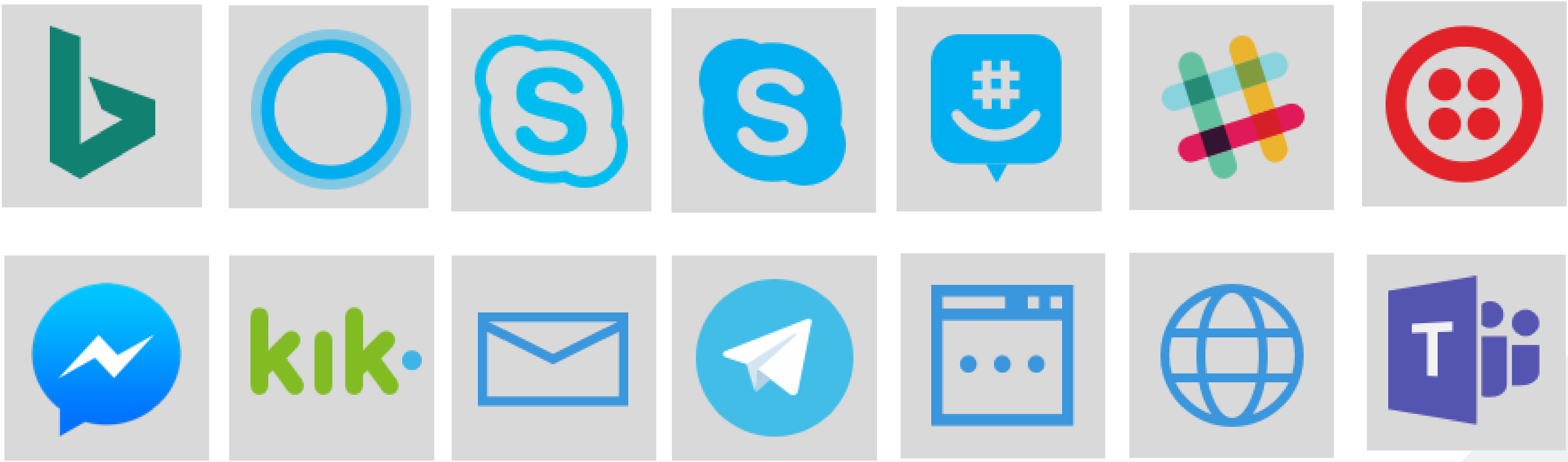
```
        $mockQueryBuilder->shouldReceive('newQuery')->once()->andReturn($query);
        $relation->expects($this->once())->method('touchIfTouching');

        $this->assertTrue($relation->detach());
    }


    public function getRelation()
    {
        list($builder, $parent) = $this->getRelationArguments();

        return new MorphToMany($builder, $parent, 'taggable', 'taggables', 'taggable_id', 'tag_id');
    }

    public function getRelationArguments()
    {
        $parent = m::mock('Illuminate\Database\Eloquent\Model');
        $parent->shouldReceive('getMorphClass')->andReturn(get_class($parent));
        $parent->shouldReceive('getKey')->andReturn(1);
```

LUIS Integration

Deployment

- The Bot is just a REST API
- Can be hosted anywhere
- Azure App Service is an easy choice
  - Azure Functions also supported

Channels



Takeaways

- Chat bots are another form of UI (NLUI)
- Microsoft Bot Framework makes it easier to write your bot
  - Standard connection to various channels
  - Bot Builder SDK
  - Integration with Cognitive Services
- The trickiest part is still designing the conversation

Q
A

Thank you!