

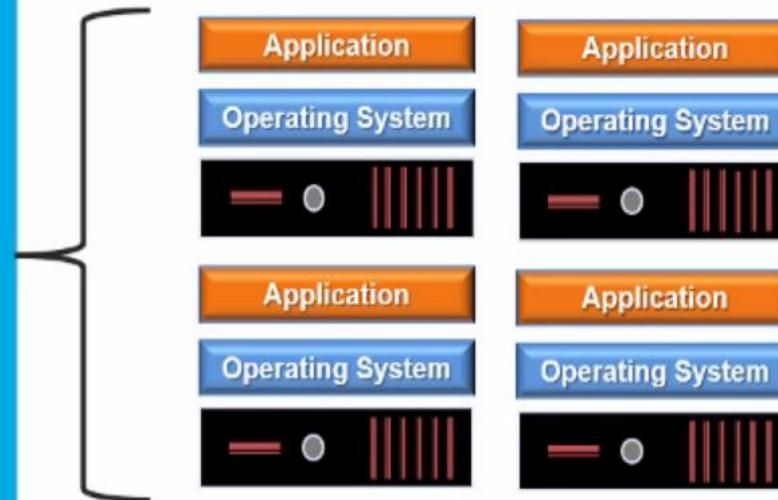
# Azure Fundamentals

# Traditional Computing Model

Host (HP, DELL, etc.) – Hardware Server made up of CPU, Memory, Network and Disk Subsystem

Operating System (Windows, Linux, etc.) – Provides the Basic Access to the Hardware and allows for Security, File Systems Management, Databases and Application Programming Interfaces.

Applications (Office, SQL , Apache) – Applications are then installed on top of the Operating Systems ranging from User applications, Web Servers, DBMS and APIs for other Servers.



## Issues with the traditional model

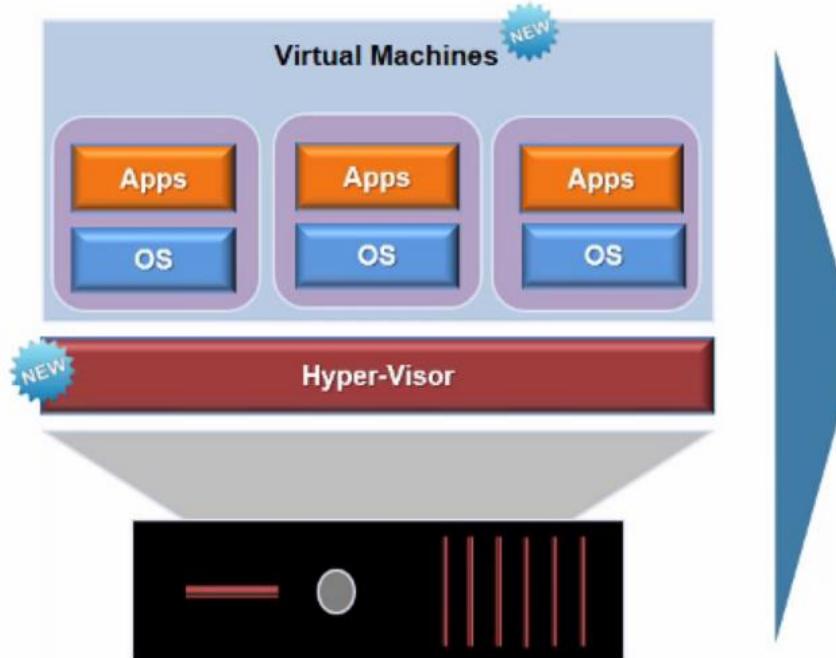
- Not Energy Efficient
- Server Sprawl
- Slow Provisioning of Servers
- Single Point of Failure
- Difficult for Disaster Recovery
- Under Utilized Servers

# Traditional Computing Model

Virtualization Software allows for the Hardware and Operating System to be Separated and Multiple Installations of an OS on the same Hardware

Hypervisor (VMware ESX, Hyper-V) – is Software which runs directly on the Host and allows for Resource Allocation and Manages Virtual Machines

Virtual Machine (VMWare) – Container that is a software based "Server" and is provided hardware resources via the Hypervisor



Enabling Technology for Optimal Use of Hardware Investments

Consolidation of Hardware (Shared Workloads)

High Availability

Reliability with Real-time Failover

Disaster Recovery

Ease of Management

## Azure

Infrastructure as a Service (IaaS)

Platform as a Service (PaaS)

Software as a Service (SaaS)



# Why IaaS?

# Why Infrastructure as a Service?

- ✓ Deploy anywhere with Azure as your data center
- ✓ Run virtually any Windows or Linux based workload
- ✓ Monitoring, scale, and automation is built in
- ✓ Flexible hybrid connectivity options



→ Let's compare a traditional vs. an Azure deployment

# Deploying a Traditional N-Tier Infrastructure

Order Hardware

Setup Network Infrastructure

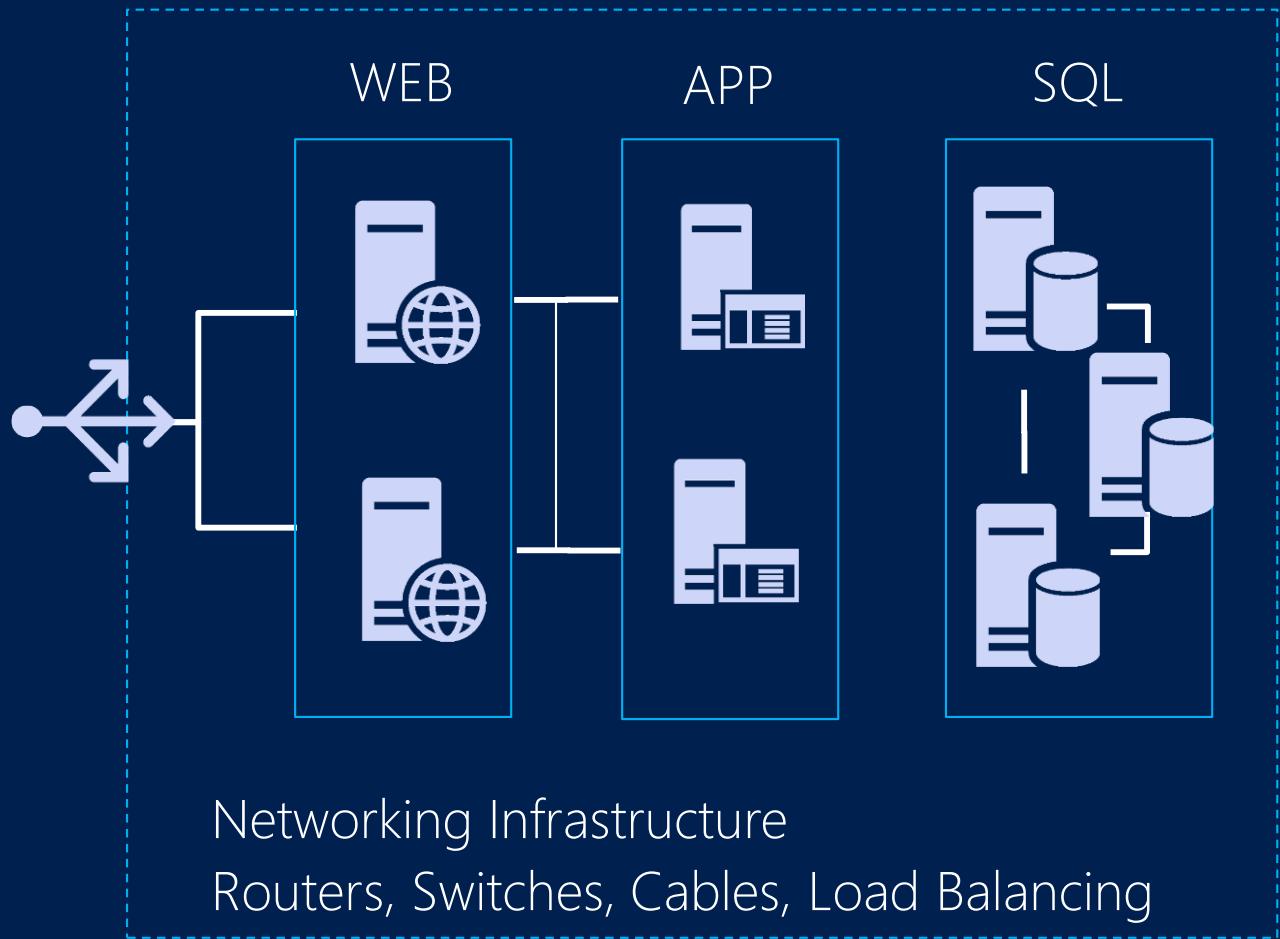
Setup Server Infrastructure

Install and Patch OS

Install and Configure Apps

Multiple Environments?

Repeat steps



# Deploying with Microsoft Azure IaaS

## Without automation

Configure Virtual Network

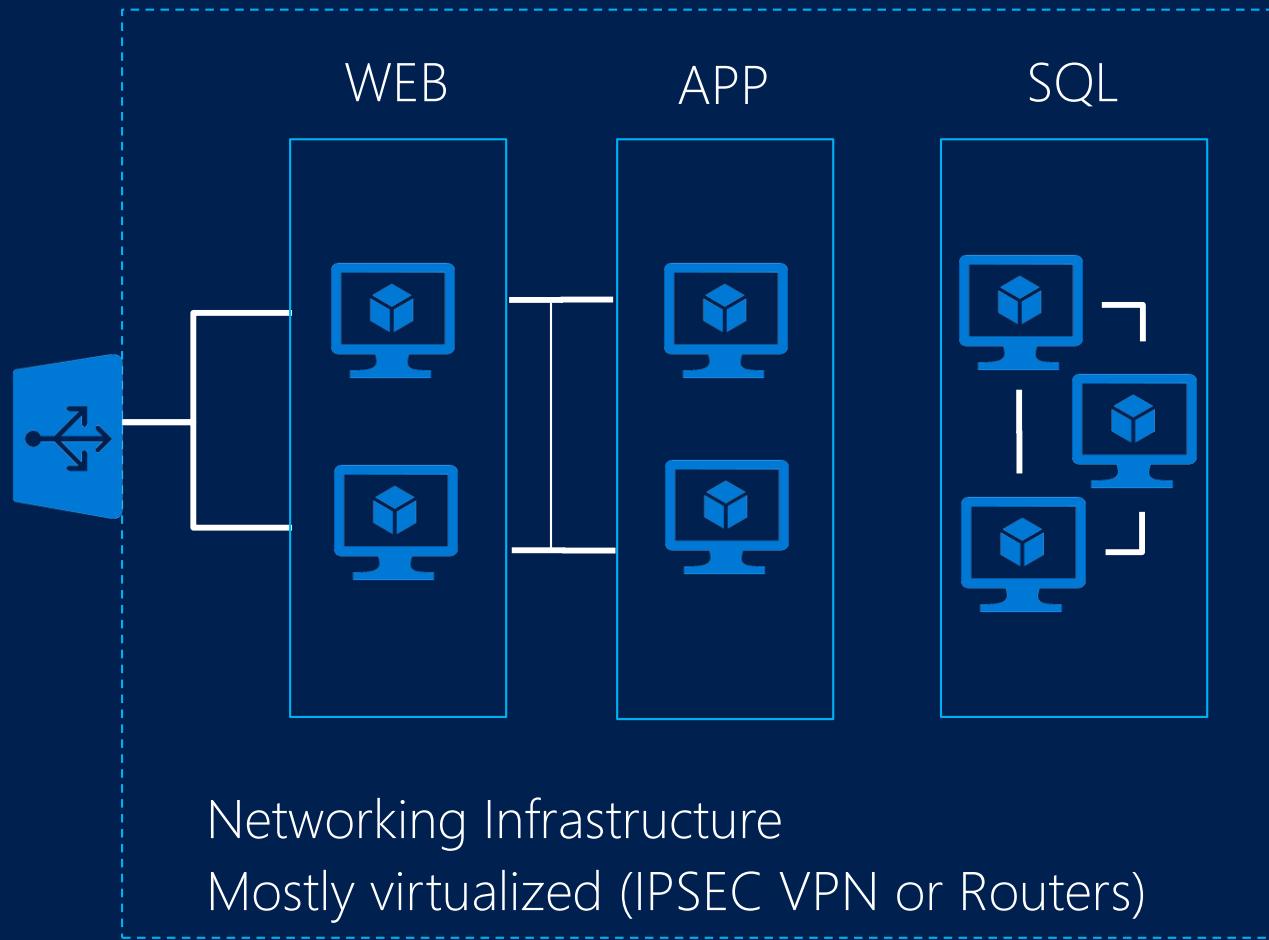
Deploy VMs and Configure Load Balancer

Patch the OS

Install and Configure Apps

Multiple Environments?

Repeat steps



# Why PaaS?

# Why Platform as a Service?

- ✓ complete development and deployment environment in the cloud
- ✓ Pay-as-you-go and access over a secure Internet connection.
- ✓ PaaS includes infrastructure—servers, storage, and networking but also middleware, development tools, business intelligence (BI) services, database management systems, and more.
- ✓ Support the complete web application lifecycle: building, testing, deploying, managing, and updating.
- ✓ Avoid the expense and complexity of buying and managing software licenses, the underlying application infrastructure and middleware or the development tools and other resources.
- ✓ Built-in Maintenance, Monitoring and Scaling

# Deploying with Microsoft Azure PaaS

## Configure PAAS Services

- App Service
  - Web App
  - Web Job
- Sql Azure

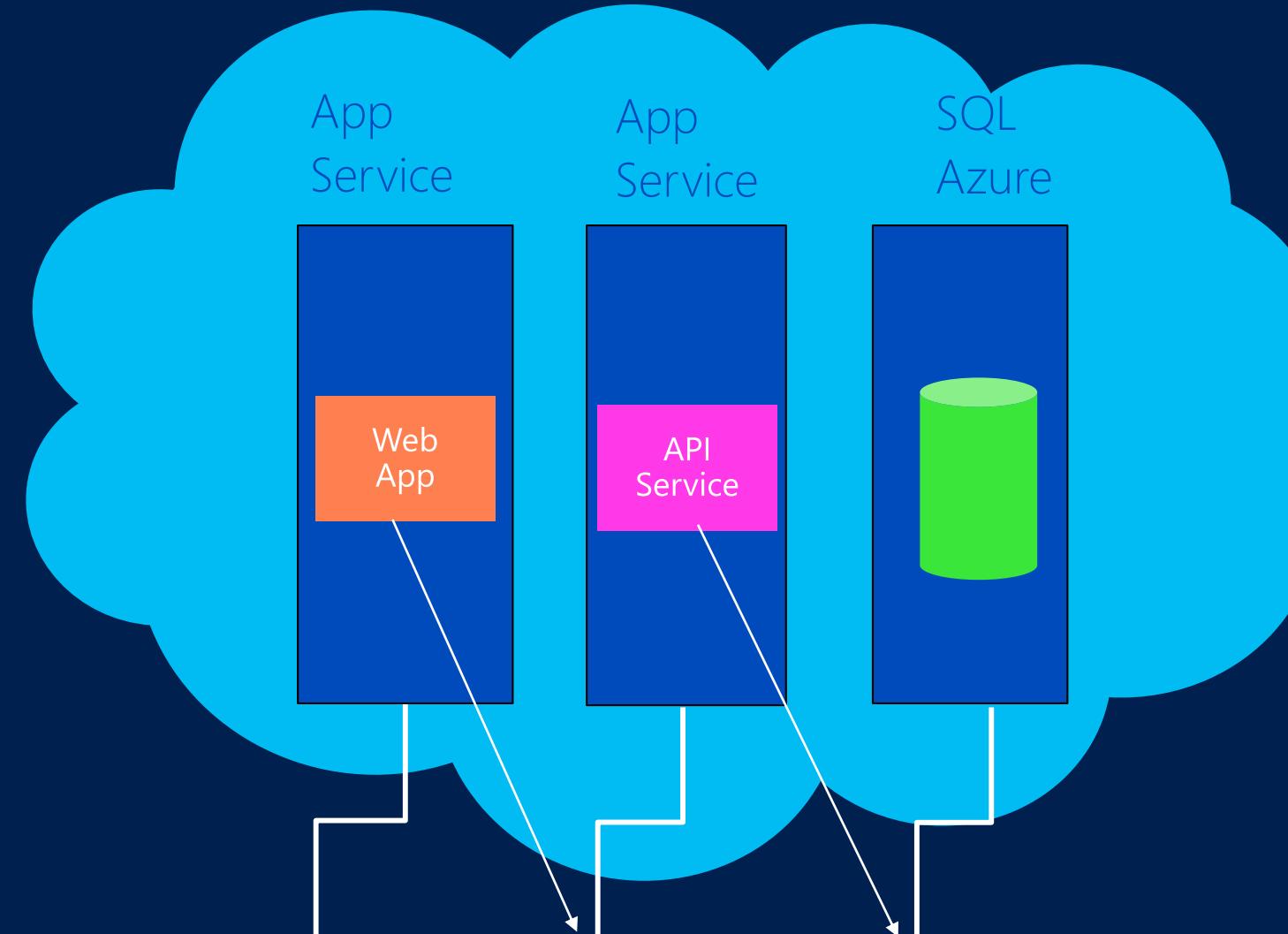
~~No need to Patch the OS or stand platform~~

Deploy Applications and Data

~~No need to deploy loadbalancers~~

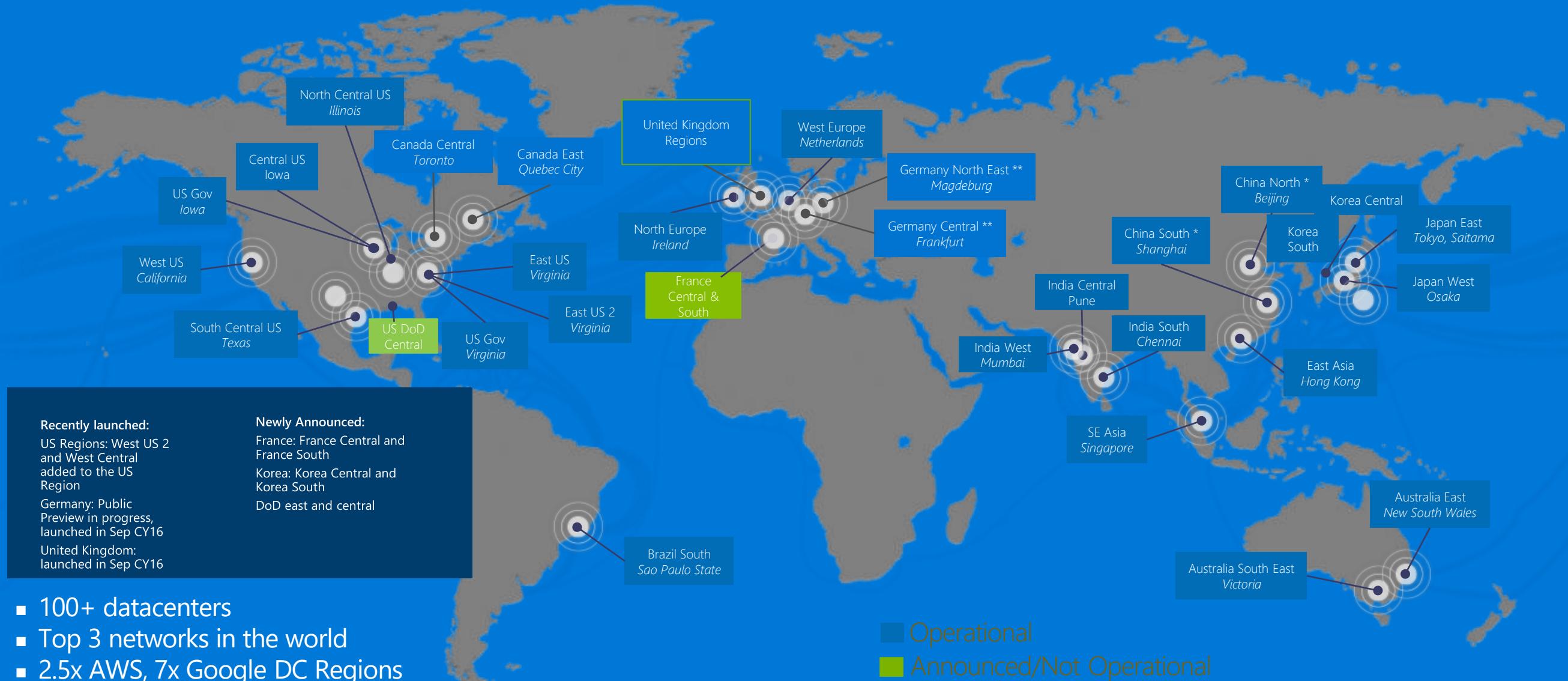
Redundancy is Built-In

Built-In support for Auto Scale based  
on schedule/performance metric/etc.



# Hyper scale Infrastructure

38+ Regions Worldwide, 30 ONLINE...huge capacity around the world...growing every year



# Paired regions for geo- redundancy

## PAIRED REGIONS PROVIDE

Isolation & Replication

Region order recovery

Sequential updates

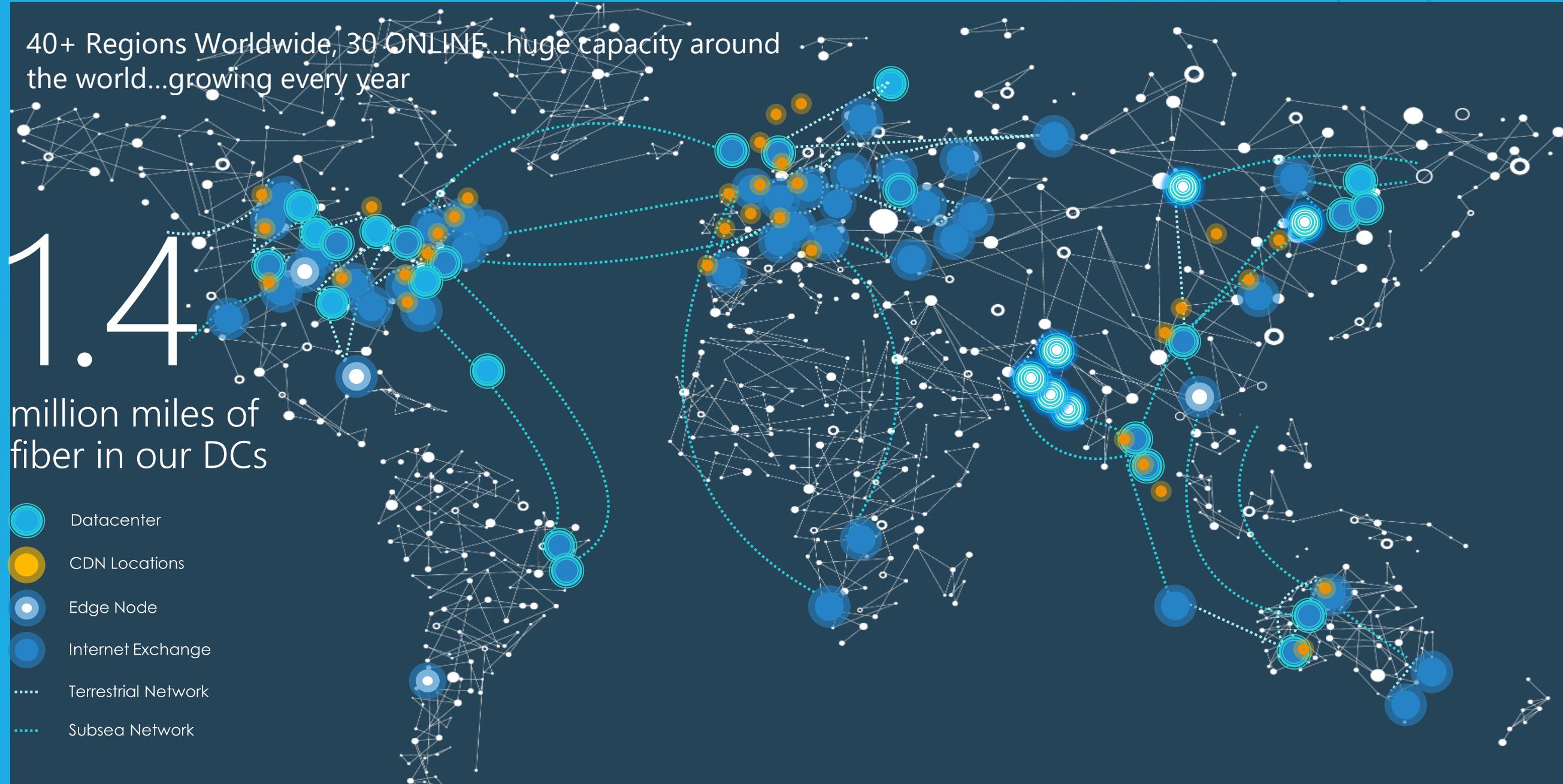
Data residency

North Central US  
East US  
West US 2  
US East 2  
US Gov Iowa  
Canada Central  
North Europe  
UK West  
Germany Central  
South East Asia  
East China  
Japan East  
Australia Southeast  
India South  
Brazil South (Primary)



South Central US  
West US  
West Central US  
Central US  
US Gov Virginia  
Canada East  
West Europe  
UK South  
Germany Northeast  
East Asia  
North China  
Japan West  
Australia East  
India Central  
South Central US

# Hyper scale Infrastructure



# Azure is an open cloud

DevOps



Clients



APACHE CORDOVA™

Management



SCALR  
CLOUD MANAGEMENT

Applications



PaaS &  
DevOps



App Frameworks  
& Tools



Databases &  
Middleware



Infrastructure



# The most trusted and compliant cloud



Trust

GLOBAL



ISO 27001



ISO 27018



ISO 27017



ISO 22301



SOC 1  
Type 2



SOC 2  
Type 2



SOC 3



CSA STAR  
Self-Assessment



CSA STAR  
Certification



CSA STAR  
Attestation

US GOV



Moderate  
JAB P-ATO



High  
JAB P-ATO



DoD DISA  
SRG Level 2



DoD DISA  
SRG Level 4



SP 800-171



FIPS 140-2



Section  
508 VPAT



ITAR



CJIS



IRS 1075

INDUSTRY



PCI DSS  
Level 1



CDSA



MPAA



FACT  
UK



Shared  
Assessments



FISC  
Japan



HIPAA /  
HITECH Act



HITRUST



GxP  
21 CFR Part 11



MARS-E



IG Toolkit  
UK



FERPA



GLBA



FFIEC

REGIONAL



Argentina  
PDPA



EU  
Model Clauses G-Cloud



UK  
GOV.UK



China  
DJCP



China  
GB 18030



China  
TRUCS



Singapore  
MTCS



Australia  
IRAP/CCSL



New  
Zealand  
GCIO



Japan  
My  
Number  
Act



ENISA  
IAF



Japan CS  
Mark Gold



Spain  
ENS



Spain  
DPA



India  
MeitY



Canada  
Privacy  
Laws



Germany  
Privacy  
Shield



Germany  
Grundschutz  
workbook

# Microsoft Azure

Microsoft Azure is sold in two different forms

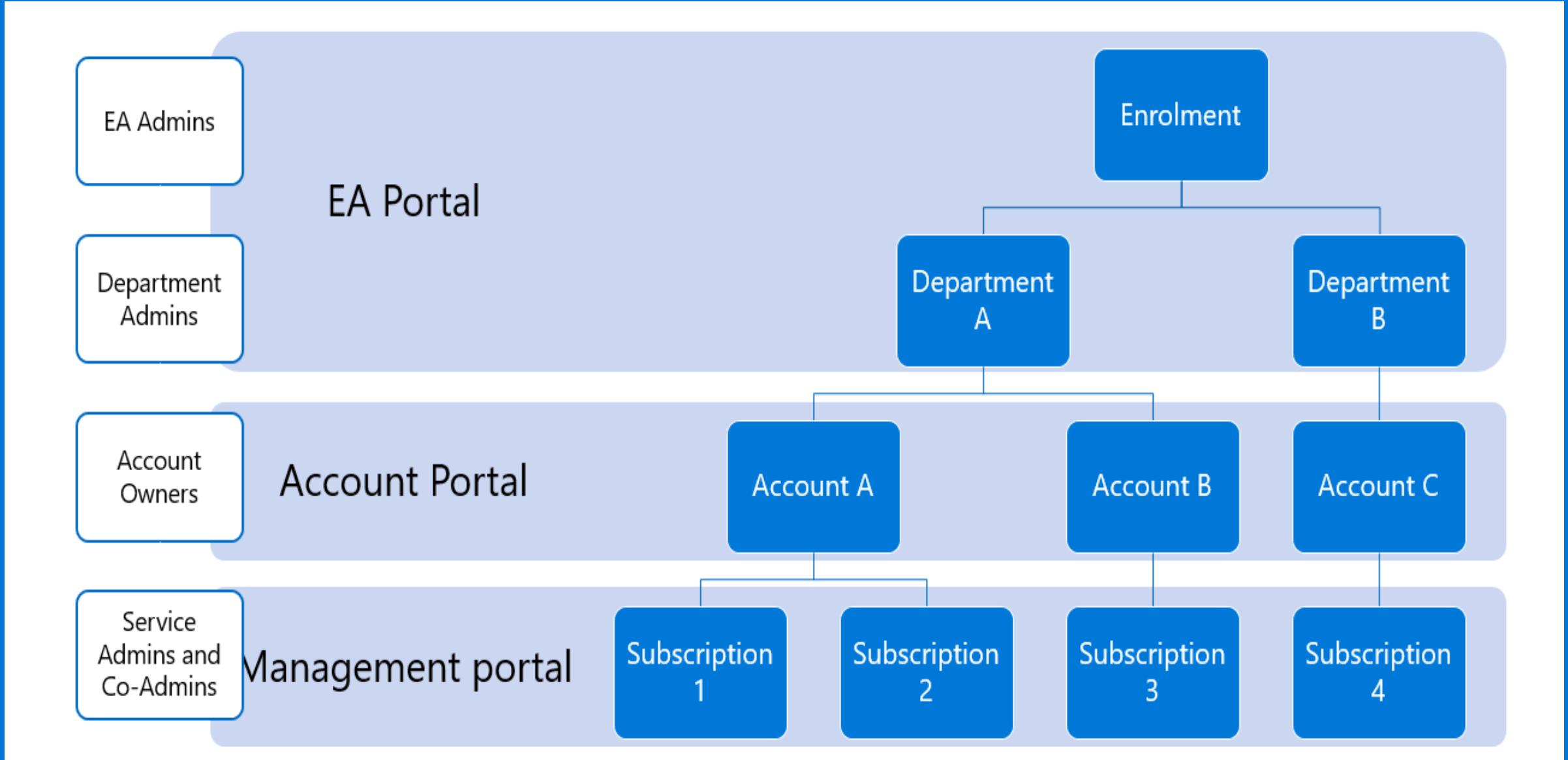
- Consumer
- Enterprise Agreement

Consumer: End user can swipe their credit card and get started.

**Enterprise Agreement:** Enterprise agreements are for enterprise customers and of course you commit to a certain amount and based on the commitment, you get discounts.

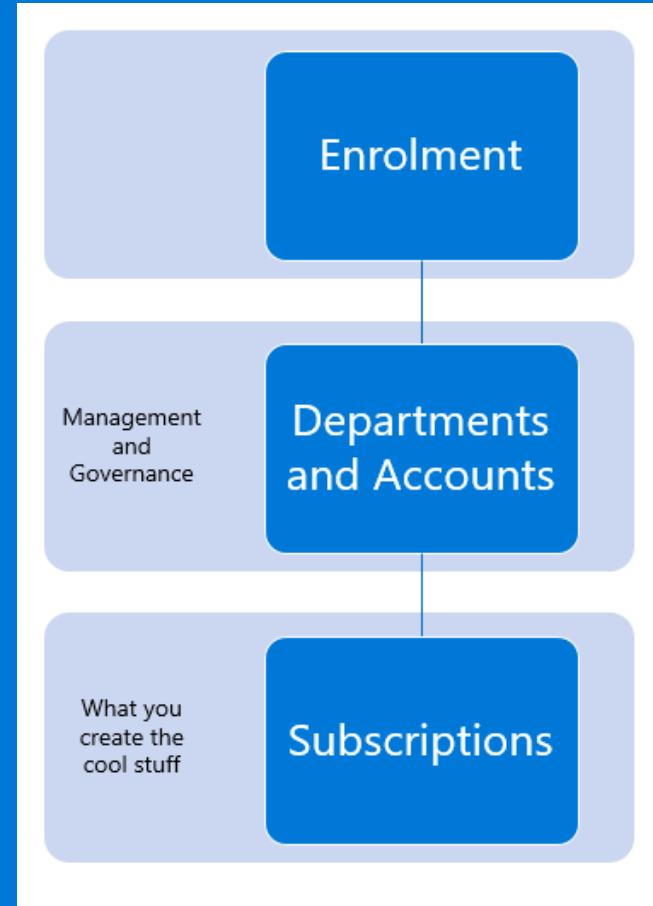
- However, for enterprise customers you get to deal with two portals and not one. The first portal you will start dealing with is the [ea.windowsazure.com](http://ea.windowsazure.com). It is also called as enterprise portal [EP].
- EP is the place where you can create departments, assign owners to these departments and under each department you can have one or more subscriptions

# Enterprise Azure Roles and Portals



# Departments

- The Department feature provides the Enterprise Administrator and Department Administrator the ability to organize and report on Enterprise Azure services and usage by Department and Cost Center.
- The Enterprise Administrator will be able to add/remove Departments, associate an Account to a Department, create Department Administrators, and dictate if Department Administrator can view price and costs.
- Email alerts are generated when spend thresholds are passed
- Accounts can then be associated against a Department



# Accounts

In an EA, Subscriptions are created under an Azure Account

- This is not a user Account

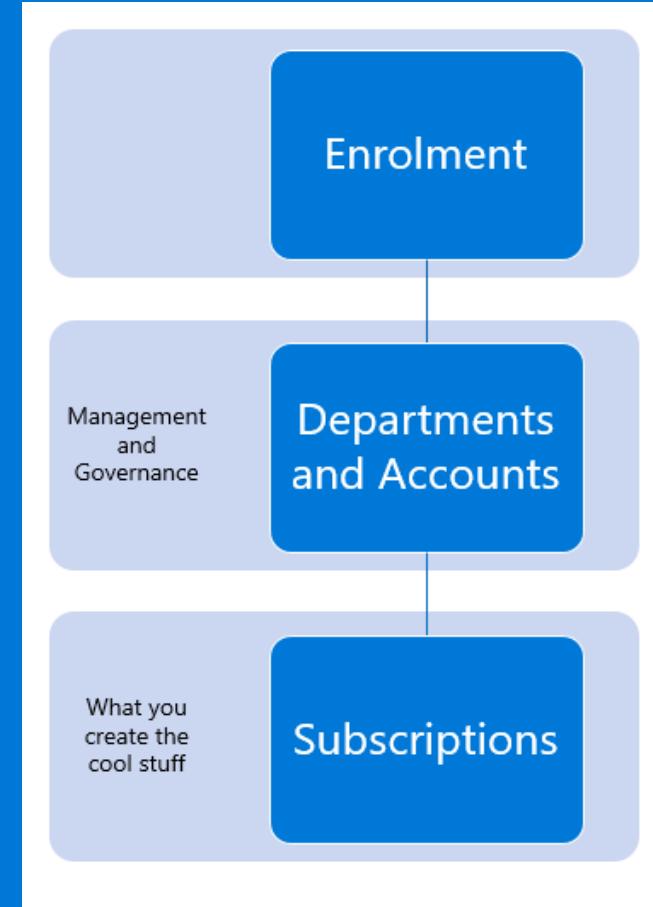
Windows Azure Account is a shell to provide;

- Reporting on usage of the services.
- Billing of the services.
- And Account Administration.

Managed by:

- Account Owner
- Managed via the Account Portal by a single Account Owner

[account.windowsazure.com](http://account.windowsazure.com)



# Subscriptions

- Subscriptions help you organize access to resources. They also help you control how resource usage is reported, billed, and paid for.
- This is actually where the magic happens, a subscription has the Azure resources in it and the Administrators can give users permissions to add, create etc. resources in that subscription.
- Every cloud service belongs to a subscription, and the subscription ID may be required for programmatic operations.
- By default subscriptions are completely independent from each other.

## Managed by:

- Classical Portal: Service Administrator, Co-Administrator,
- Portal: User Access Administrator, Subscription Owner\*
  - \*By default the Service Administrator as well as the Co-Administrator are subscription owners on the RBAC-Level.

# Roles in Enterprise Portal

To administer your Microsoft Azure Enterprise Enrollment, there are 4 distinct administrative roles:

- Enterprise Administrator
- Department Administrator
- Account Owner
- Service Administrator

Enterprise Administrator:

- Manage Accounts and Account Owners
- Manage Enterprise Administrators
- View usage across all accounts
- There may be multiple Enterprise Administrators per Enterprise Enrollment

Department Administrator:

- Create and manage Departments
- Create new Account Owners
- View Usage Details for Departments they manage
- View Costs if granted necessary permissions
- There may be multiple Department Administrators per Enterprise Enrollment

# Roles in Enterprise Portal

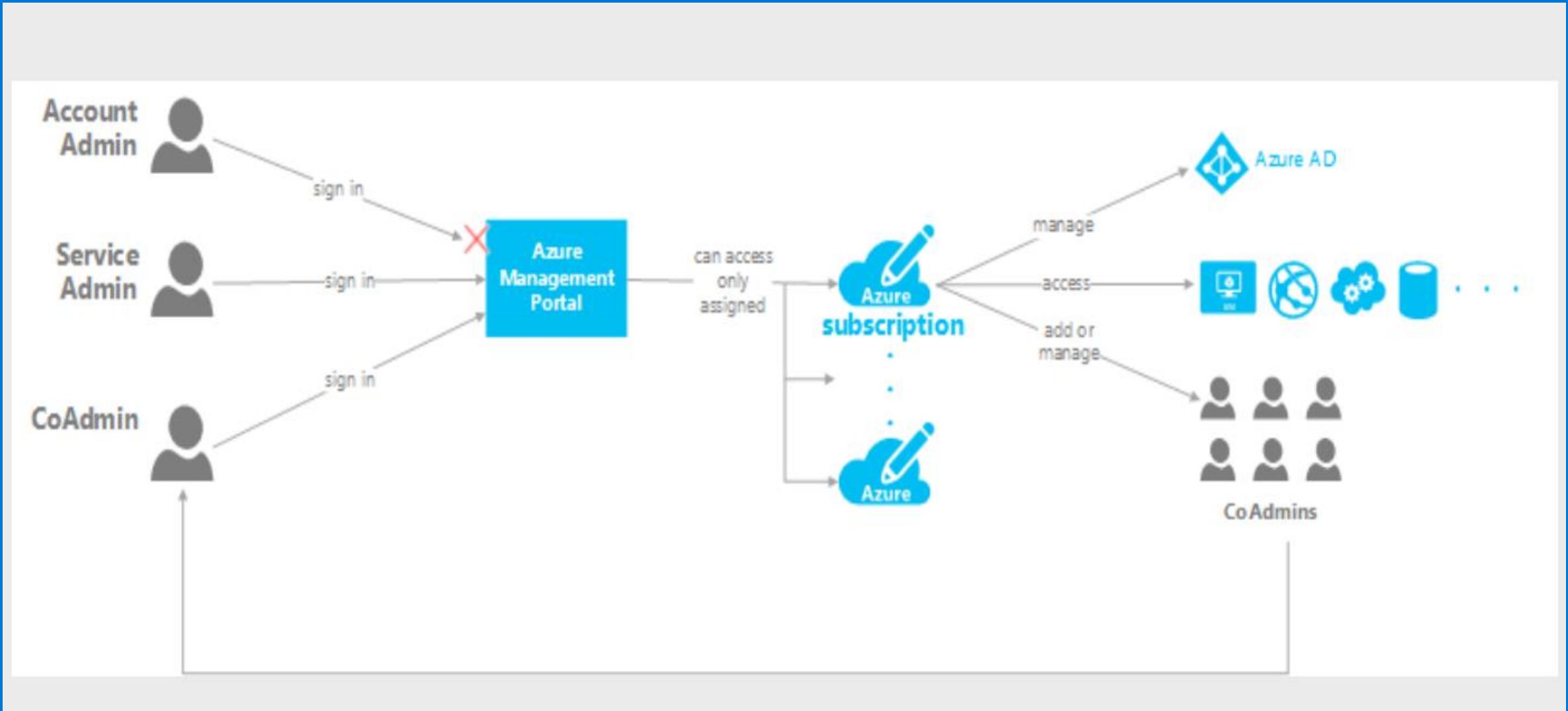
## Account Owner:

- Create and manage subscriptions
- Manage Service Administrators
- View usage for subscriptions
- Each account requires a unique Microsoft Account or Work or School Account
- Person who created Azure account is the account administrator.

## Service Administrator:

- Access and manage subscriptions and development projects on the developer portal
- A single Microsoft Account or Work or School Account may be used across subscriptions and between hierachal levels
- Account admin is the default service admin when azure subscription is created.
- Only account administrator can change service administrator.

# Subscription Access



# Subscription management starts with selecting the right identity

Azure uses both Microsoft identity services

Two services:

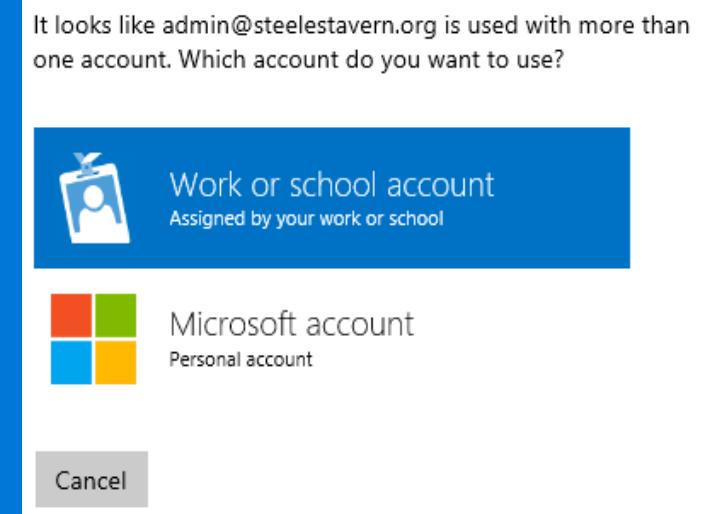
Azure Active Directory (AAD) system

Microsoft Account (MSA) system

Two types of accounts:

Work or school account (AAD)

Microsoft account (MSA)



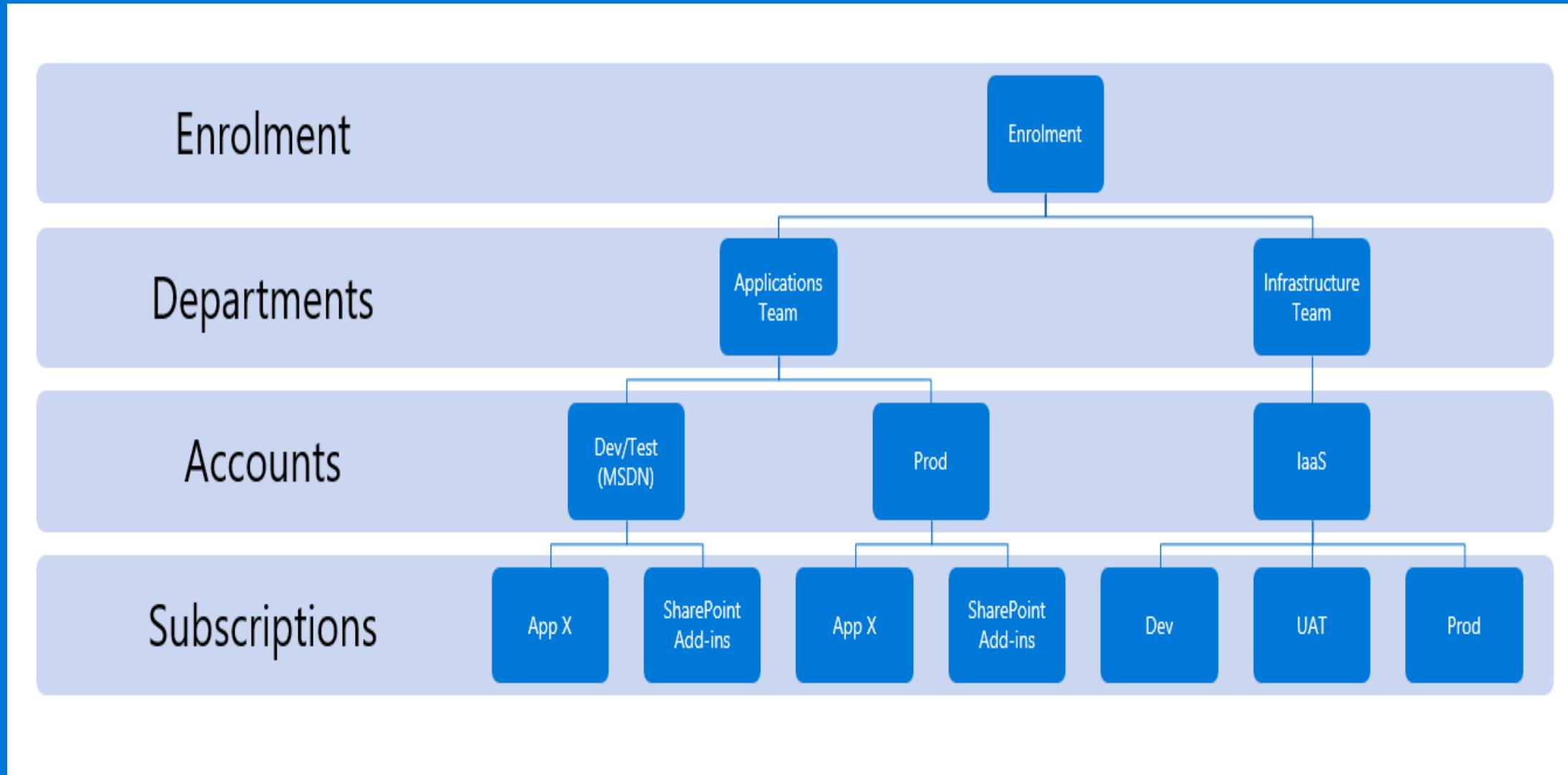
The impact:

Two different accounts can exist with the same username

Different passwords

Different access

# Example – IT



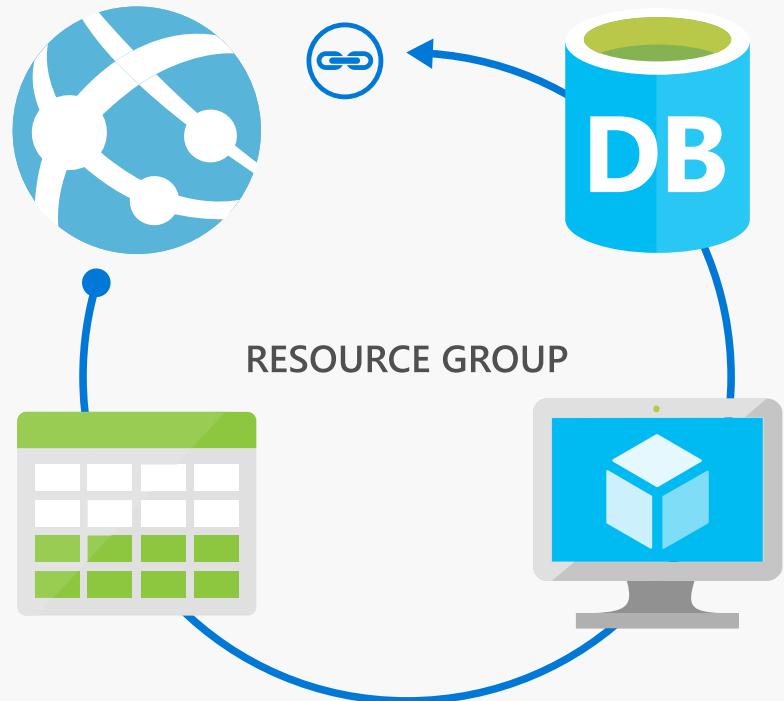
# Subscription Design Consideration

Subscription alignment by:

- Line of Business (LOB)
- Environment (PROD, QA, TEST, DEV)
- Geography
- Organizational Unit / Business Unit
- Functional

# Azure EA Portal Demo

# Azure Resource Manager (ARM)



- Resource Manager provides a consistent management layer for the tasks you perform through Azure PowerShell, Azure CLI, Azure Portal, REST API and development tools.
- Resource groups are containers that can contain multiple IaaS + PaaS resources
- Support lifecycle management with integrated Role Based Access Control (RBAC)
- Templatize application deployment and configuration
- Supports DevOps

# Terminology in Azure Resource Manager

**Resource** : A manageable item that is available through Azure. Some common resources are a virtual machine, storage account, web app, database, and virtual network, but there are many more

**Resource Group** : A container that holds related resources for an Azure solution. The resource group can include all the resources for the solution, or only those resources that you want to manage as a group

**Resource Provider** : A service that supplies the resources you can deploy and manage through Resource Manager. Each resource provider offers operations for working with the resources that are deployed. Some common resource providers are Microsoft.Compute, which supplies the virtual machine resource, Microsoft.Storage, which supplies the storage account resource, and Microsoft.Web, which supplies resources related to web apps.

# Terminology in Azure Resource Manager

**Resource Manager Template:** A JavaScript Object Notation (JSON) file that defines one or more resources to deploy to a resource group. It also defines the dependencies between the deployed resources. The template can be used to deploy the resources consistently and repeatedly.

**Declarative Syntax:** Syntax that lets you state "Here is what I intend to create" without having to write the sequence of programming commands to create it. The Resource Manager template is an example of declarative syntax.

# Azure Resource Group

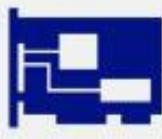
- Collection of resources with the same lifecycle
- Every resource belongs to a resource group (and only one resource group)
- Resources have types, defined by resource providers
- Role based access control
- Declarative model driven deployment
- Consistent



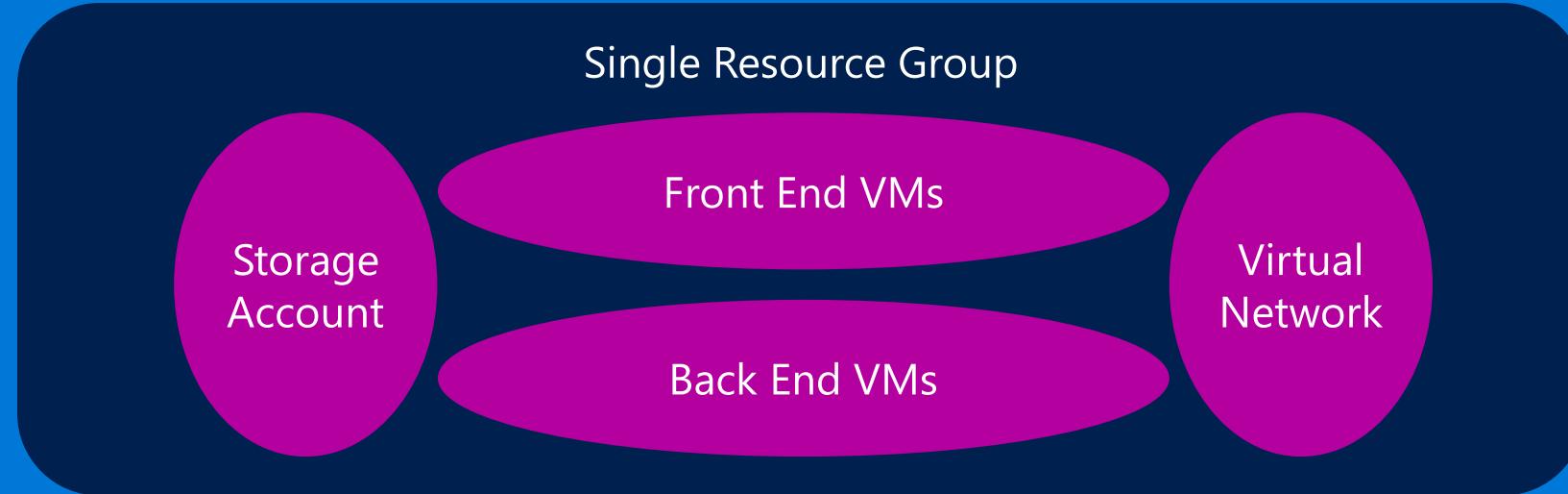
## Resource Group

### Resources

- VMs
- NICs
- Storage
- Web Apps
- SQL
- Virtual Networks



# Single or Multiple Resource Groups



# Multiple Resource Groups



## Azure Resource Manager (ARM)

- Used for managing resources created with Azure Resource Manager
- Deploy templates
- Perform imperative creates and updates
- All cmdlets have AzureRm prefix
- No more Switch-AzureMode

## Azure Service Management (ASM)

- Used for managing classic resources (cloud service based)
- No template deployment
- Perform imperative creates and updates
- All cmdlets have Azure prefix
- No more Switch-AzureMode

# RBAC in Azure

## 30 Built-In Roles

Owner, Contributor, Reader and

Resource type-specific admin roles: VM Contributor, VNet Contributor, SQL DB Contributor ...

Custom roles

## Fine-Grained Control

Grant access to AD groups, users, and services

Manage access at granular scopes (resource groups, subscriptions, and individual resources)

Delegate specific access management

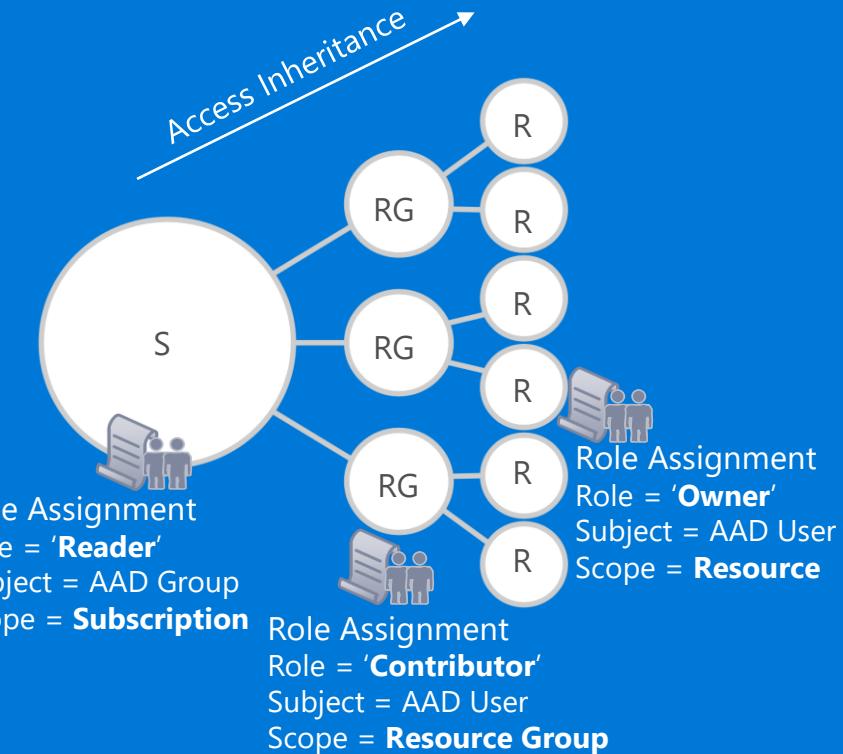
## Simple Access Management Experience

Manage access using portal, command-line, and REST APIs

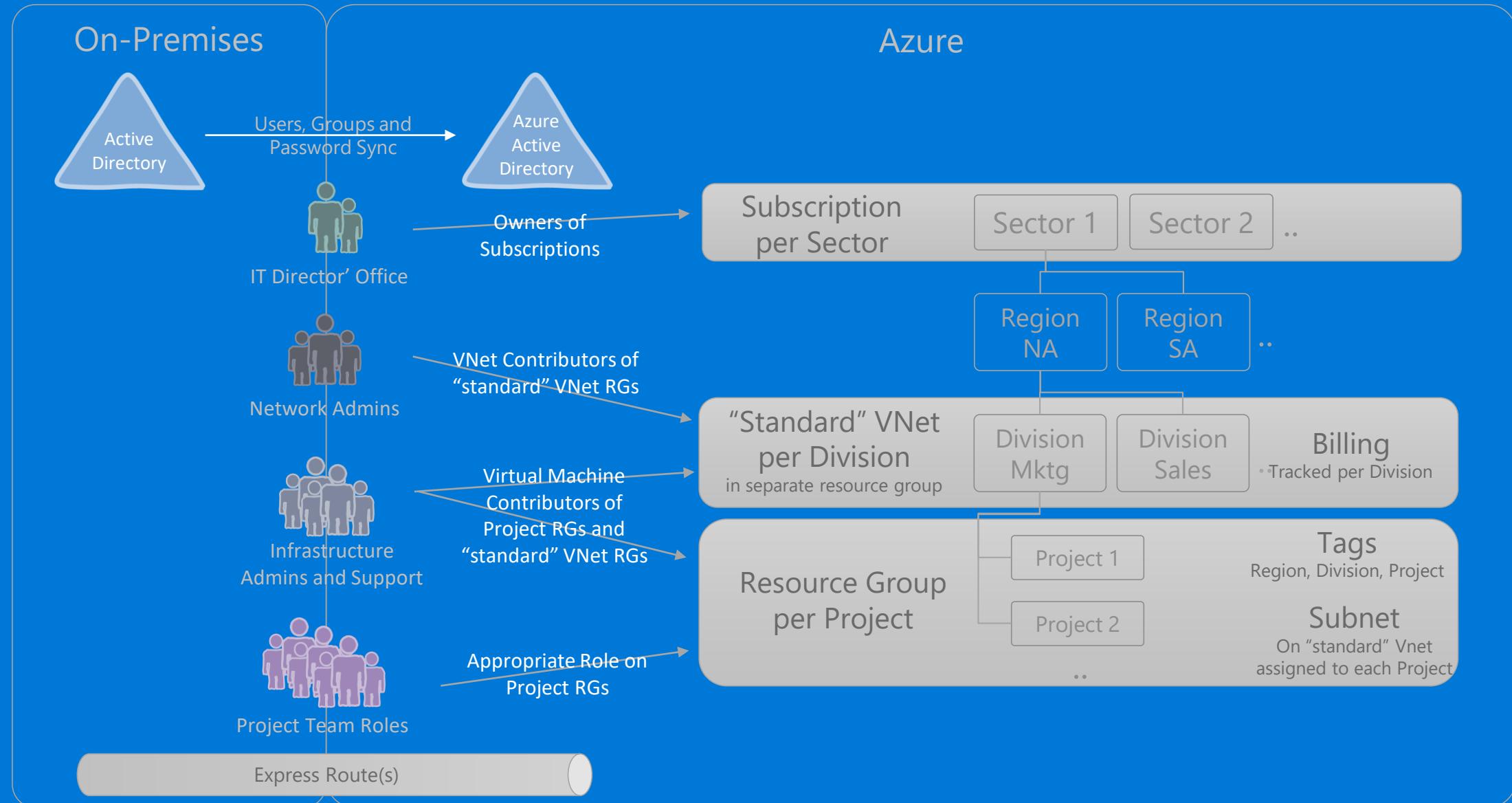
Analyze access settings and keep track of changes

Co-existence with classic service admin, and co-admin model

# Access Inheritance and Resource Hierarchy



# Actual Enterprise Implementation



# Azure Resource Policies

Resource Policies enable you to establish conventions for resources in your organization. By defining conventions, you can control costs and more easily manage your resources.

Two Concepts to understand about policies:

- policy definition - you describe when the policy is enforced and what action to take
- policy assignment - you apply the policy definition to a scope (subscription or resource group)

# Azure Portal Demo

# Azure PowerShell Installation

# PowerShell Installation

Azure PowerShell is a set of modules that provide cmdlets to manage Azure with windows PowerShell.

- You can use the cmdlets to create, test, deploy and manage solutions and services delivered through Azure platform.

Download and Install Azure PowerShell using the link below

<https://www.microsoft.com/web/handlers/webpi.ashx/getinstaller/WindowsAzurePowershellGet.3f.3f.3fnew.appids>

Enter the command below which enables you to run the PowerShell scripts.

```
Set-ExecutionPolicy -ExecutionPolicy RemoteSigned
```

# Azure CLI: Managing Azure Resources

- Azure Management tasks – UI, SDK, command lines
  - Azure management portal – web based UI, great for individual tasks, dashboard
  - Management SDKs – programmatically build custom dashboards/tools, app specific management/resources provisioning
  - Command line – automation tasks, scripting
- Command line and scripting choices – PowerShell, CLI
  - Both available on Windows, Mac, Linux
  - Personal choice based on comfort, skill level and muscle memory
- Azure CLI 2.0 and 1.0
  - Azure CLI 2.0 is the newer version
  - Co-exists with 1.0 version which continues to be maintained and supported
  - ASM service management is only available in Azure CLI 1.0

# Introducing Azure CLI 2.0



```
Welcome to the cool new Azure CLI!
Here are the base commands:
account      : Manage subscriptions.
acr          : Manage Azure container registries.
acs          : Manage Azure Container Services.
ad           : Synchronize on-premises directories and manage Azure Active Directory resources.
appservice    : Manage your App Service plans.
batch         : Manage Azure Batch.
cdn          : Manage Azure Content Delivery Networks (CDN).
cloud         : Manage the registered Azure clouds.
cognitiveservices: Manage Cognitive Services accounts in Azure Resource Manager.
component     : Manage and update Azure CLI 2.0 components.
configure     : Configure Azure CLI 2.0 or view your configuration. The command is interactive, so just type `az configure` and respond to the prompts.
cosmosdb     : Manage Azure Cosmos DB database accounts.
disk          : Manage Azure Managed Disks.
dla           : Commands to manage Data Lake Analytics accounts, jobs, and catalogs.
dls           : Commands to manage Data Lake Store accounts, and filesystems.
feature       : Manage resource provider features, such as previews.
feedback      : Loving or hating the CLI? Let us know!
find          : Find Azure CLI commands based on a given query.
functionapp   : Manage your function app.
group         : Manage resource groups and template deployments.
image         : Manage custom Virtual Machine Images.
interactive   : Start the interactive experience.
iot           : Connect, monitor, and control millions of IoT assets.
keyvault      : Safeguard and maintain control of keys, secrets, and certificates.
lab           : Commands to manage DevTest Labs.
lock          : Manage Azure locks.
login         : Log in to access Azure subscriptions.
logout        : Log out to remove access to Azure subscriptions.
managedapp    : Manage template solutions provided and maintained by the ISV using managedapp and managedapp definitions.
monitor       : Commands to manage Azure Monitor service.
mysql         : Commands to manage Azure MySQL servers.
network       : Manages Azure Network resources.
policy        : Manage resource policies.
postgres      : Commands to manage Azure PostgreSQL servers.
provider      : Manage resource providers.
redis         : Access to a secure, dedicated cache for your Azure applications.
resource      : Manage Azure resources.
role          : Use role assignments to manage access to your Azure resources.
sf            : Manage and administer a Service Fabric cluster.
snapshot      : Manage point-in-time copies of managed disks, native blobs, or other snapshots.
sql           : Manage Azure SQL Databases and Data Warehouses.
storage       : Durable, highly available, and massively scalable cloud storage.
tag           : Manage resource tags.
vm            : Provision Linux or Windows virtual machines in seconds.
vmss          : Create highly available, auto-scalable Linux or Windows virtual machines.
webapp        : Manage web apps.
```

[0] ~ \$

Azure CLI  
[CLI Docs and Samples](#)  
[Blogs](#)

# What is a Modern CLI?

- Native and easy install steps
  - Apt-get, msi, curl, pip etc options for each supported platform (Mac, Linux, Windows)
- Natural and idiomatic
  - Piping and output direction
  - Works with GREP, AWK, JQ etc
  - Productivity features like tab completion, parameter values, help all built-in
  - Environment aware (e.g. SSH keys, etc)
- Intuitive, easy to discover and learn commands
  - Predictable command structure, interactive mode, az find
  - Smart defaults
  - Business logic
  - Output format choices
  - JMESPath query language support
- Extensible and customizable
  - Out of band and 3rd party command modules, open-source and python based
- Consistent, full support of Azure services

```
$ az vm list --query
"[].{name:name,os:storageProfile.osDisk.osType}"
Name          Os
-----
myvm          Linux
VM-Data        Windows
VM-StagingWeb Linux
VM-Web         Linux
xiotest        Linux

$ az vm create -g VMGroup1 -n MyLinuxVM --image ubuntults
MacAddress           ResourceGroup      PublicIpAddress      PrivateIpAddress
ess
-----
...
00-0D-3A-34-A3-48   VmGroup1          52.160.109.58       10.0.0.9

$ ssh 52.160.109.58
[...]
Welcome to Ubuntu 14.04.4 LTS (GNU/Linux 3.19.0-65-generic x86_64)
 System load: 0.15
 2% Processes:     81
    Usage of /: 39.6% of 1.94GB   Swap usage:  0%   Users logged in: 0

0 packages can be updated.
0 dates are security updates.

jasonsha@MyLinuxVM:~$
```