Module 1 - Introduction to Cloud and Azure

Lesson 1: Cloud technology overview

Lesson 2: Overview of Azure

Lesson 3: Managing Azure with the Azure portal

Lesson 4: Managing Azure with Windows PowerShell

Lesson 5: Managing Azure with Azure CLI

Lesson 6: Overview of Azure deployment models

Lesson 7: Managing and monitoring Azure resources

Module 1 - Lesson 1 : Cloud technology overview

What is Cloud Computing?

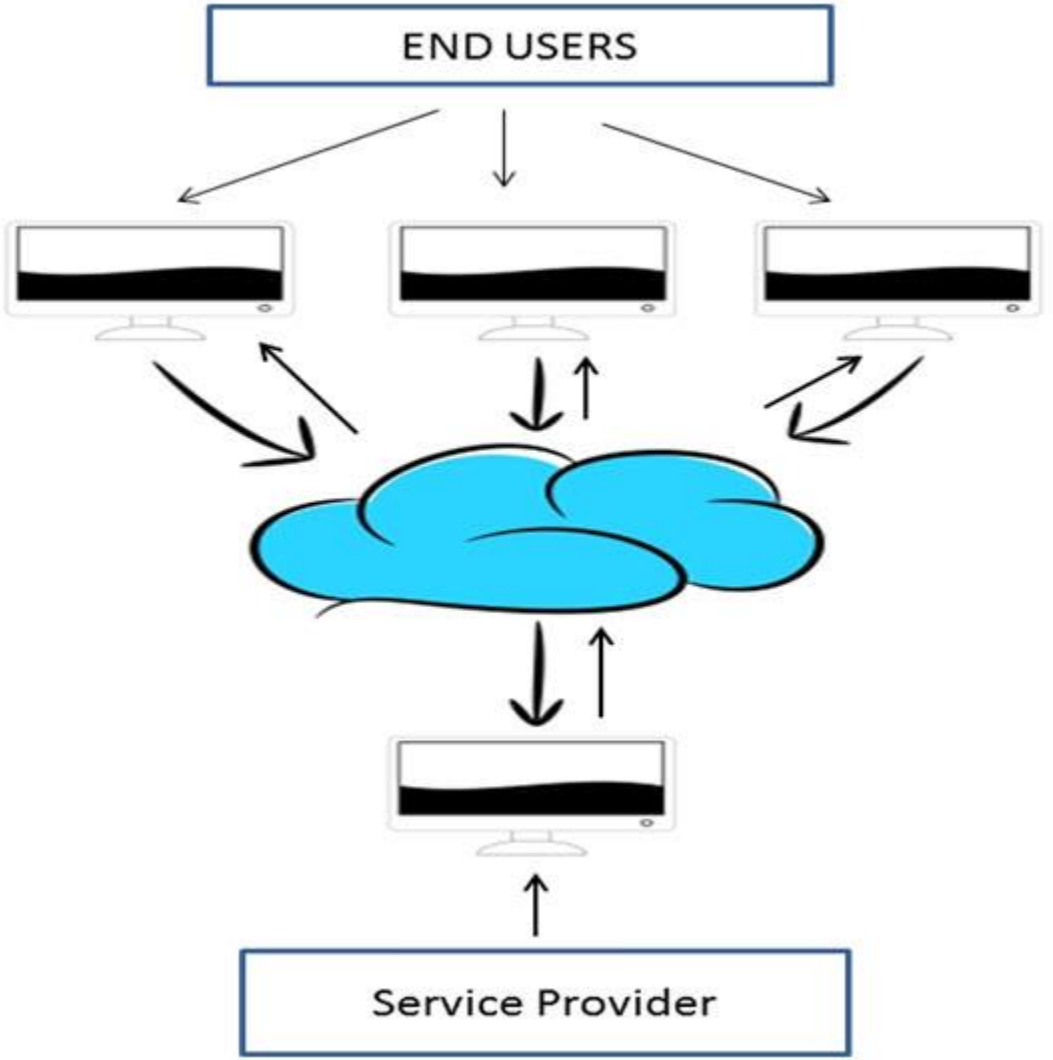
Cloud Computing can be defined as delivering computing power(CPU, RAM, Network Speeds, Storage OS software) a service over a network (usually on the internet) rather than physically having the computing resources at the customer location.

Example: AWS, Azure, Google Cloud

Cloud Computing provides us a means by which we can access the applications as utilities, over the Internet.
It allows us to create, configure, and customize applications online.
With Cloud Computing users can access database resources via the internet from anywhere for as long as they need without worrying about any maintenance or management of actual resources

Why the Name Cloud?

The term "Cloud" came from a network design that was used by network engineers to represent the location of various network devices and there inter-connection. The shape of this network design was like a cloud.



Why Cloud Computing?

With increase in computer and Mobile user's, data storage has become a priority in all fields. Large and small scale businesses today thrive on their data & they spent a huge amount of money to maintain this data. It requires a strong IT support and a storage hub. Not all businesses can afford high cost of in-house IT infrastructure and back up support services. For them Cloud Computing is a cheaper solution. Perhaps its efficiency in storing data, computation and less maintenance cost has succeeded to attract even bigger businesses as well.

Cloud computing decreases the hardware and software demand from the user's side. The only thing that user must be able to run is the cloud computing systems interface software, which can be as simple as Web browser, and the Cloud network takes care of the rest. We all have experienced cloud computing at some instant of time, some of the popular cloud services we have used or we are still using are mail services like gmail, hotmail or yahoo etc.

While accessing e-mail service our data is stored on cloud server and not on our computer. The technology and infrastructure behind the cloud is invisible. It is less important whether cloud services are based on HTTP, XML, Ruby, PHP or other specific technologies as far as it is user friendly and functional. An individual user can connect to cloud system from his/her own devices like desktop, laptop or mobile.

Cloud computing harnesses small business effectively having limited resources, it gives small businesses access to the technologies that previously were out of their reach. Cloud computing helps small businesses to convert their maintenance cost into profit. Let's see how?

In an in-house IT server, you have to pay a lot of attention and ensure that there are no flaws into the system so that it runs smoothly. And in case of any technical glitch you are completely responsible; it will seek a lot of attention, time and money for repair. Whereas, in cloud computing, the service provider takes the complete responsibility of the complication and the technical faults.

Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. This cloud model is composed of five essential characteristics, three service models, and four deployment models.

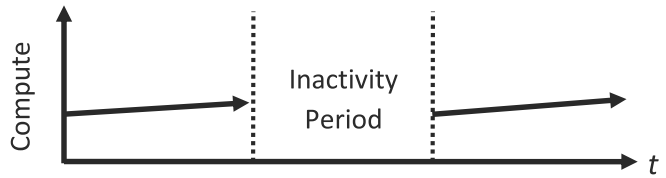
Benefits of Cloud Computing

The potential for cost saving is the major reason of cloud services adoption by many organizations. Cloud computing gives the freedom to use services as per the requirement and pay only for what you use. Due to cloud computing it has become possible to run IT operations as a outsourced unit without much in-house resources.

Following are the benefits of cloud computing:

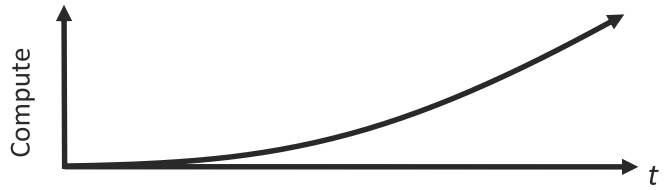
1. Lower IT infrastructure and computer costs for users
2. Improved performance
3. Fewer Maintenance issues
4. Instant software updates
5. Improved compatibility between Operating systems
6. Backup and recovery
7. Performance and Scalability
8. Increased storage capacity
9. Increase data safety

Cloud Computing Patterns



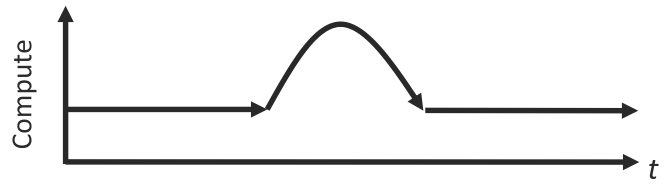
On and Off

On and off workloads (e.g. batch job)
Over provisioned capacity is wasted
Time to market can be cumbersome



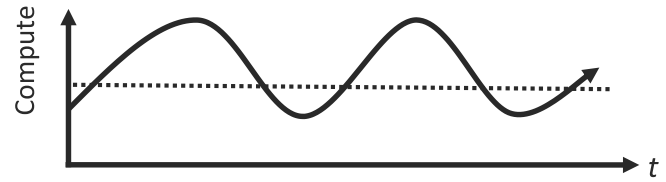
Growing Fast

Successful services needs to grow/scale
Keeping up with growth is a big IT challenge
Cannot provision hardware fast enough



Unpredictable Bursting

Unexpected/unplanned peak in demand
Sudden spike impacts performance
Cannot over provision for extreme cases

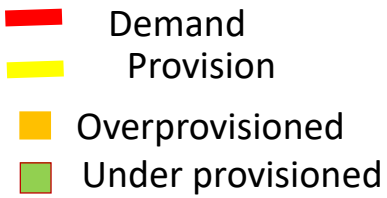


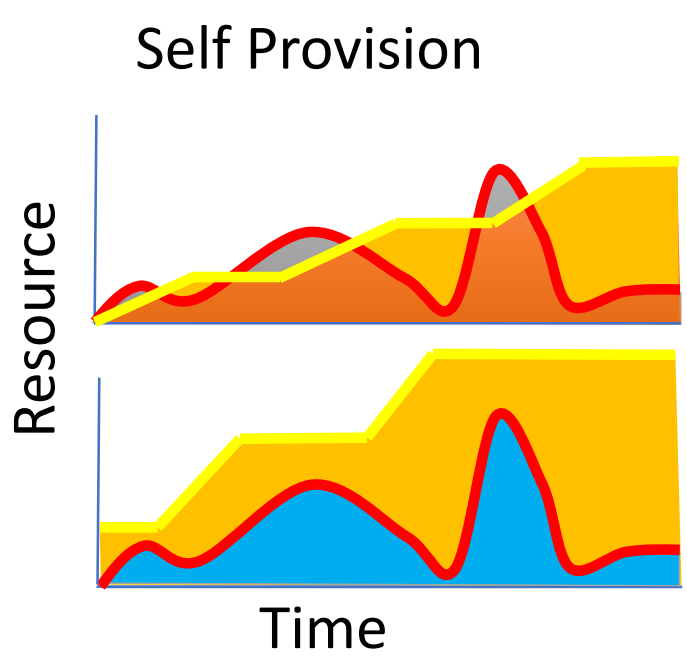
Predictable Bursting

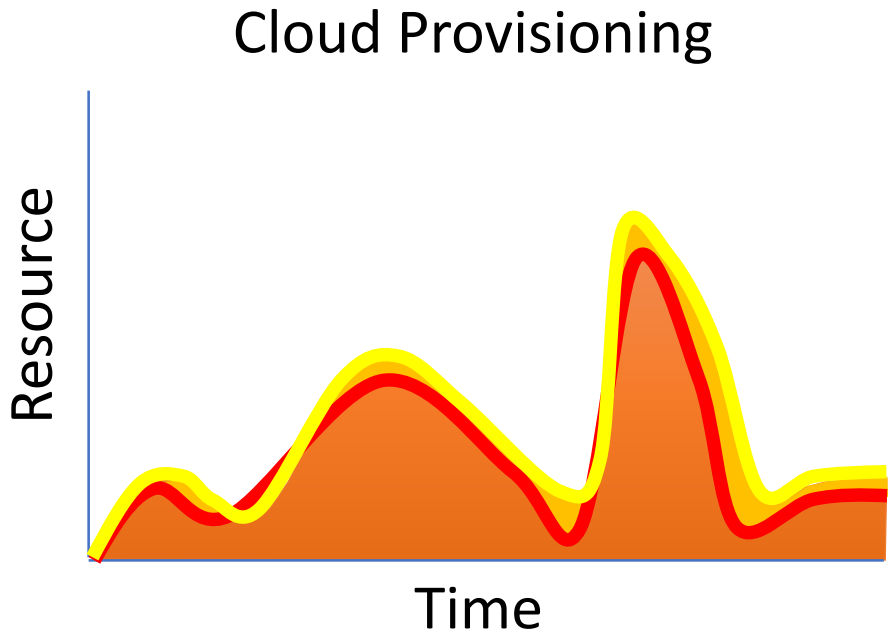
Services with micro seasonality trends
Peaks due to periodic increased demand
IT complexity and wasted capacity

Elasticity-Provision Vs Workload

- Cloud provides on-demand, scale out and in, compute, storage and network resources
- Provisioning Benefit: Reduced Costs and Improved User Experience

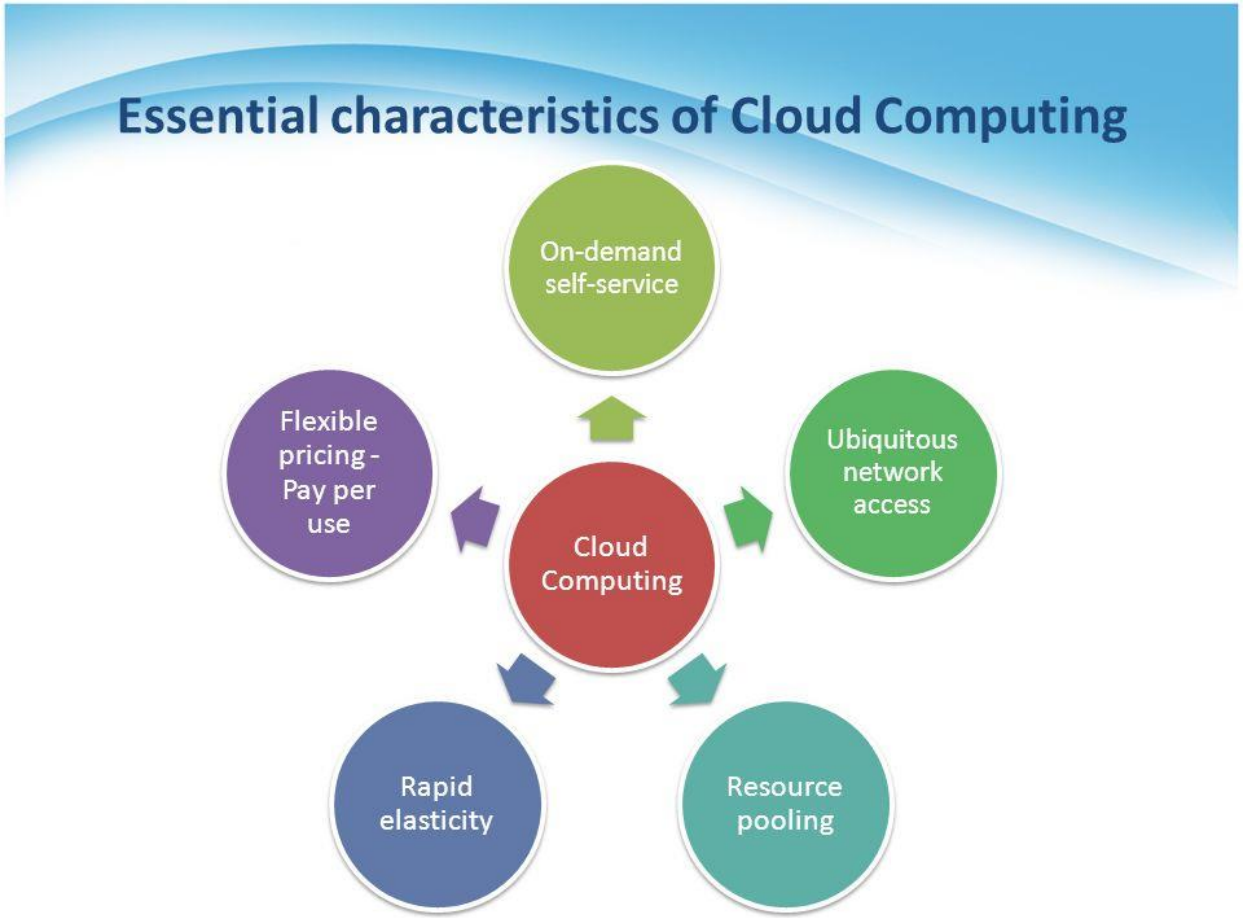


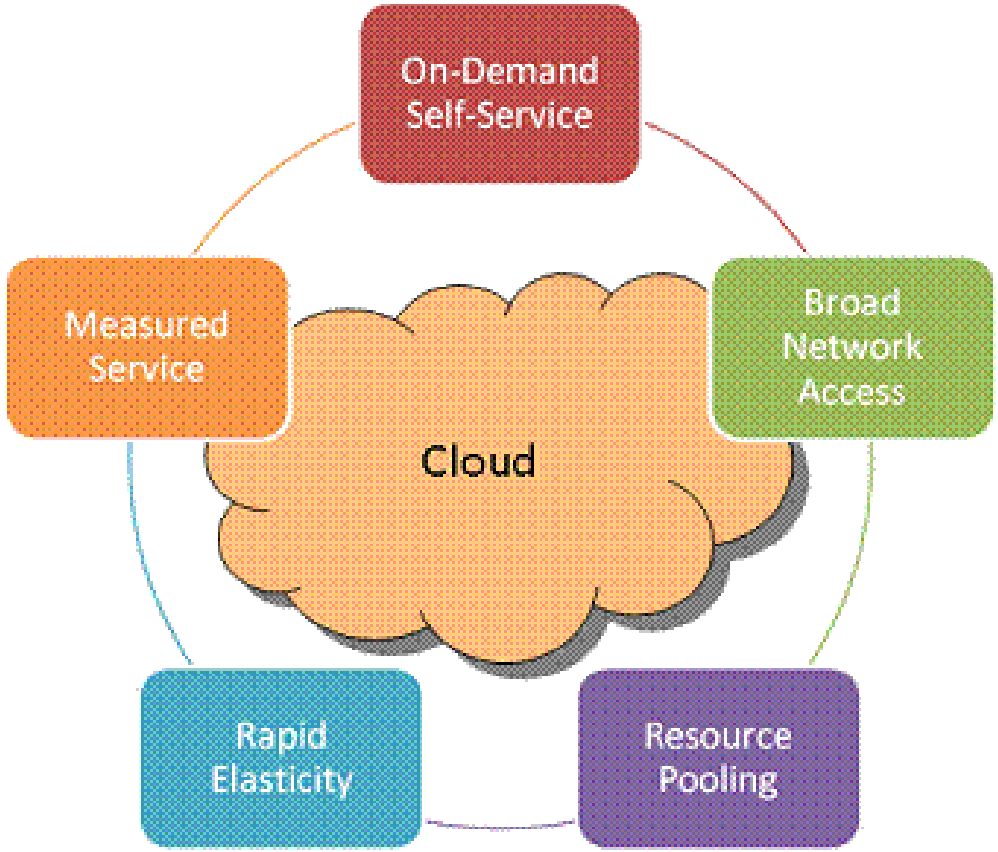




Essential Characteristics:

Cloud have five types of essential characteristics: On-demand self-service, Broad network access, Resource pooling, Rapid elasticity, Measured service.





Five Essential Characteristics:

On-demand self-service: A consumer can unilaterally provision computing capabilities, such as server time and network storage, as needed automatically without requiring human interaction with each service provider.

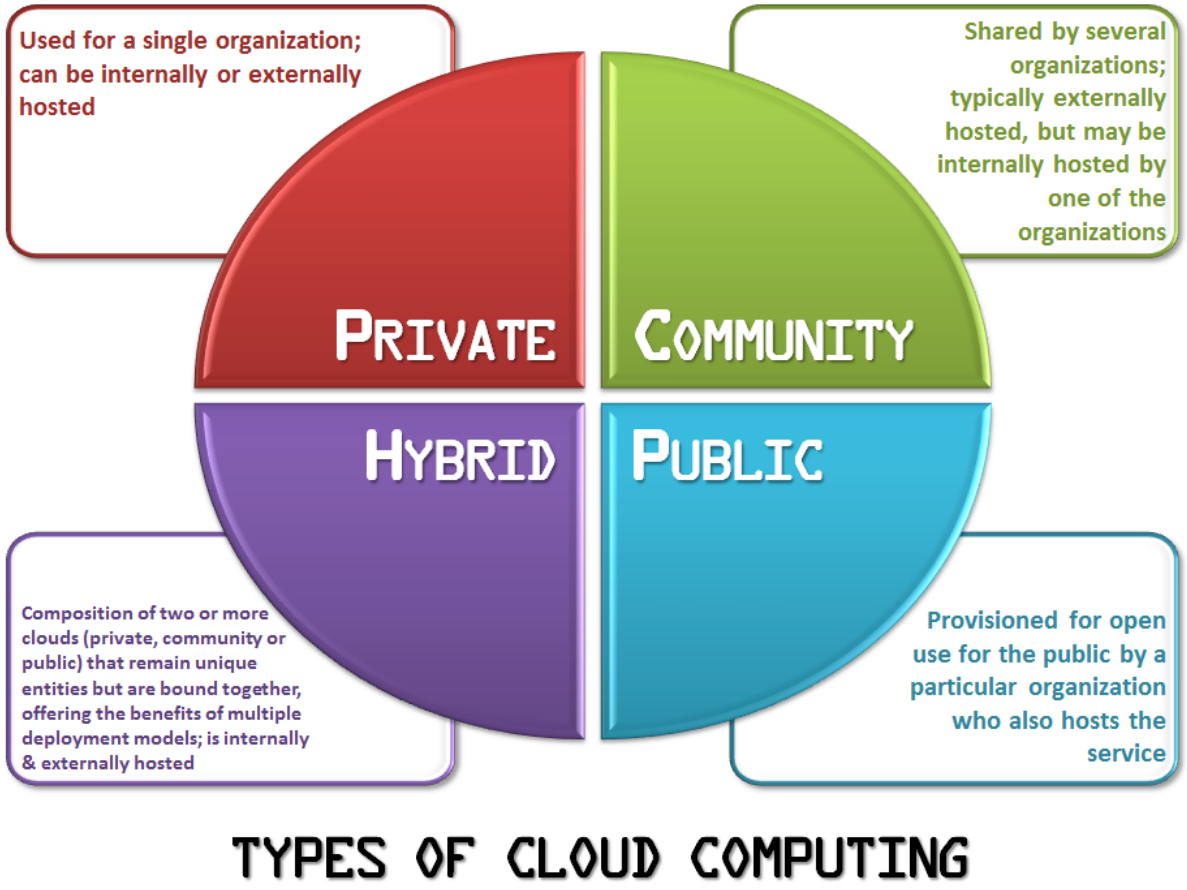
Broad network access: Your team can access business management solutions using their smartphones, tablets, laptops, and office computers. They can use these devices wherever they are located with a simple online access point.
This mobility is particularly attractive for businesses so that during business hours or on off-times, employees can stay on top of projects, contracts, and customers whether they are on the road or in the office.
Broad network access includes private clouds that operate within a company's firewall, public clouds, or a hybrid deployment.
Resource pooling: The cloud enables your employees to enter and use data within the business management software hosted in the cloud at the same time, from any location, and at any time. This is an attractive feature for multiple business offices and field service or sales teams that are usually outside the office.

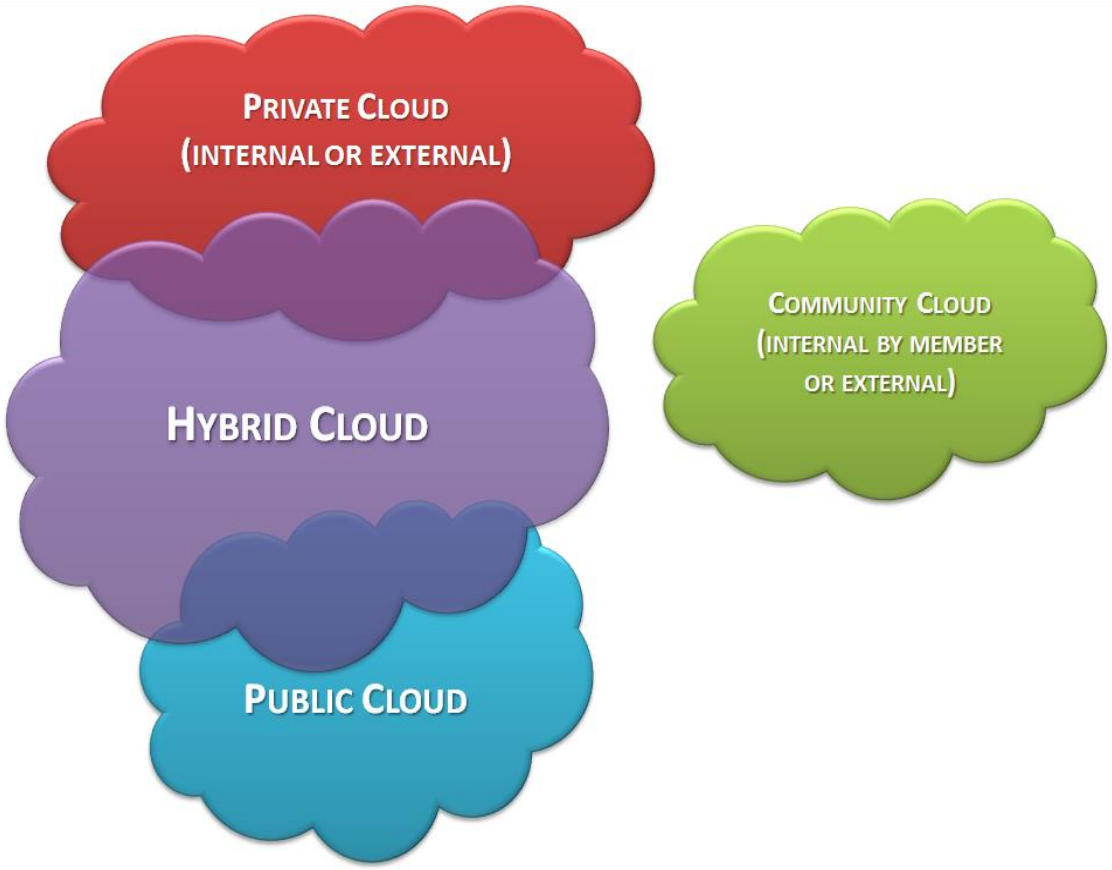
Rapid elasticity: If anything, the cloud is flexible and scalable to suit your immediate business needs. You can quickly and easily add or remove users, software features, and other resources.

Measured service: Going back to the affordable nature of the cloud, you only pay for what you use. You and your cloud provider can measure storage levels, processing, bandwidth, and the number of user accounts and you are billed appropriately. The amount of resources that you may use can be monitored and controlled from both your side and your cloud provider's side which provides transparency.

Types of Cloud : Deployment Models

Deployment models define the type of access to the cloud, i.e., how the cloud is located? Cloud can have any of the four types of access: Public, Private, Hybrid and Community.





Four Deployment Models:

Private cloud: The cloud infrastructure is provisioned for exclusive use by a single organization comprising multiple consumers (e.g., business units). It may be owned, managed, and operated by the organization, a third party, or some combination of them, and it may exist on or off premises.

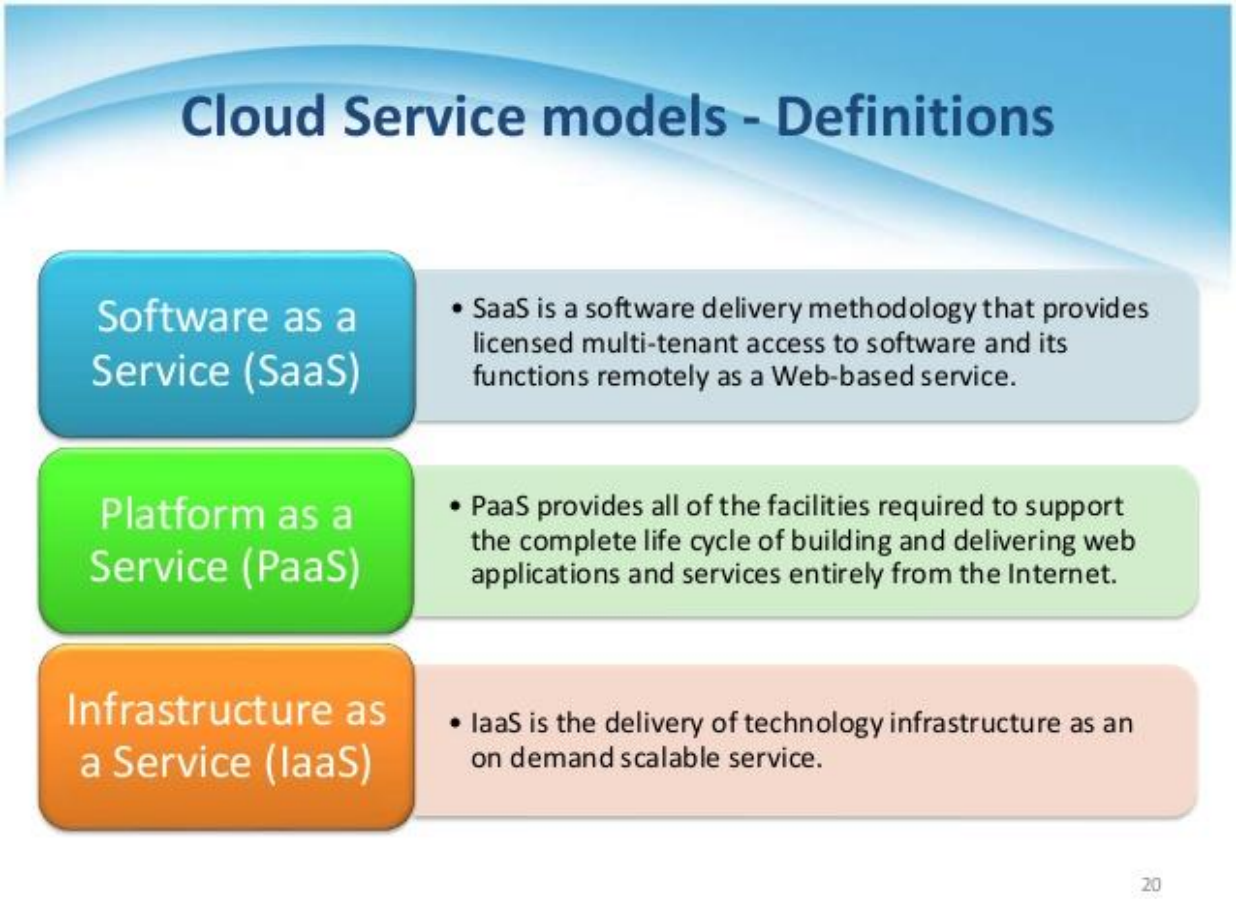
Community cloud: The cloud infrastructure is provisioned for exclusive use by a specific community of consumers from organizations that have shared concerns (e.g., mission, security requirements, policy, and compliance considerations). It may be owned, managed, and operated by one or more of the organizations in the community, a third party, or some combination of them, and it may exist on or off premises.

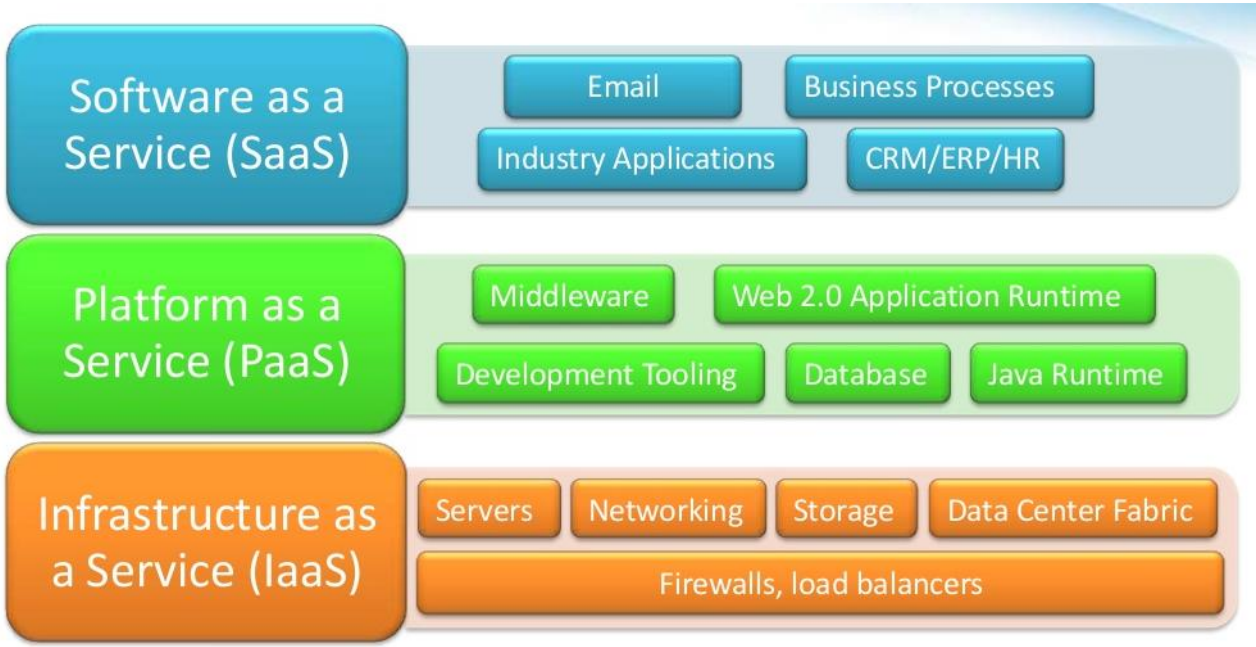
Public cloud: The cloud infrastructure is provisioned for open use by the general public. It may be owned, managed, and operated by a business, academic, or government organization, or some combination of them. It exists on the premises of the cloud provider.

Hybrid cloud: The cloud infrastructure is a composition of two or more distinct cloud infrastructures (private, community, or public) that remain unique entities, but are bound together by standardized or proprietary technology that enables data and application portability (e.g., cloud bursting for load balancing between clouds)

Service Models

Cloud have three types of Service Models: Software as a Service (SaaS), Platform as a Service (PaaS), Infrastructure as a Service (IaaS)





Three Service Models:

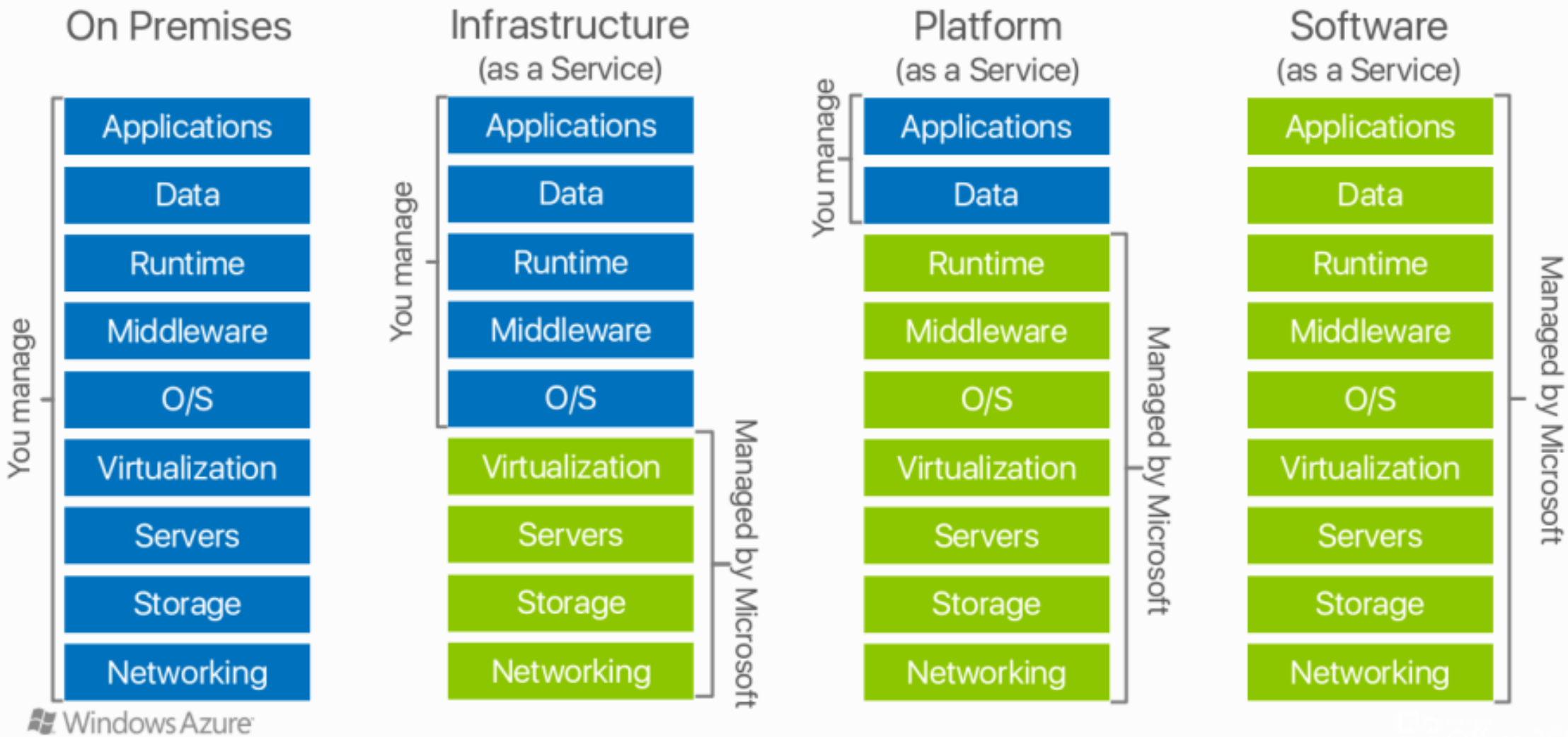
SaaS (Software as a Service): SaaS or software as a service is a software distribution model in which applications are hosted by a vendor or service provider and made available to customers over a network (internet). SaaS is becoming an increasingly prevalent delivery model as underlying technologies that supports Service Oriented Architecture (SOA) or Web Services. Through internet this service is available to users anywhere in the world. Traditionally, software application needed to be purchased upfront &then installed it onto your computer. SaaS users on the other hand, instead of purchasing the software subscribes to it, usually on monthly basis via internet.
Anyone who needs an access to a particular piece of software can be subscribe as a user, whether it is one or two people or every thousands of employees in a corporation. SaaS is compatible with all internet enabled devices.
Many important tasks like accounting, sales, invoicing and planning all can be performed using SaaS.

PaaS (Platform as a Service): Platform as a service, is referred as PaaS, it provides a platform and environment to allow developers to build applications and services. This service is hosted in the cloud and accessed by the users via internet.
To understand in a simple terms, let compare this with painting a picture, where you are provided with paint colors, different paint brushes and paper by your school teacher and you just have to draw a beautiful picture using those tools.
PaaS services are constantly updated & new features added. Software developers, web developers and business can benefit from PaaS. It provides platform to support application development. It includes software support and management services, storage, networking, deploying, testing, collaborating, hosting and maintaining applications.

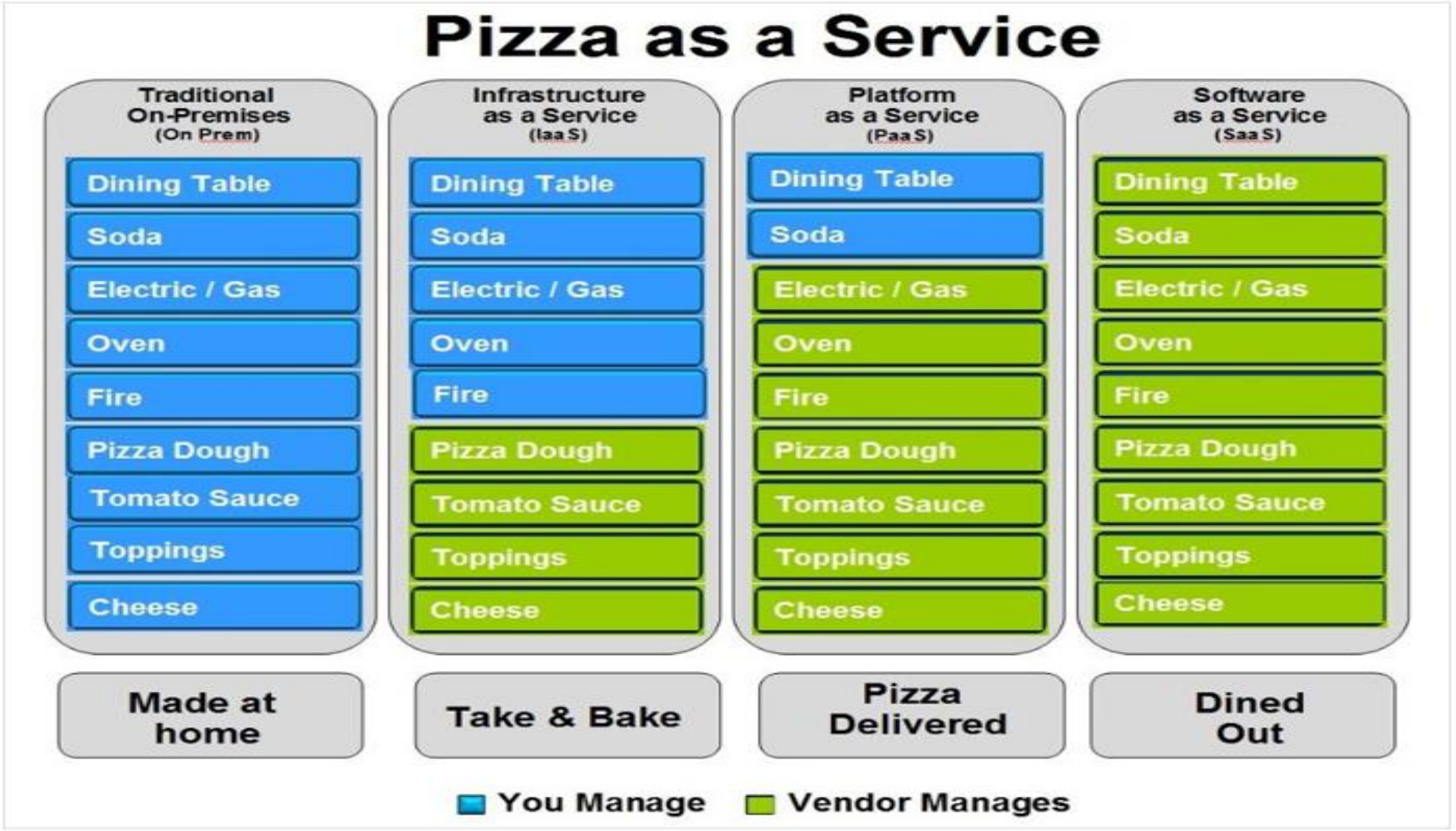
IaaS (Infrastructure as a Service): IaaS (Infrastructure As A Service) is one of the fundamental service model of cloud computing alongside PaaS(Platform as a Service). It provides access to computing resources in a virtualized environment "the cloud" on internet. It provides computing infrastructure like virtual server space, network connections, bandwidth, load balancers and IP addresses.

The pool of hardware resource is extracted from multiple servers and networks usually distributed across numerous data centers. This provides redundancy and reliability to IaaS. IaaS(Infrastructure as a service) is a complete package for computing. For small scale businesses who are looking for cutting cost on IT infrastructure, IaaS is one of the solutions. Annually a lot of money is spent in maintenance and buying new components like hard-drives, network connections, external storage device etc. which a business owner could have saved for other expenses by using IaaS.

Cloud Models



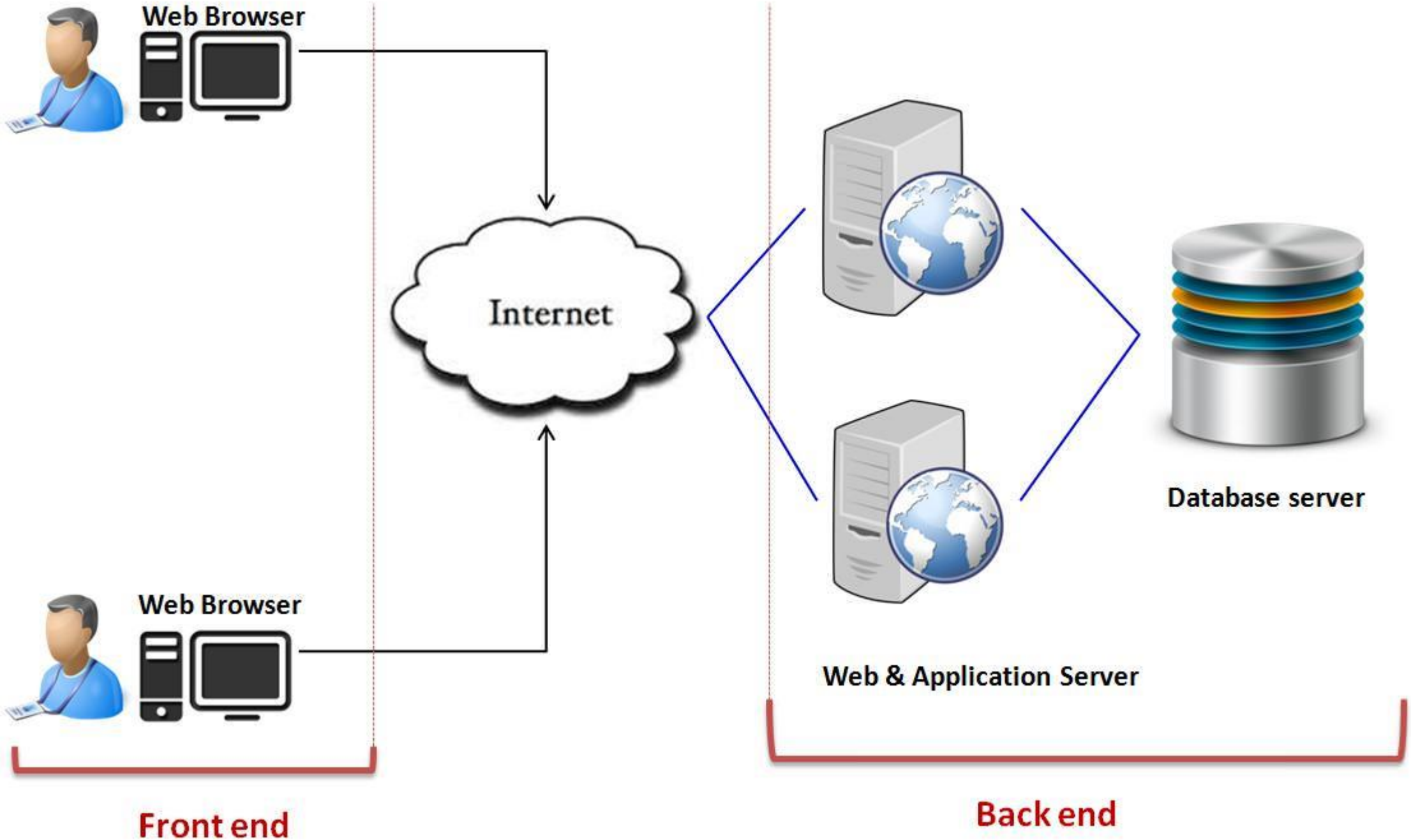
Understanding the Cloud Computing Stack (IaaS, PaaS, SaaS)



What is Cloud Computing Architecture?

Let's have a look into Cloud Computing and see what Cloud Computing is made of. Cloud computing comprises of two components front end and back end. Front end consist client part of cloud computing system. It comprise of interfaces and applications that are required to access the cloud computing platform.
While back end refers to the cloud itself, it comprises of the resources that are required for cloud computing services. It consists of virtual machines, servers, data storage, security mechanism etc. It is under providers control.

Cloud computing distributes the file system that spreads over multiple hard disks and machines. Data is never stored in one place only and in case one unit fails the other will take over automatically. The user disk space is allocated on the distributed file system, while another important component is algorithm for resource allocation. Cloud computing is a strong distributed environment and it heavily depends upon strong algorithm.


Web Browser
Internet
Web Browser
Database server
Web & Application Server
Front end
Back end

Virtualization and Cloud Computing

The main enabling technology for Cloud Computing is Virtualization. Virtualization is a partitioning of single physical server into multiple logical servers. Once the physical server is divided, each logical server behaves like a physical server and can run an operating system and applications independently. Many popular companies' like VMware and Microsoft provide virtualization services, where instead of using your personal PC for storage and computation, you use their virtual server. They are fast, cost-effective and less time consuming.
For software developers and testers virtualization comes very handy, as it allows developer to write code that runs in many different environments and more importantly to test that code.

Virtualization is mainly used for three main purposes 1) Network Virtualization 2) Server Virtualization 3) Storage Virtualization

Network Virtualization: It is a method of combining the available resources in a network by splitting up the available bandwidth into channels, each of which is independent from the others and each channel is independent of others and can be assigned to a specific server or device in real time.

Storage Virtualization: It is the pooling of physical storage from multiple network storage devices into what appears to be a single storage device that is managed from a central console. Storage virtualization is commonly used in storage area networks (SANs).

Server Virtualization: Server virtualization is the masking of server resources like processors, RAM, operating system etc., from server users. The intention of server virtualization is to increase the resource sharing and reduce the burden and complexity of computation from users.

Virtualization is the key to unlock the Cloud system, what makes virtualization so important for the cloud is that it decouples the software from the hardware. For example, PC's can use virtual memory to borrow extra memory from the hard disk. Usually hard disk has a lot more space than memory. Although virtual disks are slower than real memory, if managed properly the substitution works perfectly. Likewise, there is software which can imitate an entire computer, which means 1 computer can perform the functions equals to 20 computers.

Module 1 - Lesson 2 : Overview of Azure

Microsoft Azure

- A cloud computing platform by Microsoft
- Was announced by Microsoft back in 2008
- Used to be known as Windows Azure
- Initially offered PaaS services
- Offers over 500 cloud services and growing

Microsoft Azure Datacenters and Regions

- Azure platform is supported by network of Microsoft managed Datacenters
- Azure Datacenters are located in about 50 geographic regions (https://azure.microsoft.com/enca/regions/)
- Each region is geographic group of one or more datacenters
- Is it important to know where the Azure regions are. It is the first consideration you need to make to deploy any Azure cloud service.
- Azure region pairs – Each Azure region is paired with another region. They are connected directly. (https://docs.microsoft.com/en-us/azure/best-practices-availability-paired-regions)
- Microsoft Datacenter Tour: https://cloud-platform-assets.azurewebsites.net/datacenter/

Azure Accounts and Subscriptions

- Azure is primarily organized by Accounts and Subscriptions
- Each Account can have multiple subscriptions associated with it
- The primary account delegates privileges at subscription level
- Account administrator is the user created with the account
- Has access to the Azure Account Center (https://account.azure.com/)
- Can create the billing method
- Can create/cancel subscriptions
- Can create/change the subscription level administrator
- To consume Azure services you need subscriptions
- Subscriptions can be personal or organizational
- Subscriptions have isolation of administration & billing
- Separate privileges can be assigned to separate subscriptions
- Subscriptions have quotas and limits (https://docs.microsoft.com/en-us/azure/azuresubscription-service-limits)
- Subscriptions have 2 roles + RBAC
  Service Administrator
  Co-Administrator

Azure billing

- Azure is a pay-per-use utilization model
- PAYG – Pay As You Go Model – most of you will use this (https://azure.microsoft.com/enus/offers/ms-azr-0003p/)
- Enterprise Agreement
- Azure Compute pre-purchase plan
- Azure Hybrid benefit
- Microsoft reseller

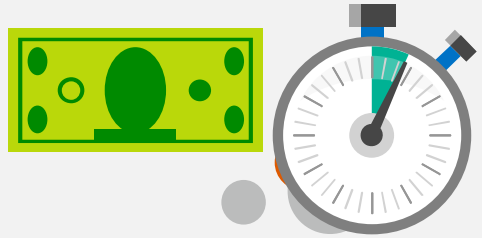
Through benefits:

- MSDN Subscription for development use
- Partners get monthly credits

Azure billing and support options

The most common Azure billing options include:

Pay-As-You-Go



Buy from a Microsoft Reseller

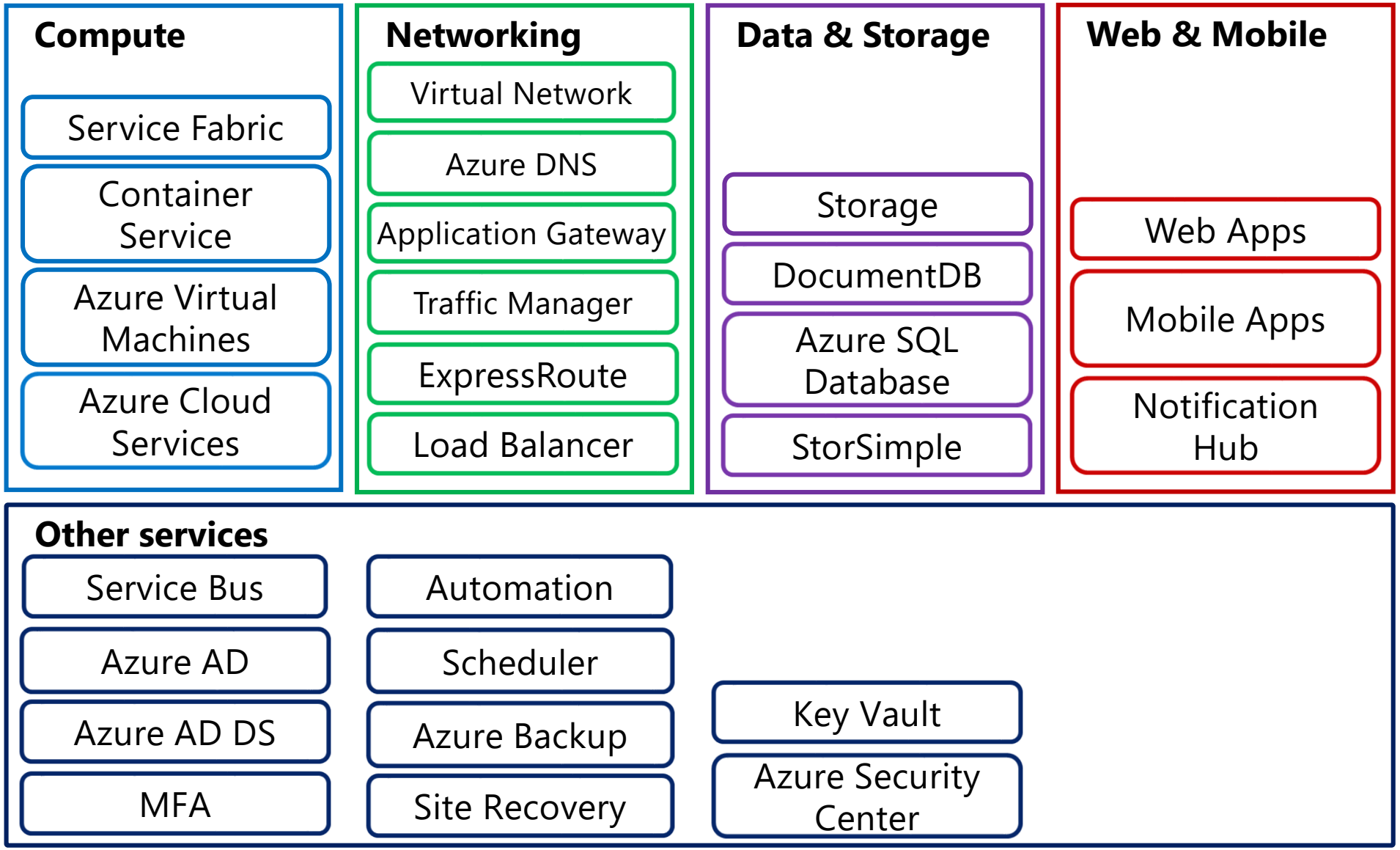
Enterprise agreements

Azure services

- Compute
- Networking
- Storage
- Web+Mobile
- Containers
- Databases
- Analytics
- AI+Machine Learning (ML)
- Internet of Things (IoT)
- Enterprise Integrations
- Security+Identity
- Developer tools
- Management tools

Exhaustive details of all Azure services: https://azure.microsoft.com/en-us/services

Overview of Azure services



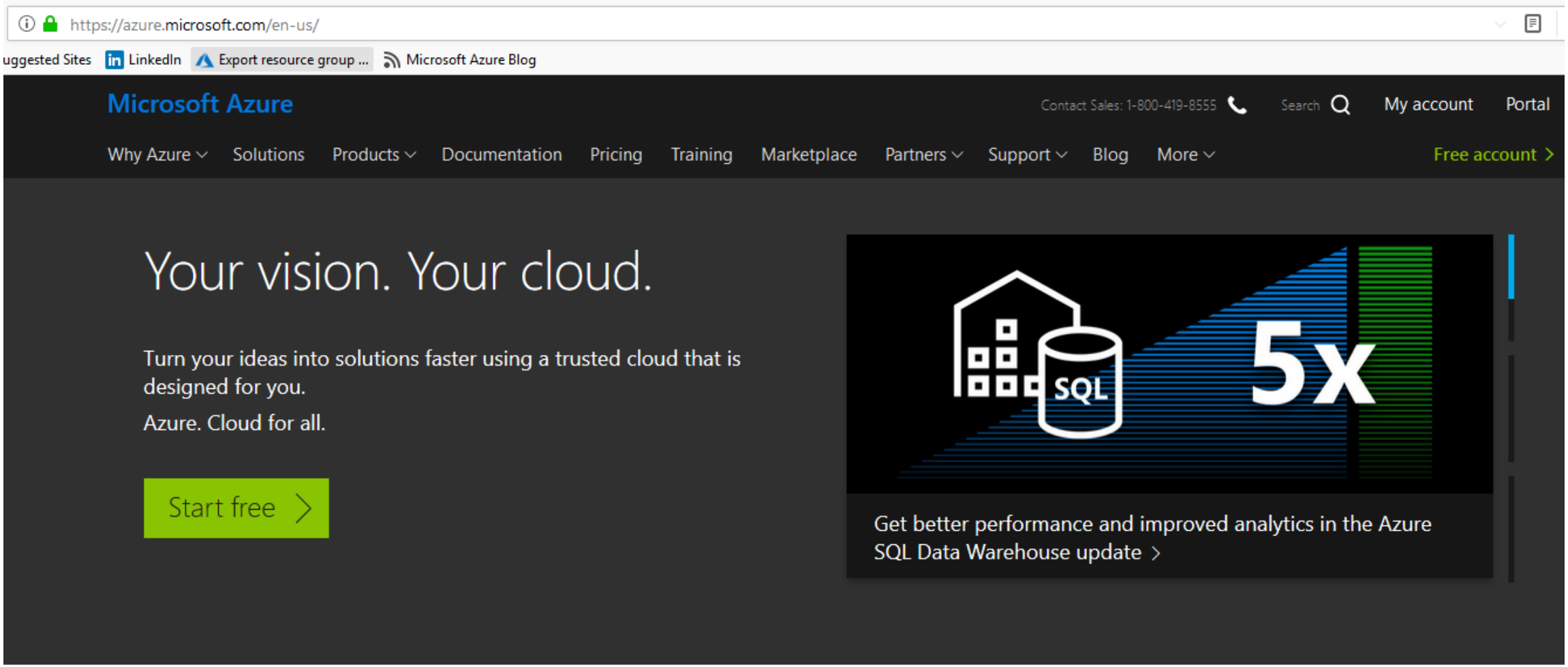
Azure Management Tools

- Azure portal (https://portal.azure.com/)
- Azure classic portal (https://manage.windowsazure.com/)
- Azure PowerShell
- Azure CLI
- Azure Cloud Shell – CLI
- Azure Cloud Shell – PowerShell)
- Visual Studio – Azure SDK

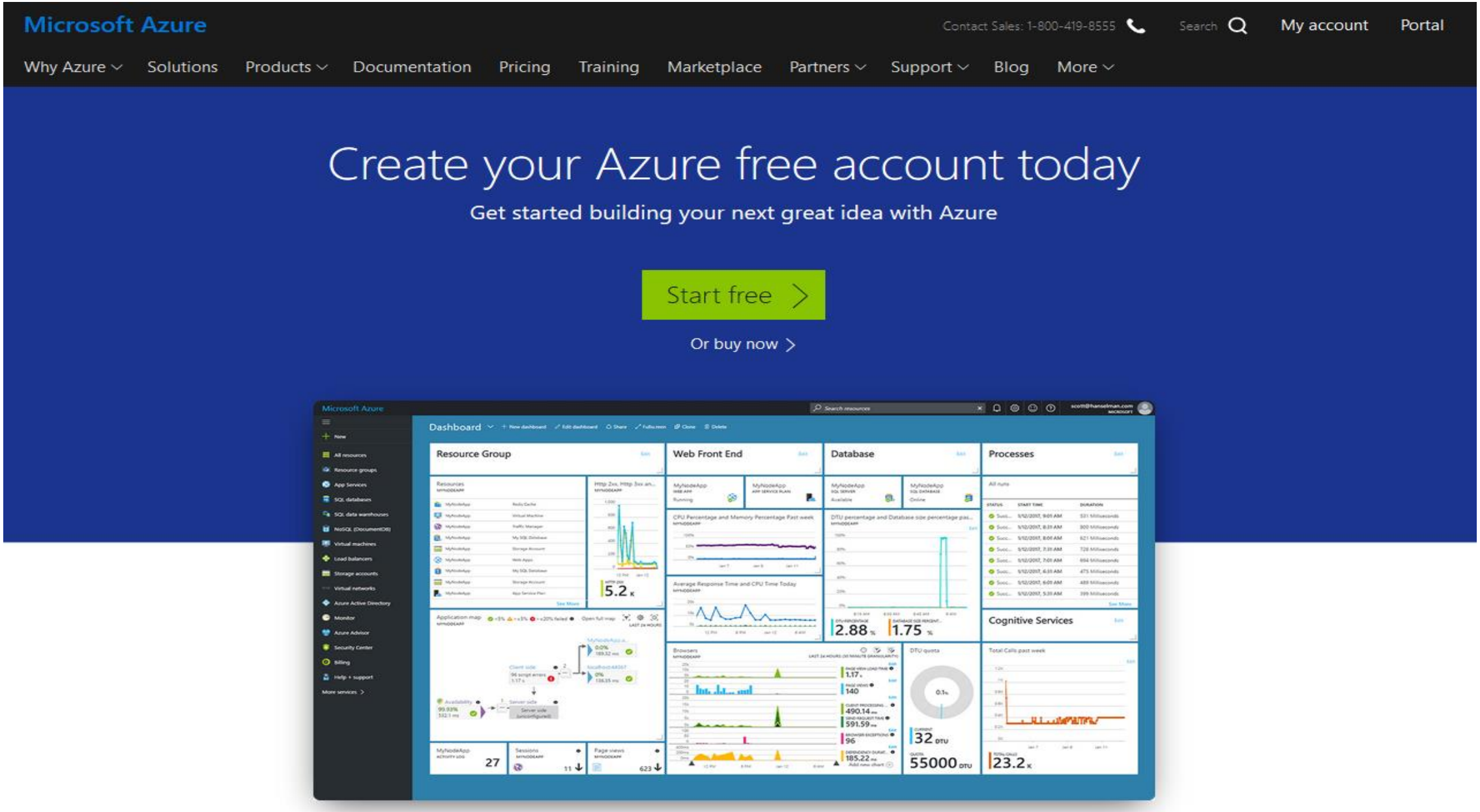
How to Create up a New Microsoft Azure Account

Microsoft Azure offers a free 30-day trial period to all new account holders.

Go to https://www.azure.com and click the green "Start free" button.



Next, click another “Start free” button.



What do I get?

With your Azure free account, you get all of this—and you won't be charged until you choose to upgrade.

$200 credit to explore services for 30 days + **12 months** of popular free services + **Always free** 25+ services

What can I do with my free account?

Here are just a few ideas of all you can do with Azure

Test and deploy enterprise apps

Use Azure Virtual Machines, Managed Disks, and SQL databases while providing high availability and network performance with Load Balancer.

Create custom mobile experiences

Build based on your customers' interests and behavior using App Service and Azure Cosmos DB, Xamarin, HockeyApp, and Traffic Manager.

Gain insights from your data

Make better decisions and create finer experiences using Machine Learning, Data Lake Analytics, and HDInsight.

Which products are free?

Get these popular products—free each month for a year

Linux Virtual Machines

COMPUTE

750 Hours

B1S VM

Create Linux virtual machines with on-demand capacity in seconds.

Windows Virtual Machines

COMPUTE

750 Hours

B1S VM

Create Windows virtual machines with on-demand capacity in seconds.

Managed Disks

STORAGE

64 GB X 2

2 P6 SDDs

Get premium, secured disk storage for Azure Virtual Machines with simplified management.

Blob Storage

STORAGE

5 GB

LRS hot block

Use massively scalable object storage for any type of unstructured data.

File Storage

STORAGE

5 GB

LRS File Storage

Migrate to simple, distributed, cross-platform file storage without changing code.

SQL Database

DATABASES

250 GB

Create an Azure SQL Database that delivers intelligence built-in.

Cosmos DB

DATABASES

5 GB

400 requests units

Build and scale your application with a globally distributed, multi-model database service.

Bandwidth (Data Transfer)

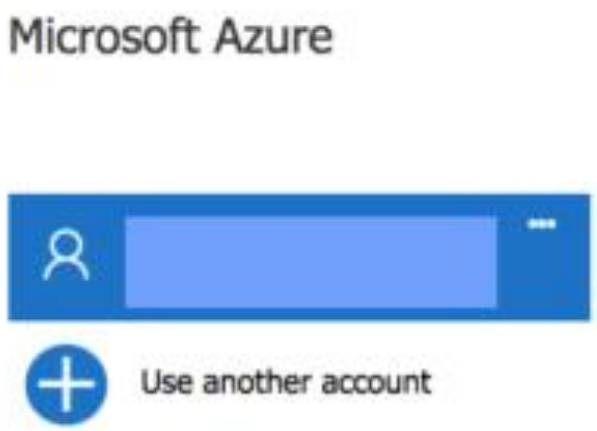
NETWORKING

15 GB

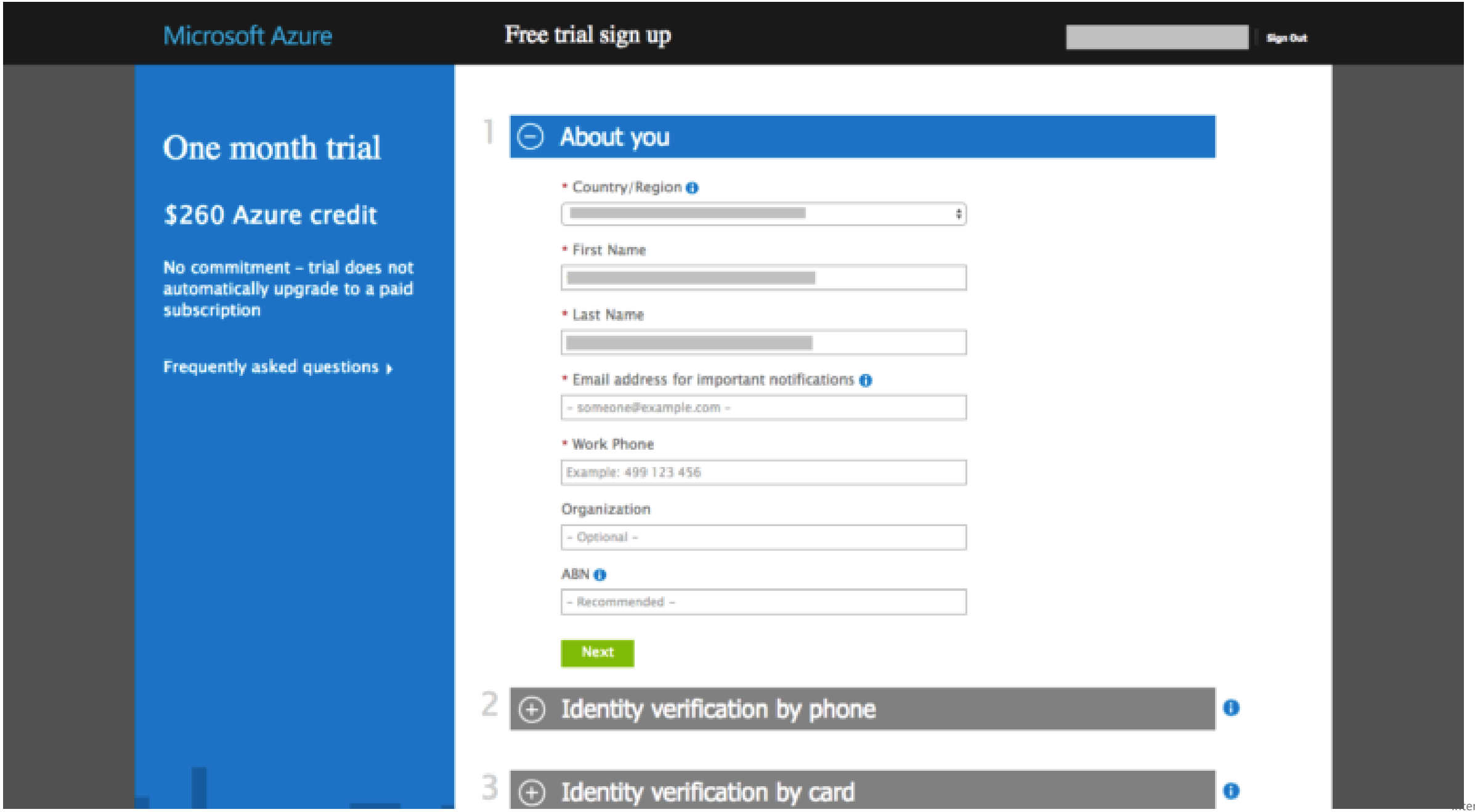
outbound

Transfer data inbound and outbound through our robust network of global data centers.

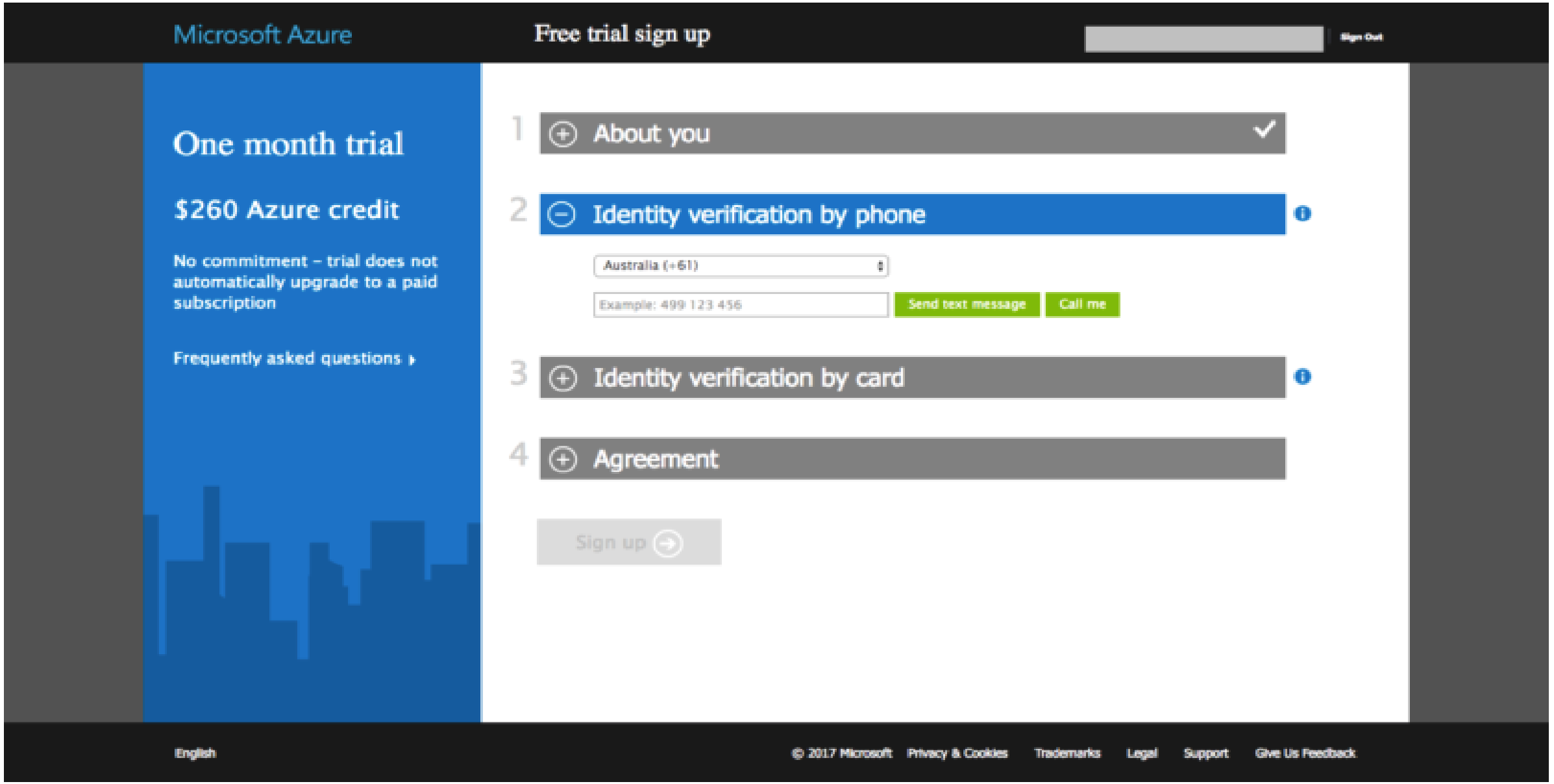
If you already have an account with Microsoft, for example, Office 365, you'll be prompted to log in.



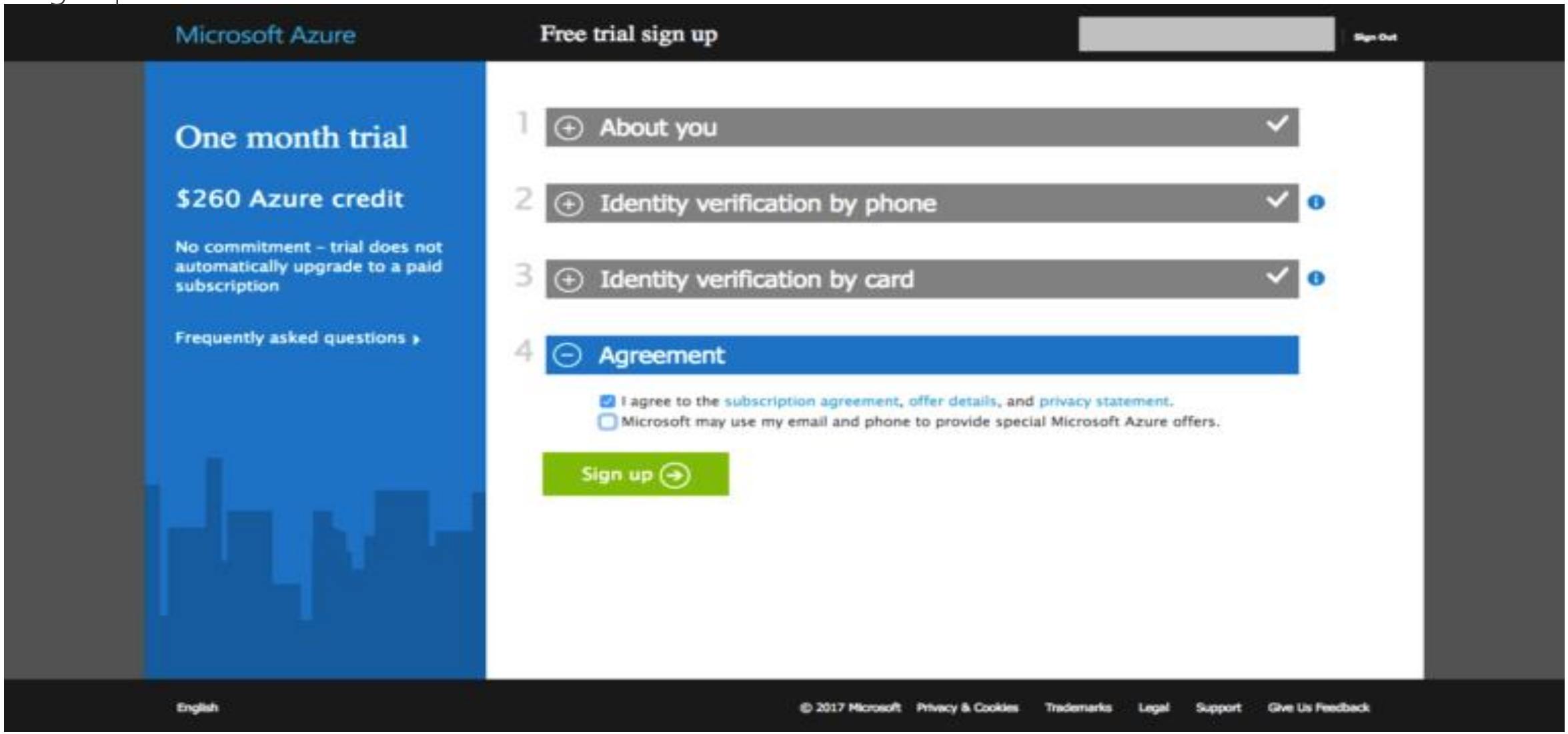
When you log in, some of your details may already be there.



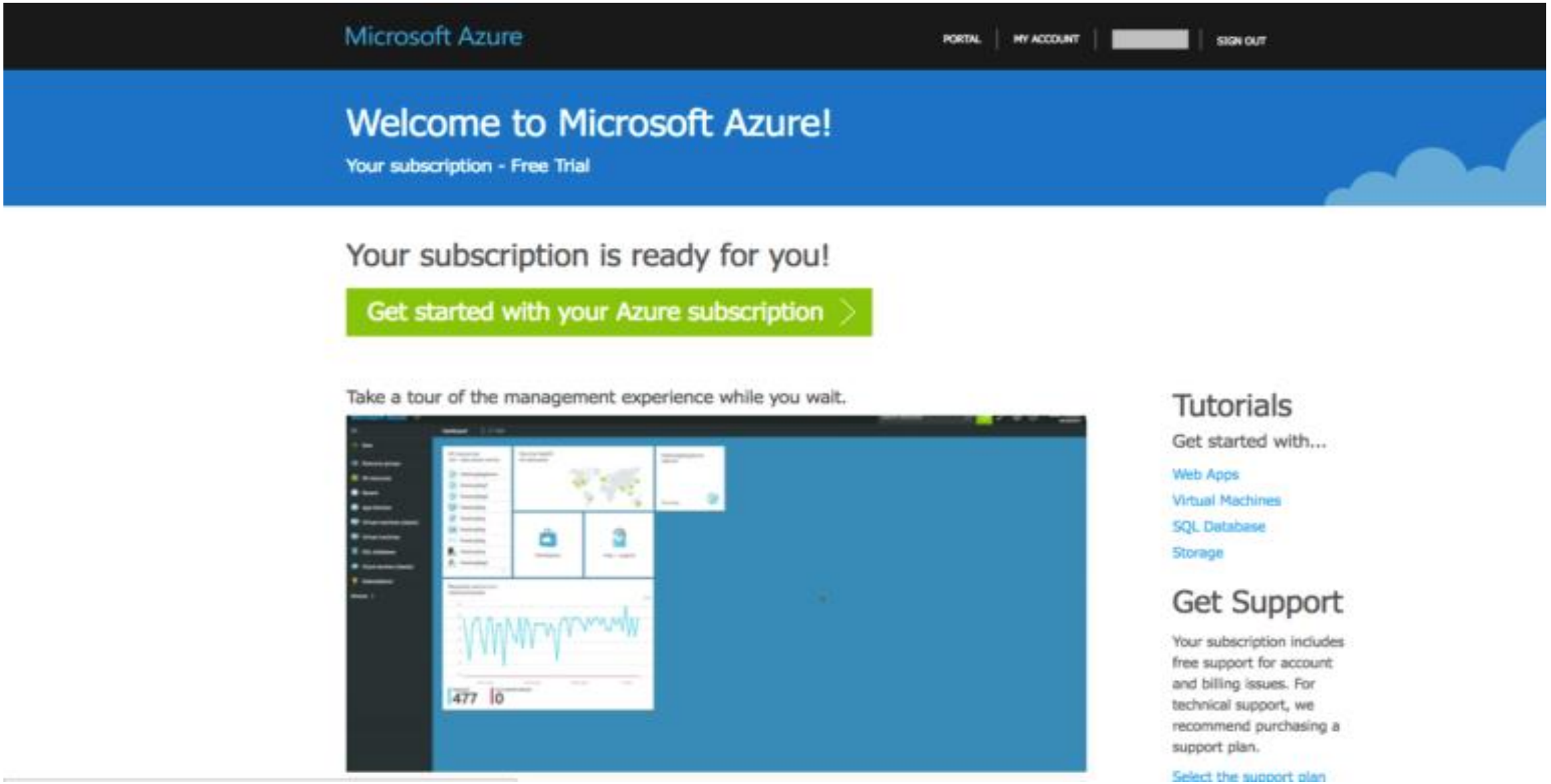
Follow the prompts to verify your account by phone (I used SMS).



You'll also need to supply a valid credit card. Prepaid credit cards won't work — you'll need a "normal" credit or debit card. There is no charge involved with the setting up of a trial account. Microsoft just wants to see your card to verify your identity. There will be, however, a record for a $0 transaction on your bank statement. Next – tick "I agree" and click "Sign Up."



Within a few seconds, your account will be ready. That's it! Your Microsoft Azure account has been created. To continue, click the "My Account" link at the top right corner or go straight to the Microsoft Azure Portal: https://portal.azure.com/



Azure Subscription Overview

Subscription Principles

Subscriptions are…

- Administrative security boundary
- Support RBAC delegation
- A billing unit
- Logical limit of scale
- First container that you create

Considerations

- Subscriptions do not cost anything
- Each subscription has its own admins, although a single account can be an admin in multiple subscriptions
- Are global

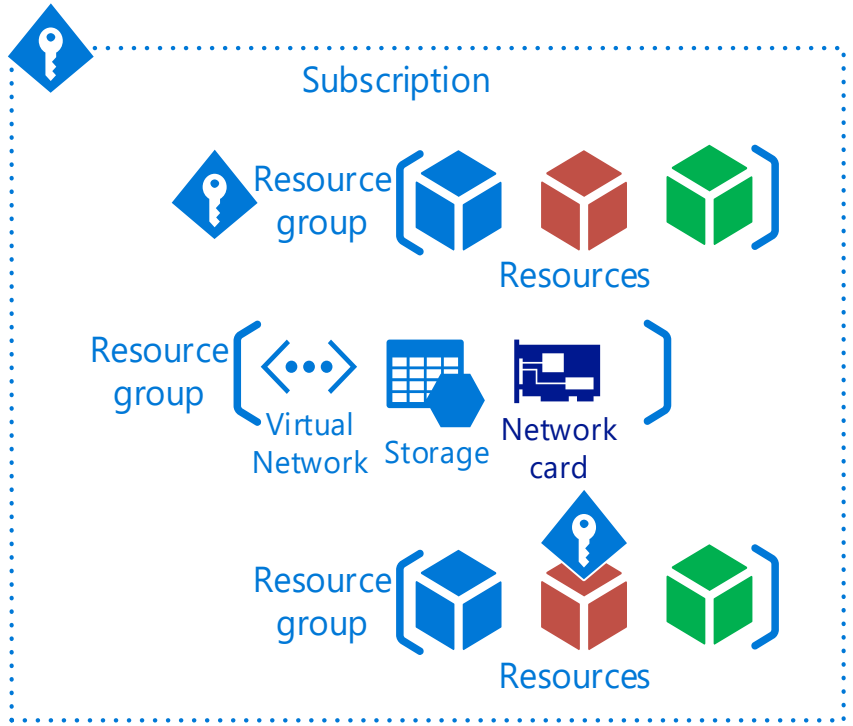
Initially a subscription was the administrative security boundary of Microsoft Azure. With the advent of Azure Resource Management (ARM) environment, a subscription now has two administrative models. Service Management and Azure Resource Management. With ARM the subscription is no longer needed as an administrative boundary. ARM provides a more granular Roles Based Access Control (RBAC) model for assigning administrative rights at the resource level. RBAC is currently being released in stages, 32 new roles have been released and user defined roles is coming in a future release. There will be some complexity during the coexistence of the service management and resource management environments and will need to be carefully considered.

A subscription additionally forms the **billing unit**. Services charges are accrued to the subscription currently, as part of the new Azure Resource Management model it will be possible to roll up costs to a resource group. A standard naming convention for Azure resource object types can be used to manage billing across projects teams, business units, or other desired view.

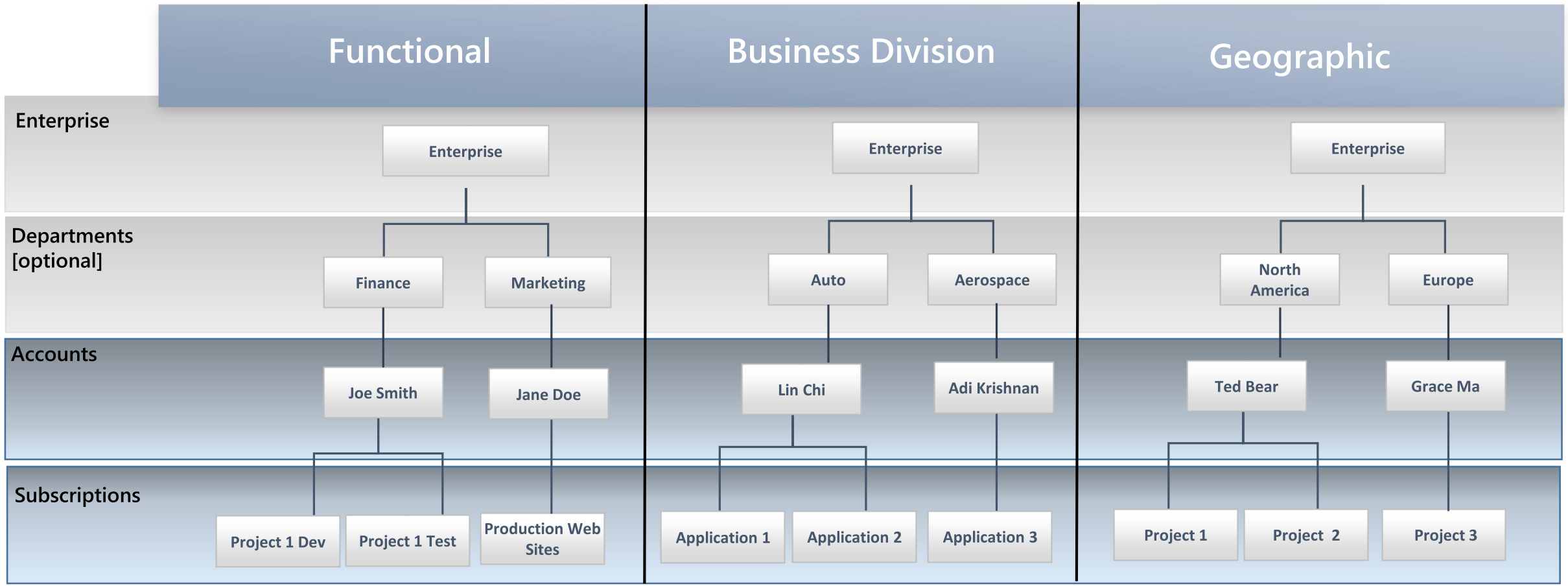
A **logical limit of scale by which resources can be allocated**, these limits include both hard and soft caps of various resource types (like 10,000 compute cores /subscription) and are changing as capacity and capabilities are updated within Azure. Scalability will continue to be a function of subscriptions and therefore is a key element to understand how the Subscription strategy will account for growth as consumption increases.

Containers and Resources

- Subscription is the top level container
- Create Resource groups in the subscription
- Place resources within the resource groups



Azure Governance Layers



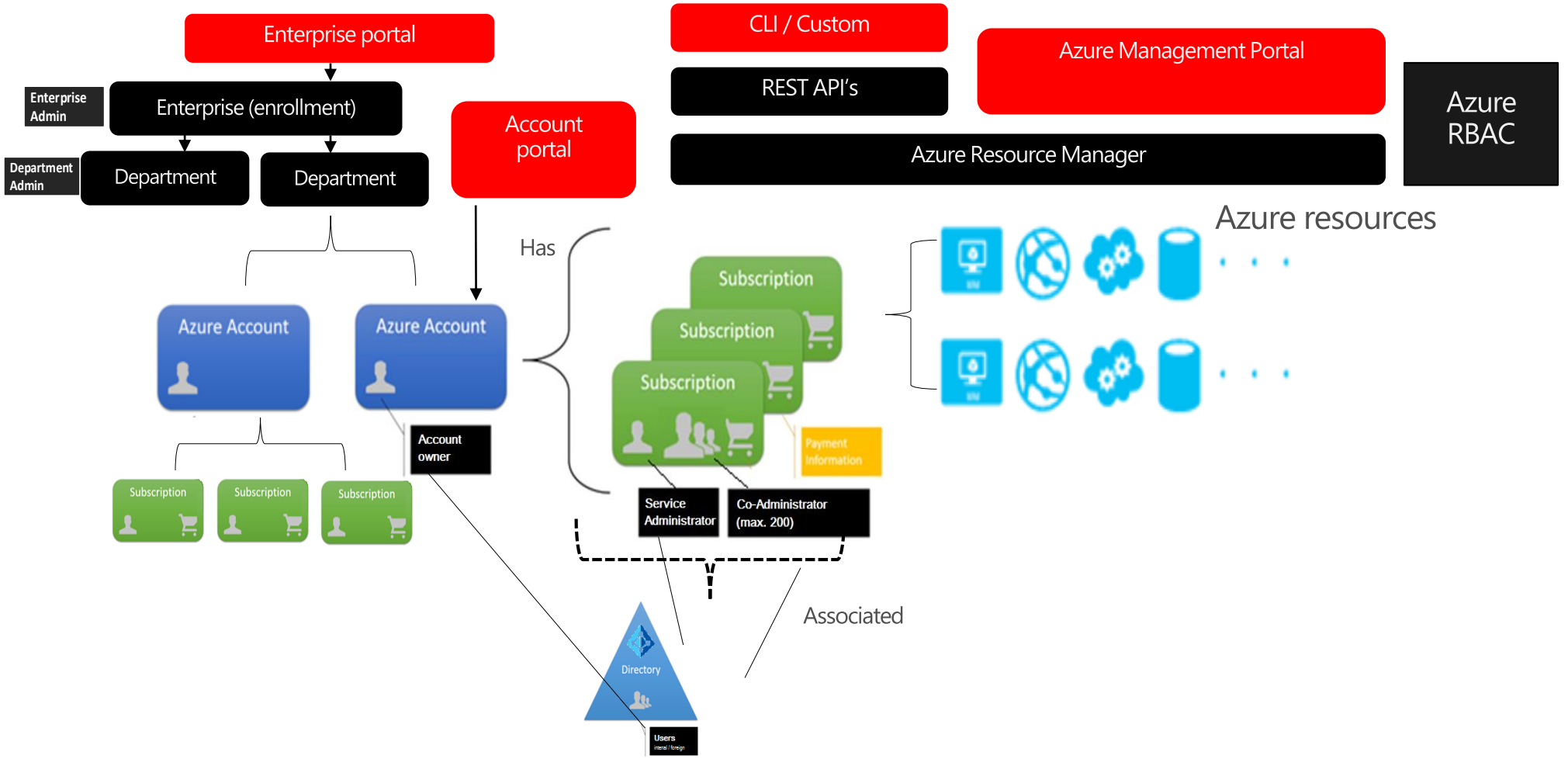
The Azure governance layers, roles, portals etc.. provide the technical means that can be used in different ways. Some customer prefer to use functional differentiation, others business division based or geographical or even a combination.

Management Portals

Portal	Location	Purpose
Enterprise Portal	https://ea.azure.com/	• Manage access • Manage accounts • Manage subscriptions • View price sheet • View usage summary • Manage usage & lifecycle email notifications • Manage Authentication Types
Account Portal	https://account.windowsazure.com	• Edit subscription details • Enroll in or enable Preview features
Management Portal	https://manage.windowsazure.com or https://portal.azure.com	• Provision/de-provision Azure services • Manage co-administrators on subscriptions • Open support tickets for issues within the subscription

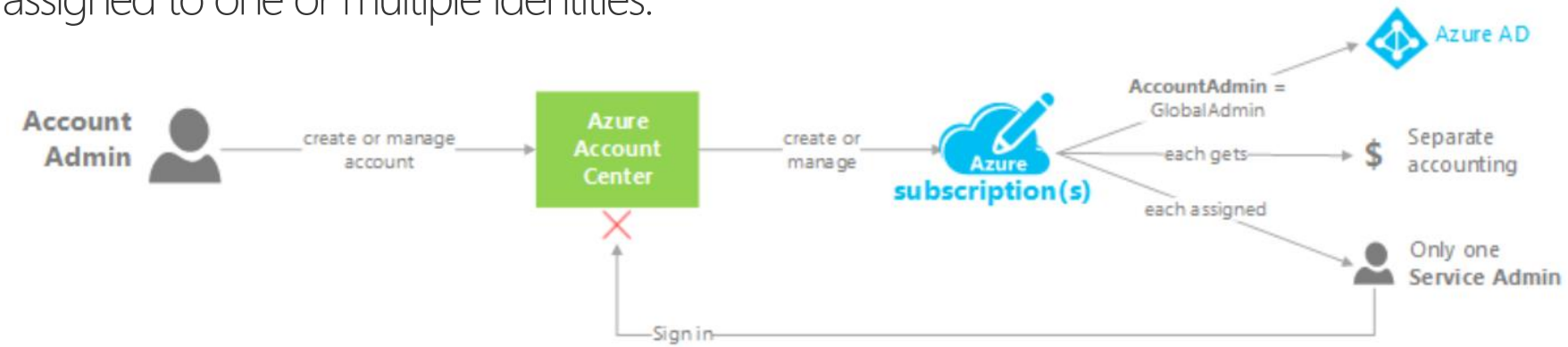
Azure governance structure

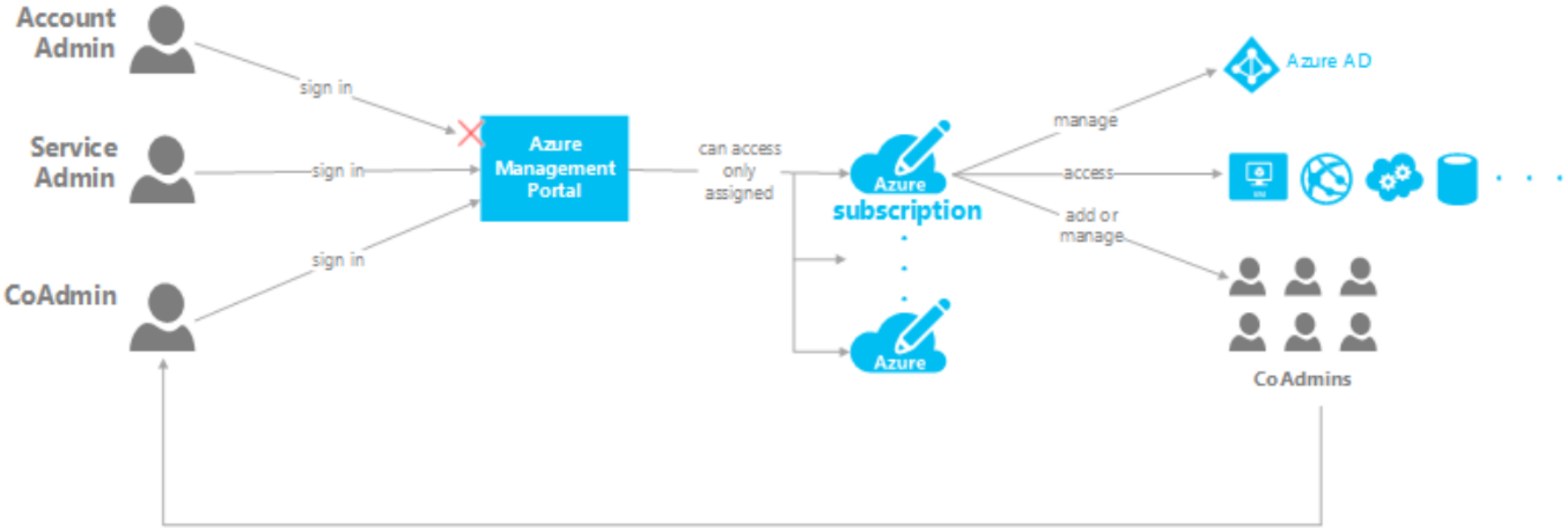
The three portals serve different audiences and needs and provide administrative boundaries. This picture can be used to provide answers on how these portals are related to one another.



Account and Subscription Management

The below diagrams explain what the account admin, service admin and co-admin roles are used for, these roles can be assigned to one or multiple identities.





Subscription Considerations

Management approach

- Single team or distributed
- RBAC

Security requirements

- Data or network security
- Environments - Sandbox, Dev, Test, UAT, Pre-Prod, Prod

Connectivity requirements

- Single point of ingress?
- Multiple regions?

Application requirements

- Data flow
- Compliance

Subscription per Department (Customer Managed)

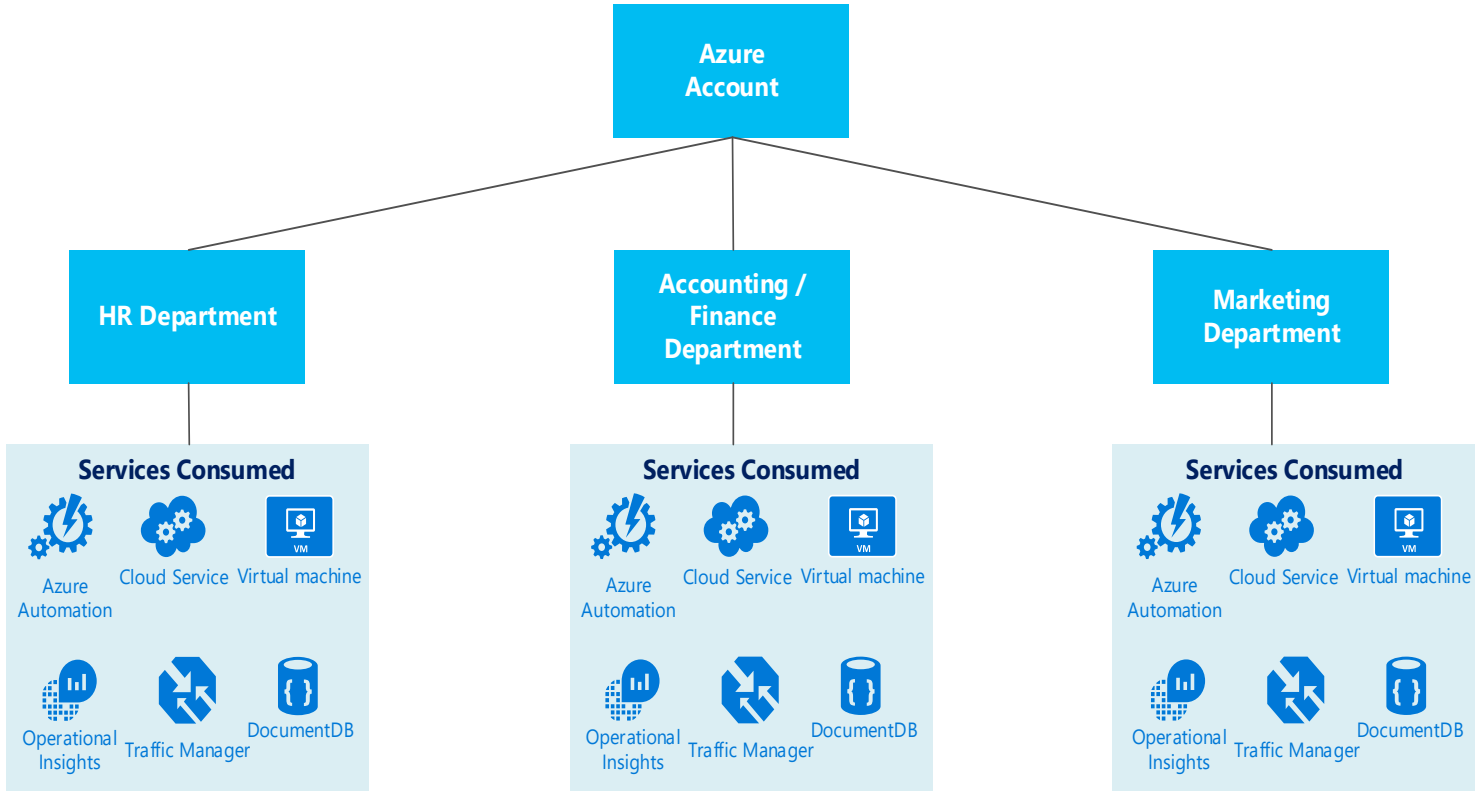
- Each department contains different types of environments (e.g. Prod, Non-Prod). Virtual Networks will wrap the different environments for traffic separation. Subnets will be created within each environment to establish required security isolation zones between applications.

Pros

- Low ExpressRoute Circuit Costs
- Simplified Subscription Management
- No Vnet Subscription Limit

Cons

- Granular RBAC model required
- Subscription Limit Issues in Cores, Storage, NSGs
- Complex Vnet addressing
- Mistake in management will affect all environments



Subscription per Environment (Customer Managed)

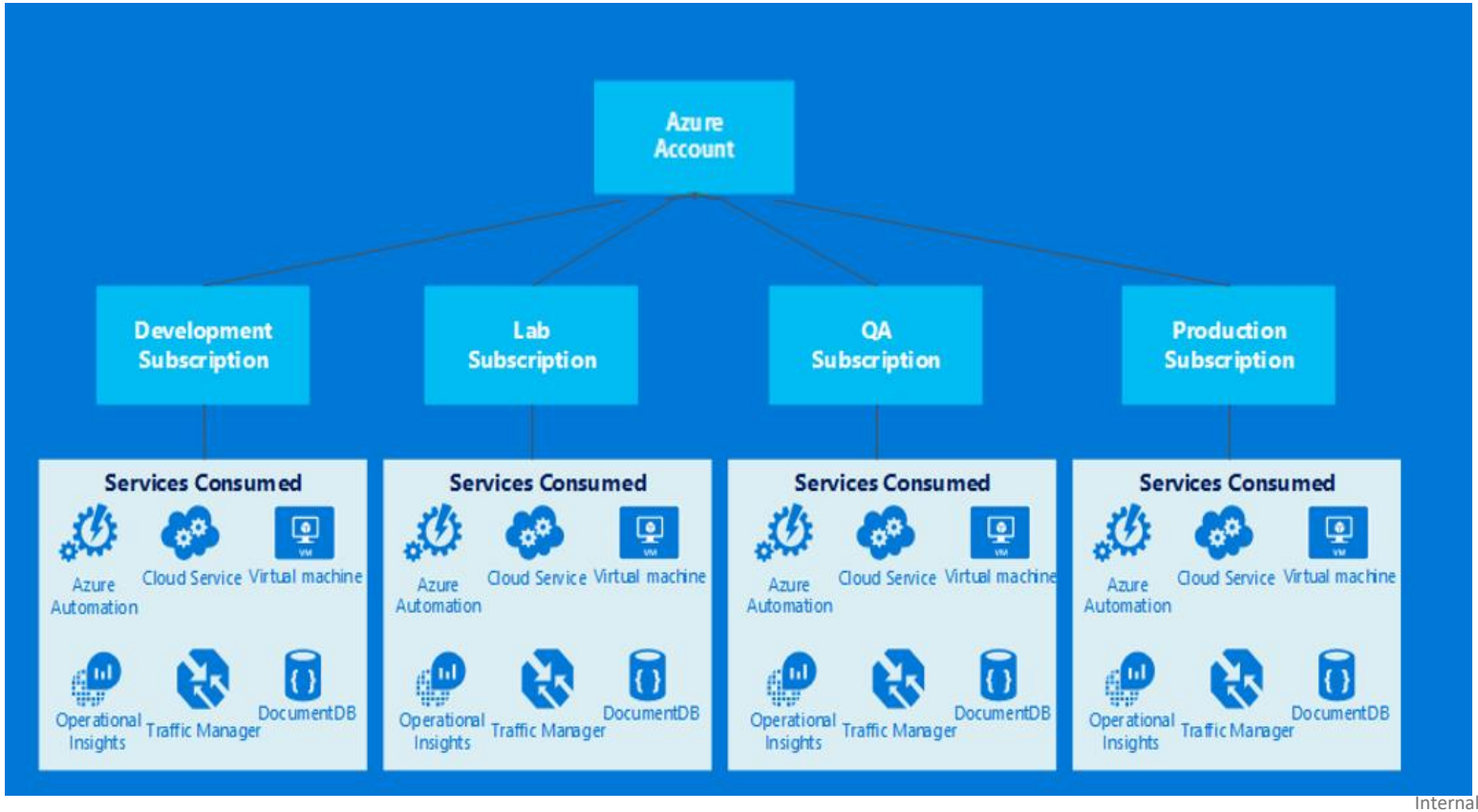
- Each environment contains the different types of applications. Virtual Networks will wrap the different applications for traffic separation. Subnets will be created within each environment to establish required security isolation zones among application tiers.

Pros

- Shared ExpressRoute circuit model
- Low Vnet subscription limit issues (Limit Per 100th application)
- Vnet address spaces can be tailored per application

Cons

- New ExpressRoute circuit required per 10th application, or ER Premium
- Granulated Application RBAC model
- Requires medium capacity planning
- Max of 10 dedicated circuits per subscription, max of 100 applications



Subscription per Application (Customer Managed)

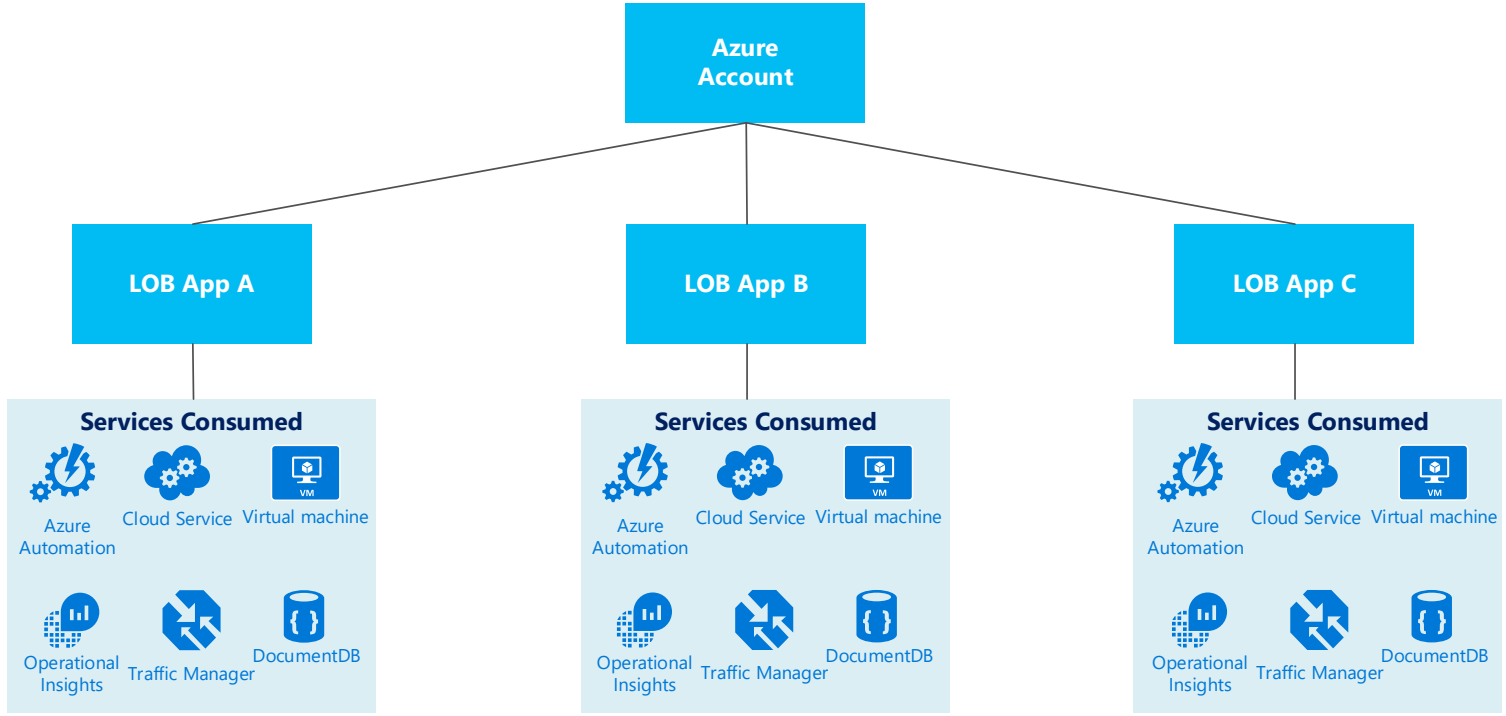
- Each application contains the different tiers. Virtual Networks will wrap the different tiers for traffic separation. Subnets will be created within each tier to establish required security isolation zones.

Pros

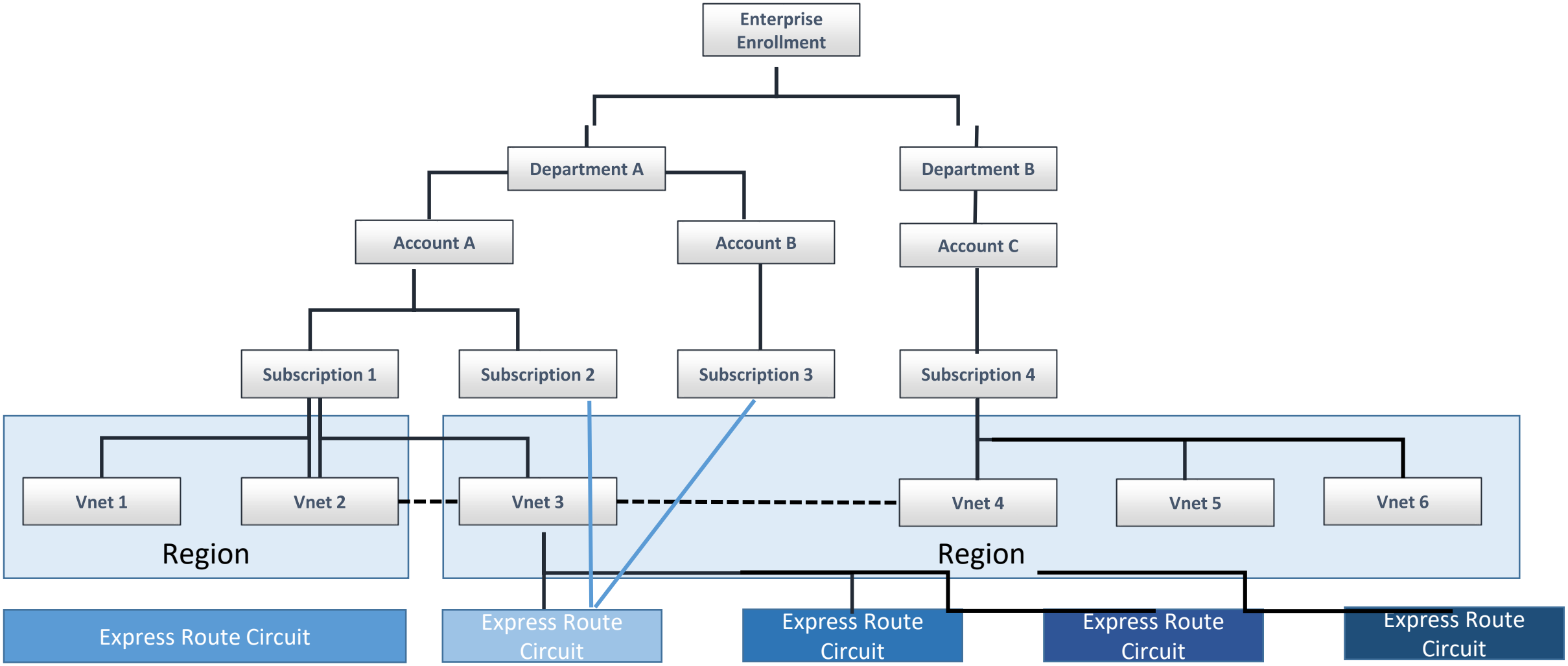
- Minimal Subscription limit issues.
- Minimal Capacity Planning
- Per Application RBAC model

Cons

- Increased Network Costs
- Management Complexity



Networking Considerations



See dashed lines ------- in above diagram

1.You MUST create or use the Azure Dynamic Routing VPN gateways to connect your virtual networks. Static Routing VPN gateways are NOT supported for VNet-to-VNet.

2.For each virtual network, you can connect up to 10 "networks"; i.e., both virtual networks and on premises sites combined cannot exceed 10.

3.You need to ensure that the address prefixes don't overlap among all the connected networks.

4. VNet-to-VNet feature works across regions and subscriptions – same or different regions, single or across subscriptions.

You can link up to 10 virtual networks to an ExpressRoute circuit. All virtual networks must be in the same continent as the ExpressRoute circuit.

You can link a single virtual network with up to 4 ExpressRoute circuits. All ExpressRoute circuits must be in the same continent. They can be ordered through different service providers and in different locations. (https://msdn.microsoft.com/en-us/library/azure/dn606292.aspx)

Multiple Subscriptions = Complexity

- Multiple Azure subscriptions means:
  - Duplicate provisioning and management: IP Address space, network circuits, gateways, vNets, Subnets, NSGs, routing
  - More connectivity to manage
  - More security to manage
  - More identities to manage
  - Potentially more ExpressRoute circuits to buy
- Multiple subscription are going to happen though, so design for it

Azure Subscription Limits

Subscription limits - Azure Resource Manager

The following limits apply when using the Azure Resource Manager and Azure Resource Groups. Limits that have not changed with the Azure Resource Manager are not listed below. Please refer to the previous table for those limits.

Resource	Default Limit	Maximum Limit
VMs per subscription	20[1] per Region	10,000 per Region
Co-administrators per subscription	Unlimited	Unlimited
Storage accounts per subscription	100	100[2]
Resource Groups per subscription	800	800
Availability Sets per subscription	2000 per Region	2000 per Region
Resource Manager API Reads	15000 per hour	15000 per hour
Resource Manager API Writes	1200 per hour	1200 per hour
Resource Manager API request size	4194304 bytes	4194304 bytes
Cloud services per subscription	Deprecated[3]	Deprecated[3]
Affinity groups per subscription	Deprecated[3]	Deprecated[3]

[1]Default limits vary by offer Category Type, such as Free Trial, Pay-As-You-Go, etc.

[2]Limit can be increased by contacting support.

[3]These features are no longer required with Azure Resource Groups and the Azure Resource Manager.

http://azure.microsoft.com/en-us/documentation/articles/azure-subscription-service-limits/

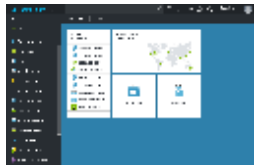
Azure Roles versus Management

RBAC

Azure Active Directory

Azure Active Directory

Customer provisions and manages Azure Object



Powershell

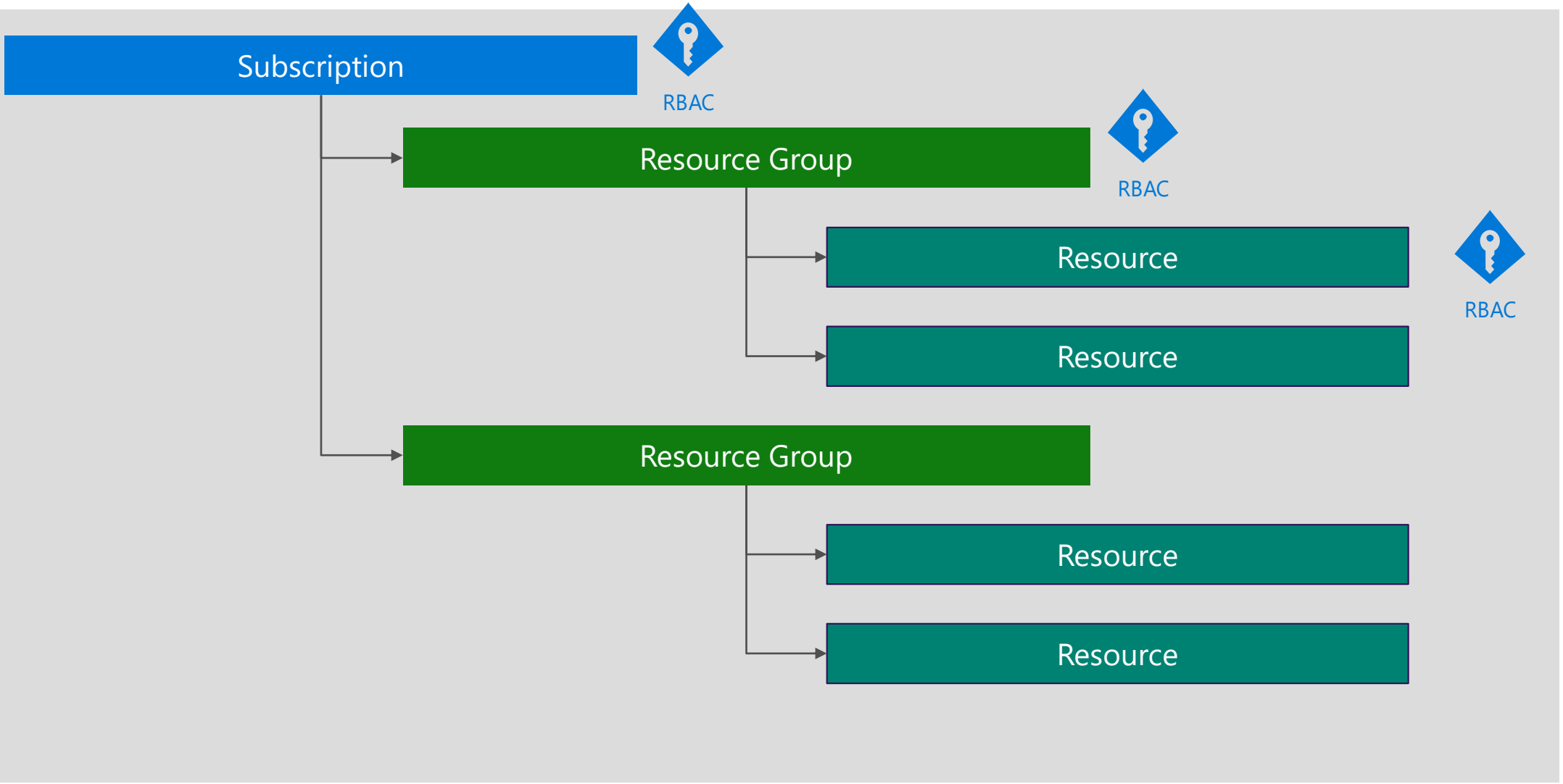
- Application
- OS
- Virtual hardware
- Physical hardware
- Fabric

Active Directory

Customer Manages the OS and app config

Microsoft manages the platform and SLA

Resource Groups and Hierarchy



Azure Resource Overview

Role Considerations

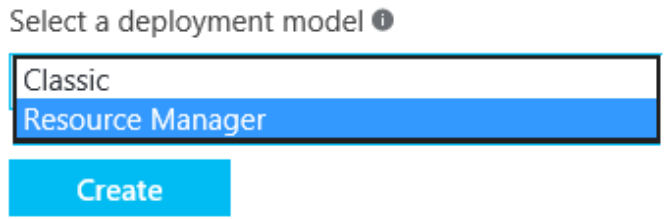
Azure Subscriptions have two administrative models:

Classic (v1)

- Azure Service Management (ASM)

Resource Manager (v2)

- Azure Resource Manager (ARM)



Azure Resource Management (ARM) environment, a subscription now has two administrative models: Service Management and Azure Resource Management. With ARM the subscription is no longer needed as an administrative boundary. ARM provides a more granular Roles Based Access Control (RBAC) model for assigning administrative rights at the resource level. RBAC is currently being released in stages, 22 new roles have been released and user defined roles is coming in a future release. There will be some complexity during the coexistence of the service management and resource management environments and will need to be carefully considered.

Resources in Azure

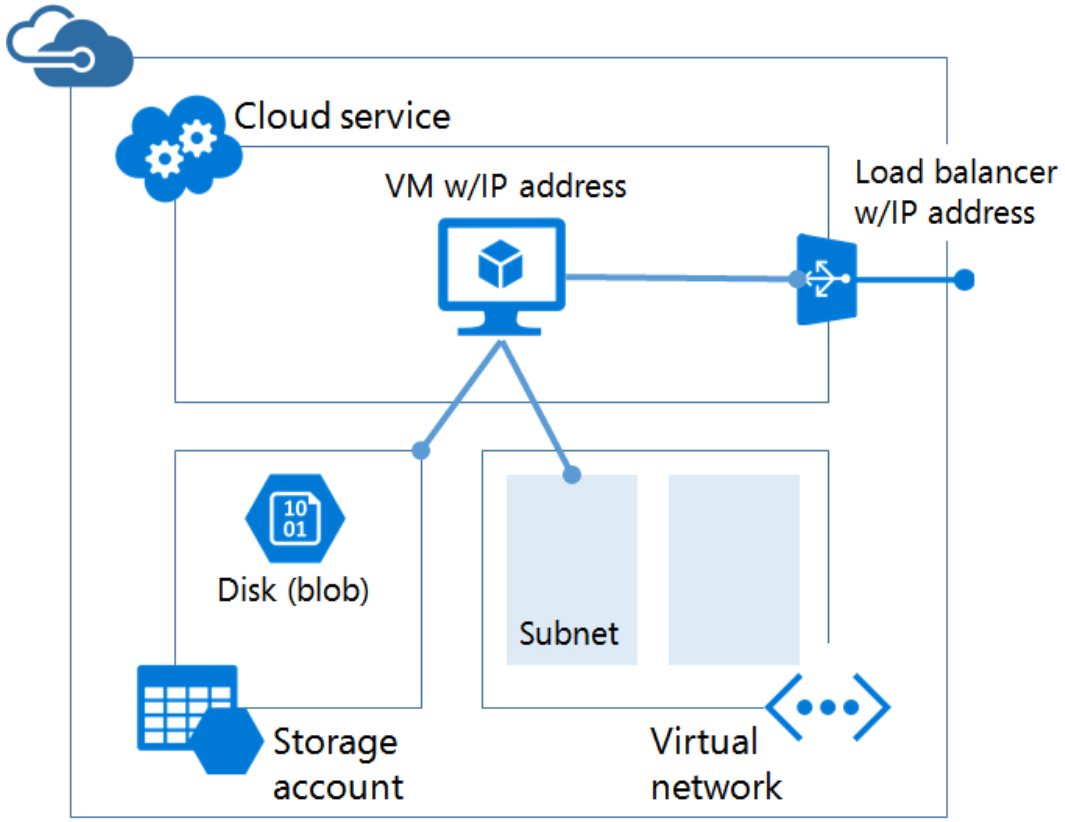
Classic (v1)

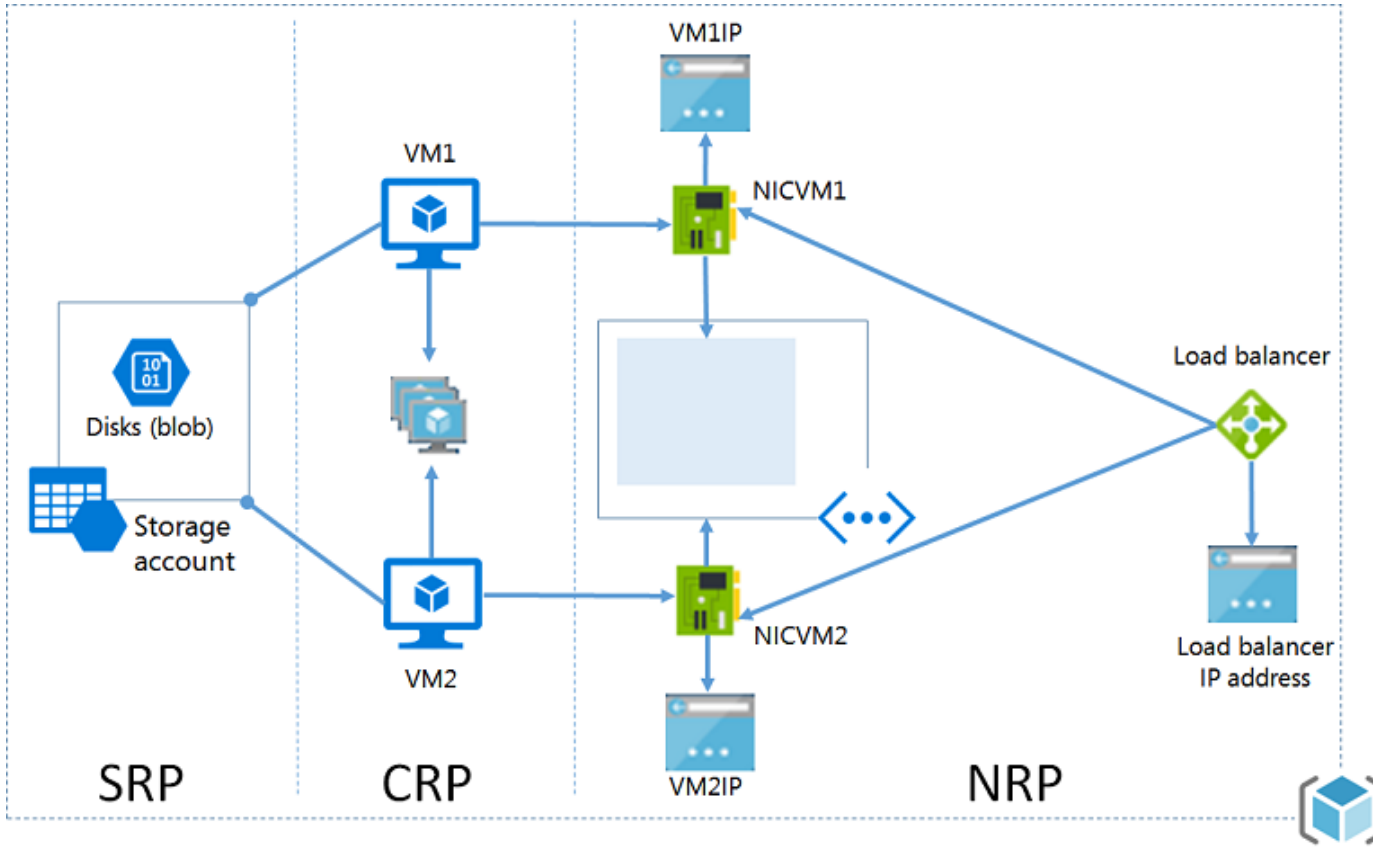
- Multiple objects combined into a single manageable instance
- Must connect to a classic network infrastructure

Resource Manager (v2)

- Each object a separately manageable
- Must connect to a RM network infrastructure
- All new development focused here

Azure Resource Manager

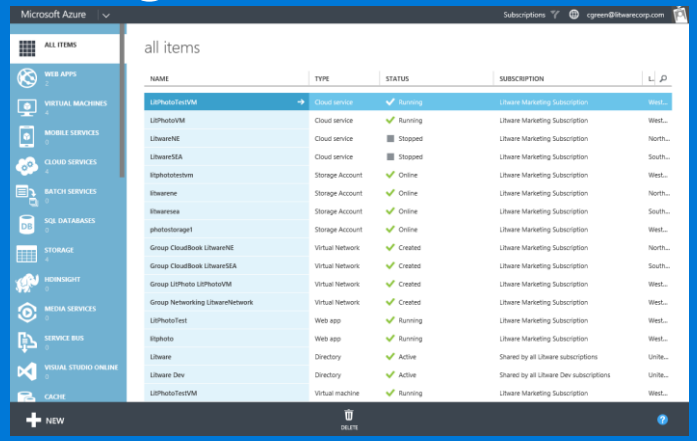
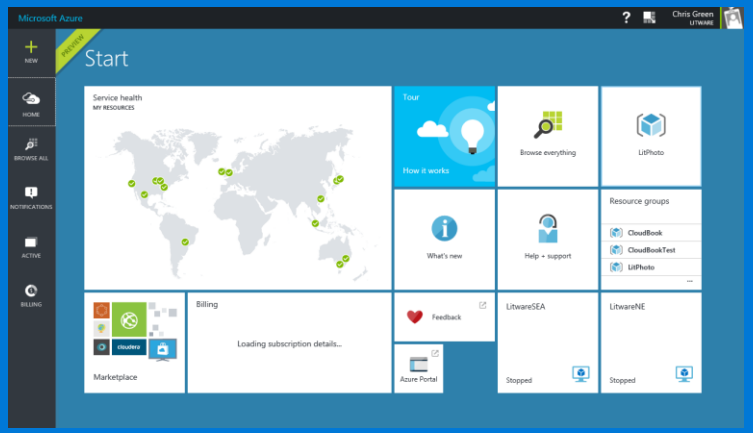




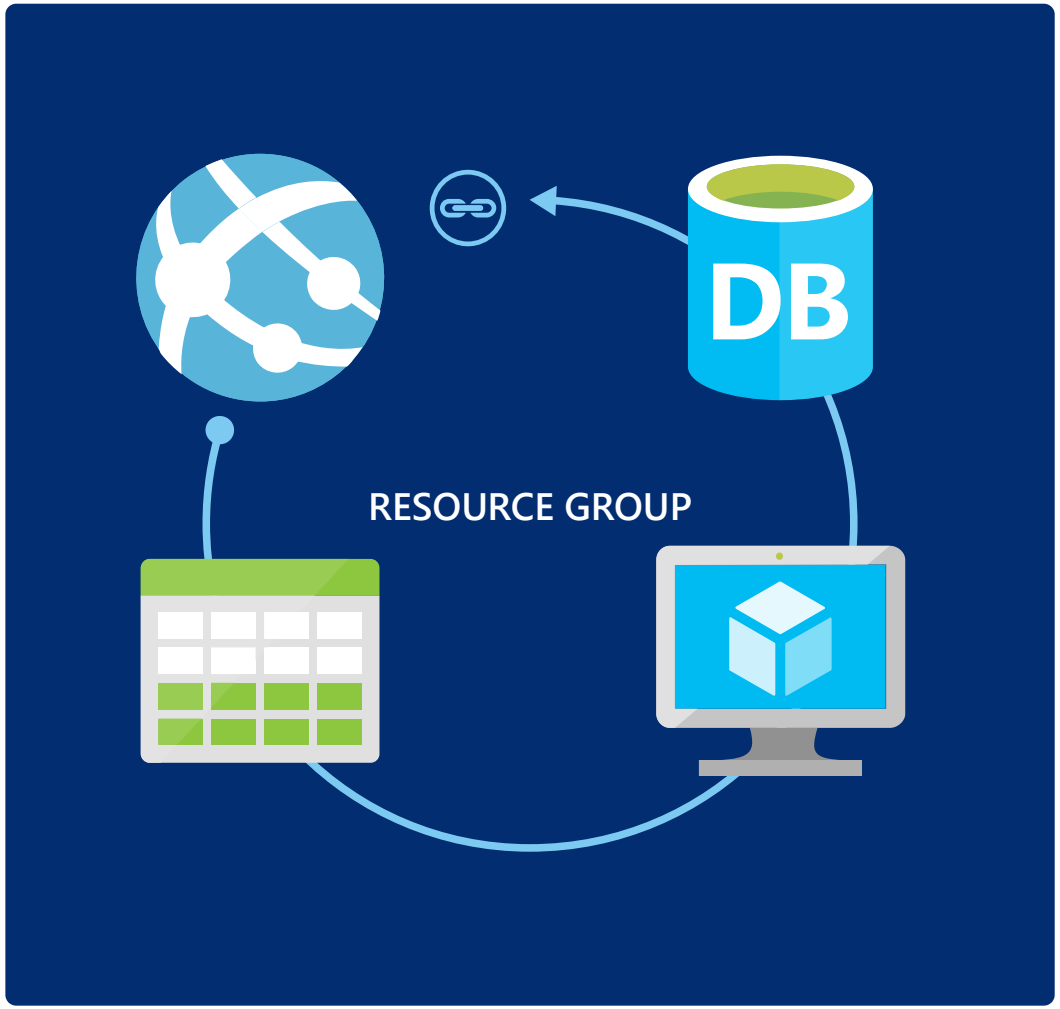
Classis (ASM)

ARM with RPs

Portal and APIs

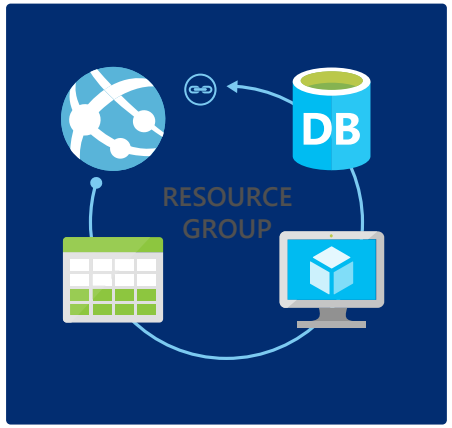
Feature	manage.windowsazure.com	portal.azure.com / ARM
		
Granularity	Subscription	Subscription, resource group, resource
Principal	User	User, directory group, application
Roles	Full control (or no access)	30+ Built-in roles Custom roles

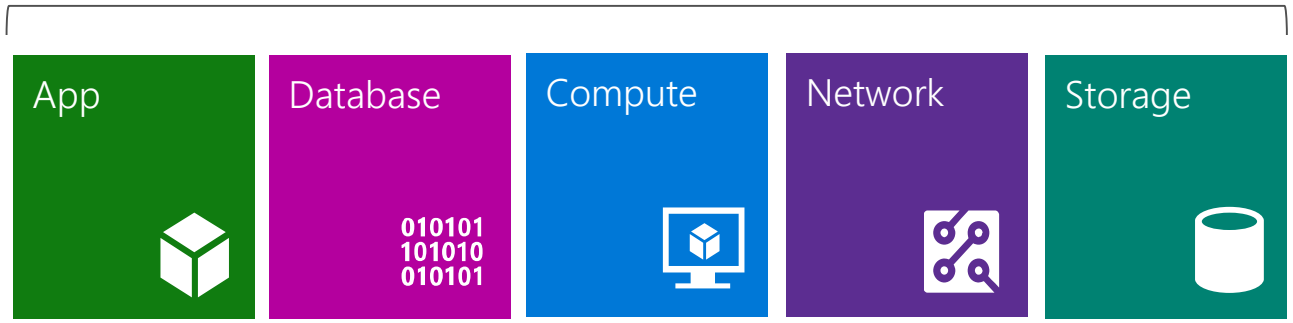
Resource Groups

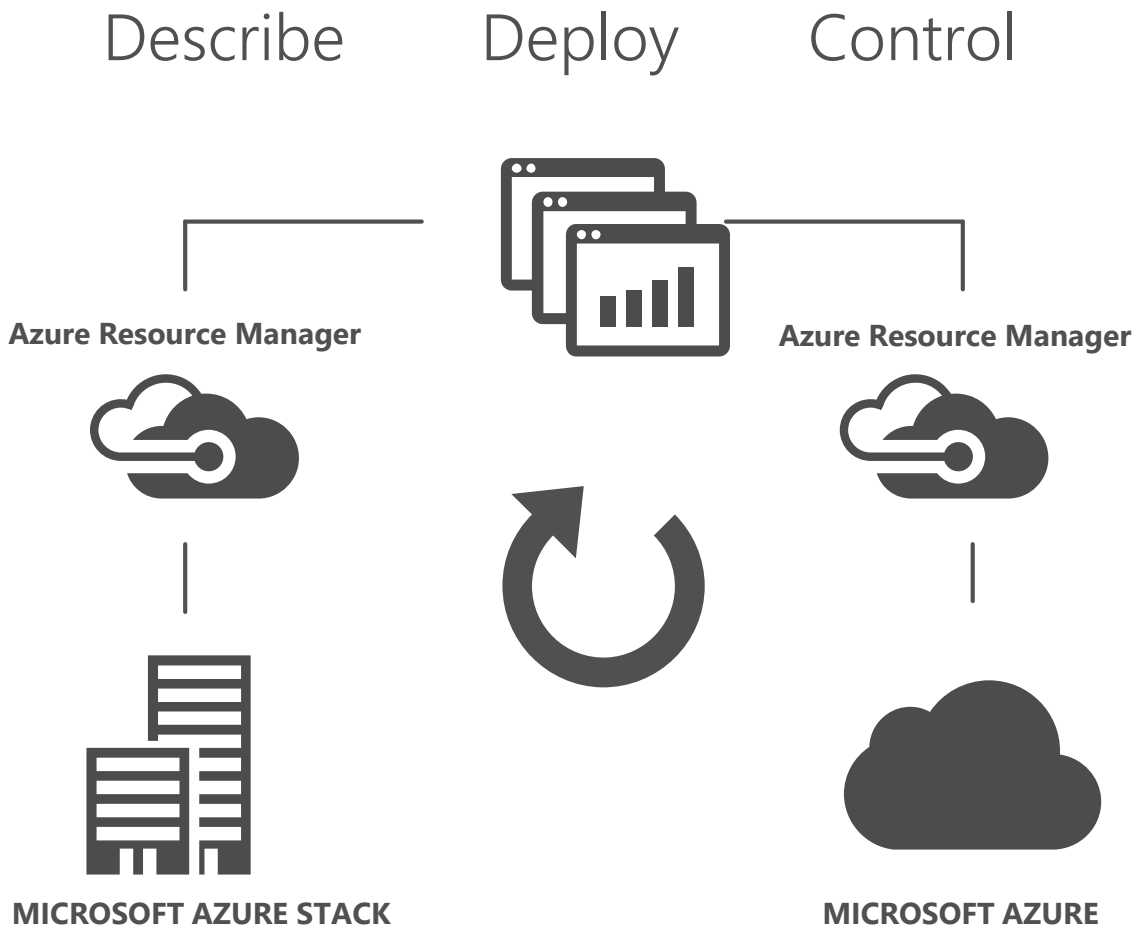


- Tightly coupled containers of multiple resources of similar or different types
- Every resource *must* exist in one and only one resource group
- Resource groups can span regions
- Nesting of Resource Groups not supported
- Only Subscription Owners can create resource groups

Azure Resource Manager



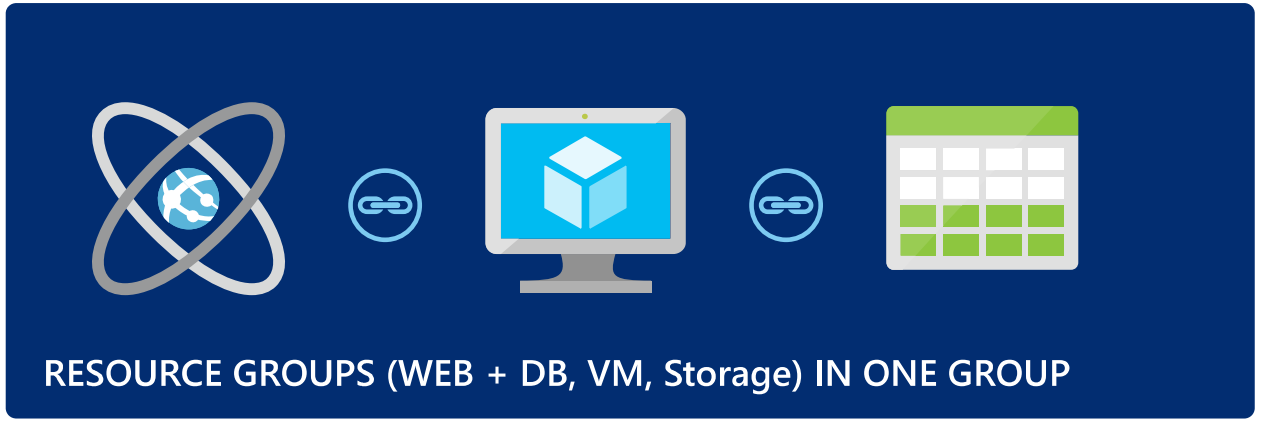




Resource Group Lifecycle

Question:

Should these resources be in the same group or a different one?



OR

Azure RBAC Overview

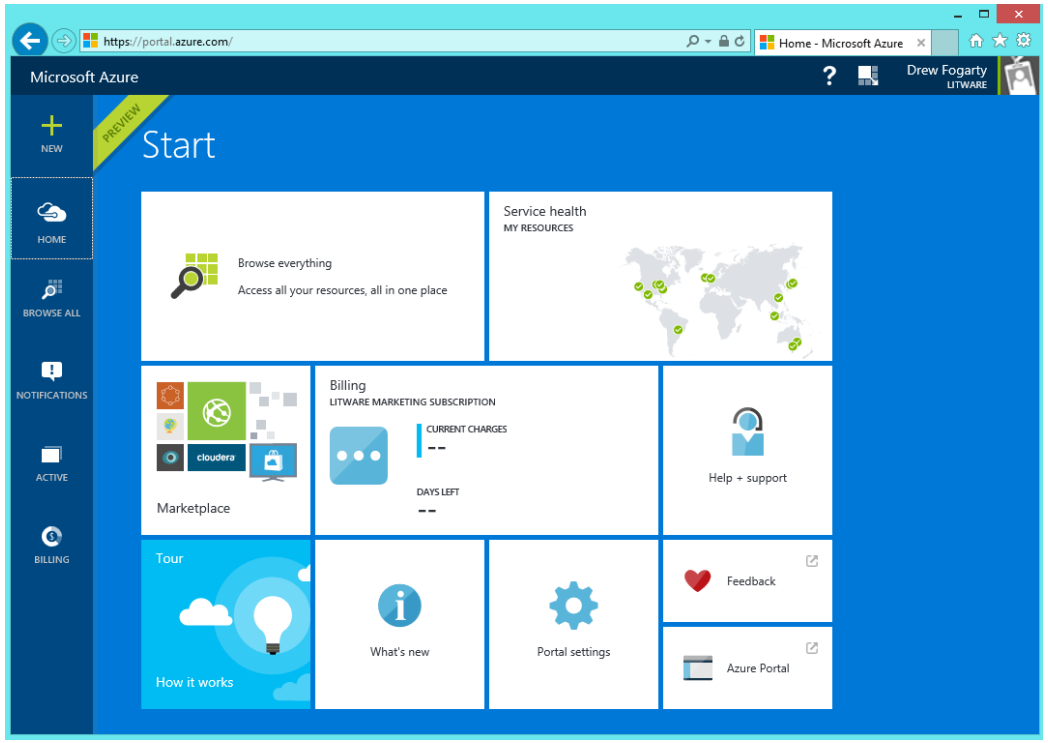
Least Privilege as a Model

Goal

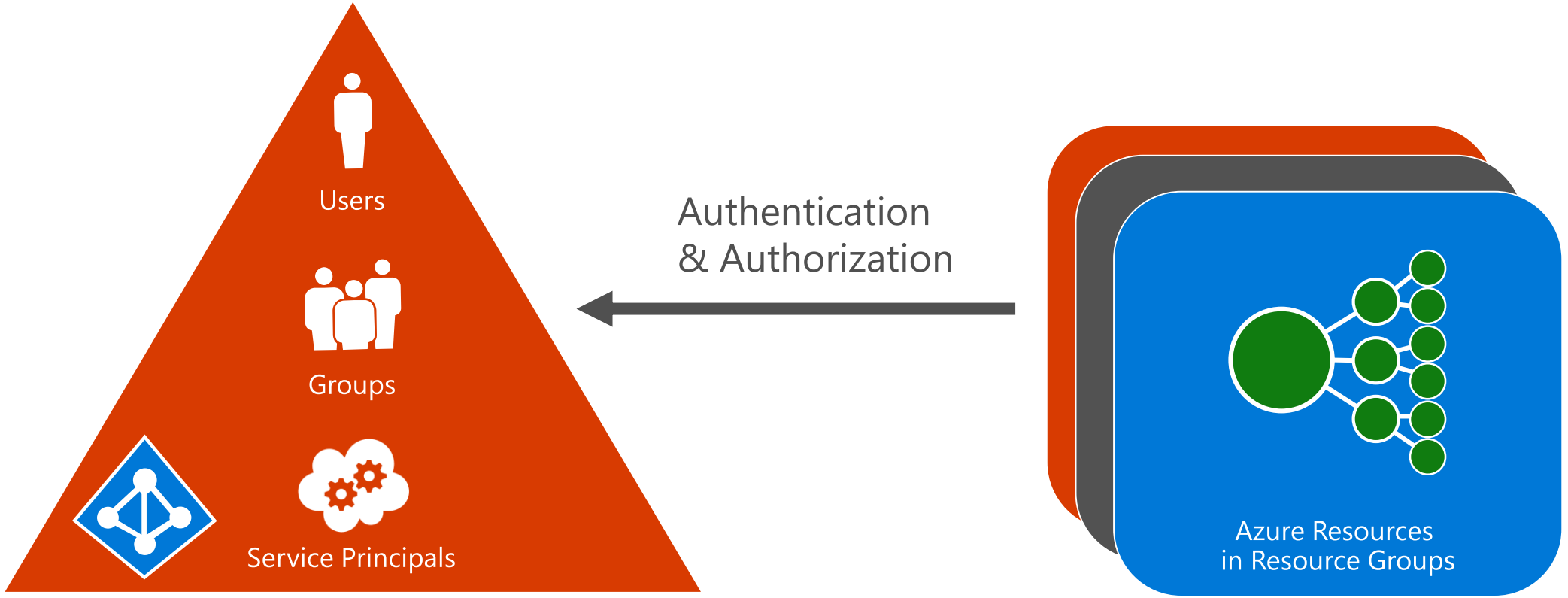
- Users can do the tasks their job requires
- But no more than that

Best practices

- Use the portal and ARM API
- Assign the right role
- Use resource groups

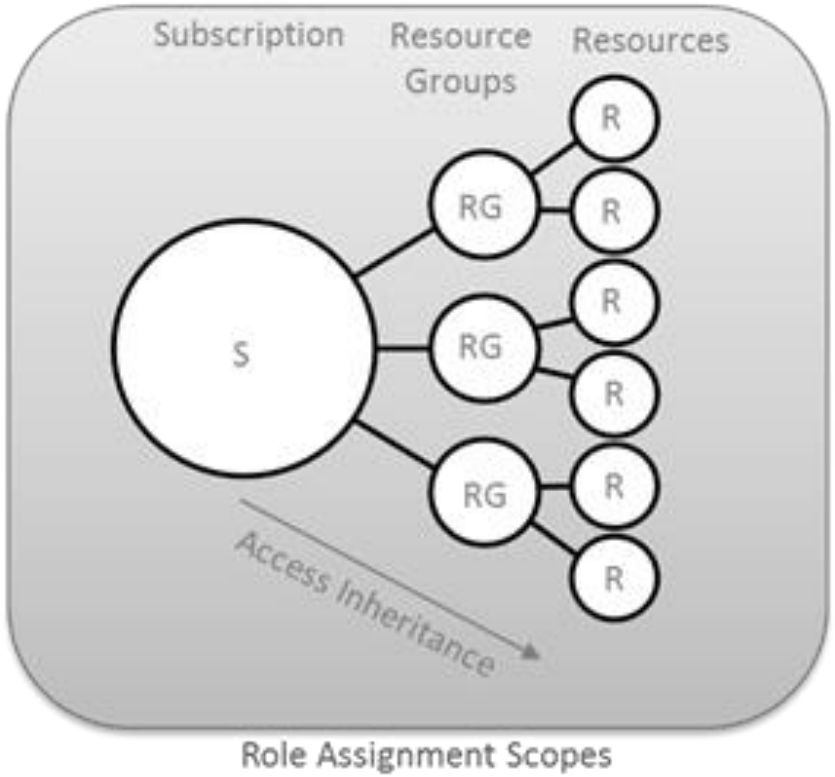


Role Based Access Control



ARM Hierarchy and RBAC Roles

ARM provides a more granular Roles Based Access Control (RBAC) model for assigning administrative rights at the resource level.



Owner

- Can perform all management operations for a resource and its child resources including access management and granting access to others.

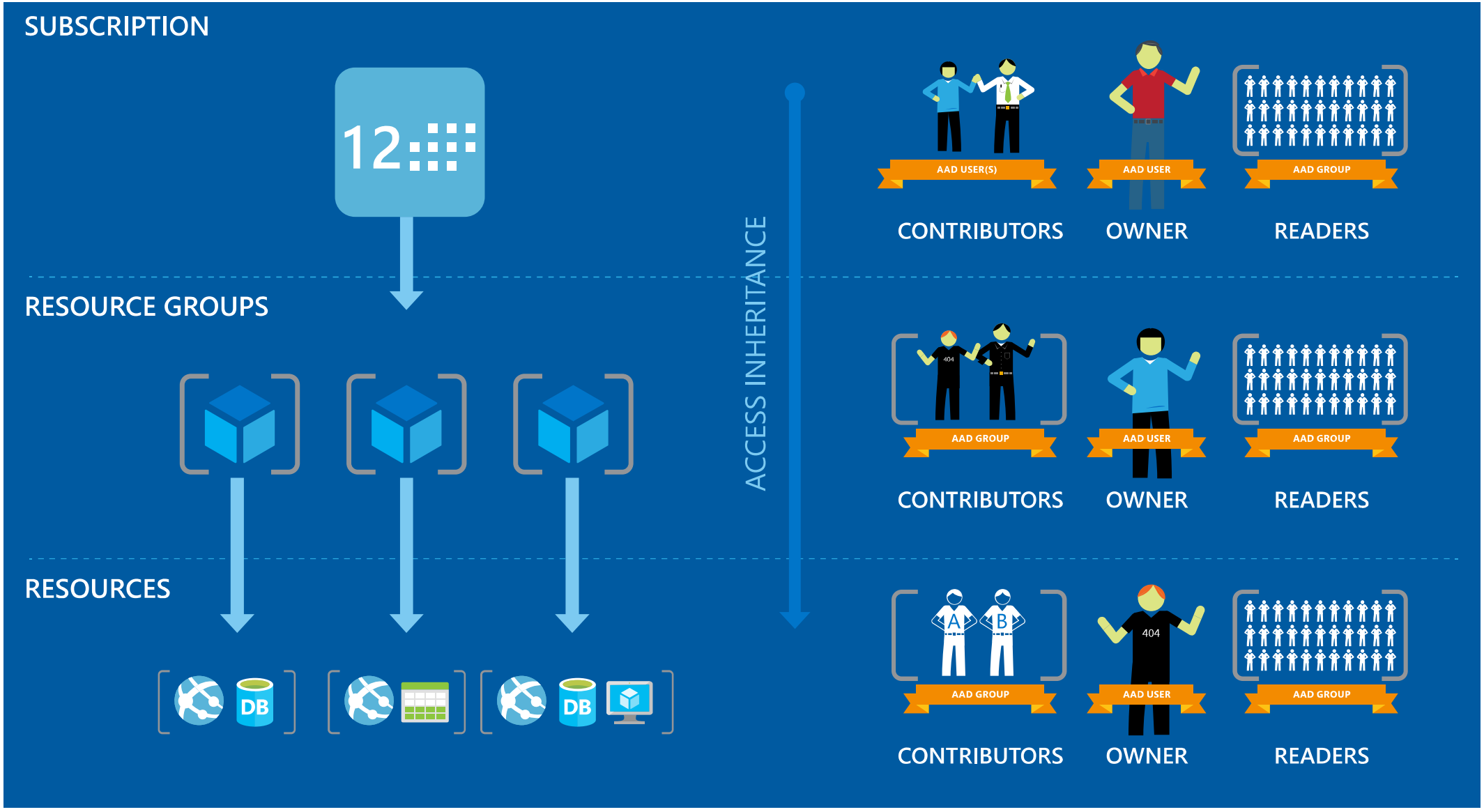
Contributor

- Can perform all management operations for a resource including create and delete resources. A contributor cannot grant access to other.

Reader

- Has read-only access to a resource and its child resources. A reader cannot read secrets.

Role Based Access Control



Key RBAC Concepts

Role Definitions

- describes the set of permissions (e.g. read actions)
- can be used in multiple assignments

Role Assignments

- associate role definitions with an identity (e.g. user/group) at a scope (e.g. resource group)
- always inherited – subscription assignments apply to all resources

RBAC - Granular Scopes

/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

Roles for Azure subscription resources

Three primary roles:

- Owner, Contributor, Reader
- Permissions on all Azure resources

30+ resource-specific roles

- Website contributor, Virtual machine contributor, etc.
- Permissions scoped to resources and actions typically required by customers
- Will add more as new Azure resources come online

Custom roles

- Allows customers to take existing actions and create a custom RBAC role
- Role must be loaded into each subscription

https://azure.microsoft.com/en-us/documentation/articles/role-based-access-built-in-roles/

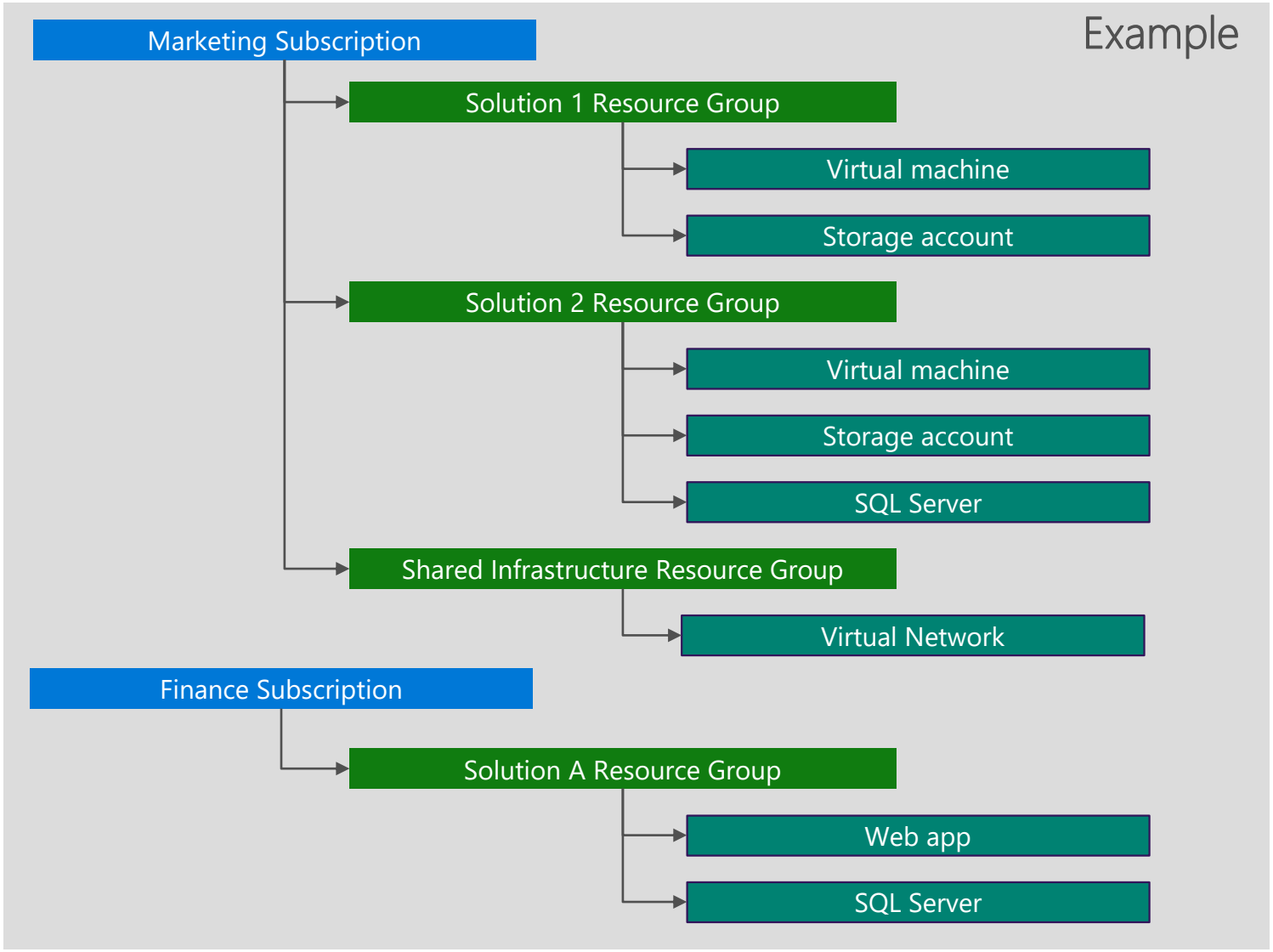
Built-in Roles

BUILT-IN ROLE	ACTIONS	NOT ACTIONS
Owner (allow all actions)	*	
Contributor (allow all actions except writing or deleting role assignments)	*	Microsoft.Authorization/*/Write, Microsoft.Authorization/*/Delete
Reader (allow all read actions)	*/Read	

Virtual Machine Contributor

Actions	Access
Microsoft.Storage/storageAccounts/read	Read storage accounts
Microsoft.Storage/storageAccounts/listKeys/action	List storage account keys
Microsoft.Network/virtualNetworks/read	Read virtual networks
Microsoft.Network/virtualNetworks/subnets/join/action	Join virtual network subnets
Microsoft.Network/loadBalancers/read	Read load balancers
Microsoft.Network/loadBalancers/backendAddressPools/join/action	Join load balancer backend address pools
Microsoft.Network/loadBalancers/inboundNatRules/join/action	Join load balancer inbound NAT Rules
Microsoft.Network/publicIPAddresses/read	Read network public IP addresses
Microsoft.Network/publicIPAddresses/join/action	Join network public IP addresses
Microsoft.Network/networkSecurityGroups/read	Read network security groups
Microsoft.Network/networkSecurityGroups/join/action	Join network security groups
Microsoft.Network/networkInterfaces/*	Create and manage network interfaces
Microsoft.Network/locations/*	Create and manage network locations
Microsoft.Network/applicationGateways/backendAddressPools/join/action	Join network application gateway backend address pools
Microsoft.Compute/virtualMachines/*	Create and manage virtual machines
Microsoft.Compute/availabilitySets/*	Create and manage compute availability sets
Microsoft.Compute/locations/*	Create and manage compute locations
Microsoft.Authorization/*/read	Read authorization
Microsoft.Resources/subscriptions/resourceGroups/read	Read subscription resource groups
Microsoft.Resources/subscriptions/resourceGroups/resources/read	Read subscription resource groups resources
Microsoft.Resources/subscriptions/resourceGroups/deployments/*	Create and manage subscription resource group deployments
Microsoft.Insights/alertRules/*	Create and manage Insights alert rules
Microsoft.Support/*	Create and manage support tickets

Resource Groups and Access Management



Best practices

- Organize resources to meet access management requirements
- Grant access at resource group when appropriate

Benefits

- More granularity
- Aligns with resource-specific roles
- Ongoing manageability

Assigning Resource-specific Roles

Requirement

- Map existing organizational roles (web, DB…) to their cloud solutions
- Many Azure resources that work together are peers in the same resource group
  - Virtual machine and storage account, web app and AppInsights
- To fully manage a resource, a user may also need to manage its related peers

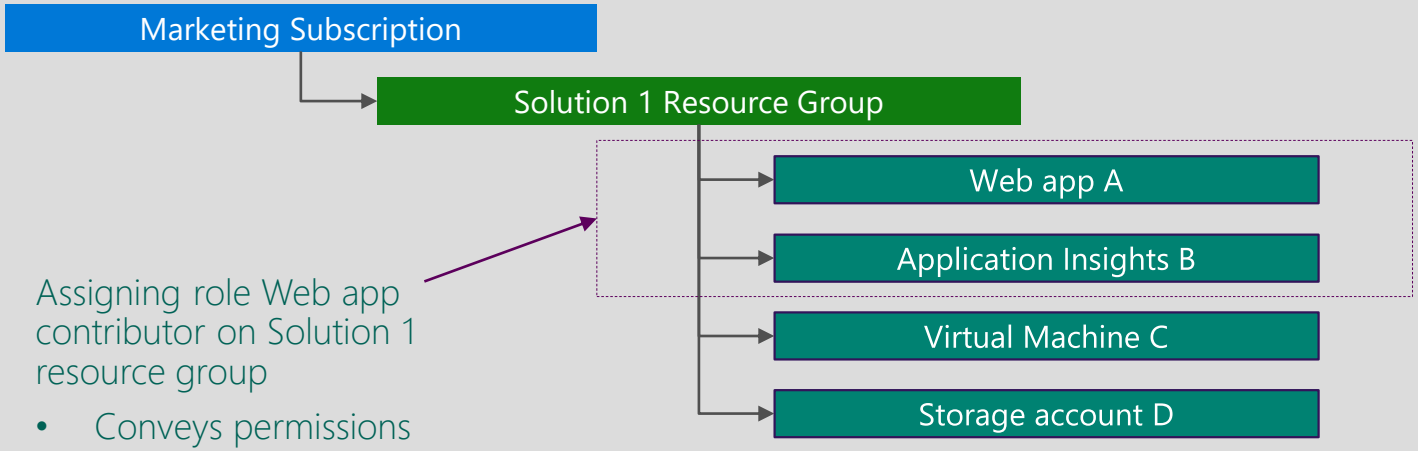
Best Practice

- Assign resource-specific role on the resource group

Alternative

- Assign access to each resource individually

Example RBAC Assignment



Assigning role Web app contributor on Solution 1 resource group

- Conveys permissions on web apps and Application Insights instances in the resource group
- Does not convey permissions on virtual machines or storage accounts

Requirement (example)

- User needs to manage "Web app A" and related resources such as "Application Insights B"
- User shouldn't manage "Virtual Machine C" or "Storage account D"

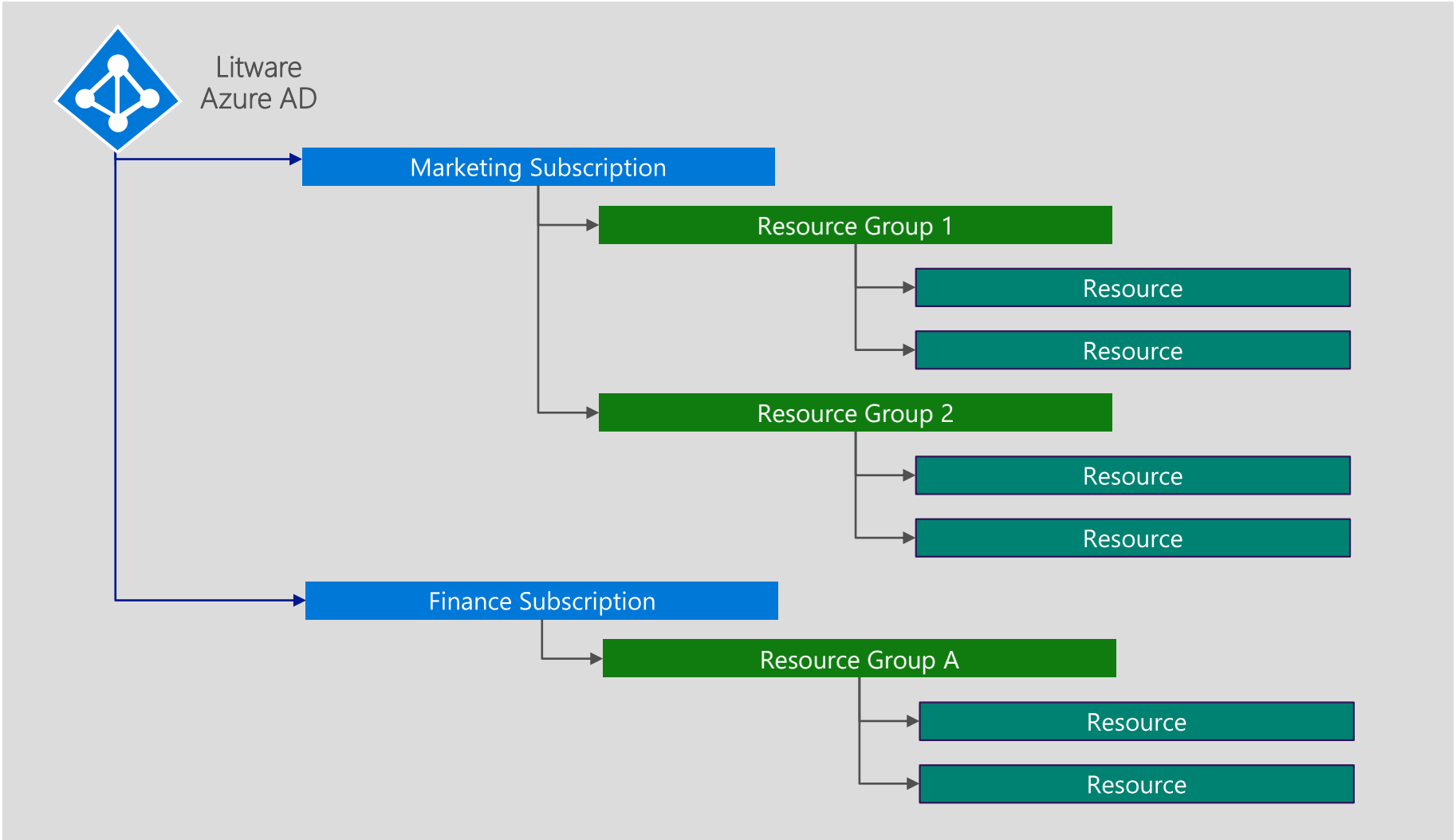
Best practice

- Assign *Web app contributor* and *Application Insights Component Contributor* role on 'Solution 1 resource group'

Alternative: Two assignments

- *Contributor* role on 'Virtual Web App" and
- *Application Insights Component Contributor* role on "App Insights B"

Azure Active Directory Integration



Best practice

- All organizational Azure subscriptions use the same Azure AD for access control.
- I.e., don't have each subscription in the organization relying on its own Default directory.

Benefits

- Manageability
- Compliance

Azure Resource Manager Resource Locks

Resource Locks

- Accidents happen. Resource locks help prevent them :)

- Resource locks allow administrators to create policies which prevent write actions or prevent accidental deletion.

Key Concepts

- Resource lock
  - Policy which enforces a "lock level" at a particular scope
- Lock level
  - Type of enforcement; current values include CanNotDelete and ReadOnly
- Scope:
  - The realm to which the lock level is applied. Expressed as a URI; can be set at the resource group, or resource scope.

Module 1 – Lesson 3 – Managing Azure with the Azure portal

Microsoft Azure Portal

- Browser based administration (https://portal.azure.com)
- Current portal that supports all Azure services
- Full support of ARM (Azure Resource Manager) resources
- Centralize view in one portal
- Personalize experience
- Fine grained access control
- Billing visibility

Azure's old/classic portal (ASM – Azure Service Manager) was located at:
https://manage.windowsazure.com

Module 1 – Lesson 4 - Managing Azure with Windows PowerShell

Azure PowerShell

- Windows PowerShell is a scripting platform
- Azure has PS modules for Azure cmdlets
- Automating IT processes with Scripts (Reusability)
- Part of larger deployments (Devops?)
- There are various PowerShell modules for Azure (we will focus on the AzureRM module)

Azure PowerShell module browser: https://docs.microsoft.com/en-us/powershell/module/
(look for AzureRM PowerShell)
Azure PS is an open-source project: https://github.com/Azure/azure-powershell/

How to get Azure PowerShell

1. Web platform installer - https://azure.microsoft.com/en-us/downloads/
a) Will get the latest version for you
b) It will take care of all the prerequisites
2. The PowerShell Gallery – depends on PowerShellGet module
a) Depends on the Windows Management Framework (WMP) :
https://www.microsoft.com/enus/download/details.aspx?id=54616
b) If you are on Windows 10, you are all set to do this. If you are not then download the WMP.
c) For all other editions download WMP and take it from there
3. Microsoft Windows Installer package (MSI)
4. Azure Cloud Shell on the Azure Portal (the easiest!)

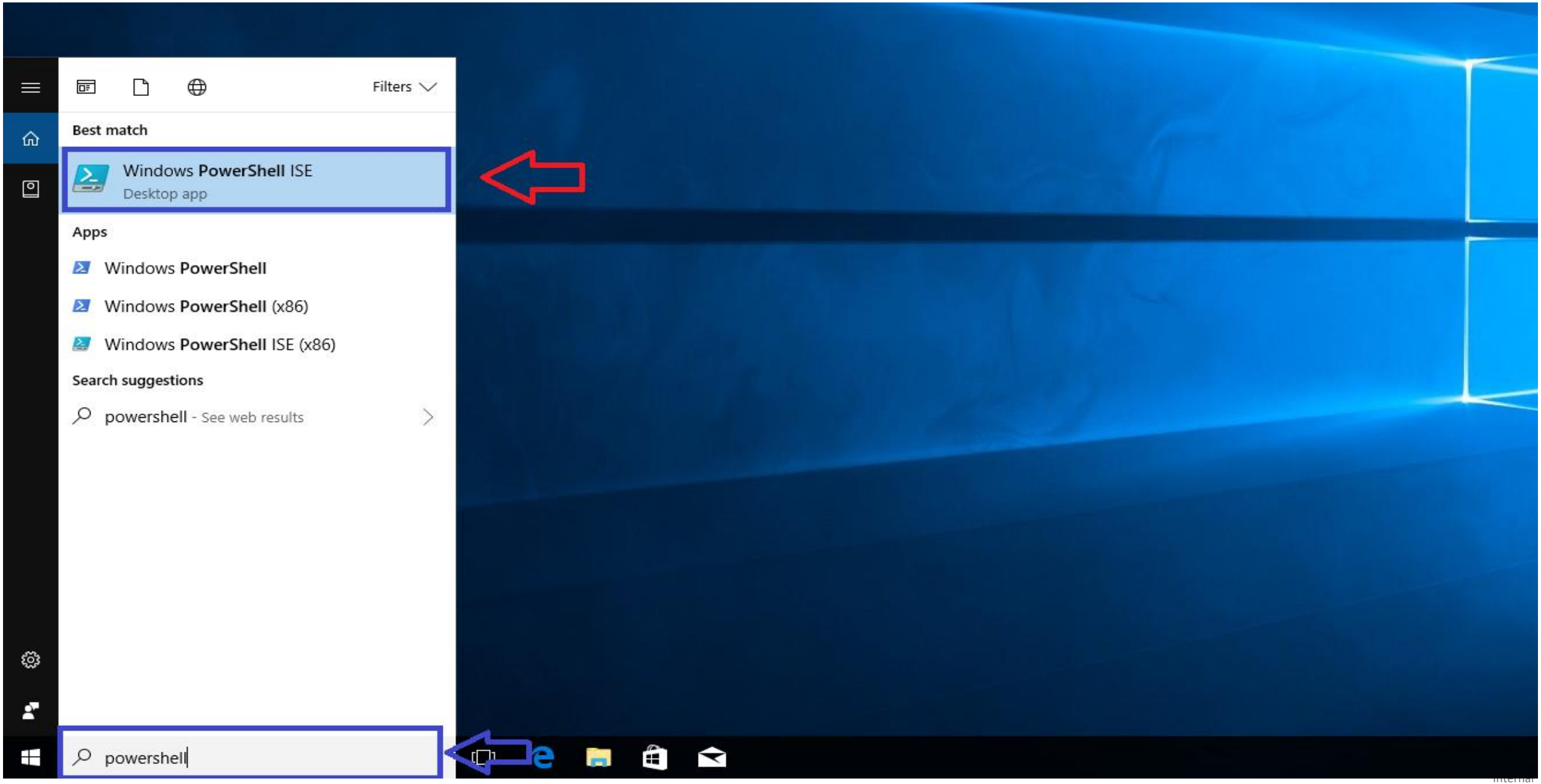
Azure PowerShell Installation

We will be using The PowerShell gallery
Check if it is already installed and if it is then which version it is:
Get-Module AzureRM -ListAvailable | Select-Object -Property Name,Version,Path
Install it :
Install-Module -Name AzureRM
Update it (if you need to in future):
Update-Module -Name AzureRM
Modules will be located at: C:\Program Files\WindowsPowerShell\Modules

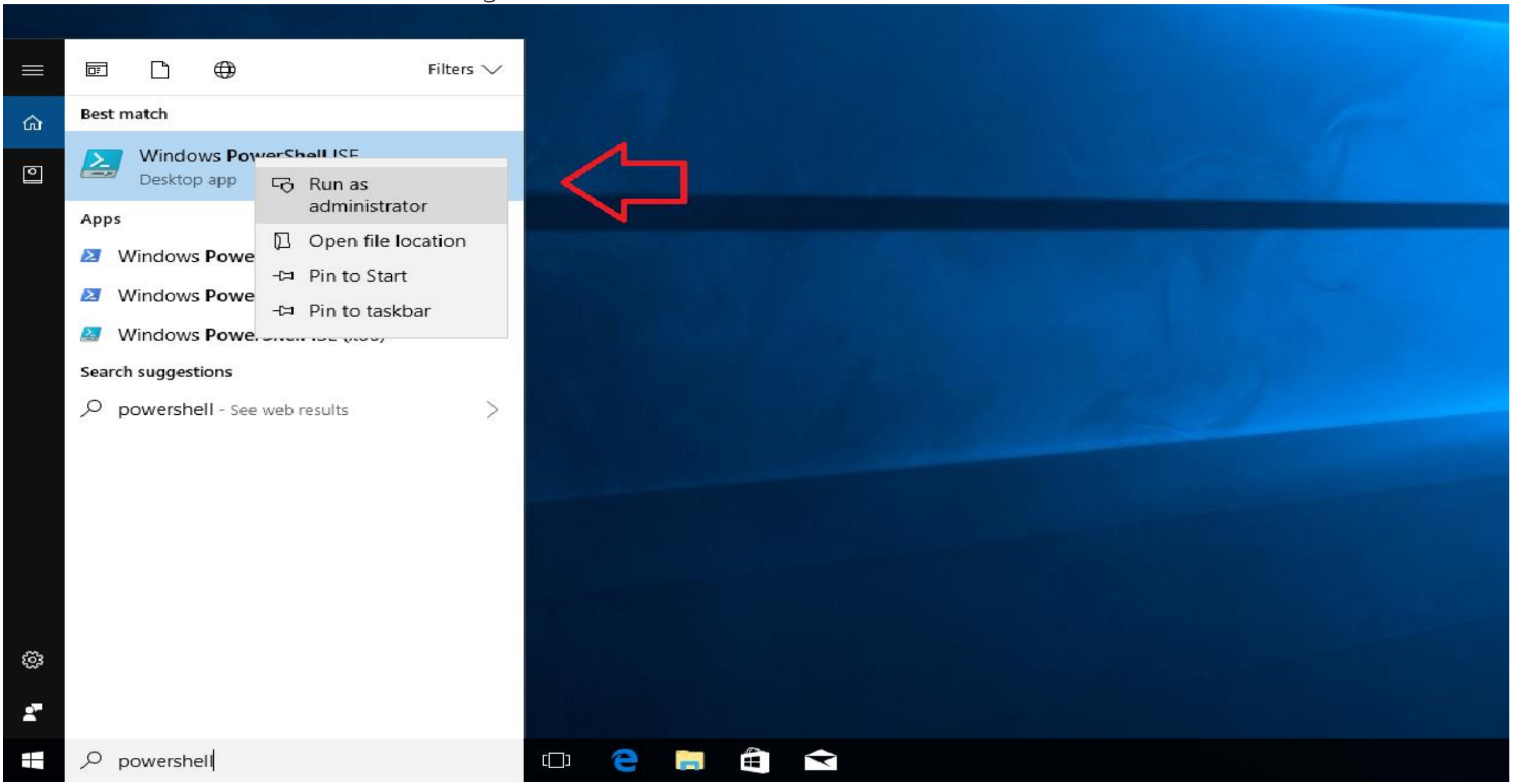
Azure PowerShell Login and subscription access

You must Authenticate to access the Azure Subscriptions
1. AD Authentication
2. Certificate based authentication (https://docs.microsoft.com/en-us/azure/azure-resource-manager/resourcegroup-authenticate-service-principal)
Using AD authentication:
Connect to Azure with an authenticated account:
Add-AzureRmAccount
Get the current context:
Get-AzureRmContext

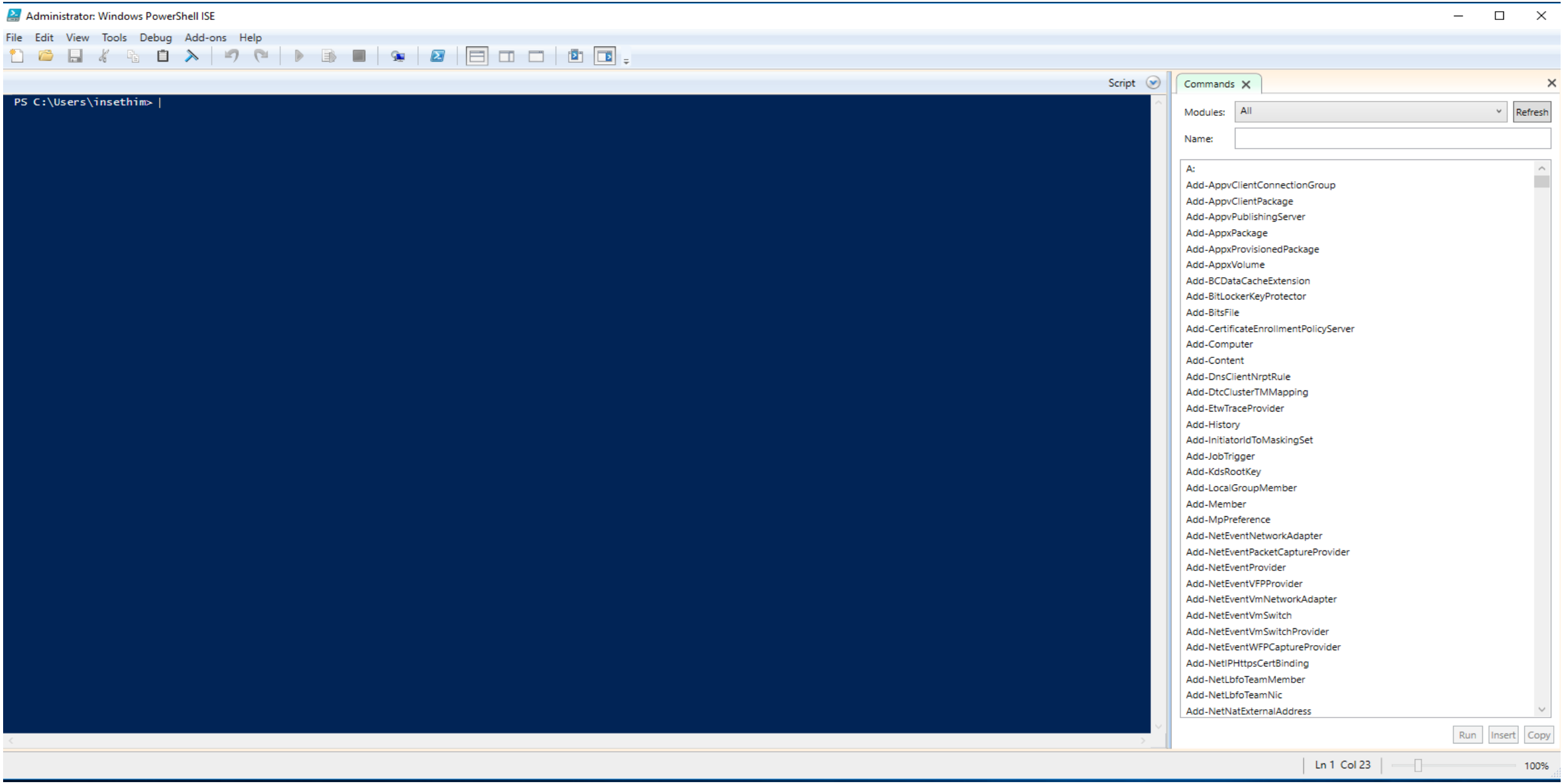
Search for PowerShell on search tab as below and you will get below options



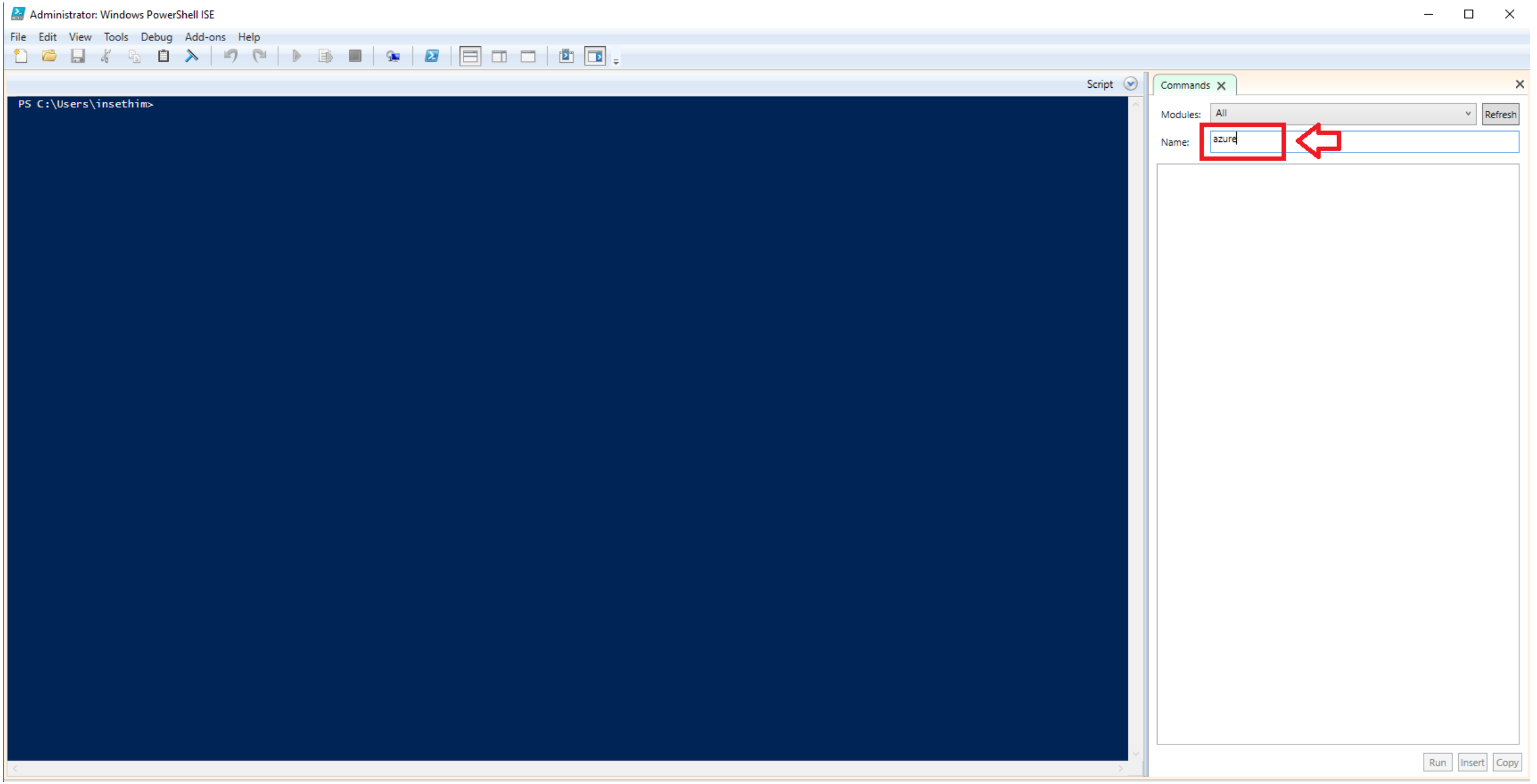
Choose Windows PowerShell ISE and right click to **Run as administrator**



The PowerShell ISE will look like below after run as administrator



This PowerShell is not having azure modules to run azure commands, so we need to install azure module in the PowerShell.

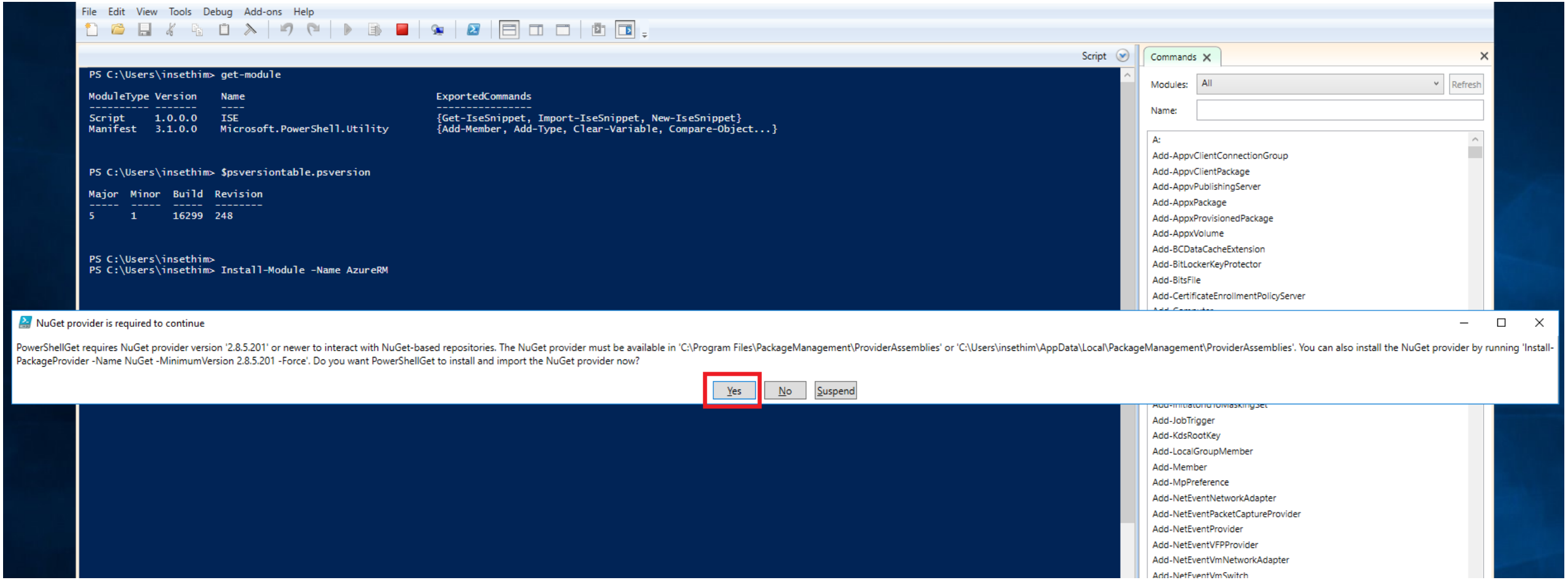


While installing Azure Module in PowerShell, Pls type all below commands shown in pic and click on **yes** button

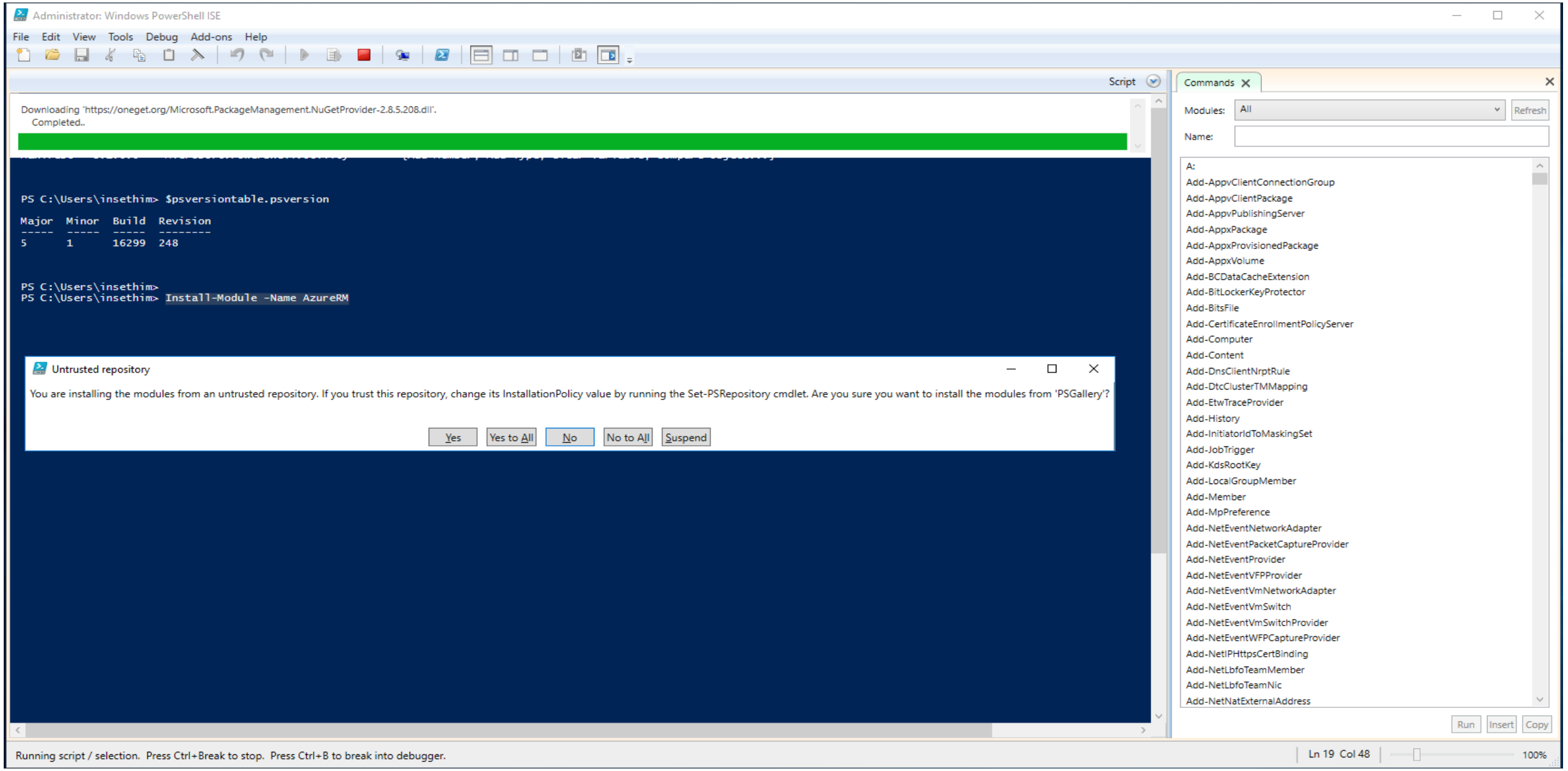
get-module

$psversiontable.psversion

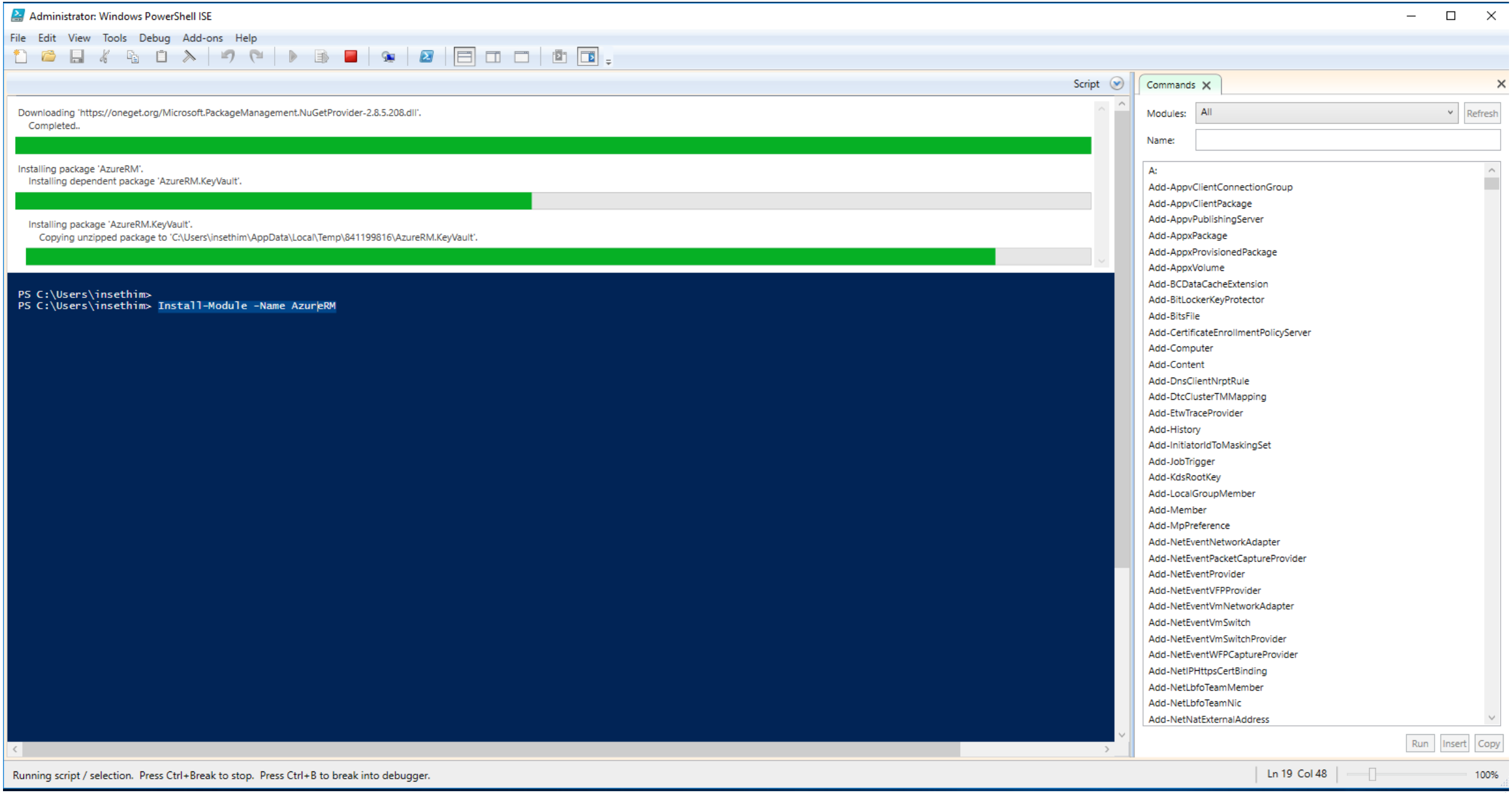
Install-Module -Name AzureRM



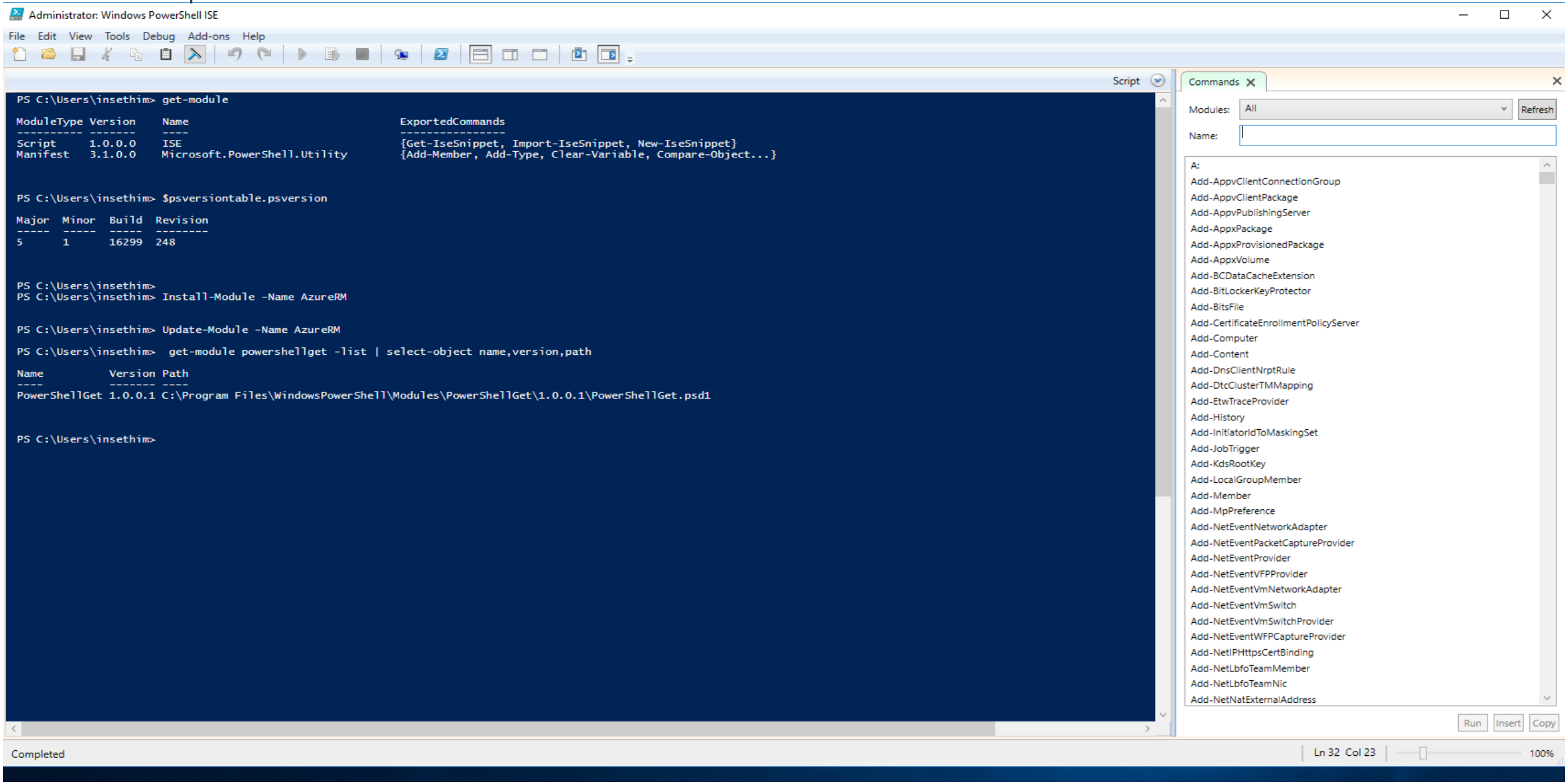
After "Install-Module -Name AzureRM" command it will download the Azure Module from internet and we need to select "Yes to all" button



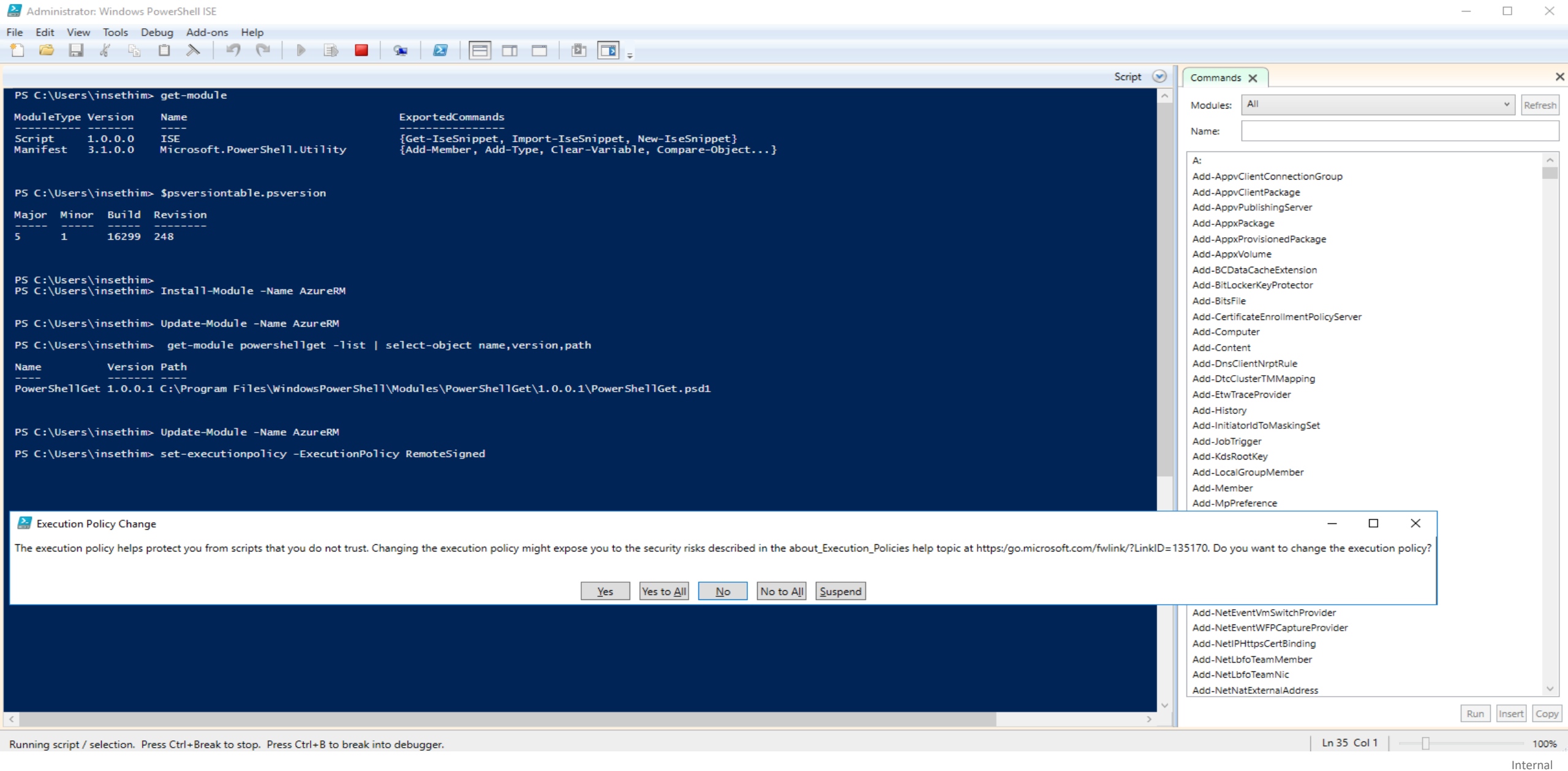
After clicking "yes to all" it will install all the packages of Azure Module on PowerShell as shown below



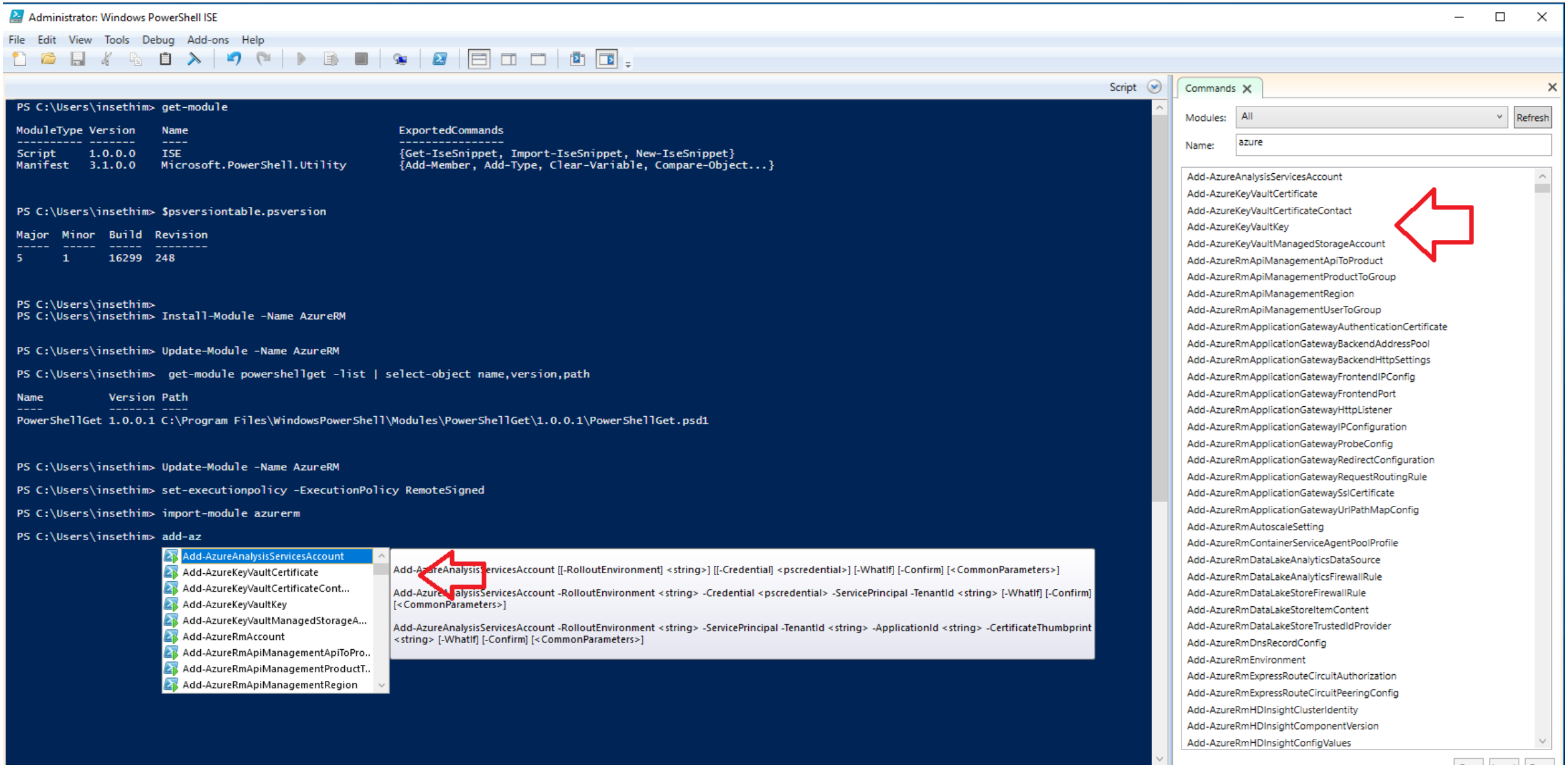
Check if it is already installed and if it is then which version it is: get-module powershellget -list | select-object name,version,path



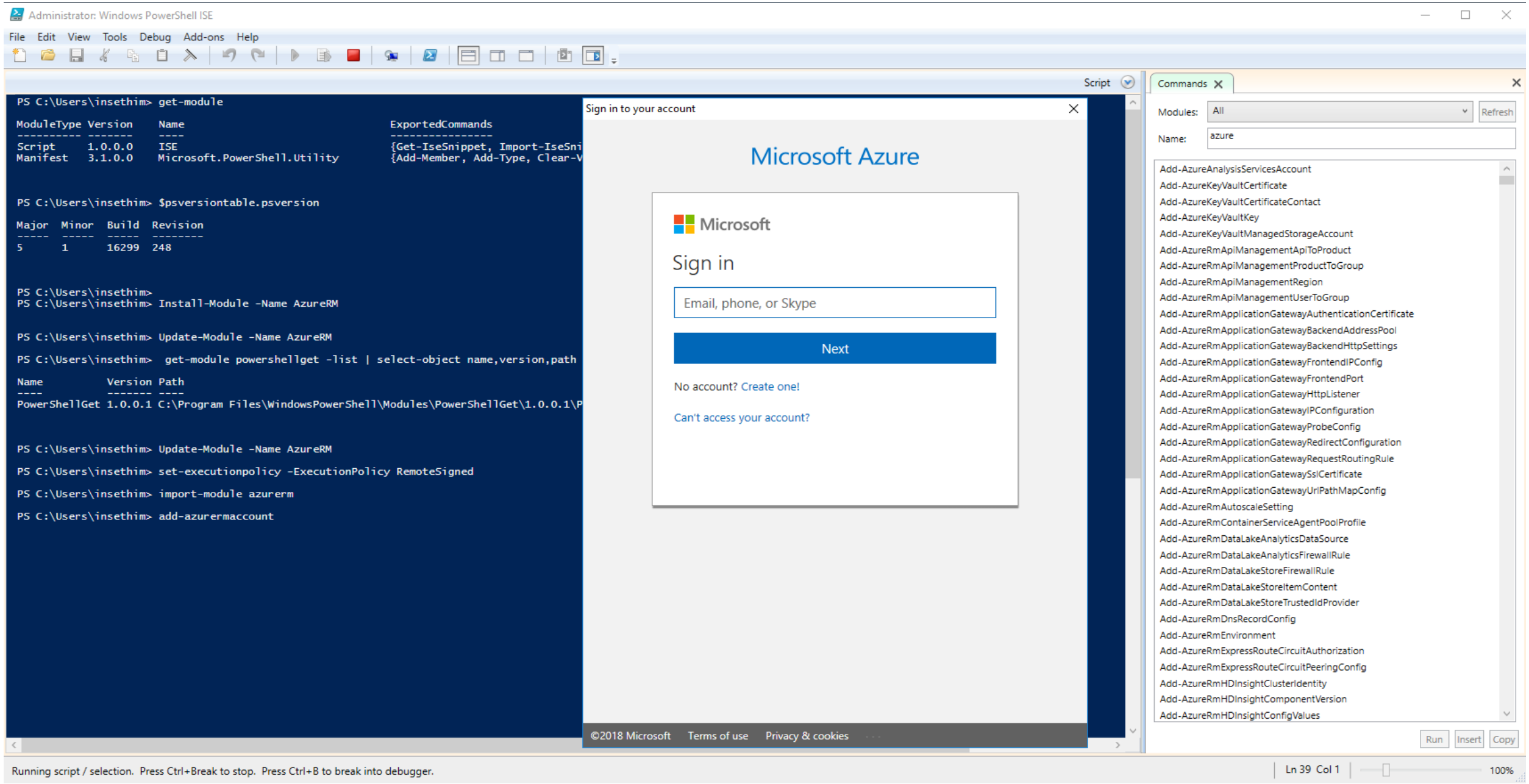
Set the permission for powershell command to run locally : **set-executionpolicy -ExecutionPolicy RemoteSigned**
And then select **"yes for all"**



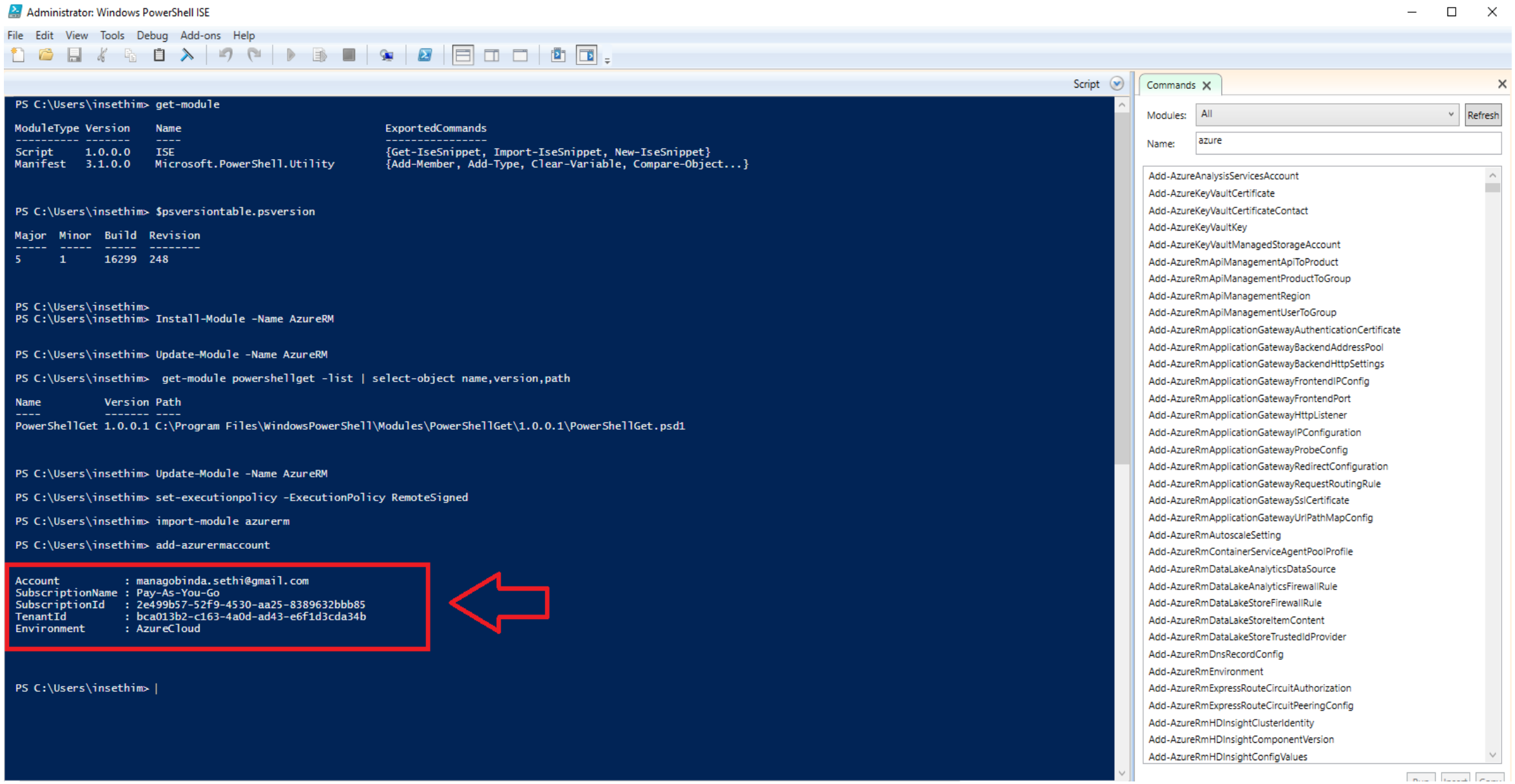
Then type "**import-module azurerm**" command to use azure commands on this PowerShell and after which search for azure command which will show on right hand side of PowerShell



Connect to Azure with an authenticated account: Add-AzureRmAccount



After successful login with authenticated account, we will get below details



Module 1 - Lesson 5 - Managing Azure with Azure CLI

Azure CLI

- From previous lesson we know that Azure PowerShell is available from Linux
- But you may prefer something more Linux native to harmonize with existing tools
- Azure CLI can be used both from Windows and Linux
- Azure first had CLI 1.0 (based on node.js). Also was known as XPLAT-CLI. It supports both Azure classic model and ARM.
- Azure CLI 2.0 (based on Python) is the latest CLI and supports only ARM

Getting Azure CLI 2.0

Linux: https://docs.microsoft.com/en-us/cli/azure/install-azure-cli?view=azure-cli-latest
Windows: https://azurecliprod.blob.core.windows.net/msi/azure-cli-latest.msi

Azure CLI 2.0 install using apt on Linux

```
# Modify the source list to get Microsoft packages
AZ_REPO=$(lsb_release -cs)
echo "deb [arch=amd64] https://packages.microsoft.com/repos/azure-cli/ $AZ_REPO main" | \
sudo tee /etc/apt/sources.list.d/azure-cli.list
# Install Azure CLI 2.0
sudo apt-key adv --keyserver packages.microsoft.com --recv-keys 52E16F86FEE04B979B07E28DB02C46DF417A0893
sudo apt-get install apt-transport-https
sudo apt-get update && sudo apt-get install azure-cli
```

Azure CLI 2.0 authenticate account

Connect to Azure subscription

```
$ az login
```

Module 1 – Lesson 6 - Overview of Azure deployment models

Azure deployment models

1. Azure Classic (ASM) deployment model - traditional approach
2. Azure Resource Manager (ARM) deployment model – latest approach

Azure Resource Manager (ARM)

1. Latest deployment model from Azure
2. A mechanism to handle all Azure “resources” as a related and interdependent parts of a single entity rather than individually
3. Azure Resource Manager provides security, auditing, and tagging features to help manage resources after deployment
4. Resource Manager provides a consistent management layer for the tasks performed through Azure PowerShell, Azure CLI, Azure portal, REST API, and development tools. Use the tool that best works for you and the purpose.

Azure Resource Manager (ARM) – Getting to know the terms

1. **Resource** – A manageable item in Azure. It is the elementary building block for Azure Services. E.g., virtual machine, virtual network, storage account and many more.
2. **Resource Group** – A logical container that holds related resources. Typically for a solution.
3. **Resource Provider** - Think of it as a service that providers the "resource" that can be deployed and managed through Resource Manager. It also offers operations for working with the resources that are deployed. E.g., For a Virtual Machine the provider is Microsoft.Compute . Typical format of the provider is {resourceprovider}/{resource-type}
4. **Tags** – Resource Manager provides tagging feature that can be applied to individual resources or resource groups or subscriptions. It can categorize resources according to requirements for managing or billing or audit, or access. Resources do not need to reside in the same resource group to share a tag
5. **Resource Manager Templates** - A JavaScript Object Notation (JSON) file that defines one or more resources to deploy to a resource group.
6. **RBACs, Policies and Locks**

Azure Resource Manager (ARM) – Resources

- The most elementary building blocks of Azure services
- Resources are provided by "Resource Providers"
- Actions can be performed on resources through resource providers. Done through interfaces or APIs
- Every resource needs to be part of a Resource Group – Resource Groups are created first before resources are created

Azure Resource Manager (ARM) – Resource Groups

- A logical container for resources
- Resources in the resource groups share the same lifecycle and can be managed together
- Permissions can be delegated on the same Resource Group through RBAC
- Costs, Audits, Utilization etc. can be measure on a Resource Group level
- A resource can be only be in one resource Group at a time
- How to allocate resources in a Resource Group is based on what works for you and your organization. Keeping all limits in mind.

Azure Resource Manager (ARM) – ARM Templates

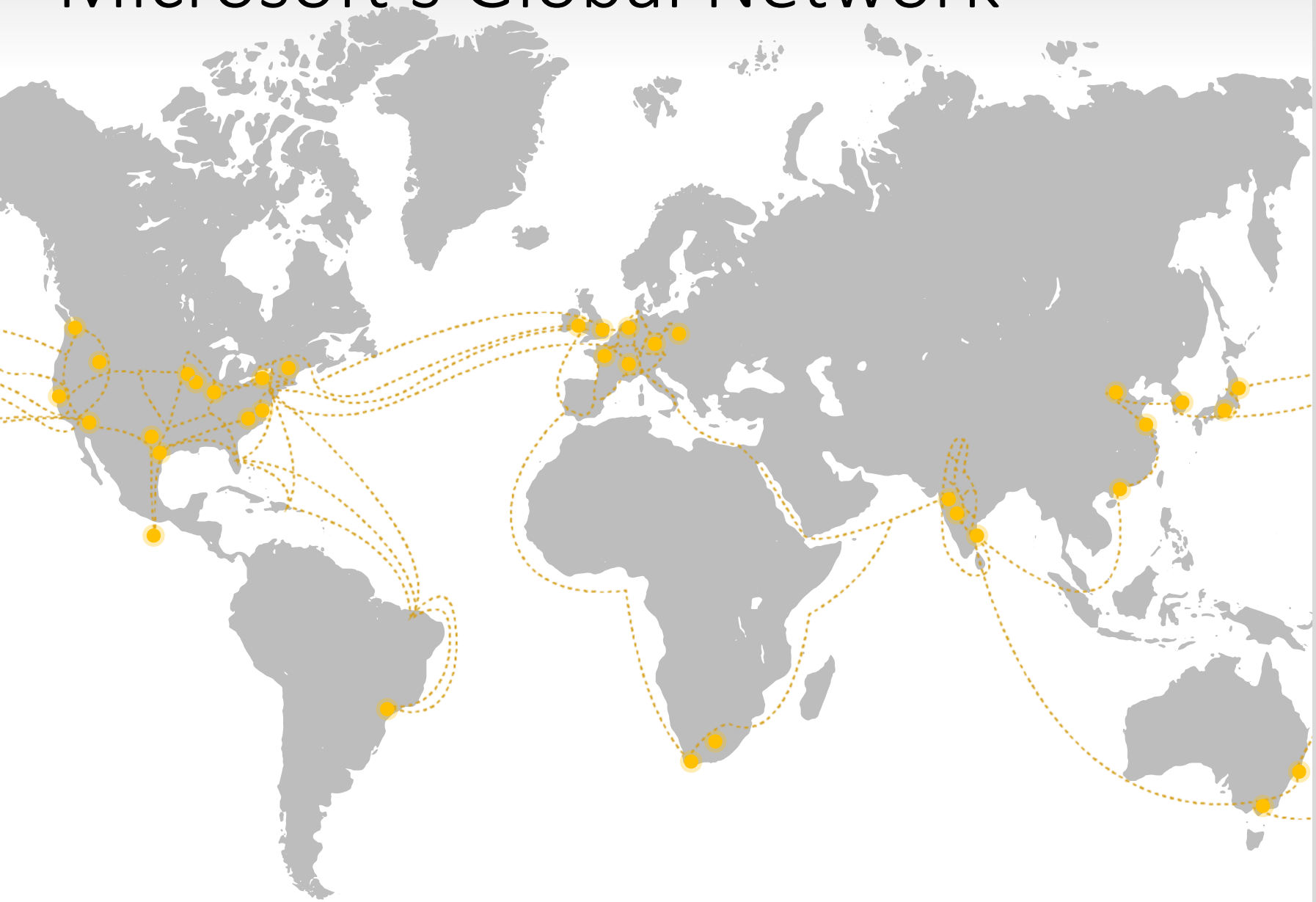
- A JavaScript Object Notation (JSON) file that defines one or more resources to deploy to a resource group. It defines the infrastructure and configuration of the Azure solution
- The template can be used to deploy the resources consistently and repeatedly
- ARM templates have a declarative syntax. “Here is what I want to create” rather than “Go create it this way”.
- When a solution is created from the portal, a deployment template is automatically created

Module 2: Implementing and managing Azure networking

Lesson 1: Overview of Azure networking
Lesson 2: Implementing and managing virtual networks
Lesson 3: Configuring an Azure virtual network
Lesson 4: Configuring virtual network connectivity
Lesson 5: Overview of Azure classic networking

Lab 1: Creating an Azure virtual network by using a deployment template
Lab 2: Creating a virtual network by using Azure PowerShell
Lab 3: Creating a virtual network by using Azure CLI
Lab 4: Create a point-to-site virtual network
Lab 5: Create a VET-to-VNET virtual network
Lab 6: Create a site-to-site virtual network

Microsoft's Global Network



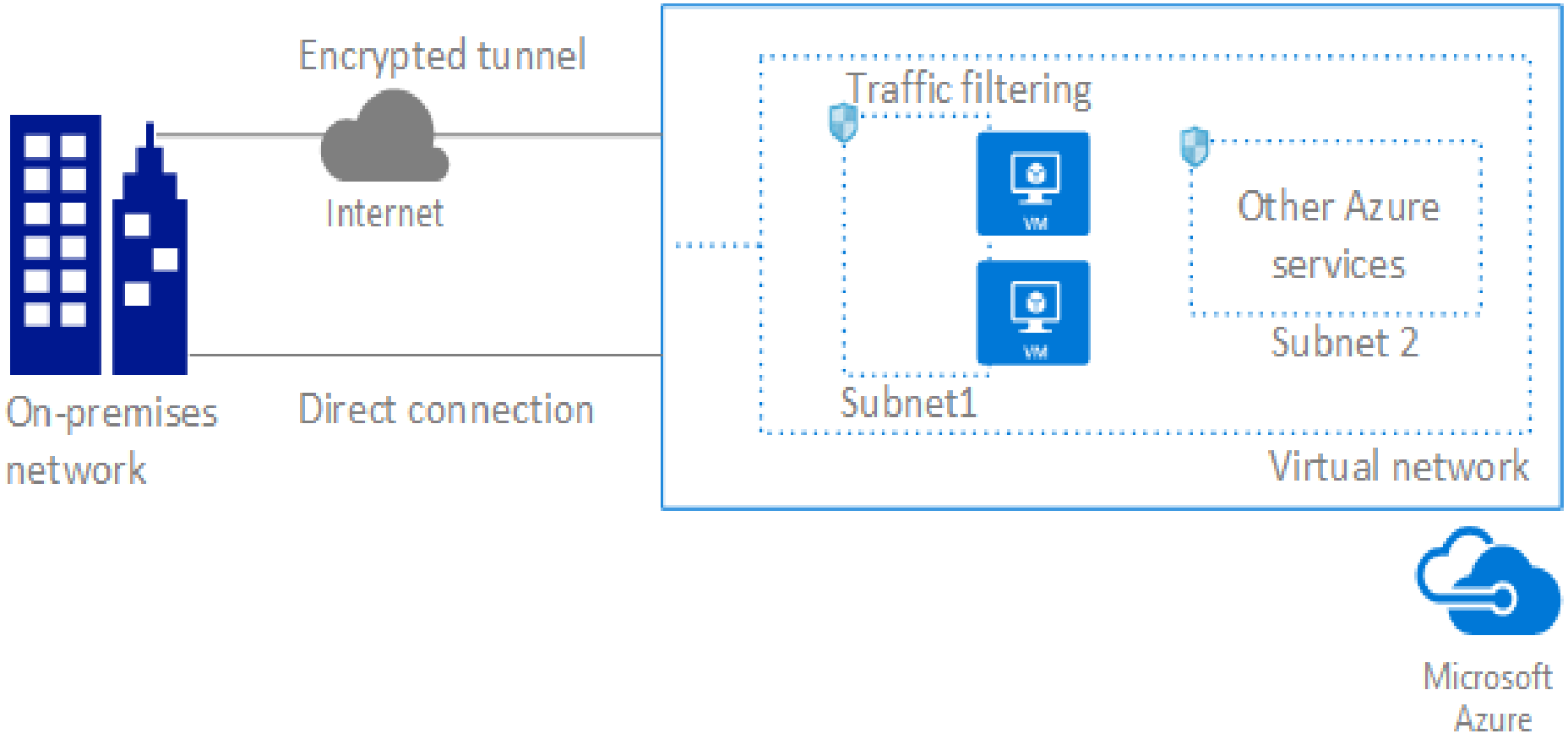
One of the largest networks in the world

- 50 Azure regions
- 8000+ ISP sessions
- 100+ edge sites
- 44 ExpressRoute locations
- 33000 miles of lit fiber

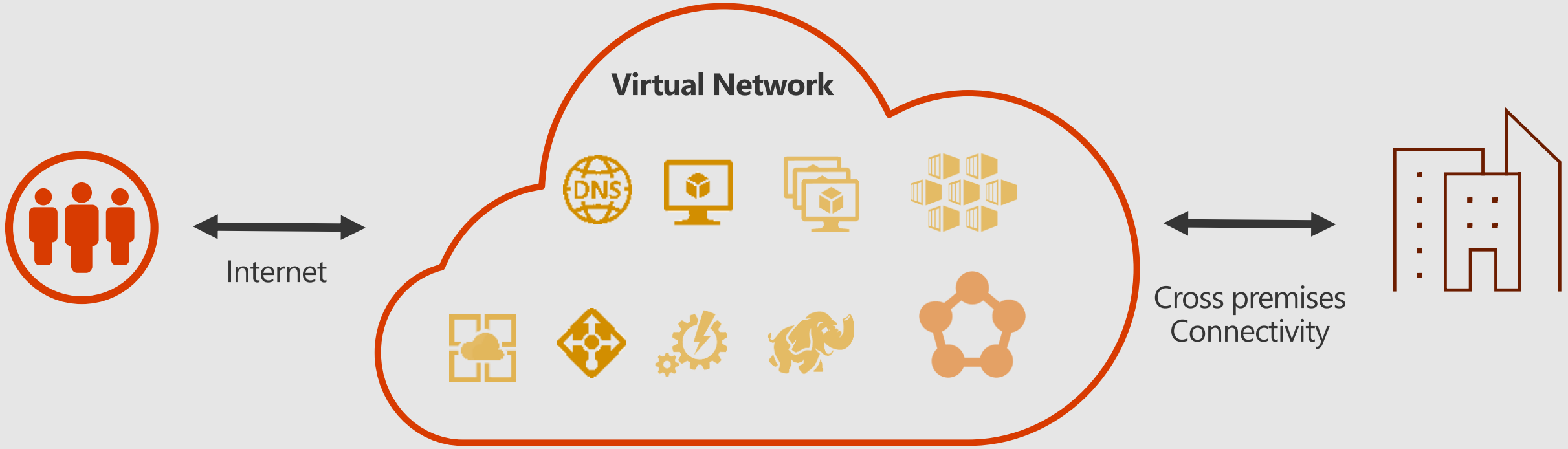
Same network for Bing, Office 365, Skype, XBOX and Azure

Azure Virtual Network

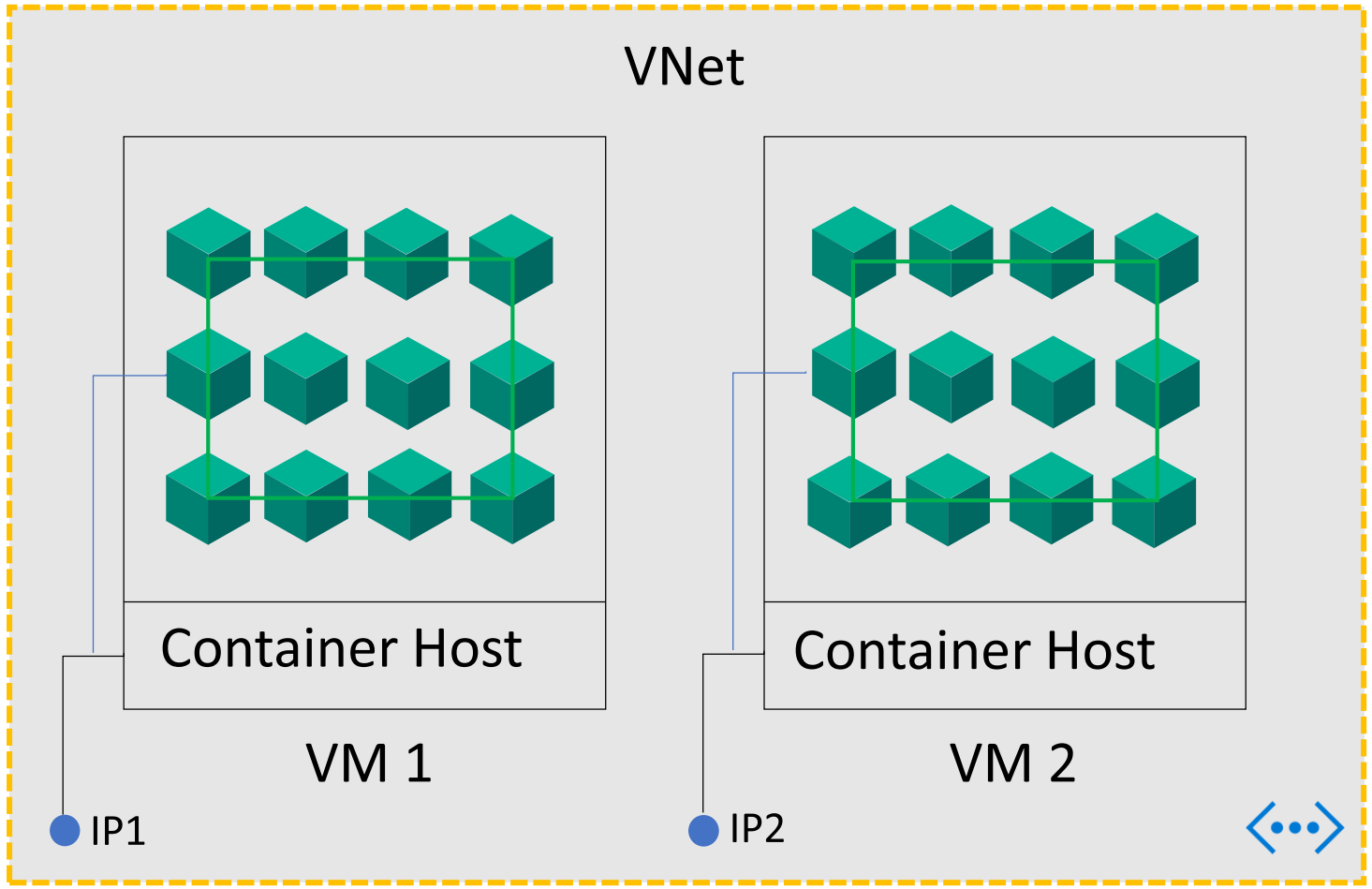
The Azure Virtual Network service enables you to securely connect Azure resources to each other with virtual networks (VNets). A VNet is a representation of your own network in the cloud. A VNet is a logical isolation of the Azure cloud dedicated to your subscription. You can also connect VNets to your on-premises network.



Your Network in Azure



Networking for Containers



Container clusters deployed in a VNet

Containers have local IP addresses

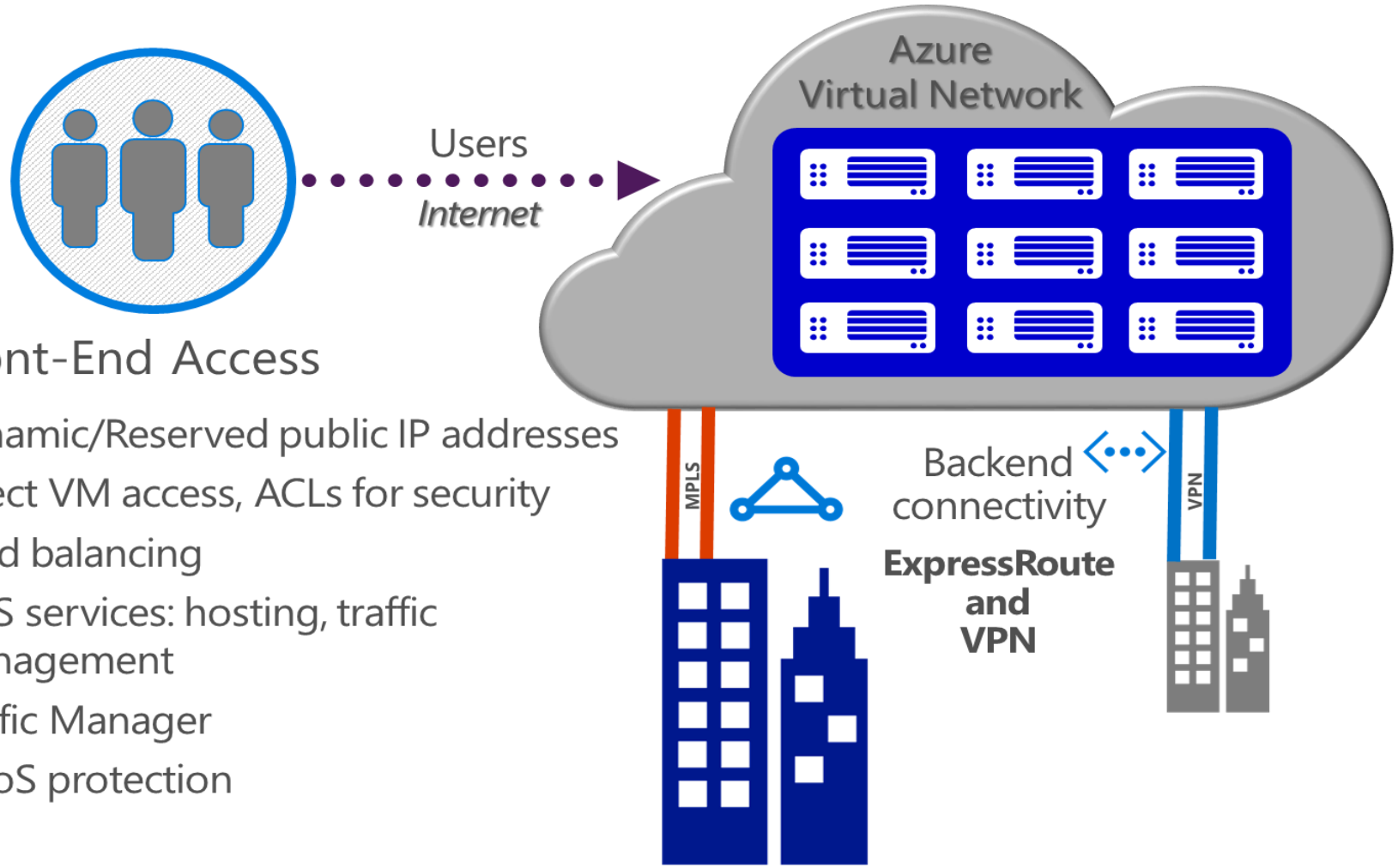
Connectivity between containers in same VM host works

Cross container, cross VM communication has to be through:

- Overlay
- Bridge Mode

Notes: Azure provides the right connectivity for all scenarios. From scalability entry points all over the world; to Virtual Software Defined Networks in Azure; and VPN or MPLS connections which make the Microsoft Network just another WAN site. Microsoft is ready.

The big (network) picture



Front-End Access

Dynamic/Reserved public IP addresses

Direct VM access, ACLs for security

Load balancing

DNS services: hosting, traffic management

Traffic Manager

DDoS protection

Virtual Network

"Bring your own network"

Segment with subnets and security groups

Control traffic flow with user defined routes

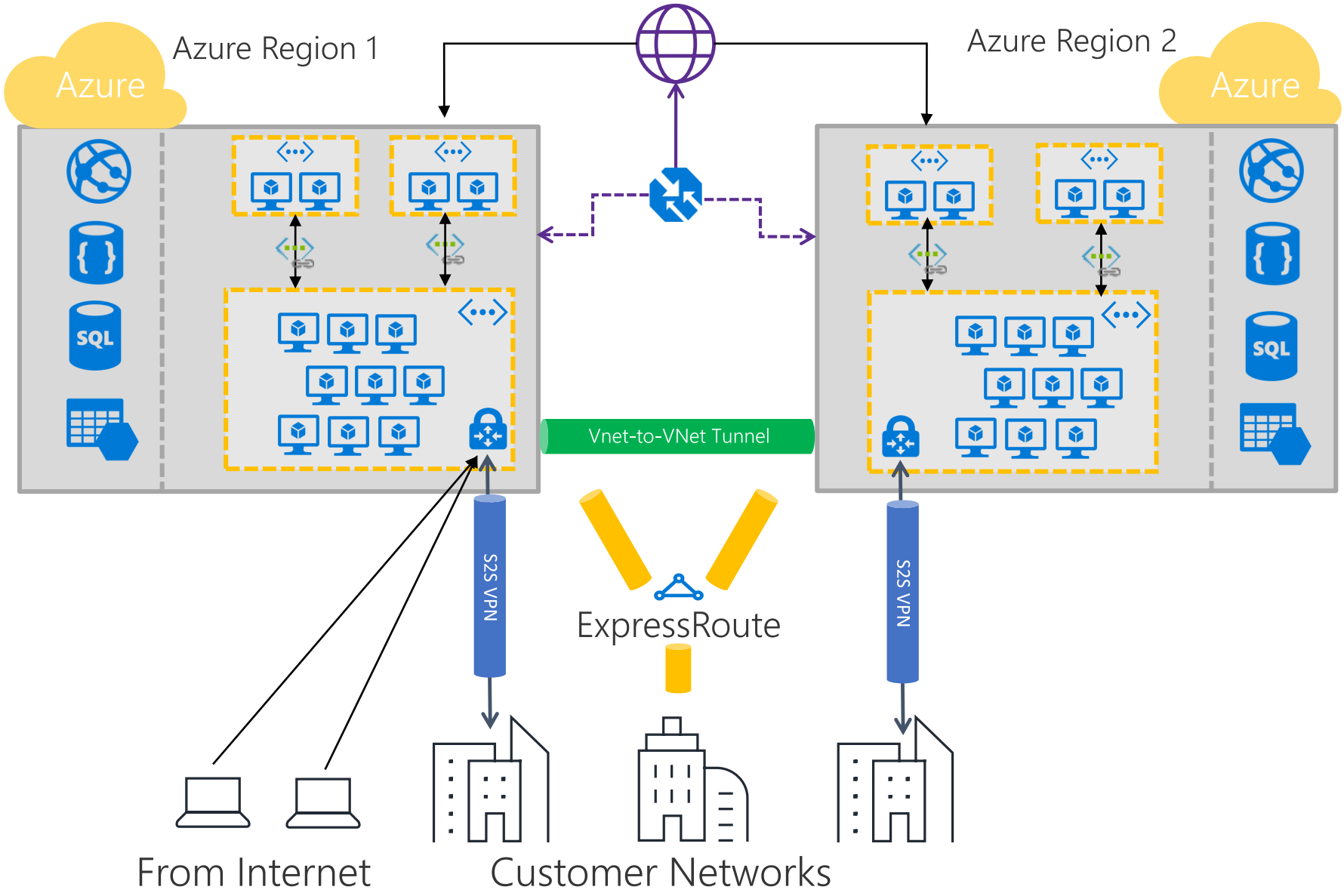
Backend connectivity

Point-to-site for dev/test

VPN Gateways for secure site-to-site connectivity

ExpressRoute for private enterprise grade connectivity

The typical architecture



DNS based traffic distribution across regions

PaaS services for cost efficiency, scale and high availability

IaaS workloads in one or more VNets in a region

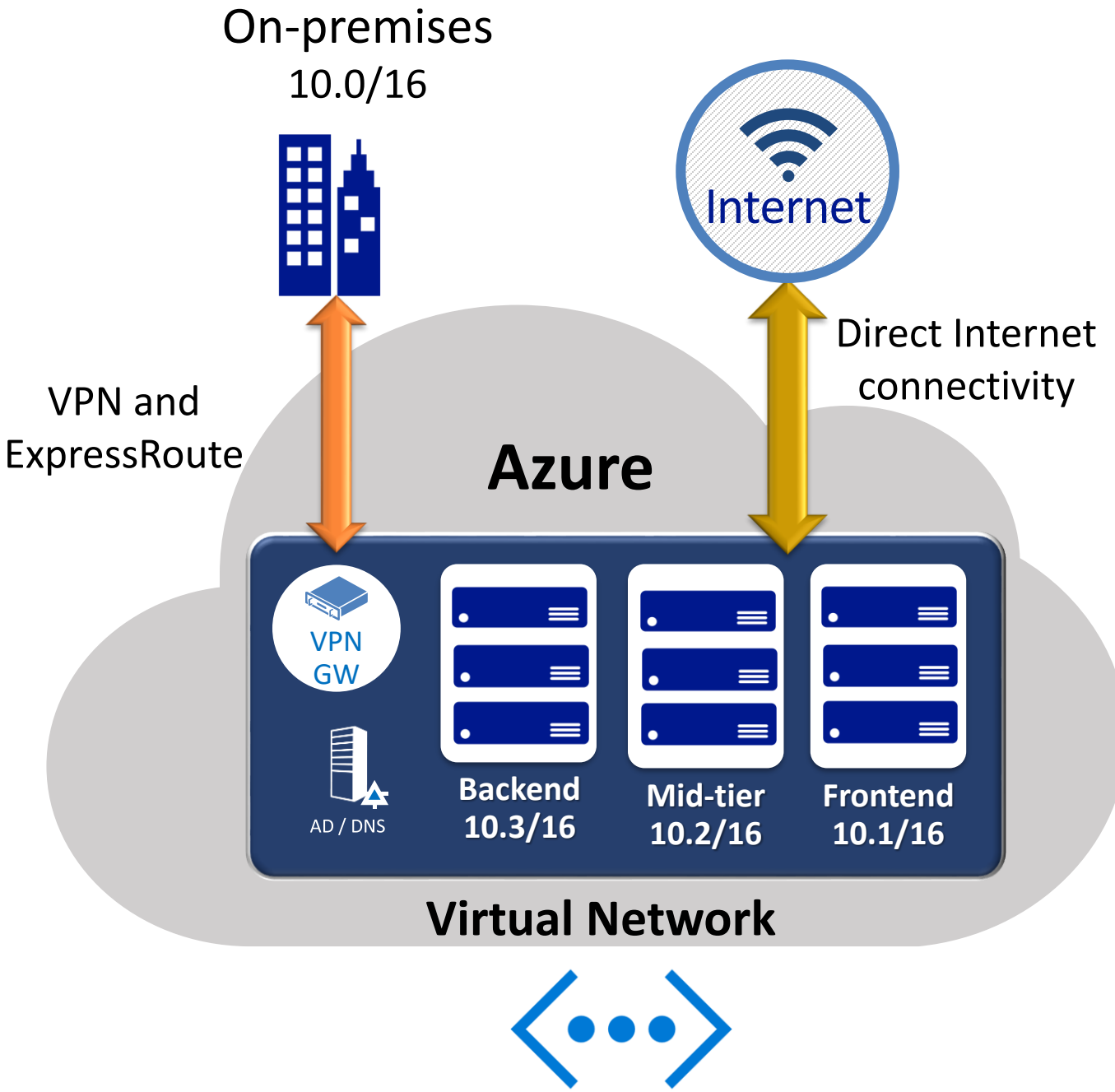
Inter-connected VNets within a region

ExpressRoute/S2S for on-premises connectivity

Vnet-to-Vnet IPSec tunnel to connect Azure regions

Virtual networks

- Bring your own network
- Create subnets with your private or public IP addresses
- Bring your own DNS or use Azure-provided DNS
- Hybrid connectivity with VPNs and/or ExpressRoute

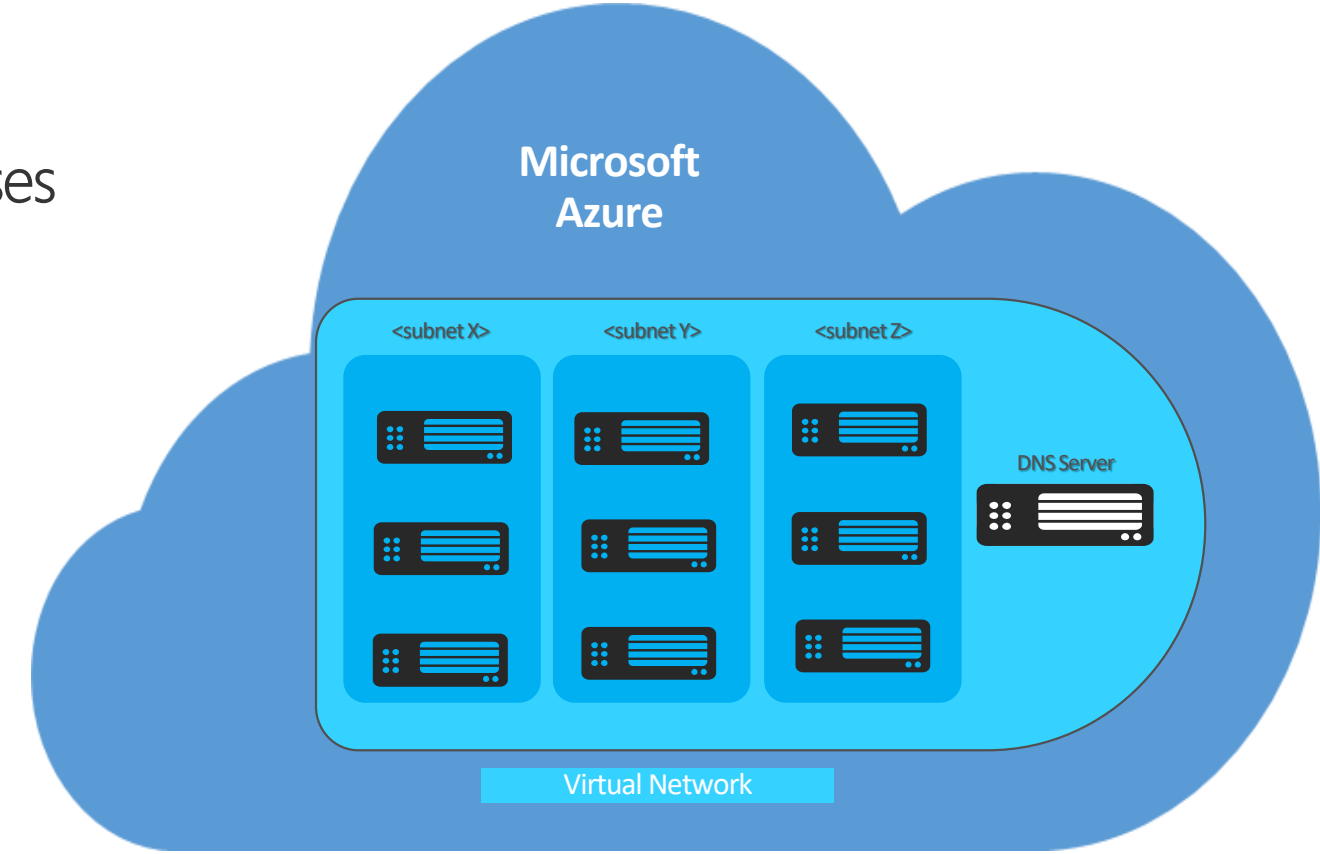


Components of a virtual network

- Address spaces (IP prefixes)
  - The range of IP addresses available in your virtual network
- Subnets
  - Named ranges of addresses assignable to virtual machines and cloud service instances
- DNS servers
  - References to DNS servers that will be assigned to virtual machines or cloud service instances in the virtual network
- Local network configuration
  - Configuration of an on-premises network connected by a site-to-site VPN connection or ExpressRoute

Azure virtual networks and subnets

- Logical isolation with control over network
- Create subnets with your private IP addresses
- Stable and persistent private IP addresses
- Bring your own DNS
- Use Azure-provided DNS
- Secure VMs with input endpoint ACLs and Network Security Groups (NSGs)



- VIP (Virtual IP address)
  - A public IP address belongs to the a machine in a virtual network. It also serves as an Azure Load Balancer which tells how network traffic should be directed before being routed to the VM.
  - It is possible to reserve an IP from the Microsoft pool
- DIP (Dynamic IP address)
  - An internal IP assigned by Microsoft Azure DHCP to the VM
  - Associated automatically with the VM when created
  - It is released when VM is deleted or deallocated (default)
  - It is possible to configure and static IP address
  - You can have more than one DIP per VM (Multi-NIC support)
- ILPIP (Instance Level Public IP)
  - A ILPIP is associated with the VM in addition to the VIP. Traffic to the ILPIP goes directly to the VM and is not routed through the Azure Load Balancer

Internet IP addresses and load balancing

Public IP addresses in Azure

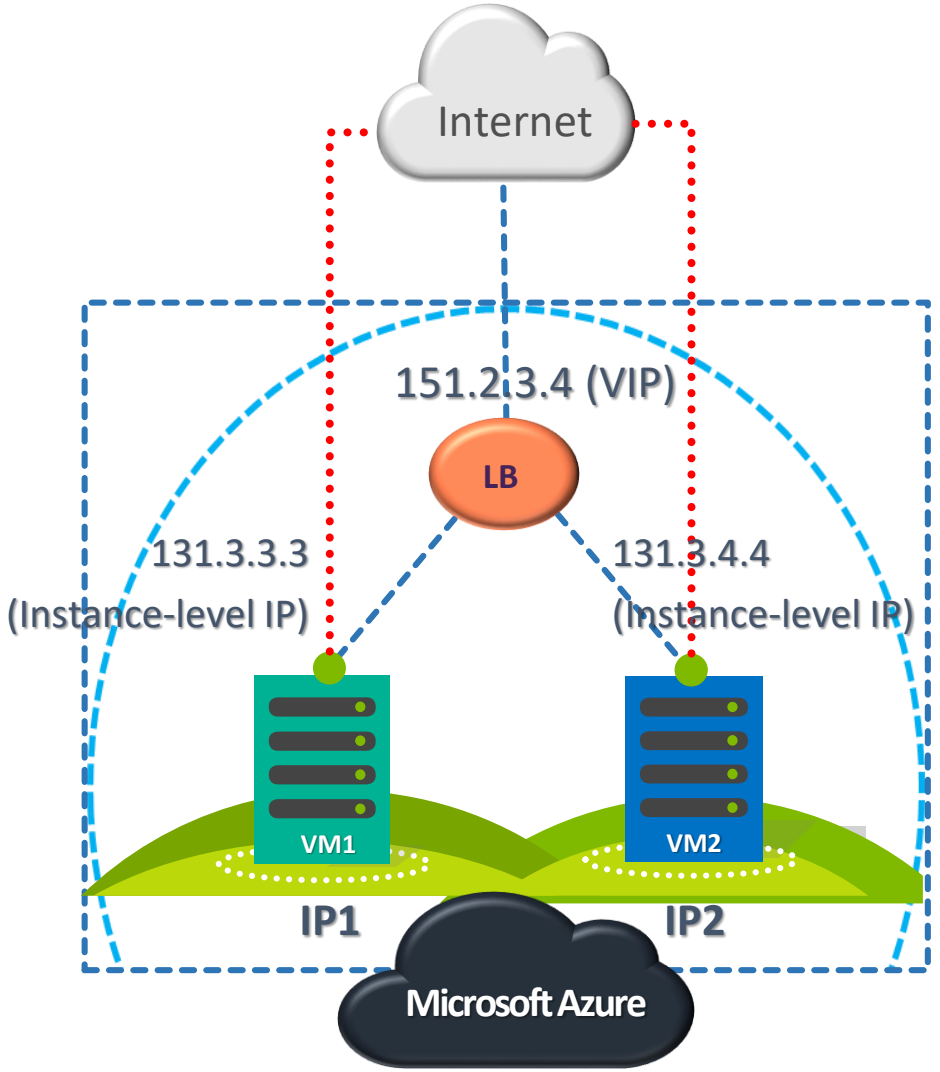
- Can be used for instance (VM) level access or load balancing

Instance-level IP

- Internet IP assigned exclusively to single VM
  Entire port range accessible by default
- Primarily for targeting a specific VM

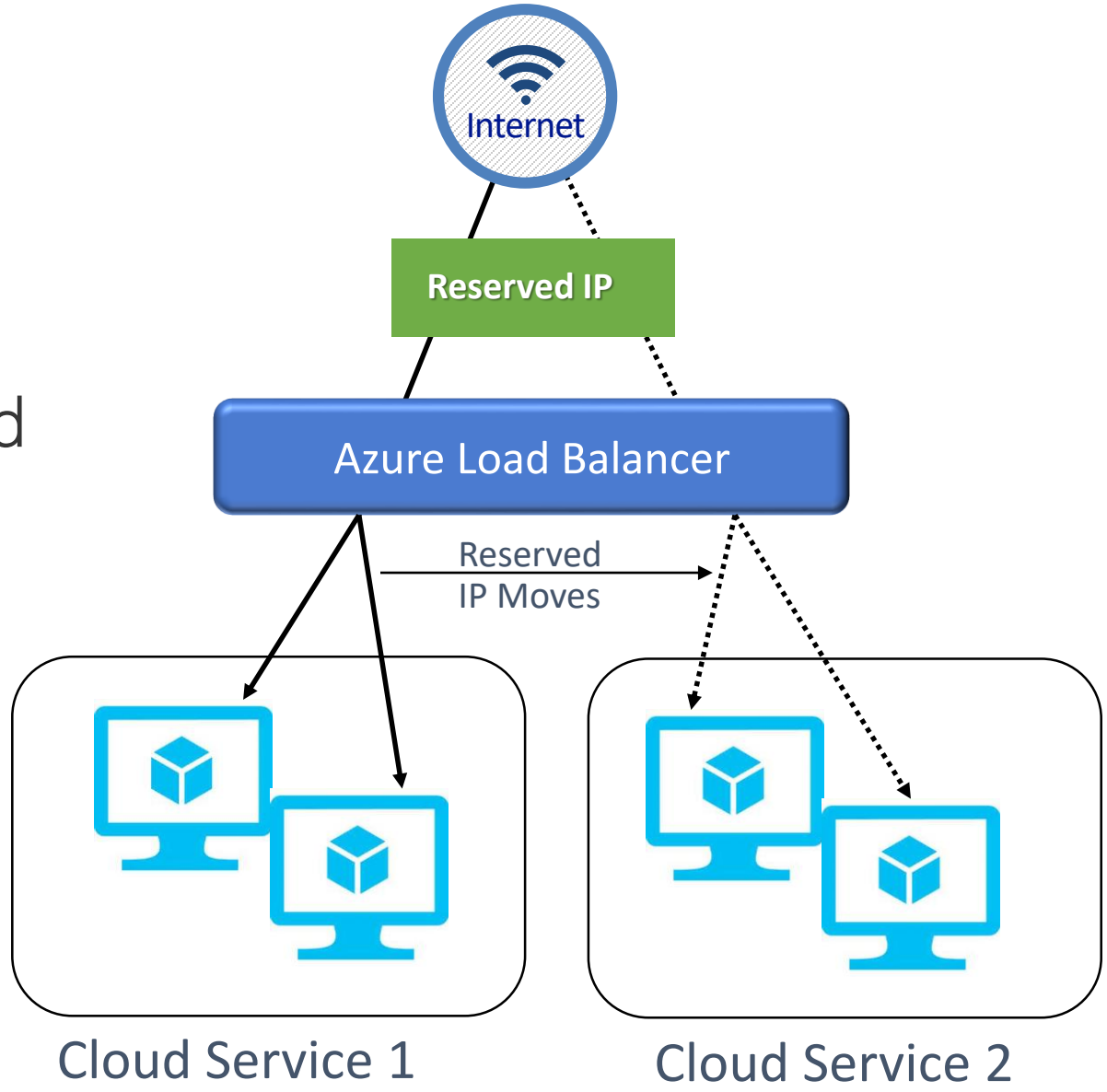
Load balanced IP (VIP)

- Internet IP load balanced among one or more VM instances
- Allows port redirection
- Primarily for load balanced, highly available, or auto-scale scenarios

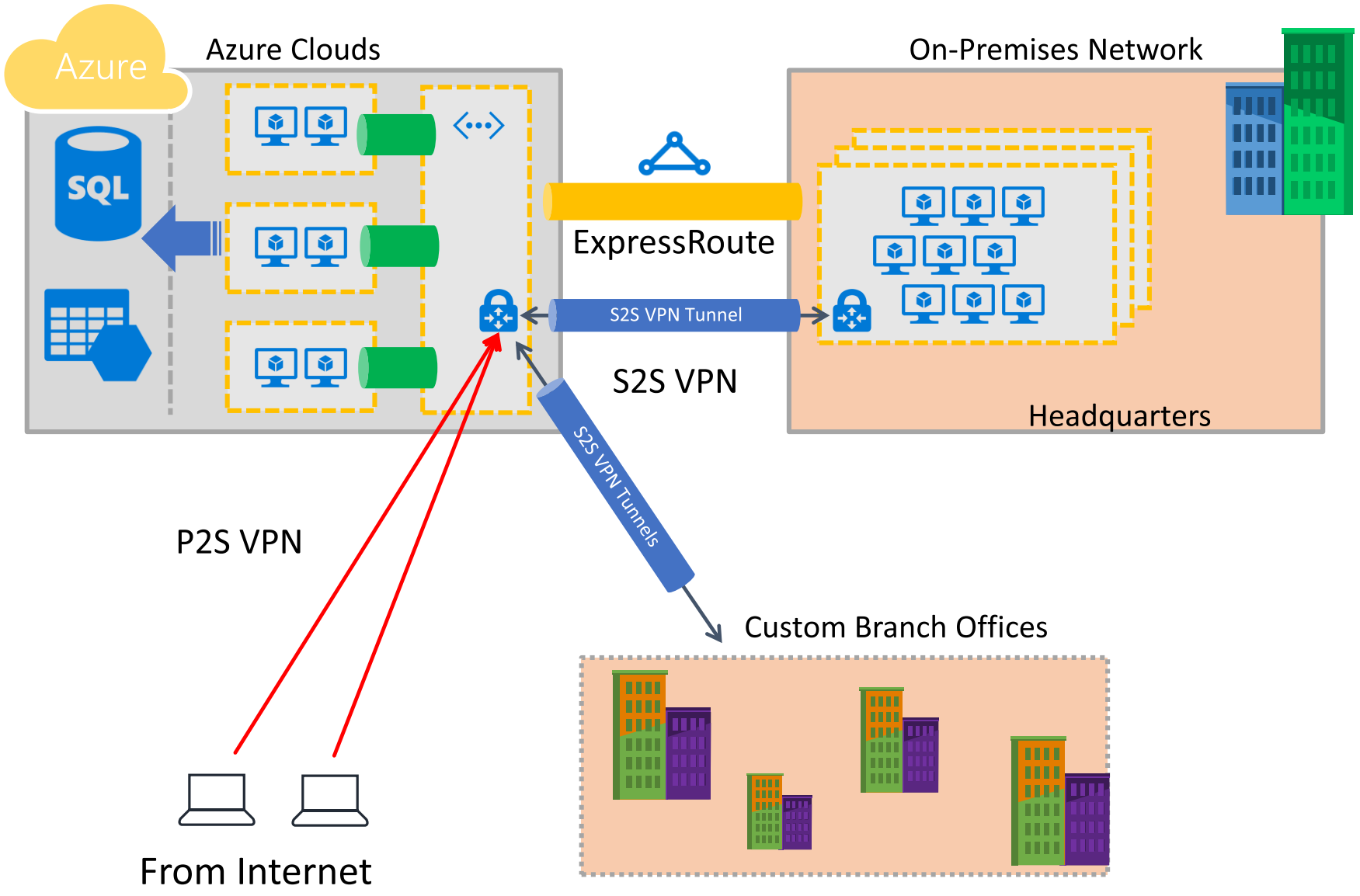


Reserved IPs

- Retain your IP addresses
- IPs on existing services can be reserved
- IPs can be moved between services in seconds



High perf, highly-available cross-premises connectivity



ExpressRoute

- ExpressRoute as primary cross-premises connectivity
- Multiple circuits for redundancy & better routing
- ExpressRoute-VPN co-existence for highly available, redundant paths

S2S VPN

- S2S VPN over Internet for remote branch locations
- BGP & active-active configuration for HA and transit

P2S VPN

- P2S VPN for mobile users & developers to connect from anywhere with macOS & Windows
- AD/radius authentication for enterprise grade security

New Features – ExpressRoute & VPN

New

Performance

- 6x VPN throughput
- Improved UltraPerf ExpressRoute gateways (latency)

Monitoring

- Azure Monitor for ExpressRoute & VPN
- OMS Network Performance Monitoring for ExpressRoute
- Network Watcher for VPN
- Resource Health Check for ExpressRoute & VPN

Security & compliance

- Custom IPsec/IKE policy for VPN

ExpressRoute Peering

- Merging Microsoft & Azure Public Peering

Enterprise grade P2S

- macOS support for P2S
- AD (Radius)-based authentication

Routing & connectivity

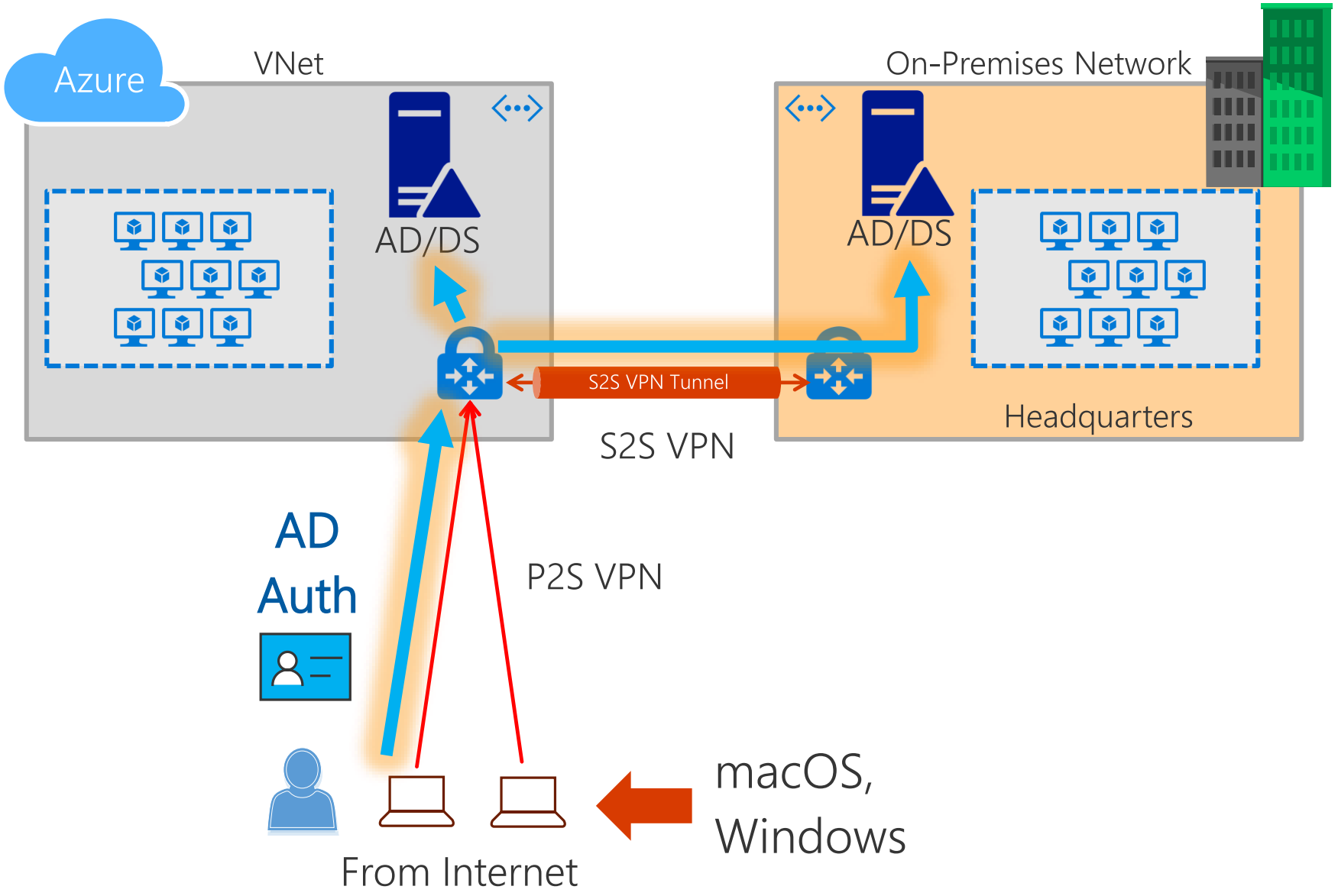
- ExpressRoute IPv6 for O365 & Azure public services
- ExpressRoute route filter for Microsoft peering – selected services
- ExpressRoute connection weight
- Multi-site for policy-based firewalls
- VPN BGP routes & peering status

New VPN Gateway SKUs – 6x faster!

SKU	Workload	Throughput*	S2S/V2V	P2S	SLA
VpnGw1	Production	**650 Mbps**	Max. 30	128	99.95%
VpnGw2	Production	**1 Gbps**	Max. 30	128	99.95%
VpnGw3	Production	**1.25 Gbps**	Max. 30	128	99.95%
Basic	Dev/Test	100 Mbps	Max. 10	128	99.9%

- Scenarios
  - High throughput, hybrid workload over VPN tunnels
  - Failover from ExpressRoute circuits to S2S VPN tunnels
  - P2S for dev/test connectivity from anywhere
- Roadmap
  - Higher throughput, more tunnels, easier migration

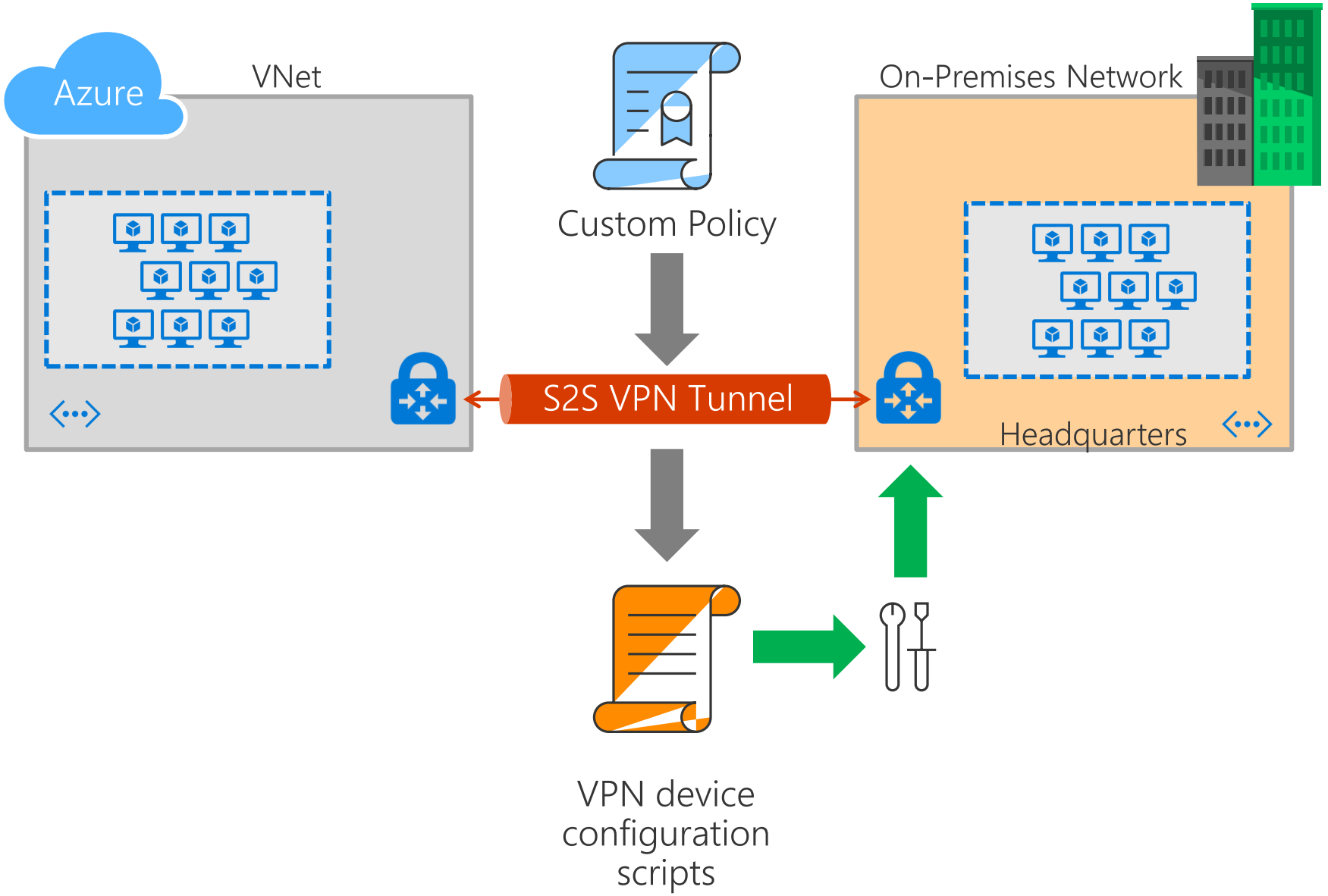
P2S for macOS & AD authentication



P2S VPN

- P2S VPN for mobile users & developers to connect from anywhere
- Now supporting macOS and Windows
- AD/Radius authentication for enterprise grade identity solution

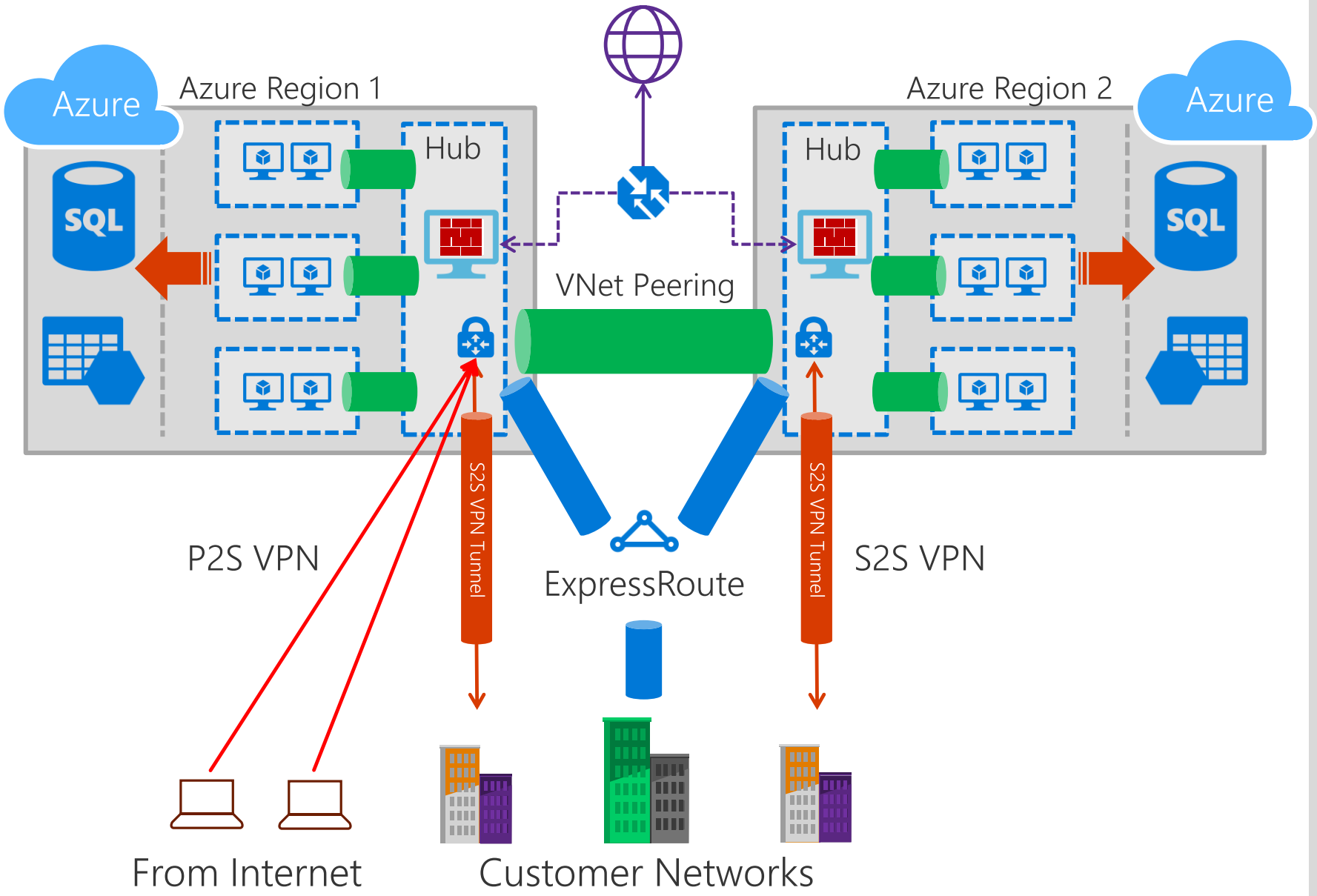
VPN download device scripts



S2S VPN

- Apply custom IPsec/IKE policy for compliance (encryption/integrity, PFS, DH)
- Download VPN device scripts for seamless configurations

Routing!



Can a hub talk to another hub?

- VNet Peering, ExpressRoute, or VPN

Can a spoke talk to another spoke in the same hub?

Can a spoke talk to another spoke in a different hub?

- Global VNet Peering, ExpressRoute, or VPN

Would Service Endpoint access get forced tunneled?

Would BGP advertised peered VNets prefixes to on-premises?

Would it take ExpressRoute or VPN to the on-premises networks?

- Multiple ExpressRoute circuits
- Co-existence
- VPN Active-active

Connectivity options and hybrid offerings

Cloud		Customer	Segment and workloads
	Internet connectivity		**Consumers** • Access over public IP • DNS resolution • Connect from anywhere
	Secure point-to-site connectivity		**Developers** • POC efforts • Small scale deployments • Connect from anywhere
	Secure site-to-site VPN connectivity		**SMB, Enterprises** • Connect to Azure compute
	ExpressRoute private connectivity		**SMB and Enterprises** • Mission critical workloads • Backup/DR, media, HPC • Connect to all Azure services

Why Load Balancing?

- Define well known endpoints for your customers
- Assure, scale, and secure your applications
- Simplify scenarios with fully managed products

Load Balancer

aka.ms/lbpreview

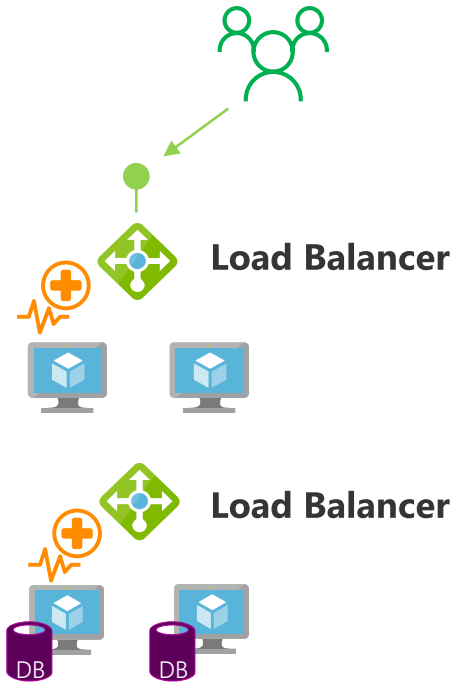
Traffic Manager

aka.ms/trafficmanager

Application Gateway

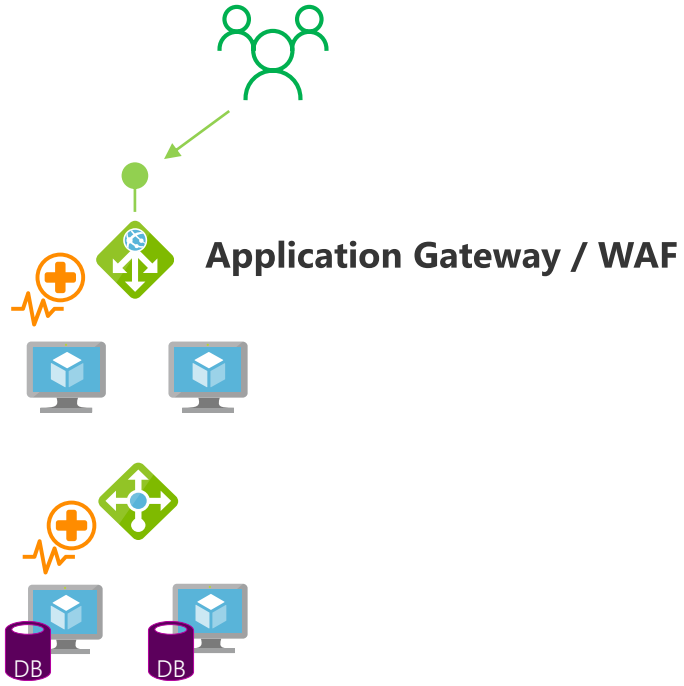
aka.ms/appgw

Load balancing scenarios



Scalable

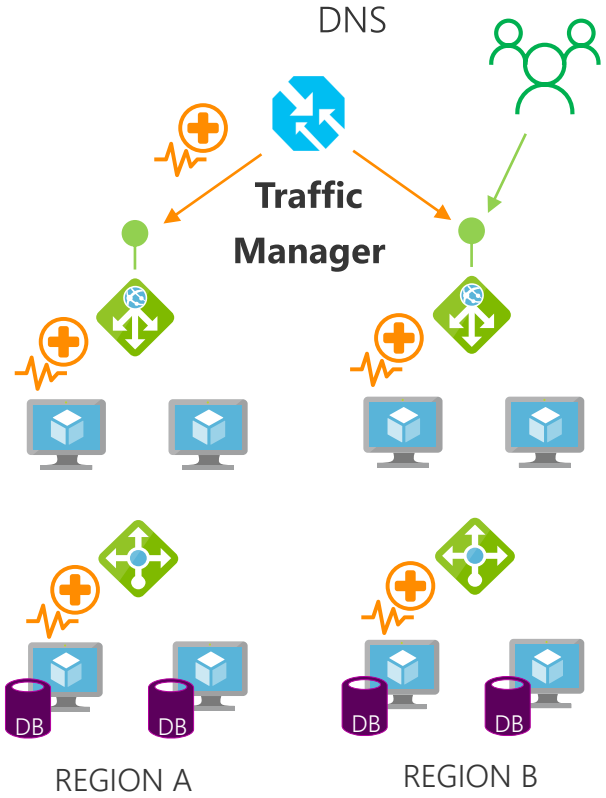
Redundant



Scalable

Redundant

+ Protection



+ Global

with multiple

Azure Regions

What's New?

Load Balancer

New Load Balancer, SKUs

Pool up to 1000 instances in VNet, incl. 1000 instance VMSS

Multiple VMSS

Zone-redundant data path with single IP

Zonal frontends

Cross-zone load balancing

Cross-zone VMSS

HA Ports for NVAs and more

Advanced analytics (Traffic Counters, Per Endpoint health probe status, Continuous in-band data plane health, Inbound connection attempts, Outbound connections)

Application Gateway

SSL Policy with cipher suites

Redirection

Multitenant (WebApp) backend

Health probe enhancements

Multiple VMSS

Path override

Idle timeout and domain label

WAF: OWASP ModSecurity CRS 3.0

WAF: Rule configurability

WAF ASC and OMS log analytics

Traffic Manager

Real User Measurements

Traffic Flow

EDNS Client Subnet

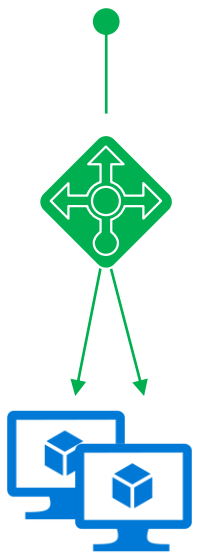
Azure Monitor Metrics

Fast Failover

Geographic Routing

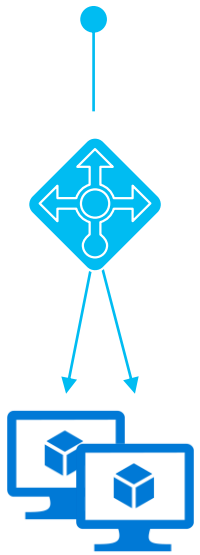
Load Balancer

3 Fundamental Scenarios



Public IP Address

Public Load Balancer



Private IP Address

Internal Load Balancer

Public IP Address

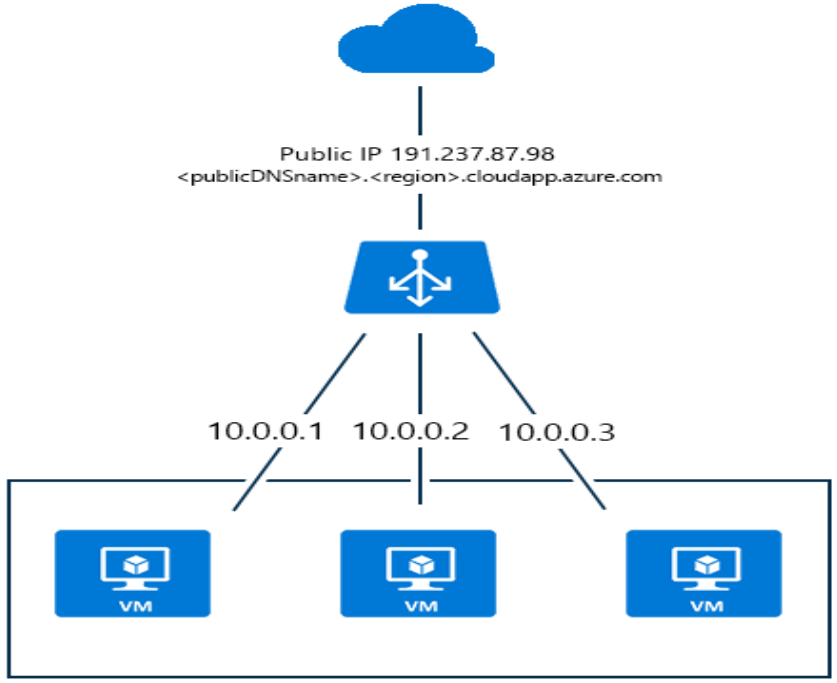
Outbound connections

- Part of the Azure SDN stack
- High performance & low latency for all TCP & UDP applications
- Flow-based Load Balancing with Health Probing
- Inbound NAT rules
- Outbound Connections

Azure Load Balancer

- Azure Load Balancer delivers high availability and network performance to your applications. It is a Layer 4 (TCP, UDP) load balancer that distributes incoming traffic among healthy instances of services defined in a load-balanced set

1.Internet-facing load balancing: Load balance incoming Internet traffic to virtual machines.



- **Front-end IP configuration** - contains public IP addresses for incoming network traffic.
- **Back-end address Pool** - contains network interfaces (NICs) for the virtual machines to receive network traffic from the load balancer.
- **Load balancing rules** - contains rules mapping a public port on the load balancer to port in the back-end address pool.
- **Inbound NAT Rules** - contains rules mapping a public port on the load balancer to a port for a specific virtual machine in the back-end address pool.
- **Probes** - contains health probes used to check availability of virtual machines instances in the back-end address pool.

2.Internal load balancing: An Azure internal load balancer (ILB) provides network load balancing between virtual machines that reside inside a cloud service, or a virtual network with a regional scope

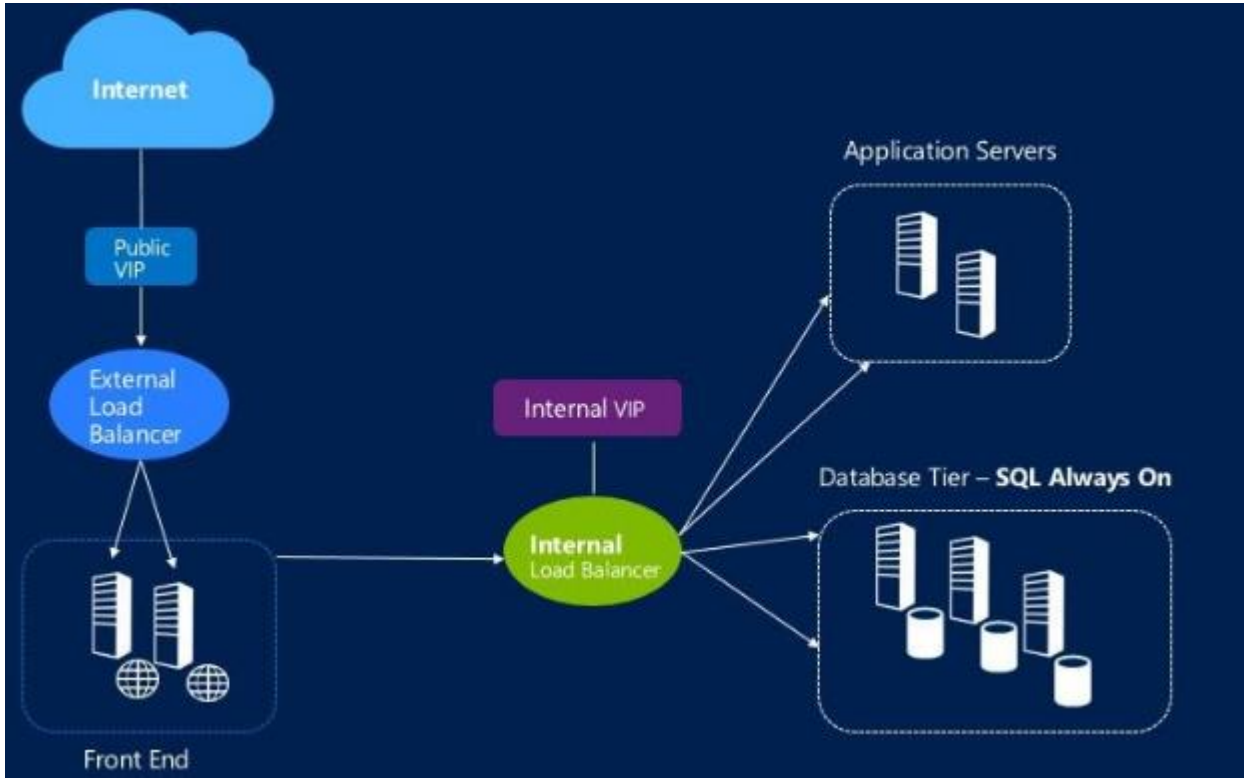
Front end IP configuration - will configure the private IP address for incoming network traffic

Backend address pool - will configure the network interfaces which will receive the load balanced traffic coming from front end IP pool

Load balancing rules - source and local port configuration for the load balancer.

Probes - configures the health status probe for the Virtual Machine instances.

Inbound NAT rules - configures the port rules to directly access one of the Virtual Machine instances.



Application Gateway

Application Gateway

Level 7 routing

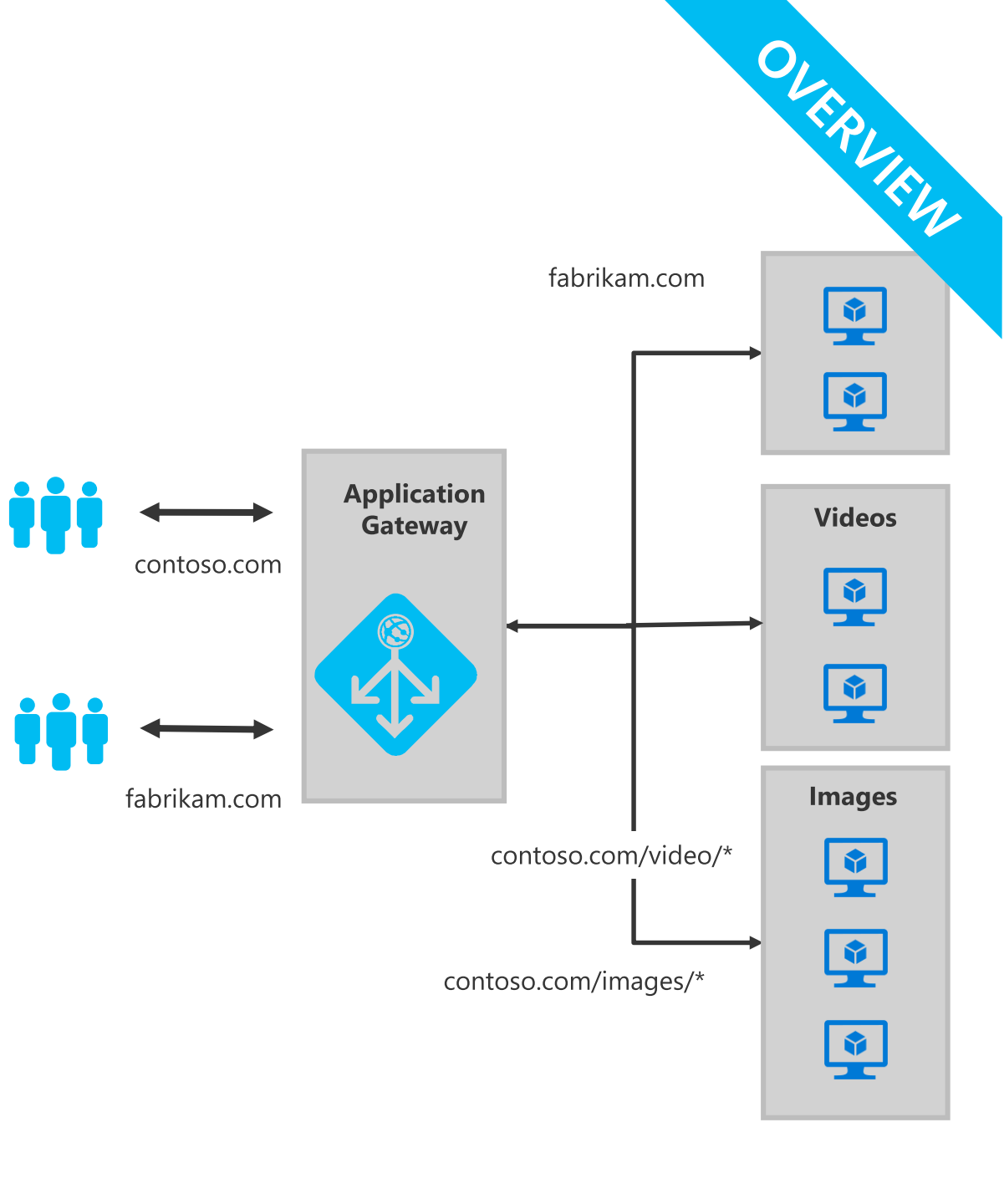
- HTTP round robin
- Cookie based session affinity
- Multi-site hosting
- URL based routing

Security

- SSL termination
- SSL Policy (protocol version and cipher)
- End to end SSL
- Web Application Firewall

Diagnostics and probes

- Rich diagnostics including access and performance logs, WAF logs, backend health log
- Custom health probes



Enhanced connectivity

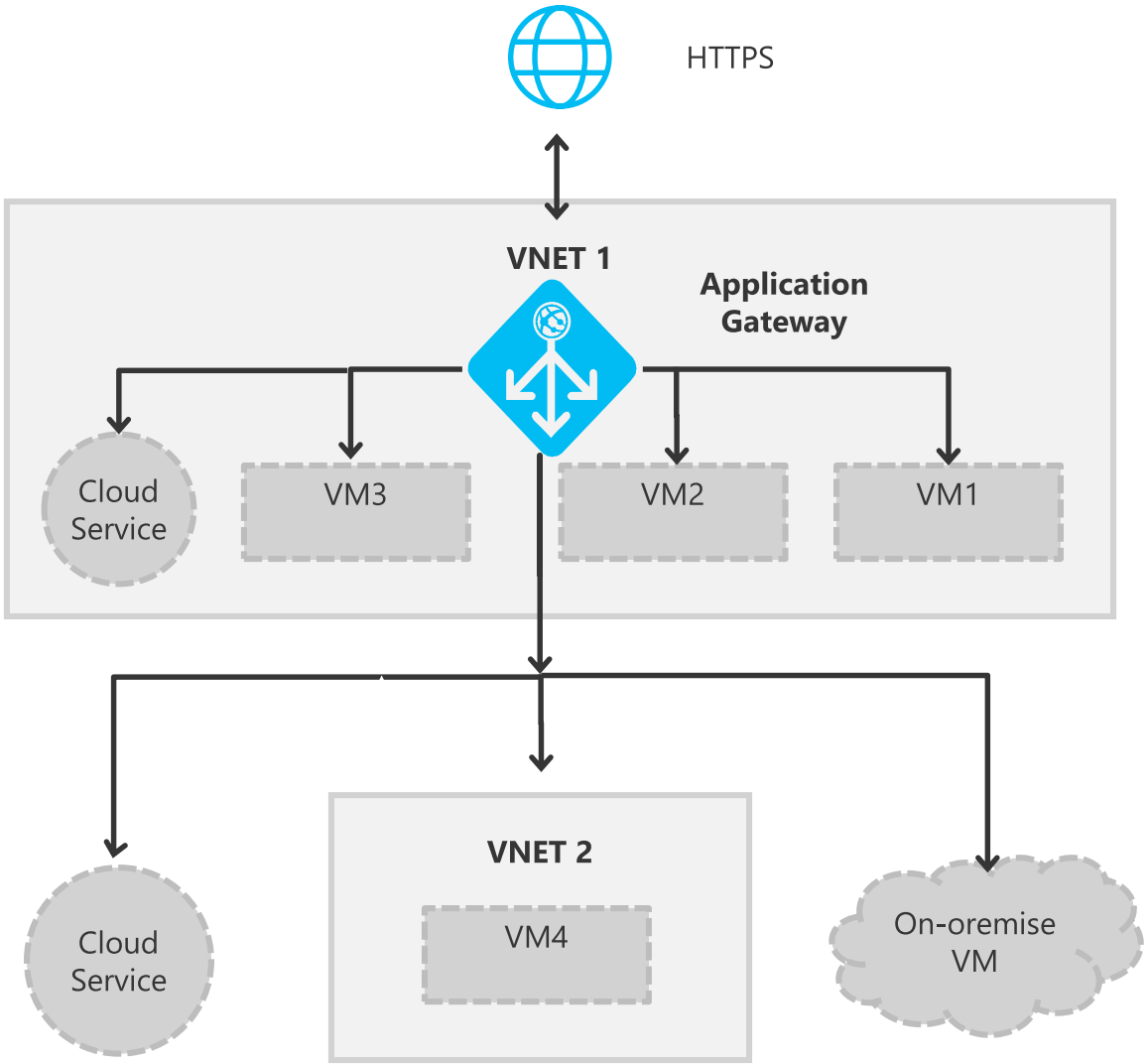
Round robin HTTP load distribution

Backend choices

- VMs via NICs
- Internal IP
- External Public IP
- VMSS
- Azure Web Apps
- Containers

Connectivity Options

- VMs in same VNet
- VMs across connected VNets
- Cloud services
- Hybrid connectivity to on premises VMs
- External servers



Application Gateway redirection

NEW

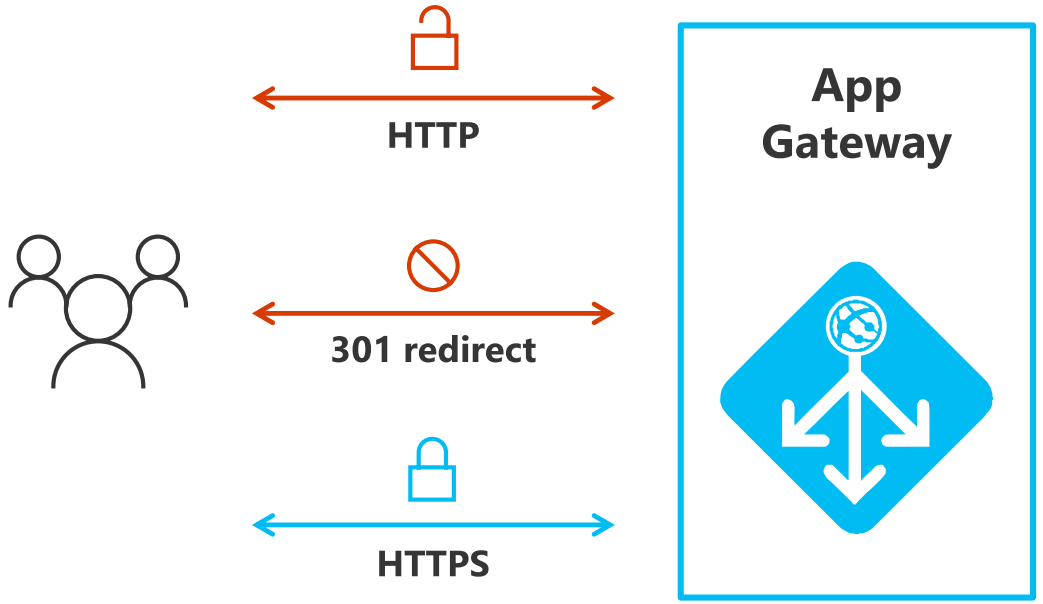
Redirect HTTP to HTTPS

- Backend pool redirection no longer required
- Global redirection for whole site
- Redirect on a specific site path
- Integrated with URL Path Maps

Redirect Listener to listener

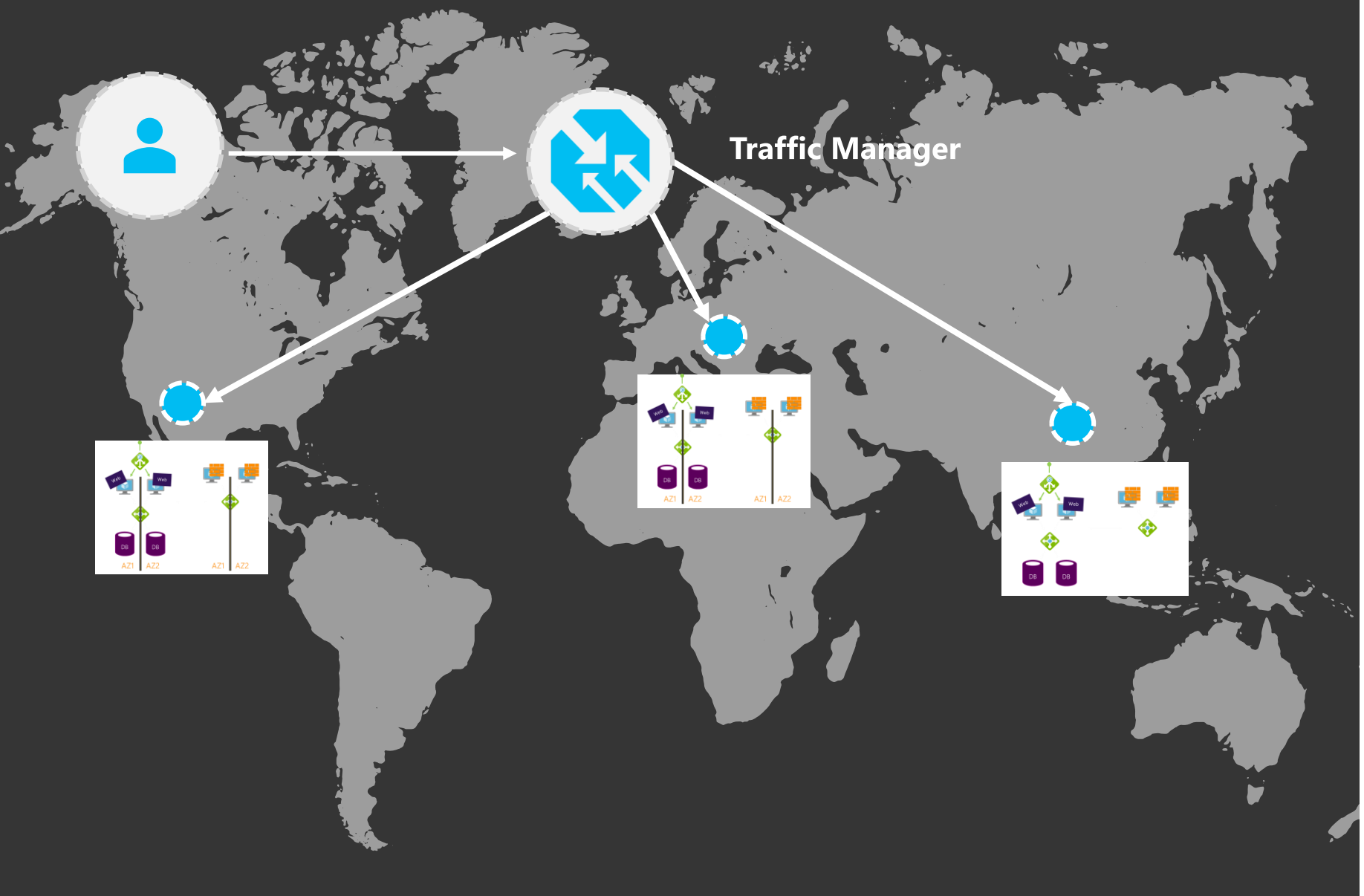
- .com to .org redirection
- One sub domain to another subdomain redirection

Redirect to external site



Traffic Manager

Global Resiliency & Performance with DNS



Easy Onboarding

Multiple Routing Methods

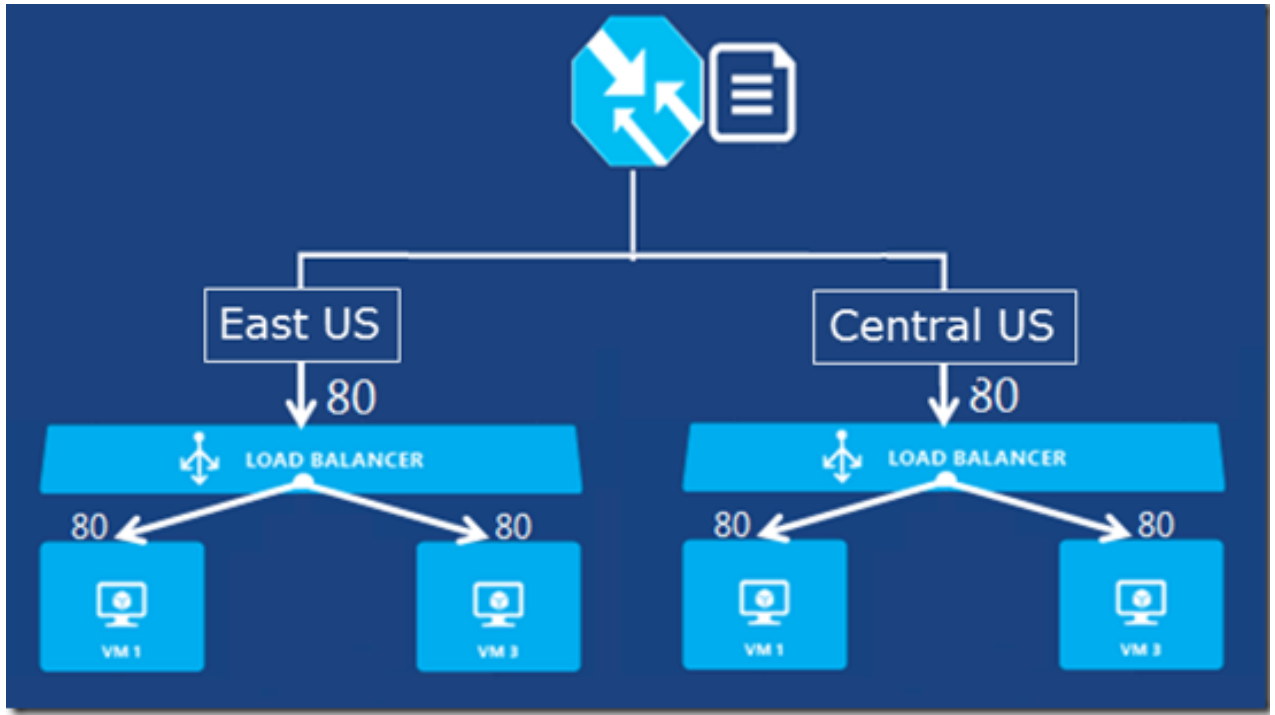
Endpoint monitoring

High resiliency

Traffic Manager

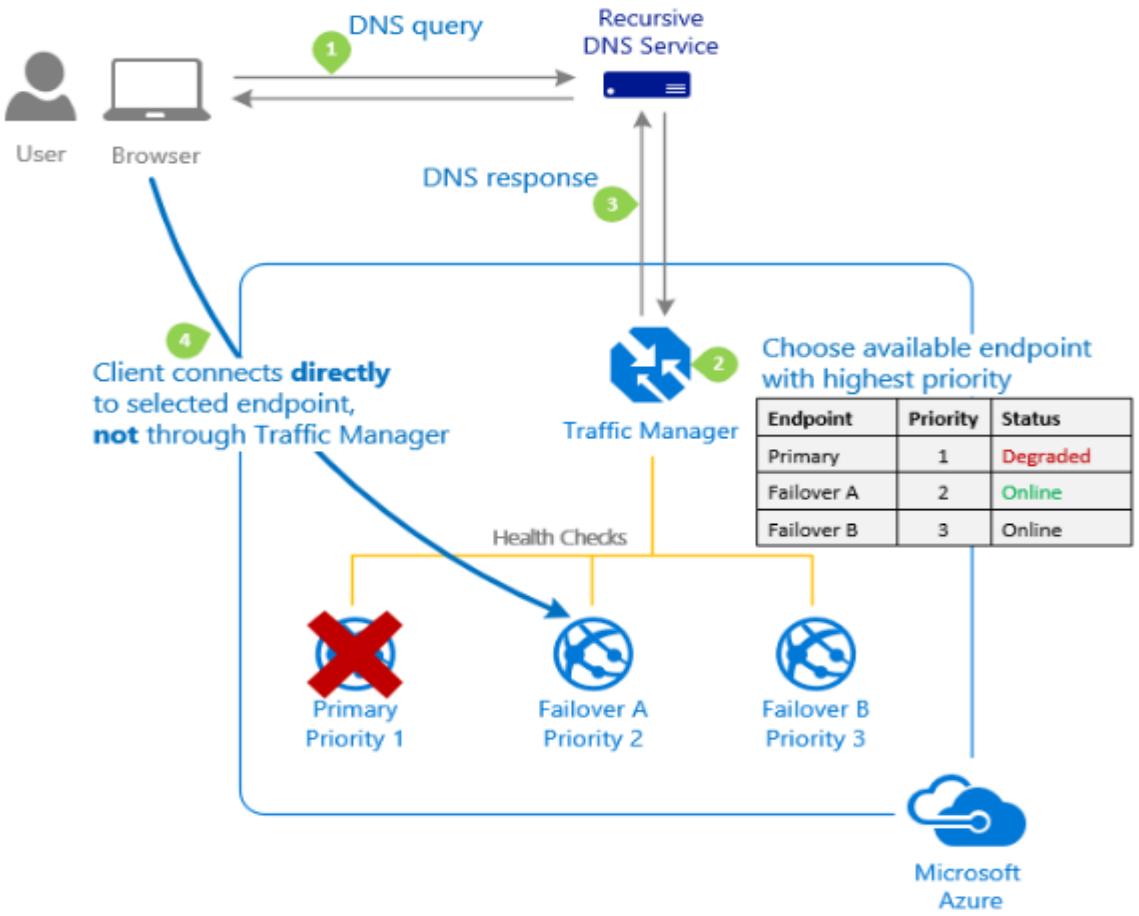
Traffic Manager balances the traffic load of services hosted in Azure. The routing policy is defined by the client and traffic to the services hosted in Azure is redirected according to set policies. Traffic manager is a DNS-based service. Thus, it will improve the availability and performance applications

Traffic Manager uses the Domain Name System (DNS) to direct client requests to the most appropriate endpoint based on a traffic-routing method and the health of the endpoints

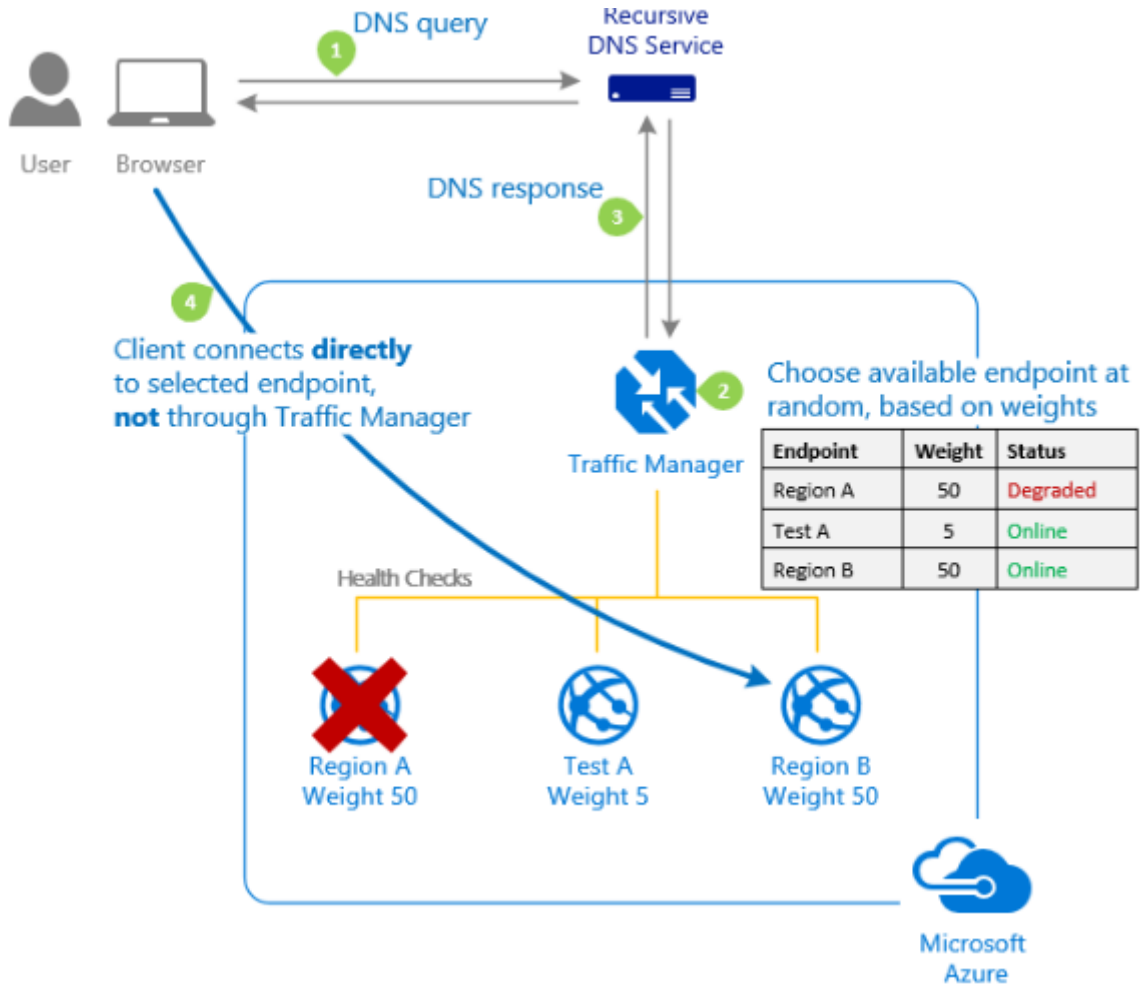


Traffic Manager Rules

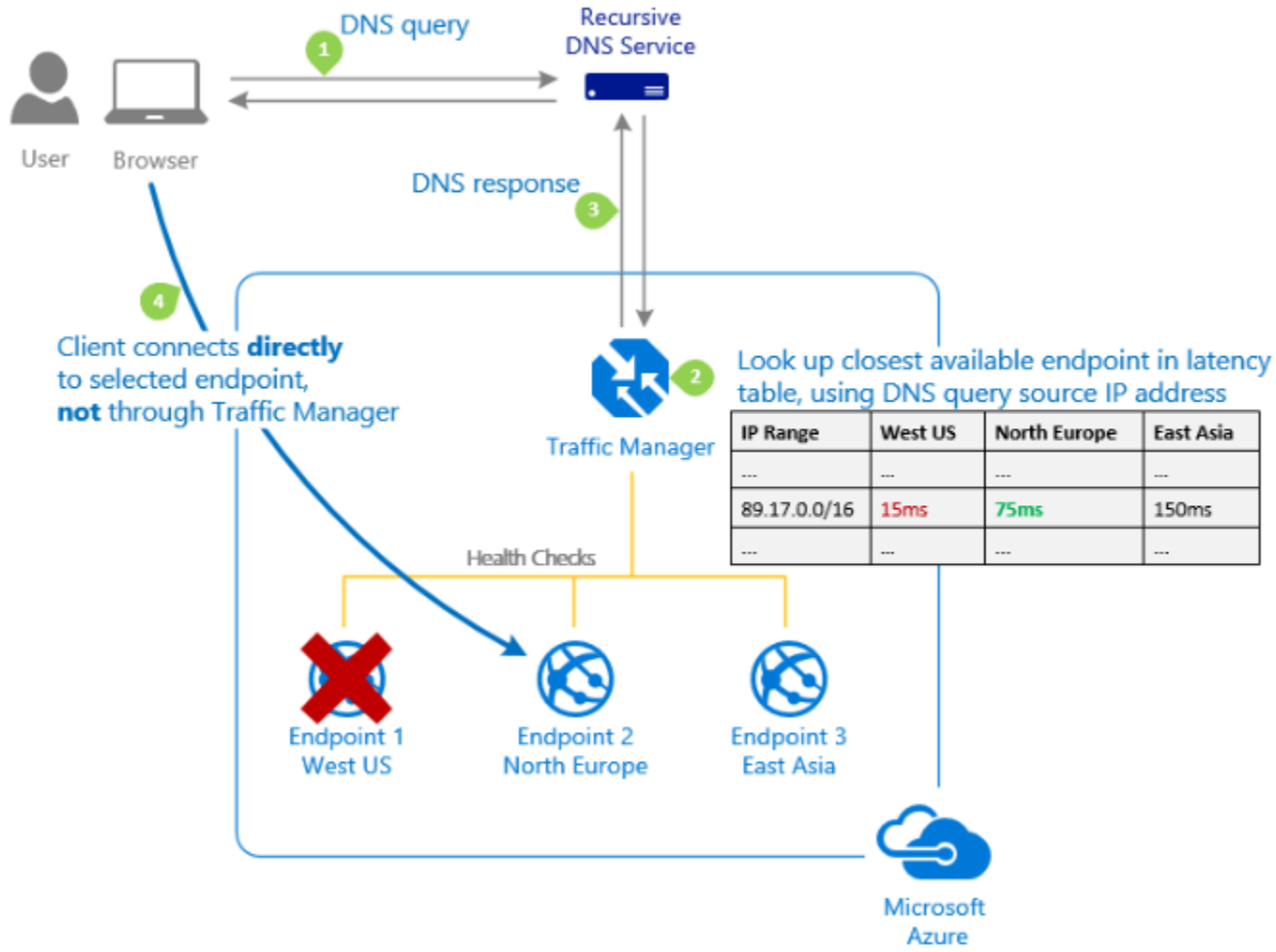
- Priority: Select Priority when you want to use a primary service endpoint for all traffic, and provide backups in case the primary or the backup endpoints are unavailable.



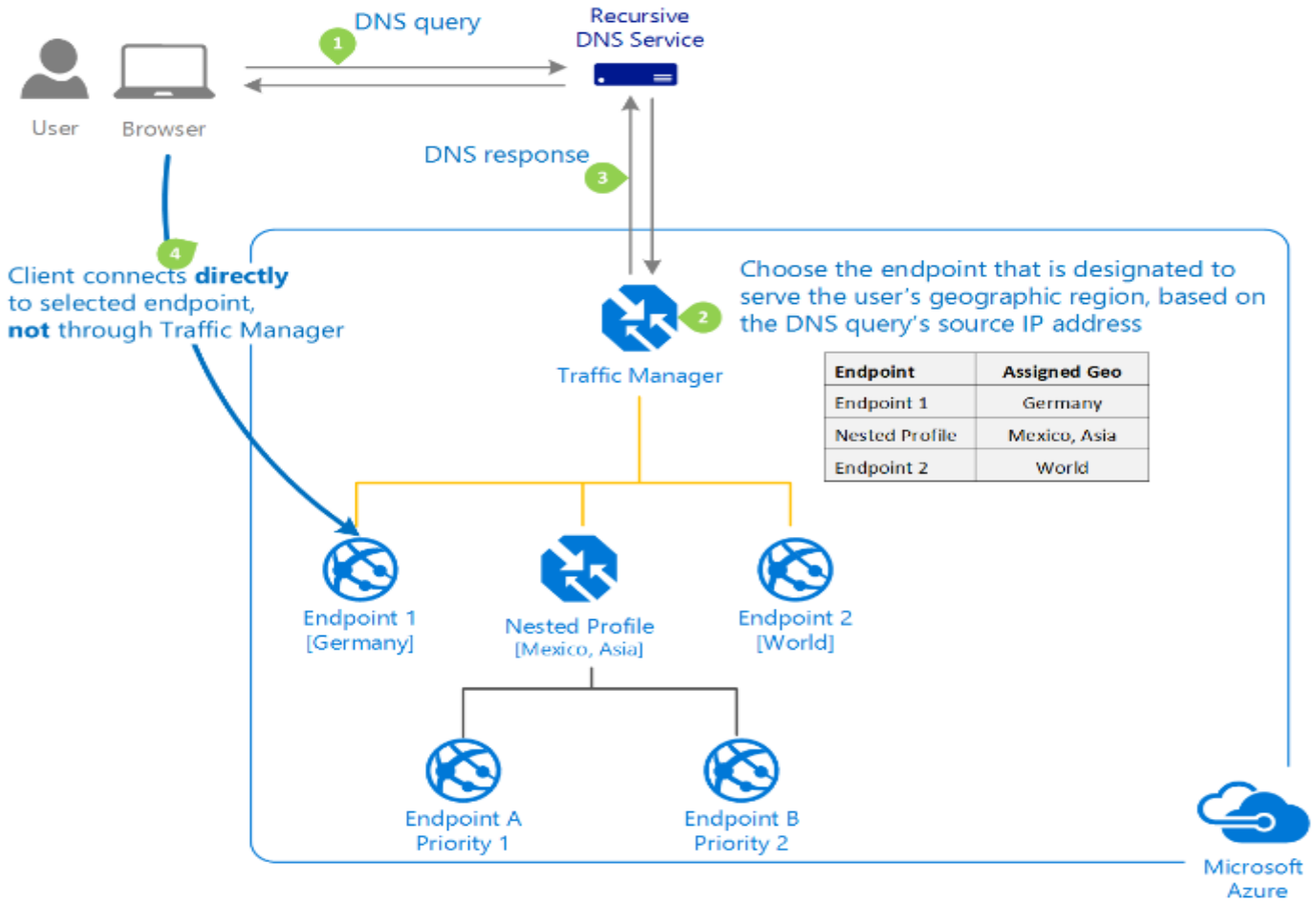
Weighted: Select Weighted when you want to distribute traffic across a set of endpoints, either evenly or according to weights, which you define.



Performance: Select Performance when you have endpoints in different geographic locations and you want end users to use the "closest" endpoint in terms of the lowest network latency.



Geographic: Select Geographic so that users are directed to specific endpoints (Azure, External, or Nested) based on which geographic location their DNS query originates from. This empowers Traffic Manager customers to enable scenarios where knowing a user's geographic region and routing them based on that is important. Examples include complying with data sovereignty mandates, localization of content & user experience and measuring traffic from different regions.

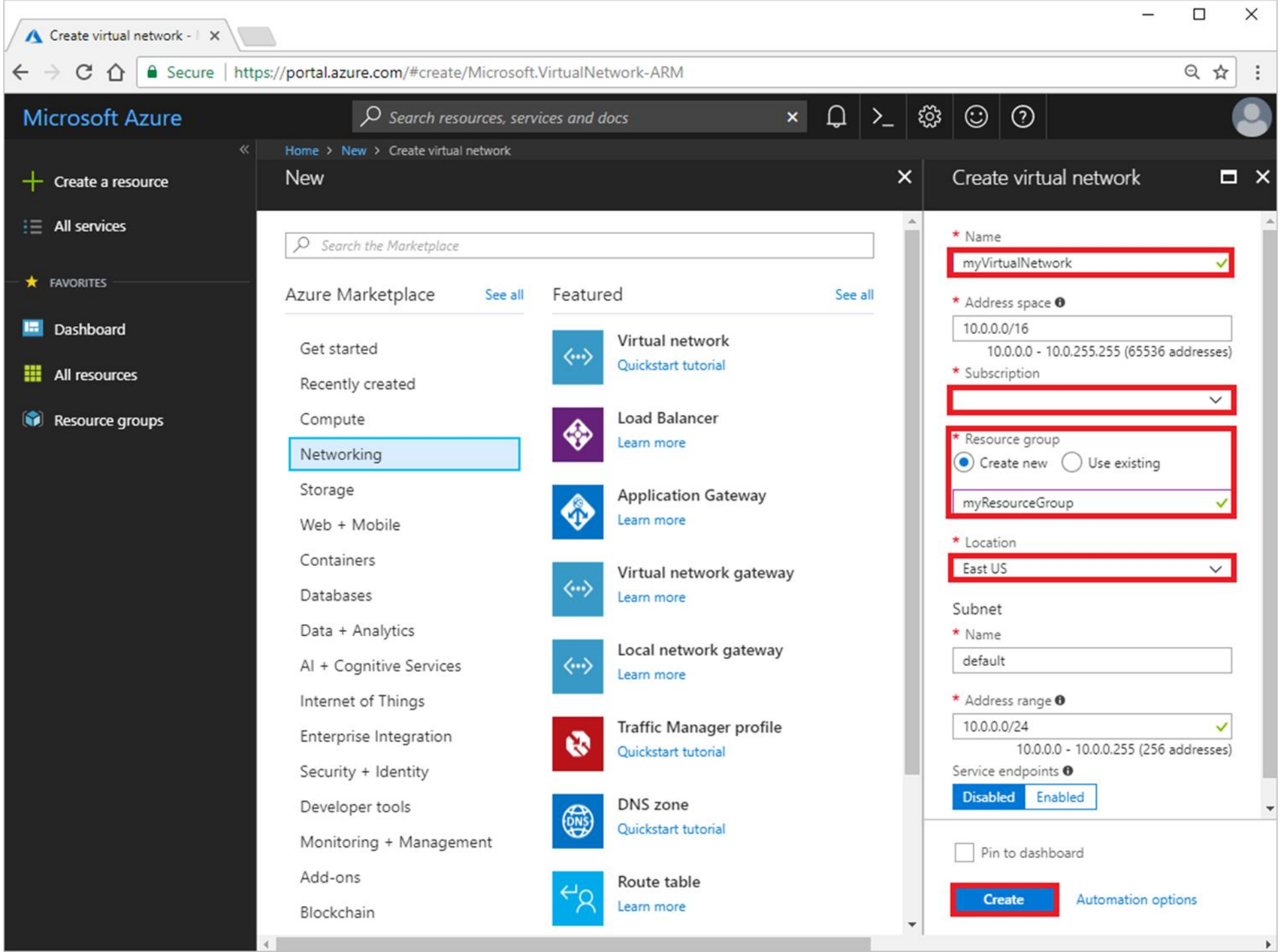


Vnet Lab

Creating VNet through Portal

Log in to the Azure portal at https://portal.azure.com.
Create a virtual network
1.Select + Create a resource on the upper, left corner of the Azure portal.
2.Select Networking, and then select Virtual network.
3.Enter, or select, the following information, accept the defaults for the remaining settings, and then select Create:

Setting	Value
Name	myVirtualNetwork
Subscription	Select your subscription.
Resource group	Select Create new and enter myResourceGroup.
Location	Select East US.


Create virtual network -
Secure
https://portal.azure.com/#create/Microsoft.VirtualNetwork-ARM
Microsoft Azure
Search resources, services and docs
Home > New > Create virtual network
Create a resource
All services
FAVORITES
Dashboard
All resources
Resource groups
New
Search the Marketplace
Azure Marketplace
See all
Get started
Recently created
Compute
Networking
Storage
Web + Mobile
Containers
Databases
Data + Analytics
AI + Cognitive Services
Internet of Things
Enterprise Integration
Security + Identity
Developer tools
Monitoring + Management
Add-ons
Blockchain
Featured
See all
Virtual network
Quickstart tutorial
Load Balancer
Learn more
Application Gateway
Learn more
Virtual network gateway
Learn more
Local network gateway
Learn more
Traffic Manager profile
Quickstart tutorial
DNS zone
Quickstart tutorial
Route table
Learn more
Create virtual network
* Name
myVirtualNetwork
* Address space
10.0.0.0/16
10.0.0.0 - 10.0.255.255 (65536 addresses)
* Subscription
* Resource group
Create new
Use existing
myResourceGroup
* Location
East US
Subnet
* Name
default
* Address range
10.0.0.0/24
10.0.0.0 - 10.0.0.255 (256 addresses)
Service endpoints
Disabled
Enabled
Pin to dashboard
Create
Automation options

Creating VNet through PowerShell

TestVnet9 - 192.168.9.0/24
FrontEnd - 192.168.9.0/26
BackEnd - 192.168.9.128/26

1. Creating New Resource Group

New-AzureRmResourceGroup -Name TestRG -Location centralus

2. Creating Virtual Network "TestVnet9" with IP Address - 192.168.9.0/24

New-AzureRmVirtualNetwork -ResourceGroupName TestRG -Name TestVNet9 -AddressPrefix 192.168.9.0/24 -Location centralus

3. Store the virtual network object in a variable:

$vnet = Get-AzureRmVirtualNetwork -ResourceGroupName TestRG -Name TestVNet9

5. Add a subnet to the new VNet variable:

```
Add-AzureRmVirtualNetworkSubnetConfig -Name FrontEnd -VirtualNetwork $vnet -AddressPrefix 192.168.9.0/26
```

6. Repeat step 5 above for each subnet you want to create. The following command creates the BackEnd subnet for the scenario:

```
Add-AzureRmVirtualNetworkSubnetConfig -Name BackEnd -VirtualNetwork $vnet -AddressPrefix 192.168.9.128/26
```

7. Although you create subnets, they currently only exist in the local variable used to retrieve the VNet you create in step 4 above. To save the changes to Azure, run the following command:

```
Set-AzureRmVirtualNetwork -VirtualNetwork $vnet
```

Connectivity

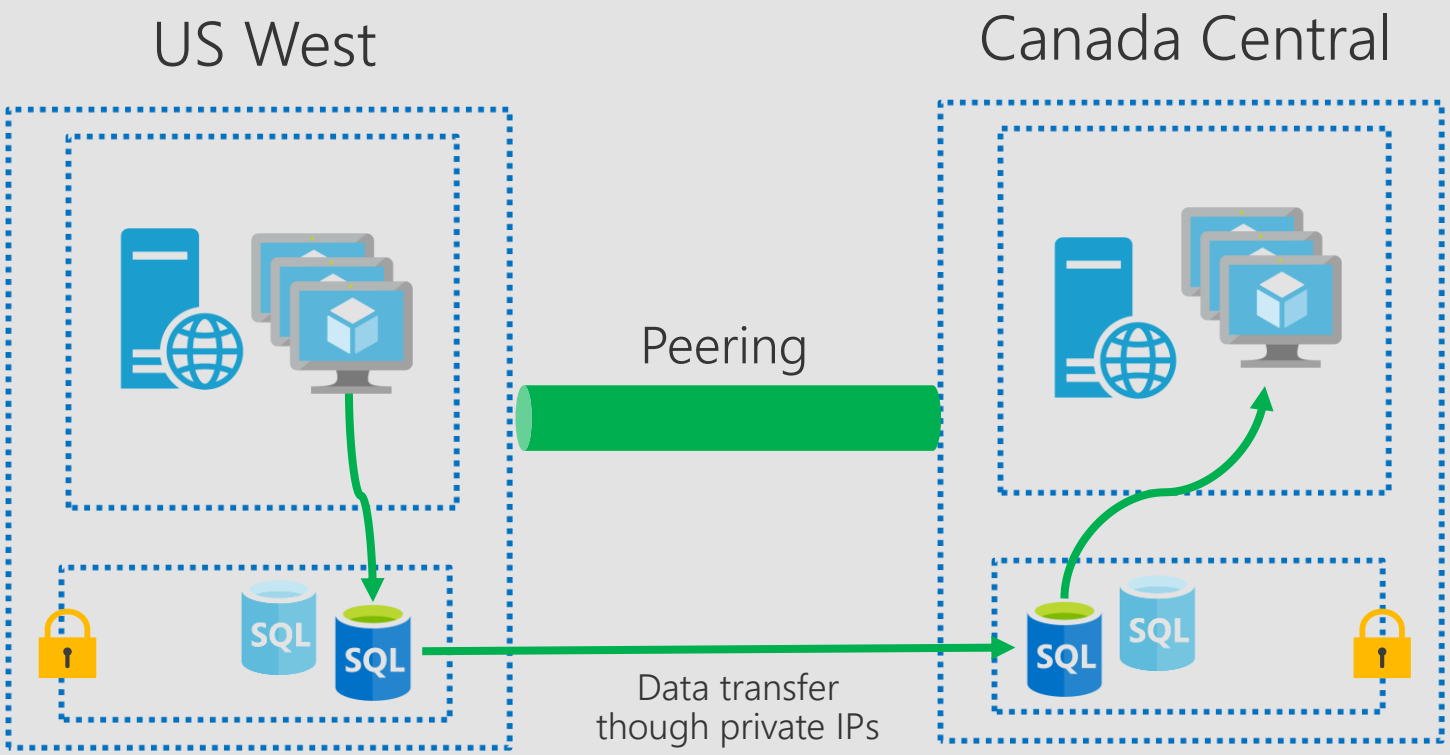
Azure offers the following VPN gateway SKUs:

SKU	S2S/VNet-to-VNet Tunnels	P2S Connections	Aggregate Throughput Benchmark
VpnGw1	Max. 30	Max. 128	650 Mbps
VpnGw2	Max. 30	Max. 128	1 Gbps
VpnGw3	Max. 30	Max. 128	1.25 Gbps
Basic	Max. 10	Max. 128	100 Mbps

Workload	SKUs
Production, critical workloads	VpnGw1, VpnGw2, VpnGw3
Dev-test or proof of concept	Basic

SKU	Features
Basic	**Route-based VPN**: 10 tunnels with P2S **Policy-based VPN**: (IKEv1): 1 tunnel; no P2S
VpnGw1, VpnGw2, and VpnGw3	**Route-based VPN**: up to 30 tunnels (*), P2S, BGP, active-active, custom IPsec/IKE policy, ExpressRoute/VPN co-existence

Global VNet Peering



New

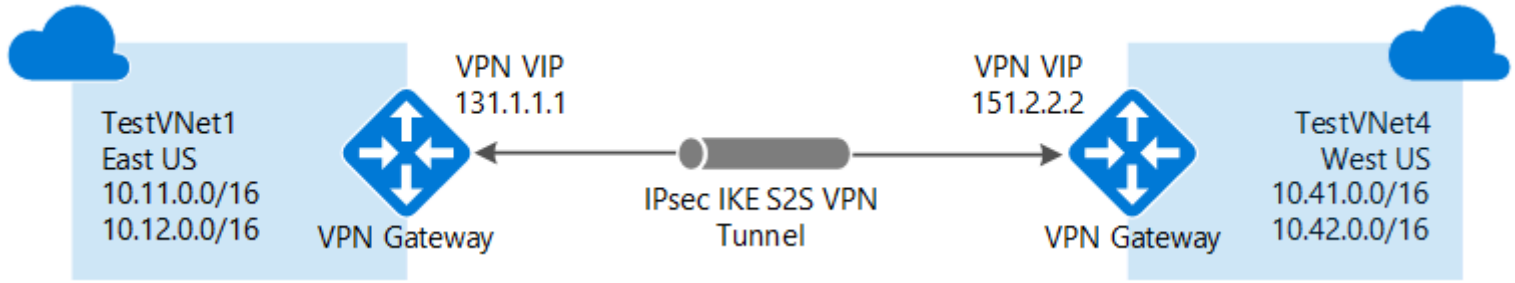
- Peer VNets cross regions
- Simple to setup (just a few clicks) and manage
- Direct VM-to-VM connectivity across regions
- Through Microsoft Backbone: no public internet
- No bandwidth limitations
- No extra hops
- Scenarios:
  - Data replication
  - Database failover

VNet to VNet

Configuring a VNet-to-VNet connection is a good way to easily connect VNets. Connecting a virtual network to another virtual network using the VNet-to-VNet connection type (VNet2VNet) is similar to creating a Site-to-Site IPsec connection to an on-premises location. Both connectivity types use a VPN gateway to provide a secure tunnel using IPsec/IKE, and both function the same way when communicating. The difference between the connection types is the way the local network gateway is configured. When you create a VNet-to-VNet connection, you do not see the local network gateway address space. It is automatically created and populated. If you update the address space for one VNet, the other VNet automatically knows to route to the updated address space. Creating a VNet-to-VNet connection is typically faster and easier than creating a Site-to-Site connection between VNets

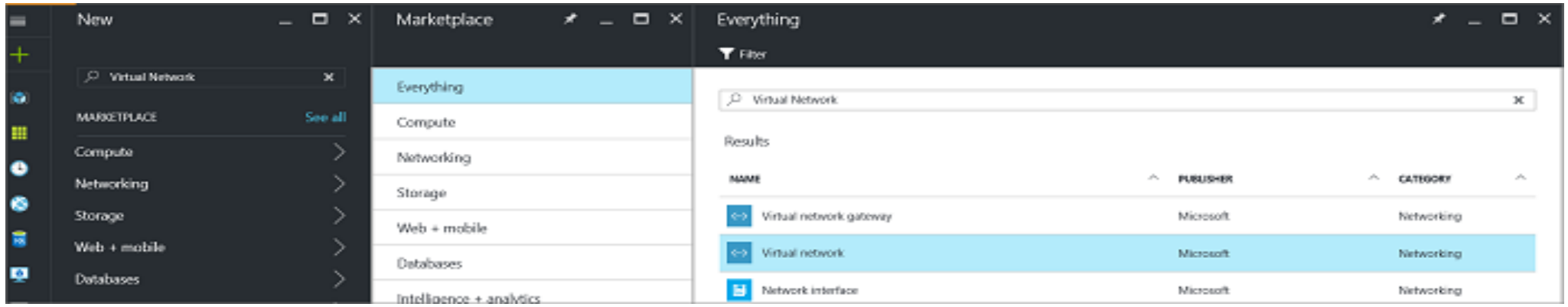
Steps:

1. Create Virtual Network
2. Create Gateway Subnet
3. Create Virtual Network Gateway
4. Create Vnet-to-Vnet Peering

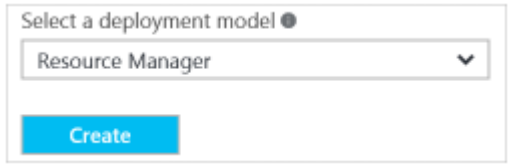


1. Create and configure TestVNet1

1.From a browser, navigate to the Azure portal and, if necessary, sign in with your Azure account.
2.Click +. In the **Search the marketplace** field, type **"Virtual Network"**. Locate Virtual Network from the returned list and click to open the **Virtual Network** page.



3.Near the bottom of the Virtual Network page, from the **Select a deployment model** list, select **Resource Manager**, and then click **Create**.

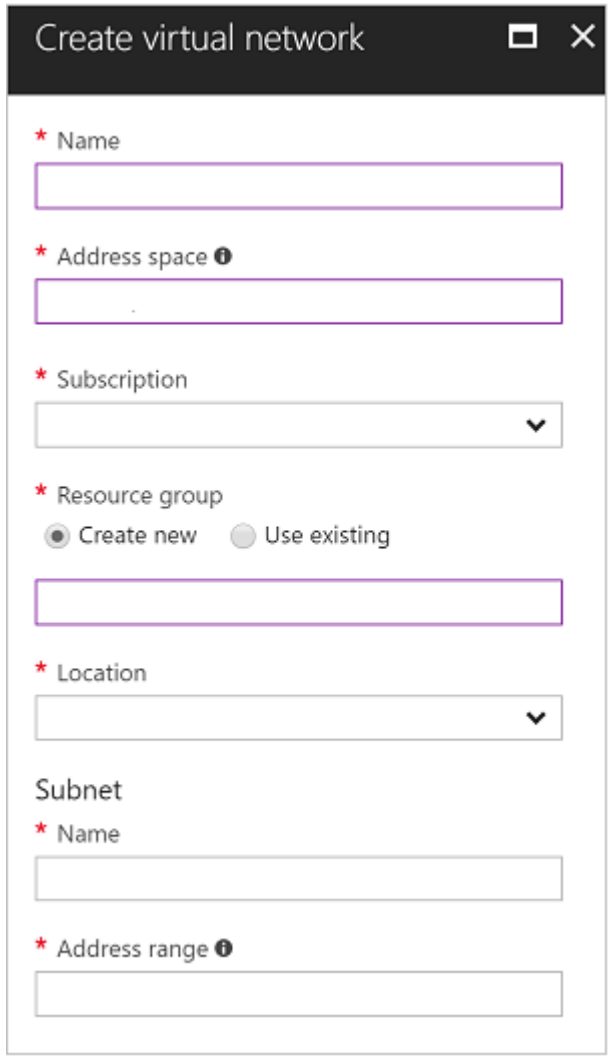


Example settings

Values for TestVNet1:

- VNet Name: TestVNet1
- Address space: 10.11.0.0/16
- Subscription: Select the subscription you want to use
- Resource Group: TestRG1
- Location: East US
- Subnet Name: FrontEnd
- Subnet Address range: 10.11.0.0/24
- Gateway Subnet name: GatewaySubnet (this will auto-fill in the portal)
- Gateway Subnet address range: 10.11.255.0/27
- DNS Server: Use the IP address of your DNS Server
- Virtual Network Gateway Name: TestVNet1GW
- Gateway Type: VPN
- VPN type: Route-based
- SKU: Select the Gateway SKU you want to use
- Public IP address name: TestVNet1GWIP
- Connection Name: TestVNet1toTestVNet4
- Shared key: You can create the shared key yourself. For this example, we'll use abc123. The important thing is that when you create the connection between the VNets, the value must match.

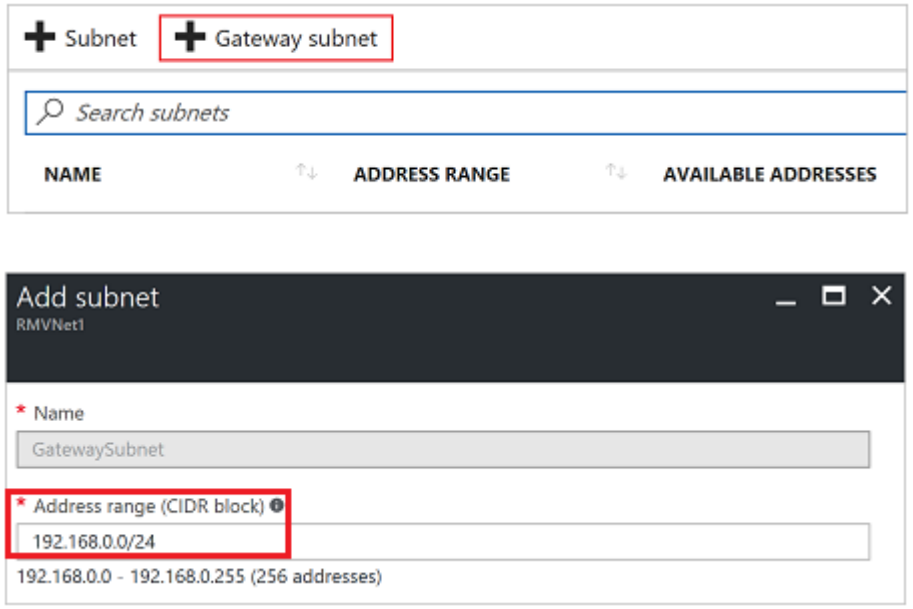
4. On the **Create virtual network** page, configure the VNet settings. When you fill in the fields, the red exclamation mark becomes a green check mark when the characters entered in the field are valid. There may be values that are auto-filled. If so, replace the values with your own. The **Create virtual network** page looks similar to the following example:



5.**Name:** Enter the name for your virtual network
6.**Address space:** Enter the address space. If you have multiple address spaces to add, add your first address space. You can add additional address spaces later, after creating the VNet.
7.**Subscription:** Verify that the Subscription listed is the correct one. You can change subscriptions by using the drop-down.
8.**Resource group:** Select an existing resource group, or create a new one by typing a name for your new resource group. If you are creating a new group, name the resource group according to your planned configuration values. For more information about resource groups, visit Azure Resource Manager Overview.
9.**Location:** Select the location for your VNet. The location determines where the resources that you deploy to this VNet will reside.
10.**Subnet:** Add the subnet name and subnet address range. You can add additional subnets later, after creating the VNet.
11.Select **Pin to dashboard** if you want to be able to find your VNet easily on the dashboard, and then click **Create.**

2. Create a gateway subnet

Before creating a virtual network gateway for your virtual network, you first need to create the gateway subnet. The gateway subnet contains the IP addresses that are used by the virtual network gateway. If possible, it's best to create a gateway subnet using a CIDR block of /28 or /27 in order to provide enough IP addresses to accommodate additional future configuration requirements.

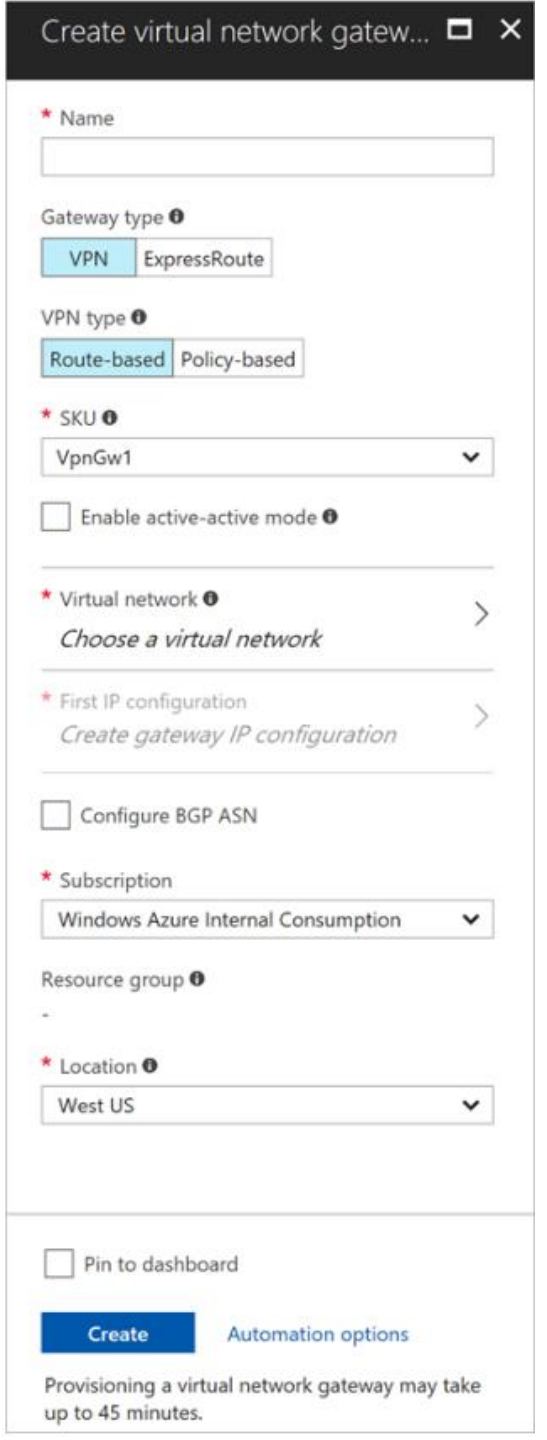


1.In the portal, navigate to the Resource Manager virtual network for which you want to create a virtual network gateway.
2.In the **Settings** section of your VNet page, click **Subnets** to expand the Subnets page.
3.On the **Subnets** page, click **+Gateway subnet** to open the **Add subnet** page.
4.The **Name** for your subnet is automatically filled in with the value **'GatewaySubnet'**. This value is required in order for Azure to recognize the subnet as the gateway subnet. Adjust the auto-filled **Address range** values to match your configuration requirements, then click **OK** at the bottom of the page to create the subnet.

3.Create a virtual network gateway

In this step, you create the virtual network gateway for your VNet. Creating a gateway can often take 45 minutes or more, depending on the selected gateway SKU.

1. In the portal, on the left side, click + and type **'virtual network gateway'** in search.
Locate **Virtual network gateway** in the search return and click the entry. On the Virtual network gateway page, click **Create** at the bottom of the page to open the **Create virtual network gateway** page.
2. On the **Create virtual network gateway** page, fill in the values for your virtual network gateway.
3.On the **Create virtual network gateway** page, specify the values for your virtual network gateway.
•**Name:** Name your gateway. This is not the same as naming a gateway subnet. It's the name of the gateway object you are creating.
•**Gateway type:** Select VPN. VPN gateways use the virtual network gateway type VPN.
•**VPN type:** Select the VPN type that is specified for your configuration. Most configurations require a Route-based VPN type.
•**SKU:** Select the gateway SKU from the dropdown. The SKUs listed in the dropdown depend on the VPN type you select. For more information about gateway SKUs, see Gateway SKUs.



•**Location:** You may need to scroll to see Location. Adjust the **Location** field to point to the location where your virtual network is located. If the location is not pointing to the region where your virtual network resides, when you select a virtual network in the next step, it will not appear in the drop-down list.
•**Virtual network:** Choose the virtual network to which you want to add this gateway. Click **Virtual network** to open the 'Choose a virtual network' page. Select the VNet. If you don't see your VNet, make sure the Location field is pointing to the region in which your virtual network is located.

•**Gateway subnet address range:** You will only see this setting if you did not previously create a gateway subnet for your virtual network. If you previously created a valid gateway subnet, this setting will not appear.
•**First IP configuration:** The 'Choose public IP address' page creates a public IP address object that gets associated to the VPN gateway. The public IP address is dynamically assigned to this object when the VPN gateway is created. VPN Gateway currently only supports Dynamic Public IP address allocation. However, this does not mean that the IP address changes after it has been assigned to your VPN gateway. The only time the Public IP address changes is when the gateway is deleted and re-created. It doesn't change across resizing, resetting, or other internal maintenance/upgrades of your VPN gateway.

- First, click **Create gateway IP configuration** to open the 'Choose public IP address' page, then click **+Create new** to open the 'Create public IP address' page.
- Next, input a **Name** for your public IP address. Leave the SKU as **Basic** unless there is a specific reason to change it to something else, then click **OK** at the bottom of this page to save your changes.

5.Verify the settings. You can select **Pin to dashboard** at the bottom of the page if you want your gateway to appear on the dashboard.
6.Click **Create** to begin creating the VPN gateway. The settings are validated and you'll see the "Deploying Virtual network gateway" tile on the dashboard. Creating a gateway can take up to 45 minutes. You may need to refresh your portal page to see the completed status.

After the gateway is created, view the IP address that has been assigned to it by looking at the virtual network in the portal. The gateway appears as a connected device. You can click the connected device (your virtual network gateway) to view more information.

4.Create and configure TestVNet4

Once you've configured TestVNet1, create TestVNet4 by repeating the previous steps, replacing the values with those of TestVNet4. You don't need to wait until the virtual network gateway for TestVNet1 has finished creating before configuring TestVNet4. If you are using your own values, make sure that the address spaces don't overlap with any of the VNets that you want to connect to.

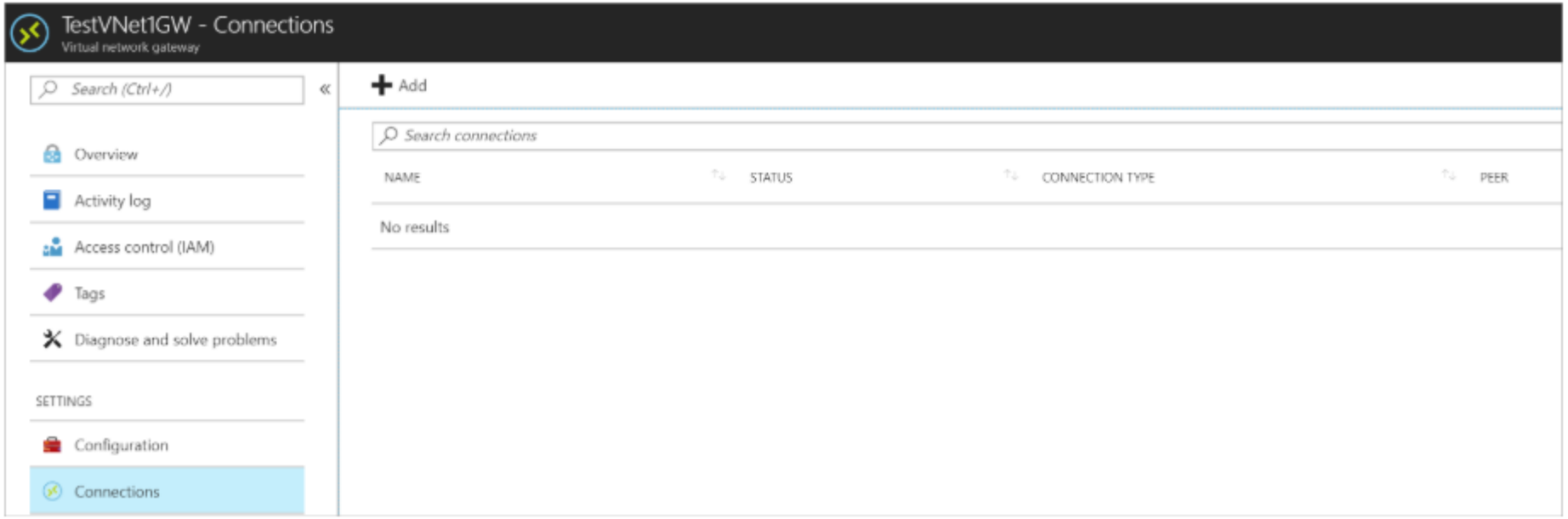
Values for TestVNet4:

- VNet Name: TestVNet4
- Address space: 10.41.0.0/16
- Subscription: Select the subscription you want to use
- Resource Group: TestRG4
- Location: West US
- Subnet Name: FrontEnd
- Subnet Address range: 10.41.0.0/24
- GatewaySubnet name: GatewaySubnet (this will auto-fill in the portal)
- GatewaySubnet address range: 10.41.255.0/27
- DNS Server: Use the IP address of your DNS Server
- Virtual Network Gateway Name: TestVNet4GW
- Gateway Type: VPN
- VPN type: Route-based
- SKU: Select the Gateway SKU you want to use
- Public IP address name: TestVNet4GWIP
- Connection Name: TestVNet4toTestVNet1
- Shared key: You can create the shared key yourself. For this example, we'll use abc123. The important thing is that when you create the connection between the VNets, the value must match.

5.Configure the TestVNet1 gateway connection

When the virtual network gateways for both TestVNet1 and TestVNet4 have completed, you can create your virtual network gateway connections. In this section, you create a connection from VNet1 to VNet4. These steps work only for VNets in the same subscription. If your VNets are in different subscriptions, you must use PowerShell to make the connection. However, if your VNets are in different resource groups in the same subscription, you can connect them using the portal.

1.In **All resources,** navigate to the virtual network gateway for your VNet. For example, **TestVNet1GW**. Click **TestVNet1GW** to open the virtual network gateway page.



2.Click **+Add** to open the **Add connection** page.

3.On the **Add connection** page, in the name field, type a name for your connection. For example, **TestVNet1toTestVNet4**.

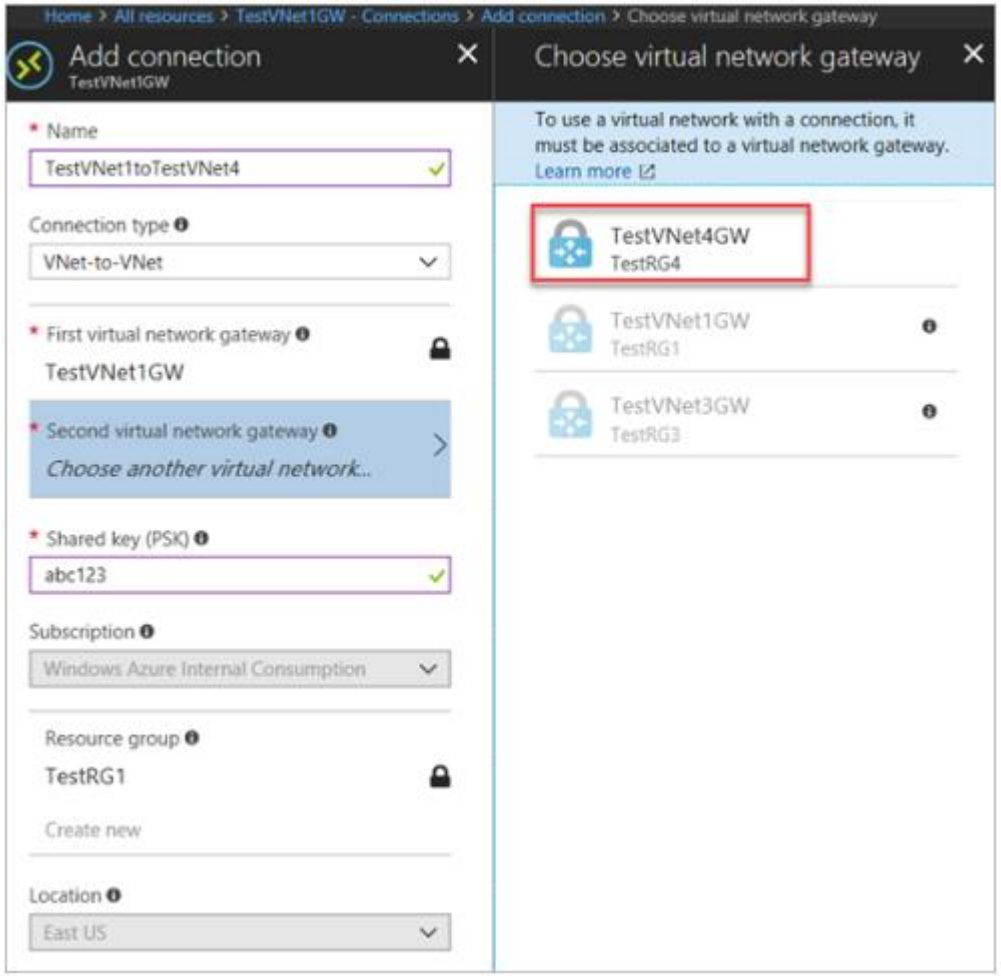
4.For **Connection type**, select **VNet-to-VNet** from the dropdown.

5.The **First virtual network gateway** field value is automatically filled in because you are creating this connection from the specified virtual network gateway.

6.The **Second virtual network gateway** field is the virtual network gateway of the VNet that you want to create a connection to. Click **Choose another virtual network gateway** to open the **Choose virtual network gateway** page.

7.View the virtual network gateways that are listed on this page. Notice that only virtual network gateways that are in your subscription are listed.

8.Click the virtual network gateway that you want to connect to.



9.In the **Shared key** field, type a shared key for your connection. You can generate or create this key yourself. In a site-to-site connection, the key you use would be exactly the same for your on-premises device and your virtual network gateway connection. The concept is similar here, except that rather than connecting to a VPN device, you are connecting to another virtual network gateway.

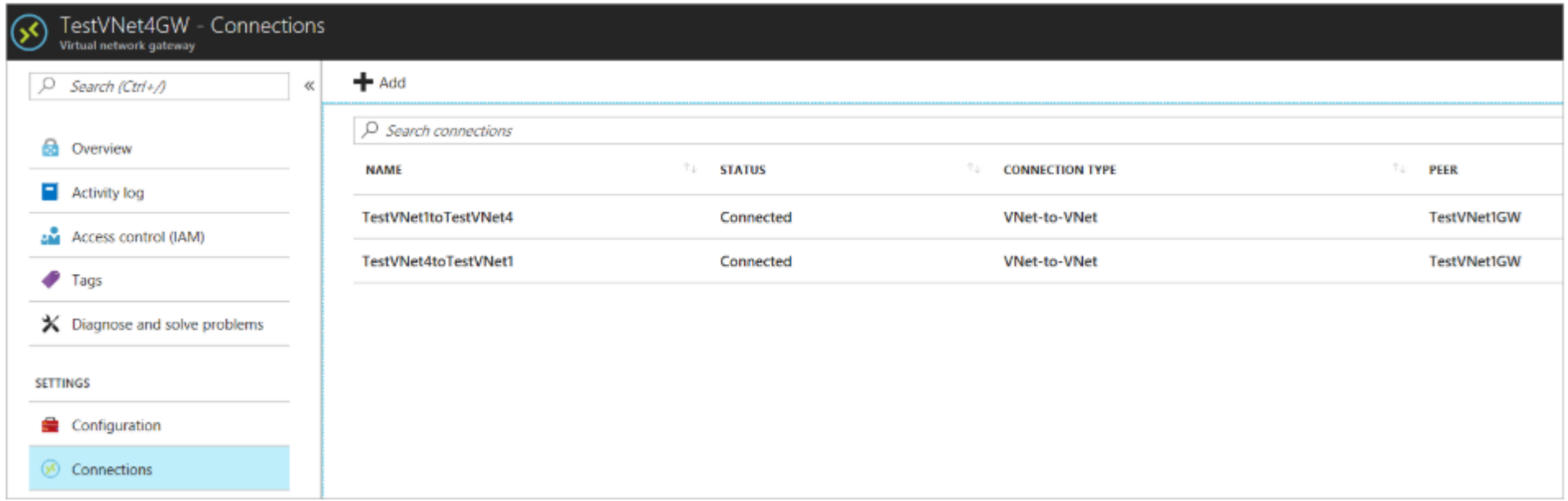
10.Click **OK** at the bottom of the page to save your changes.

6.Configure the TestVNet4 gateway connection

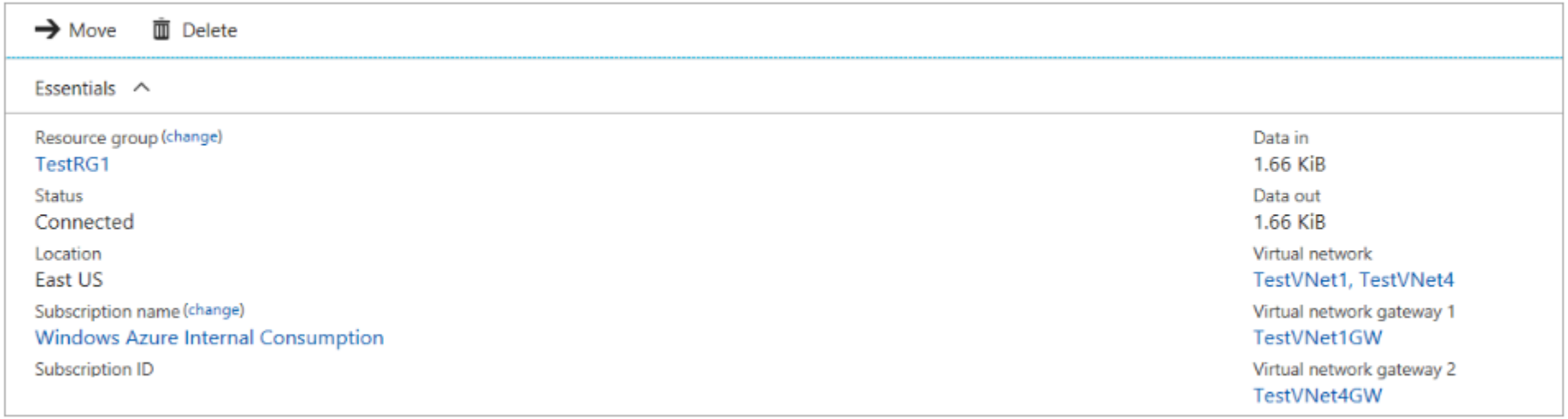
Next, create a connection from TestVNet4 to TestVNet1. In the portal, locate the virtual network gateway associated with TestVNet4. Follow the steps from the previous section, replacing the values to create a connection from TestVNet4 to TestVNet1. Make sure that you use the same shared key.

7. Verify your connections

Locate the virtual network gateway in the portal. On the virtual network gateway page, click **Connections** to view the connections page for the virtual network gateway. Once the connection is established, you see the Status values change to **Succeeded** and **Connected**. You can double-click a connection to open the **Essentials** page and view more information.



When data begins flowing, you see values for Data in and Data out.



To add additional connections

If you want to add additional connections, navigate to the virtual network gateway that you want to create the connection from, then click **Connections.** You can create another VNet-to-VNet connection, or create an IPsec Site-to-Site connection to an on-premises location. Be sure to adjust the **Connection type** to match the type of connection you want to create. Before creating additional connections, verify that the address space for your virtual network does not overlap with any of the address spaces that you want to connect to.

Connectivity – Cross-Premises

Point to Site VPN Connectivity

A **Point-to-Site (P2S)** VPN gateway connection lets you create a secure connection to your virtual network from an individual client computer. A P2S connection is established by starting it from the client computer. This solution is useful for telecommuters who want to connect to Azure VNets from a remote location, such as from home or a conference. P2S VPN is also a useful solution to use instead of S2S VPN when you have only a few clients that need to connect to a VNet.

What protocol does P2S use?

Point-to-site VPN can use one of the following protocols:

- **Secure Socket Tunneling Protocol (SSTP)**, a proprietary SSL-based VPN protocol. An SSL VPN solution can penetrate firewalls, since most firewalls open TCP port 443, which SSL uses. SSTP is only supported on Windows devices. Azure supports all versions of Windows that have SSTP (Windows 7 and later).
- **IKEv2 VPN**, a standards-based IPsec VPN solution. IKEv2 VPN can be used to connect from Mac devices (OSX versions 10.11 and above).

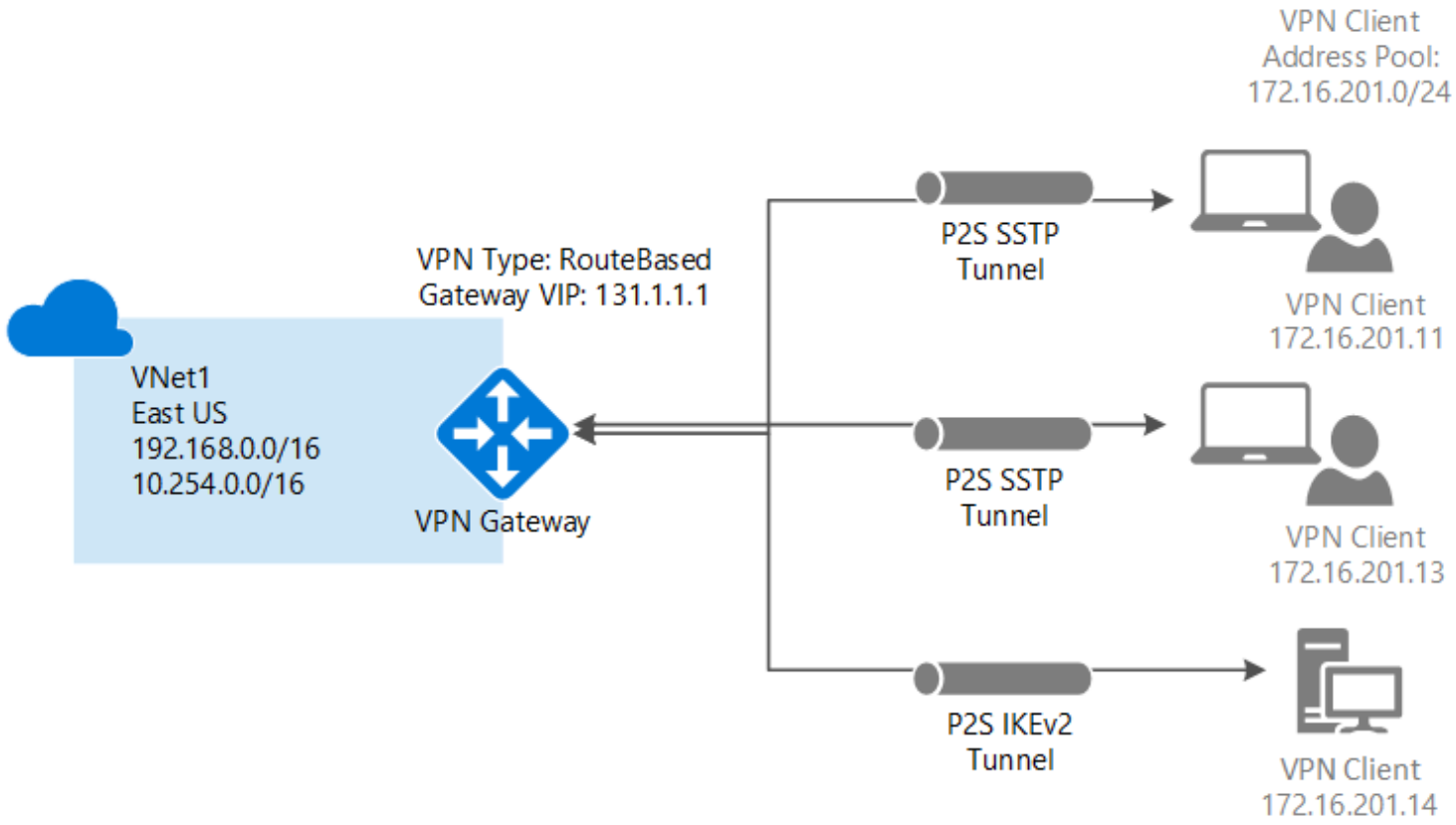
If you have a mixed client environment consisting of Windows and Mac devices, configure both SSTP and IKEv2.

Note:
IKEv2 for P2S is available for the Resource Manager deployment model only. It is not available for the classic deployment model.

Point to Site VPN Connectivity

Steps:

1.Create Virtual Network
2.Create Gateway Subnet
3.Create Virtual Network Gateway
4.Virtual Network Gateway-Setting-Point-to-site configuration
5.Provide private IP address for Clients
6.Generate Root and Client Certificate through PowerShell
7.Upload the root certificate public certificate data
8.Generate and install the VPN client configuration package



Architecture

Point-to-Site native Azure certificate authentication connections use the following items, which you configure in this exercise:

- A **RouteBased** VPN gateway.
- The **public key (.cer file)** for a **root certificate**, which is uploaded to Azure. Once the certificate is uploaded, it is considered a trusted certificate and is used for authentication.
- A **client certificate** that is generated from the **root certificate**. The client certificate installed on each client computer that will connect to the VNet. This certificate is used for **client authentication**.
- A **VPN client configuration**. The VPN client configuration files contain the necessary information for the client to connect to the VNet. The files configure the existing VPN client that is native to the operating system. Each client that connects must be configured using the settings in the configuration files

How are P2S VPN clients authenticated?

Before Azure accepts a P2S VPN connection, the user has to be authenticated first. There are two mechanisms that Azure offers to authenticate a connecting user.

Authenticate using native Azure certificate authentication

When using the native Azure certificate authentication, a client certificate that is present on the device is used to authenticate the connecting user. Client certificates are generated from a trusted root certificate and then installed on each client computer. You can use a root certificate that was generated using an Enterprise solution, or you can generate a self-signed certificate. The validation of the client certificate is performed by the VPN gateway and happens during establishment of the P2S VPN connection. The root certificate is required for the validation and must be uploaded to Azure.

Authenticate using Active Directory (AD) Domain Server

AD Domain authentication allows users to connect to Azure using their organization domain credentials. It requires a RADIUS server that integrates with the AD server. Organizations can also leverage their existing RADIUS deployment. The RADIUS server could be deployed on-premises or in your Azure VNET. During authentication, the Azure VPN Gateway acts as a pass through and forwards authentication messages back and forth between the RADIUS server and the connecting device. So Gateway reachability to the RADIUS server is important. If the RADIUS server is present on-premises, then a VPN S2S connection from Azure to the on-premises site is required for reachability. The RADIUS server can also integrate with AD certificate services. This lets you use the RADIUS server and your enterprise certificate deployment for P2S certificate authentication as an alternative to the Azure certificate authentication. The advantage is that you don't need to upload root certificates and revoked certificates to Azure.
A RADIUS server can also integrate with other external identity systems. This opens up plenty of authentication options for P2S VPN, including multi-factor options.

Configuration requirements for client devices

Users use the native VPN clients on Windows and Mac devices for P2S. Azure provides a VPN client configuration zip file that contains settings required by these native clients to connect to Azure.

- For Windows devices, the VPN client configuration consists of an installer package that users install on their devices.
- For Mac devices, it consists of the mobile config file that users install on their devices.

The zip file also provides the values of some of the important settings on the Azure side that you can use to create your own profile for these devices. Some of the values include the VPN gateway address, configured tunnel types, routes, and the root certificate for gateway validation.

Obtain the .cer file for the root certificate

- **Enterprise certificate:** If you are using an enterprise solution, you can use your existing certificate chain. Obtain the .cer file for the root certificate that you want to use.
- **Self-signed root certificate:** If you aren't using an enterprise certificate solution, you need to create a self-signed root certificate.

Create Self Signed Certificate: From PowerShell

ROOT CERTIFICATE

```
$cert = New-SelfSignedCertificate -Type Custom -KeySpec Signature `
-Subject "CN=P2SRootCert" -KeyExportPolicy Exportable `
-HashAlgorithm sha256 -KeyLength 2048 `
-CertStoreLocation "Cert:\CurrentUser\My" -KeyUsageProperty Sign -KeyUsage CertSign
```

CLIENTCERTIFICATE

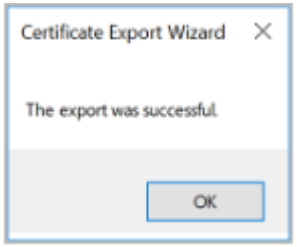
```
New-SelfSignedCertificate -Type Custom -KeySpec Signature `
-Subject "CN=P2SChildCert" -KeyExportPolicy Exportable `
-HashAlgorithm sha256 -KeyLength 2048 `
-CertStoreLocation "Cert:\CurrentUser\My" `
-Signer $cert -TextExtension @("2.5.29.37={text}1.3.6.1.5.5.7.3.2")
```

Export the root certificate public key (.cer)

Open certificate manager in windows platform: Start->Run-> then type certmgr.msc

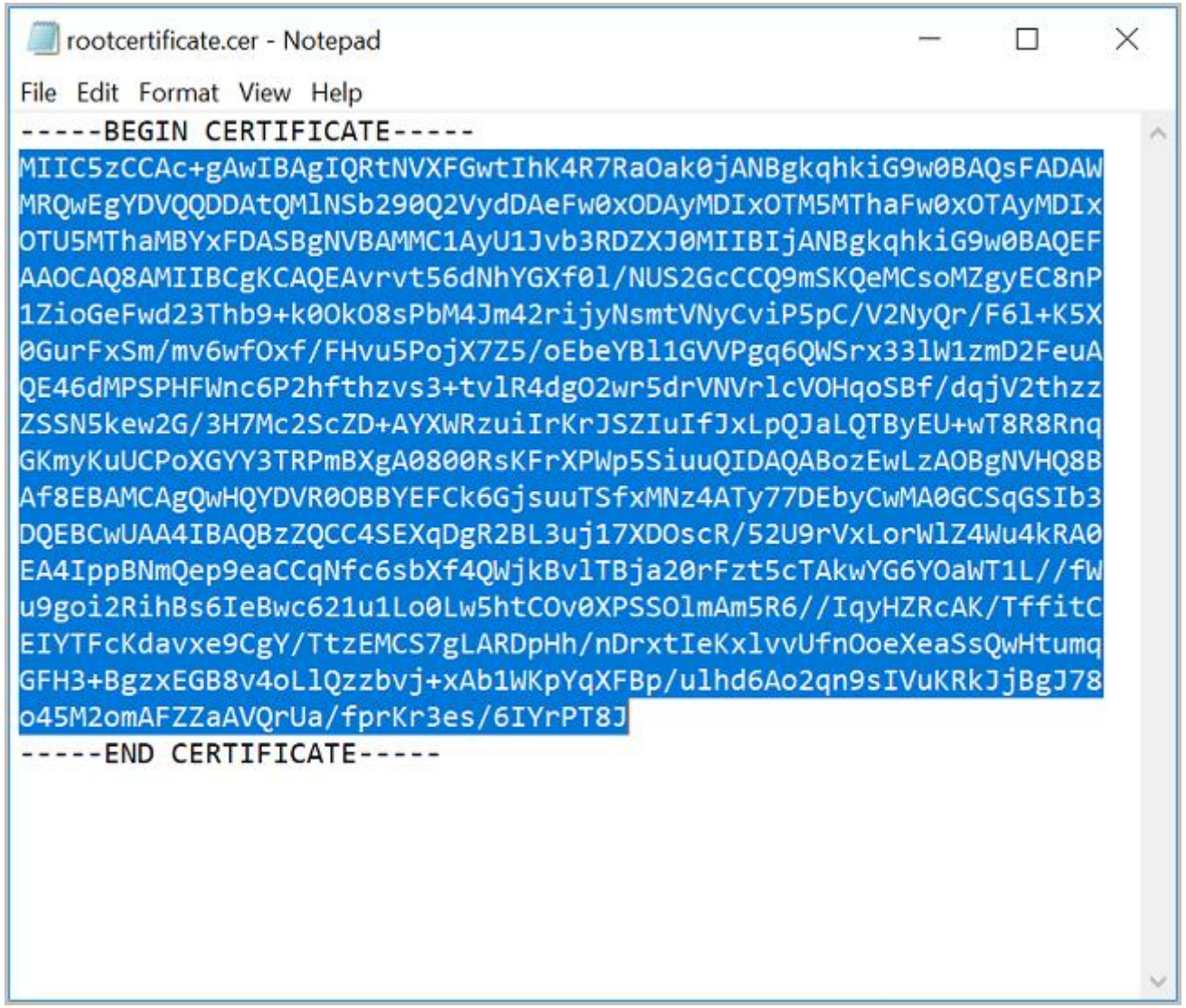
To obtain a .cer file from the certificate, open **Manage user certificates.** Locate the self-signed root certificate, typically in 'Certificates - Current User\Personal\Certificates', and right-click. Click **All Tasks,** and then click **Export.** This opens the **Certificate Export Wizard.**
In the Wizard, click **Next.** Select **No, do not export the private key,** and then click **Next.**
On the **Export File Format** page, select **Base-64 encoded X.509 (.CER).,** and then click **Next.**
On the **File to Export, Browse to** the location to which you want to export the certificate. For **File name,** name the certificate file. Then, click **Next.**
Click **Finish** to export the certificate. You see **The export was successful.** Click **OK** to close the wizard.

Certificate is successfully exported**.**



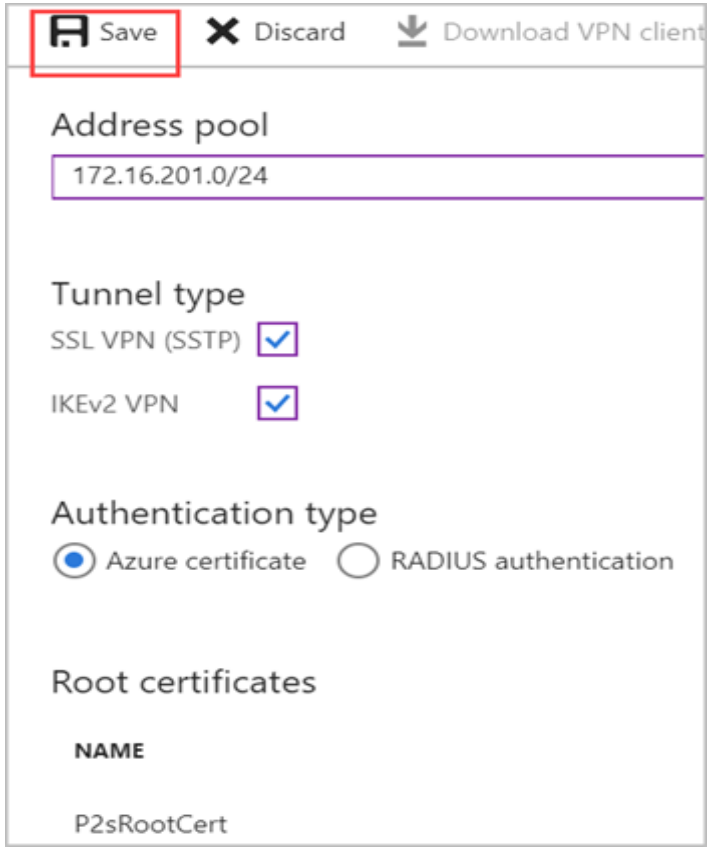
The exported certificate looks similar to this:

If you open the exported certificate using Notepad, you see something similar to this example. The section in blue contains the information that is uploaded to Azure. If you open your certificate with Notepad and it does not look similar to this, typically this means you did not export it using the **Base-64 encoded X.509(.CER)** format. Additionally, if you want to use a different text editor, understand that some editors can introduce unintended formatting in the background. This can create problems when uploaded the text from this certificate to Azure.



Paste the certificate data into the **Public Certificate Data** field. Name the certificate, and then click **Save**. You can add up to 20 trusted root certificates.

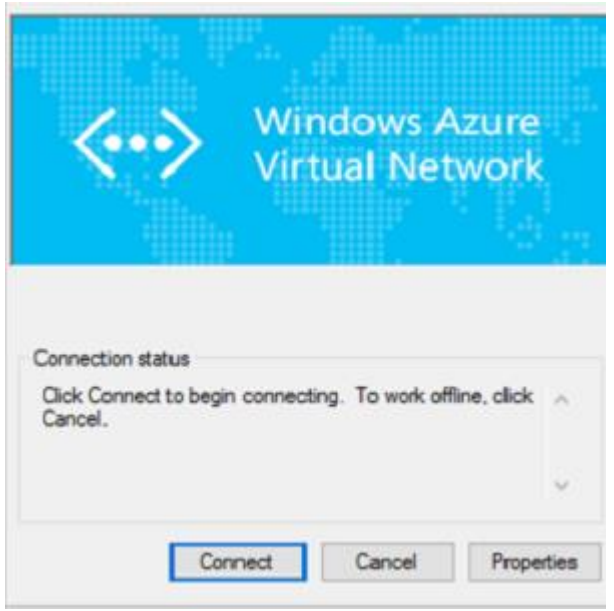
Click **Save** at the top of the page to save all of the configuration settings.



Download and install the VPN client configuration package

The VPN client configuration files contain settings to configure devices to connect to a VNet over a P2S connection.

Connect to Azure



1.To connect to your VNet, on the client computer, navigate to VPN connections and locate the VPN connection that you created. It is named the same name as your virtual network.
Click **Connect.** A pop-up message may appear that refers to using the certificate.
Click **Continue** to use elevated privileges.
2.On the **Connection status** page, click **Connect** to start the connection. If you see a **Select Certificate screen**, verify that the client certificate showing is the one that you want to use to connect. If it is not, use the drop-down arrow to select the correct certificate, and then click **OK.**

Your connection is established.

To verify your connection

1.To verify that your VPN connection is active, open an elevated command prompt, and run *ipconfig/all.*

```
PPP adapter VNet1:
   Connection-specific DNS Suffix .:
   Description.....................: VNet1
   Physical Address................:
   DHCP Enabled....................: No
   Autoconfiguration Enabled.......: Yes
   IPv4 Address....................: 172.16.201.3(Preferred)
   Subnet Mask.....................: 255.255.255.255
   Default Gateway.................:
   NetBIOS over Tcpip..............: Enabled
```

Export the client certificate

When you generate a client certificate, it's automatically installed on the computer that you used to generate it. If you want to install the client certificate on another client computer, you need to export the client certificate that you generated.

1.To export a client certificate, open **Manage user certificates.** The client certificates that you generated are, by default, located in 'Certificates - Current User\Personal\Certificates'. Right-click the client certificate that you want to export, click **all tasks**, and then click **Export** to open the **Certificate Export Wizard.**

2.In the **Certificate Export Wizard**, click **Next** to continue.

3.Select **Yes**, export the **private key,** and then click **Next.**

4.On the **Export File Format** page, leave the defaults selected. Make sure that **Include all certificates in the certification path if possible** is selected. This setting additionally exports the root certificate information that is required for successful client authentication. Without it, client authentication fails because the client doesn't have the trusted root certificate. Then, click **Next.**

5.On the **Security** page, you must protect the private key. If you select to use a password, make sure to record or remember the password that you set for this certificate. Then, click **Next.**

6.On the **File to Export**, Browse to the location to which you want to export the certificate. For File name, name the certificate file. Then, click **Next.**

7.Click **Finish** to export the certificate.

Install certificate - Windows

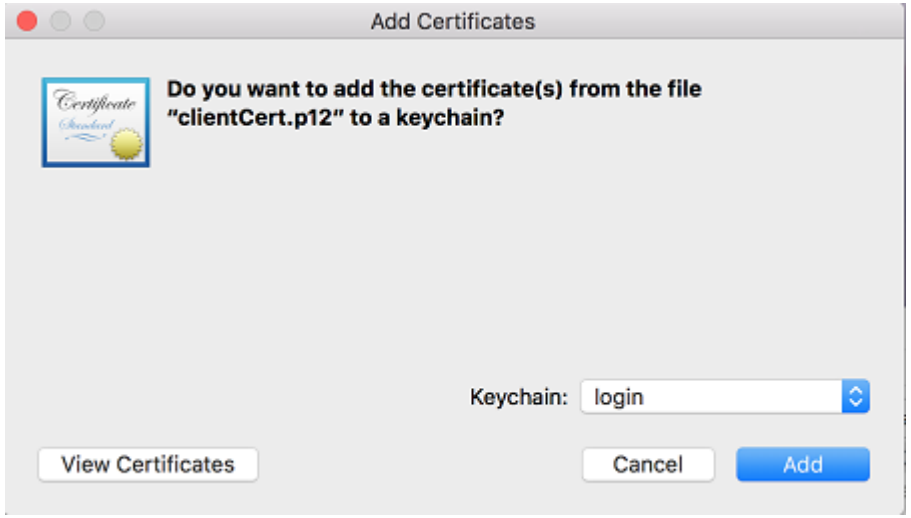
If you want to create a P2S connection from a client computer other than the one you used to generate the client certificates, you need to install a client certificate. When installing a client certificate, you need the password that was created when the client certificate was exported.

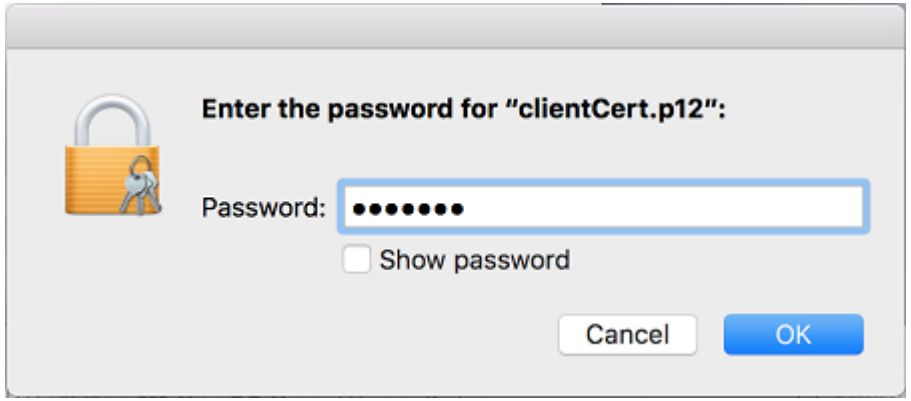
1.Locate and copy the .pfx file to the client computer. On the client computer, double-click the .pfx file to install. Leave the **Store Location as Current User,** and then click **Next.**
2.On the **File** to import page, don't make any changes. Click **Next.**
3.On the **Private key protection page**, input the password for the certificate, or verify that the security principal is correct, then click **Next.**
4.On the **Certificate Store** page, leave the default location, and then click **Next.**
5.Click **Finish.** On the **Security Warning for** the certificate installation, click **Yes.** You can feel comfortable clicking 'Yes' because you generated the certificate. The certificate is now successfully imported.

Install certificate - Mac

Mac VPN clients are supported for the Resource Manager deployment model only. They are not supported for the classic deployment model. When installing a client certificate, you need the password that was created when the client certificate was exported.
1.Locate the .pfx certificate file and copy it to your Mac. You can get the certificate to the Mac in several ways, for example, you can email the certificate file.
2.After the certificate copied to the Mac, double-click the certificate to open the **Add Certificates** box, the click **Add** to begin the install.

3.Enter the password that you created when the client certificate was exported. The password protects the private key of the certificate. Click OK to complete the installation.





Site to Site VPN Connectivity

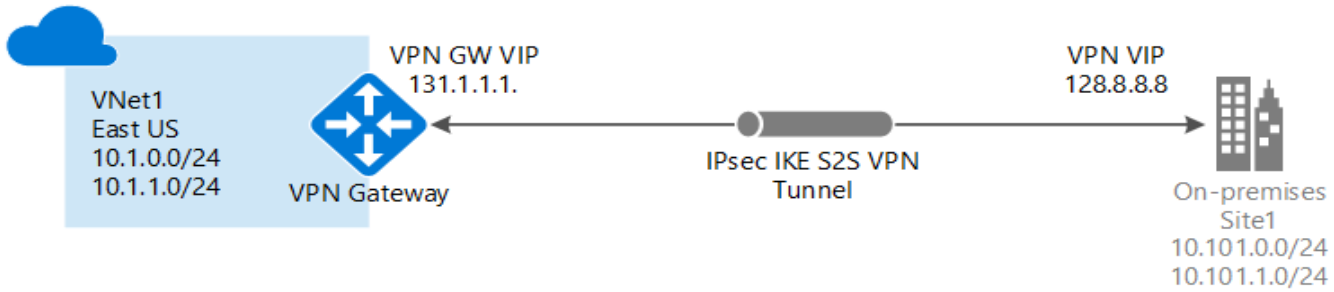
A Site-to-Site VPN gateway connection is used to connect your on-premises network to an Azure virtual network over an IPsec/IKE (IKEv1 or IKEv2) VPN tunnel. This type of connection requires a VPN device located on-premises that has an externally facing public IP address assigned to it.

Make sure you have a compatible VPN device and someone who is able to configure it. For more information about compatible VPN devices and device configuration

Verify that you have an externally facing public IPv4 address for your VPN device. This IP address cannot be located behind a NAT.

Steps:

1. Create Virtual Network
2. Create Gateway Subnet
3. Create Virtual Network Gateway
4. Create Local Network Gateway
5. Create Site-to-Site VPN



Example values

The examples in this article use the following values. You can use these values to create a test environment, or refer to them to better understand the examples in this article. For more information about VPN Gateway settings in general,

- **VNet Name:** TestVNet1
- **Address Space:** 10.1.0.0/16
- **Subscription:** The subscription you want to use
- **Resource Group:** TestRG1
- **Location:** East US
- **Subnet: FrontEnd:** 10.1.0.0/24, BackEnd: 10.1.1.0/24 (optional for this exercise)
- **Gateway Subnet name:** GatewaySubnet (this will auto-fill in the portal)
- **Gateway Subnet address range:** 10.1.255.0/27
- **DNS Server:** 8.8.8.8 - Optional. The IP address of your DNS server.
- **Virtual Network Gateway Name:** VNet1GW
- **Public IP:** VNet1GWIP
- **VPN Type:** Route-based
- **Connection Type:** Site-to-site (IPsec)
- **Gateway Type:** VPN
- **Local Network Gateway Name:** Site1
- **Connection Name:** TestVNet1toSite1
- **Shared key:** For this example, we use **abc123**. But, you can use whatever is compatible with your VPN hardware. The important thing is that the values match on both sides of the connection.

1. Create a virtual network

1. From a browser, navigate to the Azure portal and sign in with your Azure account.
2. Click **Create a resource**. In the **Search the marketplace** field, type 'virtual network'. Locate **Virtual network** from the returned list and click to open the **Virtual Network** page.
3. Near the bottom of the Virtual Network page, from the **Select a deployment model** list, select **Resource Manager,** and then click **Create.** This opens the 'Create virtual network' page.

4. On the **Create virtual network** page, configure the VNet settings. When you fill in the fields, the red exclamation mark becomes a green check mark when the characters entered in the field are valid.

Name: Enter the name for your virtual network. In this example, we use VNet1.

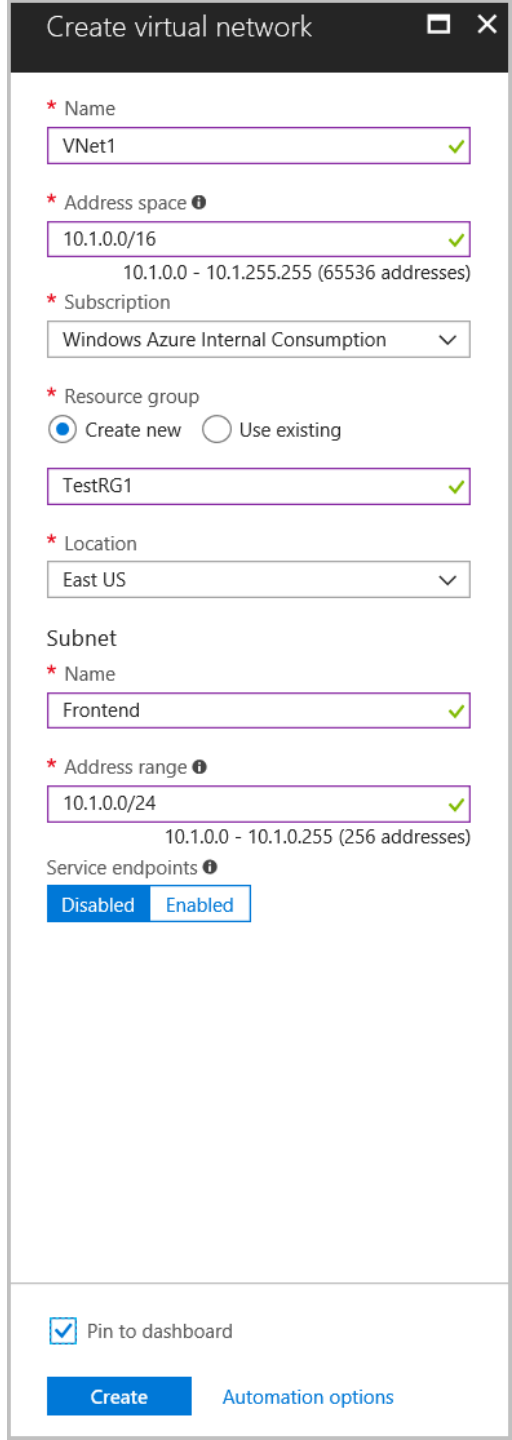
Address space: Enter the address space. If you have multiple address spaces to add, add your first address space. You can add additional address spaces later, after creating the VNet. Make sure that the address space that you specify does not overlap with the address space for your on-premises location.

Subscription: Verify that the subscription listed is the correct one. You can change subscriptions by using the drop-down.

Resource group: Select an existing resource group, or create a new one by typing a name for your new resource group. If you are creating a new group, name the resource group according to your planned configuration values. For more information about resource groups, visit Azure Resource Manager Overview.

Location: Select the location for your VNet. The location determines where the resources that you deploy to this VNet will reside.

Subnet: Add the first subnet name and subnet address range. You can add additional subnets and the gateway subnet later, after creating this VNet.



5. Select Pin to dashboard if you want to be able to find your VNet easily on the dashboard, and then click Create. After clicking Create, you will see a tile on your dashboard that will reflect the progress of your VNet. The tile changes as the VNet is being created.

2. Specify a DNS server

DNS is not required to create a Site-to-Site connection. However, if you want to have name resolution for resources that are deployed to your virtual network, you should specify a DNS server. This setting lets you specify the DNS server that you want to use for name resolution for this virtual network. It does not create a DNS server.

1. On the **Settings** page for your virtual network, navigate to **DNS Servers** and click to open the **DNS servers** page.

   DNS Servers: Select Custom.
   Add DNS server: Enter the IP address of the DNS server that you want to use for name resolution.

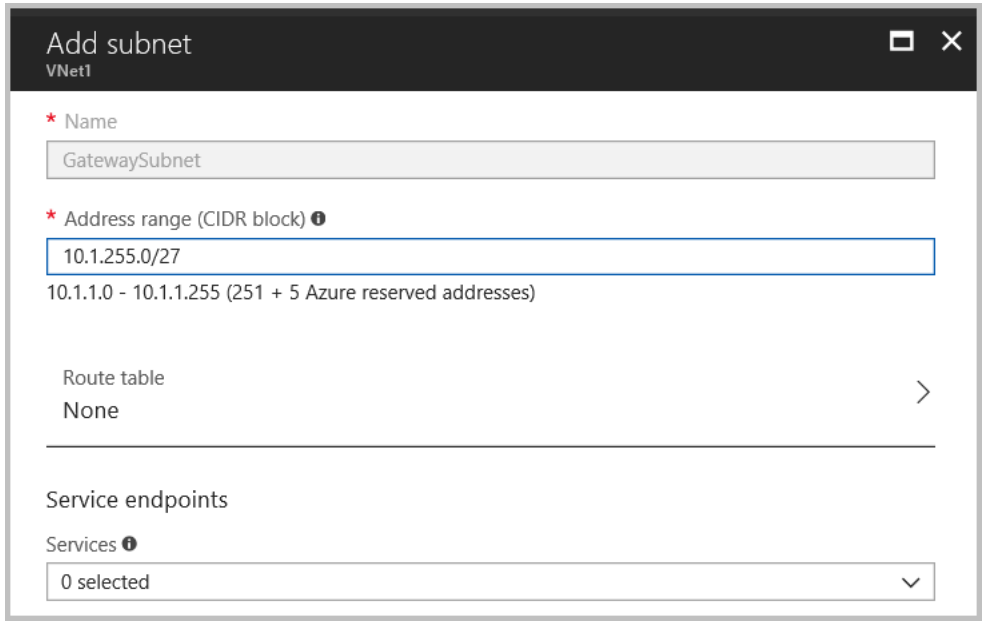
2. When you are done adding DNS servers, click Save at the top of the page.

3. Create the gateway subnet

The virtual network gateway uses specific subnet called the gateway subnet. The gateway subnet is part of the virtual network IP address range that you specify when configuring your virtual network. It contains the IP addresses that the virtual network gateway resources and services use. The subnet must be named 'GatewaySubnet' in order for Azure to deploy the gateway resources. You can't specify a different subnet to deploy the gateway resources to. If you don't have a subnet named 'GatewaySubnet', when you create your VPN gateway, it will fail.

When you create the gateway subnet, you specify the number of IP addresses that the subnet contains. The number of IP addresses needed depends on the VPN gateway configuration that you want to create. Some configurations require more IP addresses than others. We recommend that you create a gateway subnet that uses a /27 or /28.

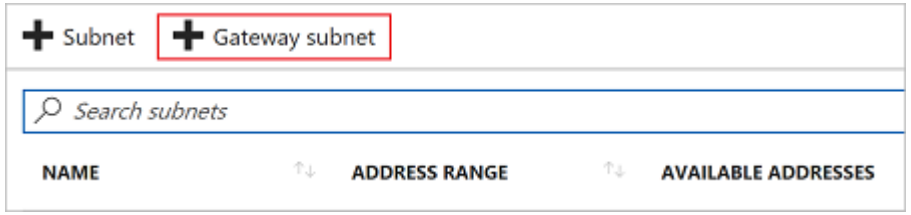
If you see an error that specifies that the address space overlaps with a subnet, or that the subnet is not contained within the address space for your virtual network, check your VNet address range. You may not have enough IP addresses available in the address range you created for your virtual network. For example, if your default subnet encompasses the entire address range, there are no IP addresses left to create additional subnets.



You can either adjust your subnets within the existing address space to free up IP addresses, or specify an additional address range and create the gateway subnet there.

1.In the portal, navigate to the virtual network for which you want to create a virtual network gateway.

2.In the **Settings** section of your VNet page, click **Subnets** to expand the Subnets page.



3.On the **Subnets** page, click **+Gateway subnet** at the top to open the **Add subnet** page.

4.The Name for your subnet is automatically filled in with the value 'GatewaySubnet'. The GatewaySubnet value is required in order for Azure to recognize the subnet as the gateway subnet. Adjust the auto-filled Address range values to match your configuration requirements.

5.To create the subnet, click **OK** at the bottom of the page.

4. Create the VPN gateway

1.On the left side of the portal page, click + and type 'Virtual Network Gateway' in search. In **Results**, locate and click **Virtual network gateway.**

2.At the bottom of the 'Virtual network gateway' page, click **Create.** This opens the **Create virtual network gateway** page.

3.On the **Create virtual network gateway** page, specify the values for your virtual network gateway.

Name: Name your gateway. This is not the same as naming a gateway subnet. It's the name of the gateway object you are creating.

Gateway type: Select **VPN.** VPN gateways use the virtual network gateway type VPN.

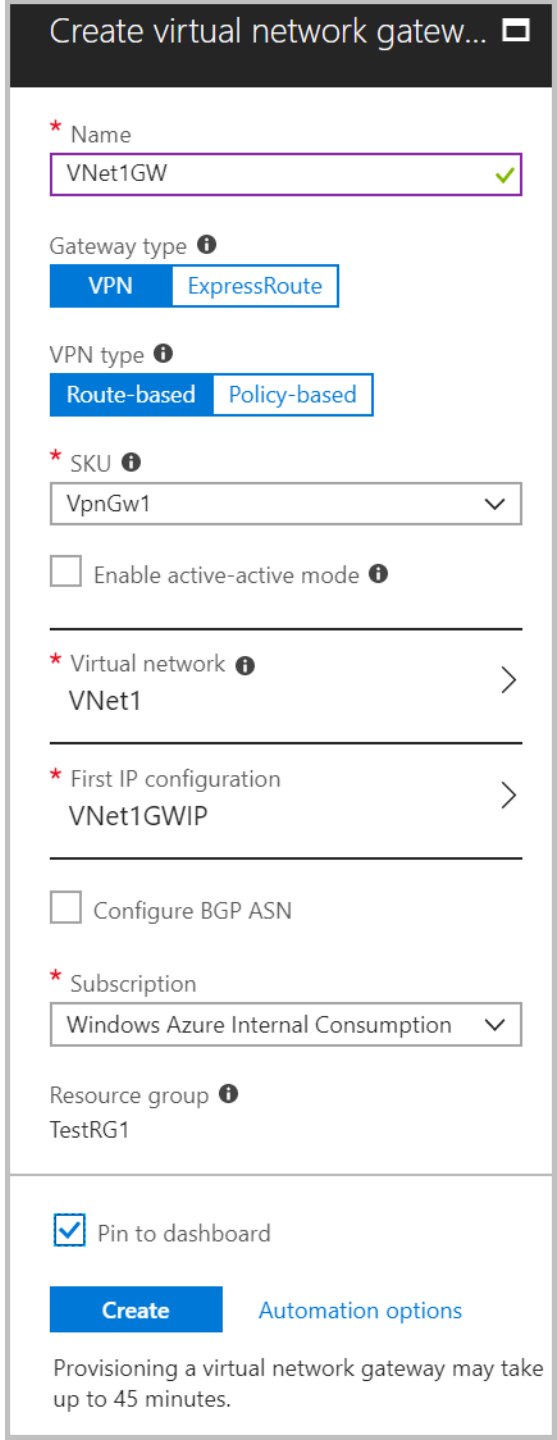
VPN type: Select the VPN type that is specified for your configuration. Most configurations require a Route-based VPN type.

SKU: Select the gateway SKU from the dropdown. The SKUs listed in the dropdown depend on the VPN type you select. For more information about gateway SKUs, see Gateway SKUs.

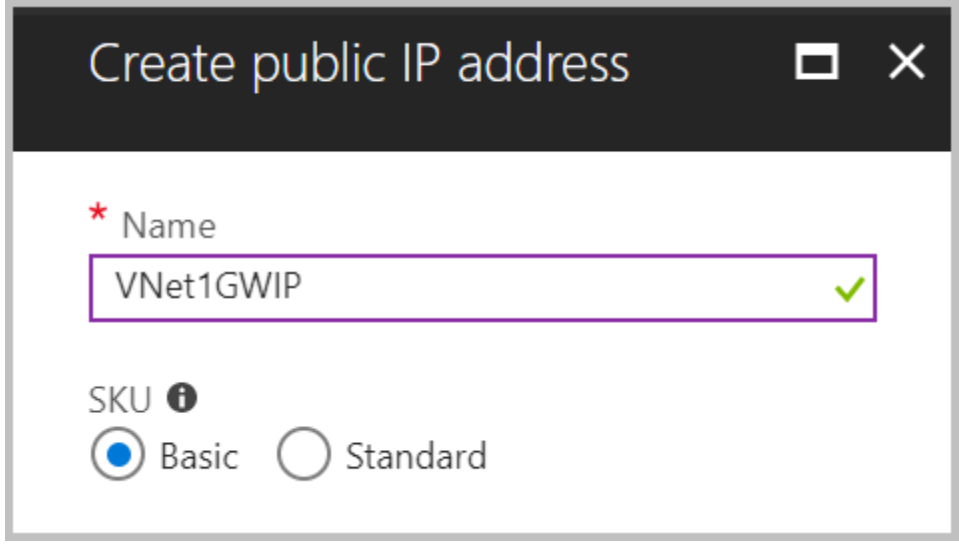
Location: You may need to scroll to see Location. Adjust the Location field to point to the location where your virtual network is located. If the location is not pointing to the region where your virtual network resides, when you select a virtual network in the next step, it will not appear in the drop-down list.

Virtual network: Choose the virtual network to which you want to add this gateway. Click Virtual network to open the 'Choose a virtual network' page. Select the VNet. If you don't see your VNet, make sure the Location field is pointing to the region in which your virtual network is located.

Gateway subnet address range: You will only see this setting if you did not previously create a gateway subnet for your virtual network. If you previously created a valid gateway subnet, this setting will not appear.



First IP configuration: The 'Choose public IP address' page creates a public IP address object that gets associated to the VPN gateway. The public IP address is dynamically assigned to this object when the VPN gateway is created. VPN Gateway currently only supports Dynamic Public IP address allocation. However, this does not mean that the IP address changes after it has been assigned to your VPN gateway. The only time the Public IP address changes is when the gateway is deleted and re-created. It doesn't change across resizing, resetting, or other internal maintenance/upgrades of your VPN gateway.



•First, click **Create gateway IP configuration** to open the 'Choose public IP address' page, then click **+Create new** to open the 'Create public IP address' page.

•Next, input a **Name** for your public IP address. Leave the SKU as **Basic** unless there is a specific reason to change it to something else, then click **OK** at the bottom of this page to save your changes.

4.Verify the settings. You can select **Pin to dashboard** at the bottom of the page if you want your gateway to appear on the dashboard.
5.Click **Create** to begin creating the VPN gateway. The settings are validated and you'll see the "Deploying Virtual network gateway" tile on the dashboard. Creating a gateway can take up to 45 minutes. You may need to refresh your portal page to see the completed status.

After the gateway is created, view the IP address that has been assigned to it by looking at the virtual network in the portal. The gateway appears as a connected device. You can click the connected device (your virtual network gateway) to view more information.

5. Create the local network gateway

The local network gateway typically refers to your on-premises location. You give the site a name by which Azure can refer to it, then specify the IP address of the on-premises VPN device to which you will create a connection. You also specify the IP address prefixes that will be routed through the VPN gateway to the VPN device. The address prefixes you specify are the prefixes located on your on-premises network. If your on-premises network changes or you need to change the public IP address for the VPN device, you can easily update the values later.

1. In the portal, click **+Create a resource.**
2. In the search box, type **Local network gateway,** then press **Enter** to search. This will return a list of results. Click **Local network gateway,** then click the Create button to open the **Create local network gateway** page.
3.On the **Create local network gateway** page, specify the values for your local network gateway.

Name: Specify a name for your local network gateway object.

IP address: This is the public IP address of the VPN device that you want Azure to connect to. Specify a valid public IP address. The IP address cannot be behind NAT and has to be reachable by Azure. If you don't have the IP address right now, you can use the values shown in the example, but you'll need to go back and replace your placeholder IP address with the public IP address of your VPN device. Otherwise, Azure will not be able to connect.

Address Space refers to the address ranges for the network that this local network represents. You can add multiple address space ranges. Make sure that the ranges you specify here do not overlap with ranges of other networks that you want to connect to. Azure will route the address range that you specify to the on-premises VPN device IP address. Use your own values here if you want to connect to your on-premises site, not the values shown in the example.

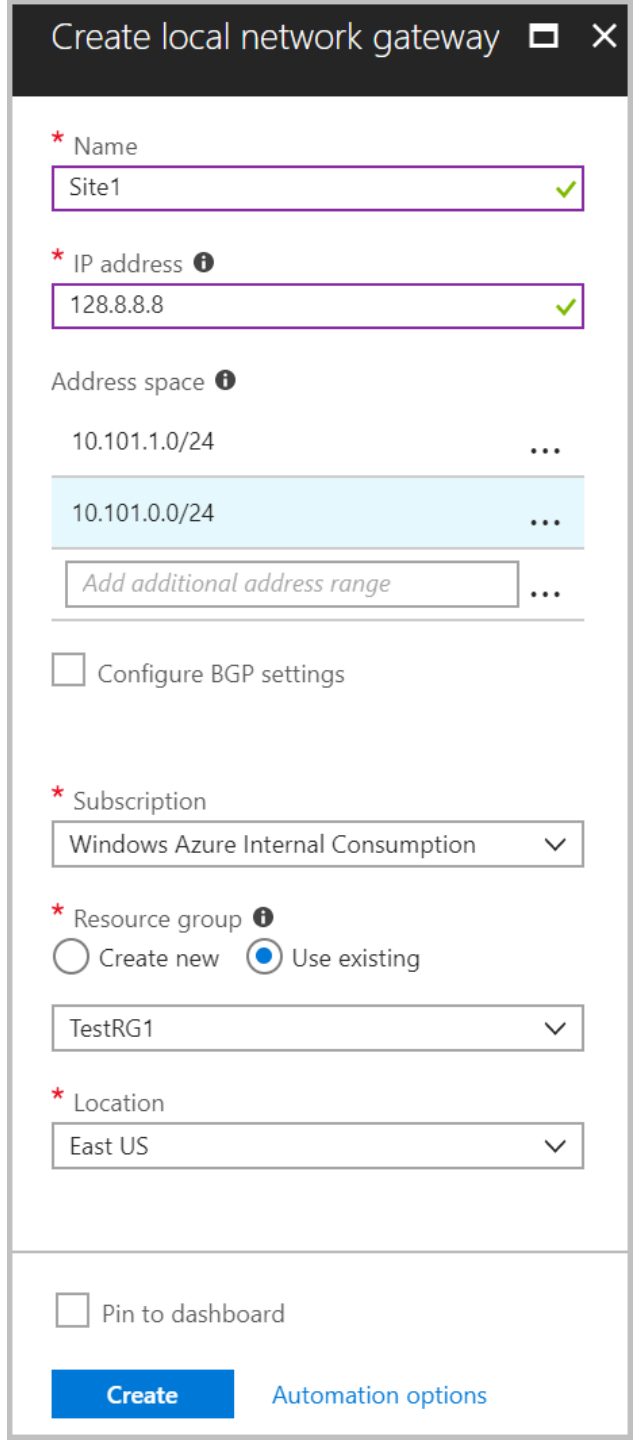
Configure BGP settings: Use only when configuring BGP. Otherwise, don't select this.

Subscription: Verify that the correct subscription is showing.

Resource Group: Select the resource group that you want to use. You can either create a new resource group, or select one that you have already created.

Location: Select the location that this object will be created in. You may want to select the same location that your VNet resides in, but you are not required to do so.

When you have finished specifying the values, click the Create button at the bottom of the page to create the local network gateway.



6. Configure your VPN device

Site-to-Site connections to an on-premises network require a VPN device. In this step, you configure your VPN device. When configuring your VPN device, you need the following:

A shared key. This is the same shared key that you specify when creating your Site-to-Site VPN connection. In our examples, we use a basic shared key. We recommend that you generate a more complex key to use.

The Public IP address of your virtual network gateway. You can view the public IP address by using the Azure portal, PowerShell, or CLI. To find the Public IP address of your VPN gateway using the Azure portal, navigate to **Virtual network gateways**, then click the name of your gateway.

7. Create the VPN connection

Create the Site-to-Site VPN connection between your virtual network gateway and your on-premises VPN device.

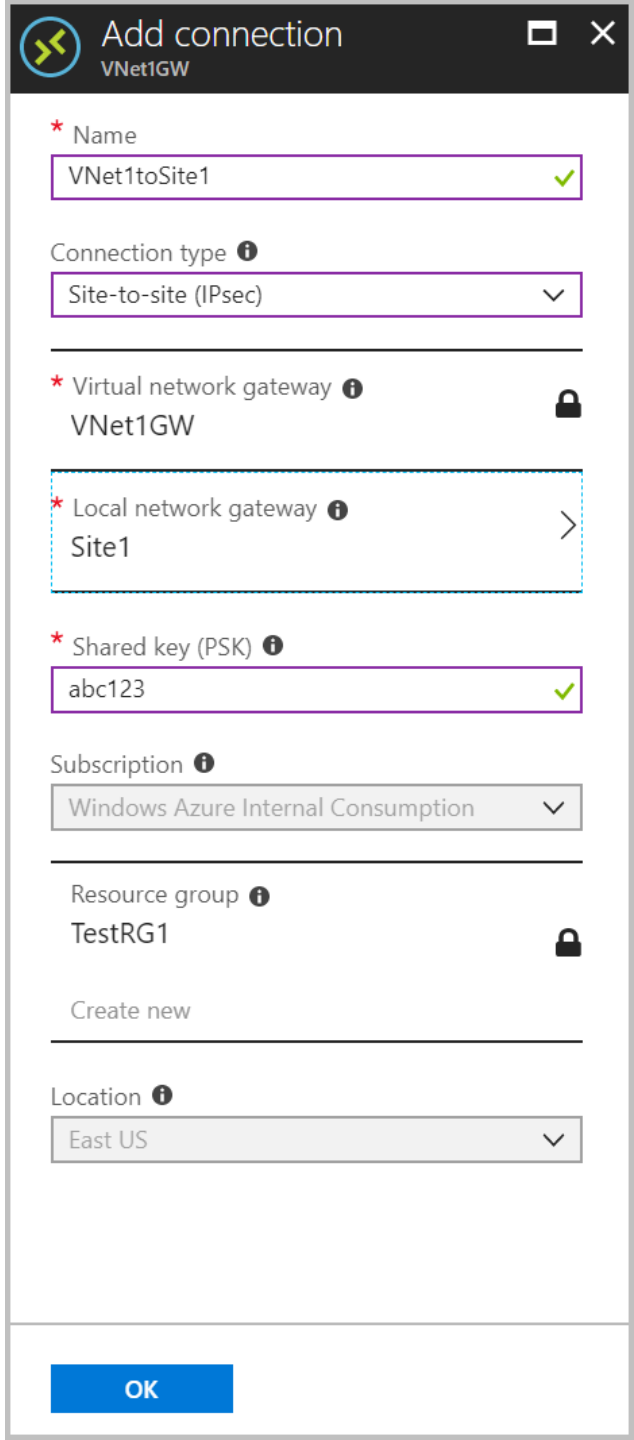
1. Navigate to and open the page for your virtual network gateway. There are multiple ways to navigate. You can navigate to the gateway 'VNet1GW' by going to **TestVNet1 -> Overview -> Connected devices -> VNet1GW.**

2. On the page for VNet1GW, click **Connections.** At the top of the Connections page, click **+Add** to open the **Add connection** page.

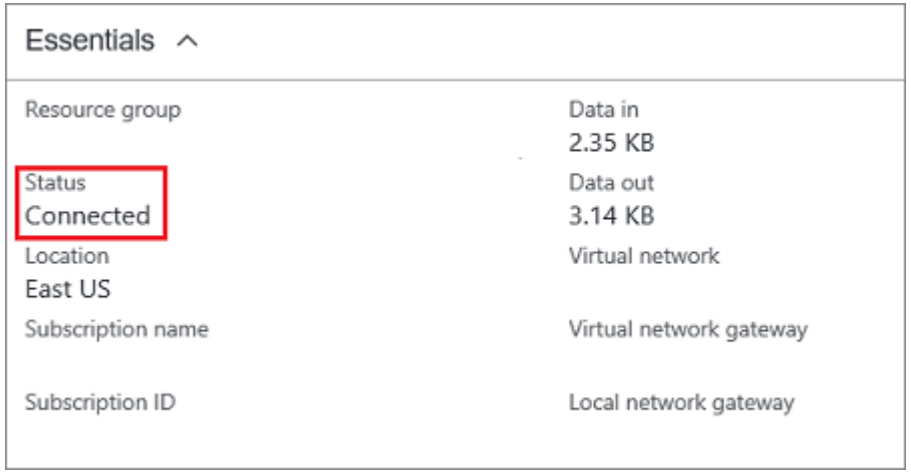
3.On the **Add connection** page, configure the values for your connection.

Name: Name your connection.
Connection type: Select Site-to-site(IPSec).
Virtual network gateway: The value is fixed because you are connecting from this gateway.
Local network gateway: Click Choose a local network gateway and select the local network gateway that you want to use.
Shared Key: the value here must match the value that you are using for your local on-premises VPN device. The example uses 'abc123', but you can (and should) use something more complex. The important thing is that the value you specify here must be the same value that you specify when configuring your VPN device.
The remaining values for **Subscription, Resource Group**, and **Location** are fixed.

Click **OK** to create your connection. You'll see Creating Connection flash on the screen.
You can view the connection in the **Connections** page of the virtual network gateway. The Status will go from Unknown to Connecting, and then to Succeeded.



8. Verify the VPN connection



1. In the Azure portal, you can view the connection status of a Resource Manager VPN Gateway by navigating to the connection. The following steps show one way to navigate to your connection and verify.

2.In the Azure portal, click All resources and navigate to your virtual network gateway.
On the blade for your virtual network gateway, click Connections. You can see the status of each connection.

3. Click the name of the connection that you want to verify to open Essentials. In Essentials, you can view more information about your connection. The Status is 'Succeeded' and 'Connected' when you have made a successful connection.

Module 3: Implementing virtual machines

Lesson 1: Overview of Azure VMs
Lesson 2: Planning deployment of Azure VMs
Lesson 3: Deploying Azure VMs
Lesson 4: Overview of classic Azure VMs

Lab: Deploying Azure VMs

- Creating Azure VMs by using the Azure portal, Azure PowerShell, and Azure CLI
- Validating Azure VM deployment

Module 4: Managing Azure VMs

Lesson 1:Configuring Azure VMs
Lesson 2:Managing disks of Azure VMs
Lesson 3:Managing and monitoring Azure VMs
Lesson 4:Managing classic Azure VMs

Lab : Managing Azure VMs

- Implementing Desired State Configuration (DSC)
- Implementing Storage Spaces–based volumes

What are Azure VMs?

- Use Azure VMs to:
  - Extend your datacenter to increase agility
  - Migrate your workloads from on-premises datacenters or from other cloud providers
  - Implement test or development
- Key differences when using Azure VMs:
  - Read-only VM console access
- You can create Azure VMs by using:
  - The Azure Portal
  - Azure PowerShell or Azure CLI
  - Azure Resource Manager templates

Azure VM sizes

- A-series:
  - Basic: No load balancing or auto-scaling support
  - Standard:
    - A0-A7, general computing
    - A8-A11, compute intensive
- D-series:
  - Faster CPUs and local Hyper-V host SSD (temporary disk)
- Dv2 series:
  - 35% faster CPU than D-series
- G-series:
  - Largest VMs (up to 448 GB of RAM and 64 data disks)
- DS, DSv2, and GS series:
  - Support for Premium Storage (SSD for operating system and data disks)

Burstable VM Series (Preview)

Low cost VM Sku

Flexible CPU Usage

Workloads:

- Small Databases
- Dev/Test Environments
- Web Servers

Credit-based system

Region Availability

- US West 2
- US East
- Europe West
- Southeast Asia

Size	vCPU's	Memory: GiB	Local SSD: GiB	Baseline CPU Performance of VM	Max CPU Performance of VM
B1s	1	1	4	10%	100%
B1ms	1	2	4	20%	100%
B2s	2	4	8	40%	200%
B2ms	2	8	16	60%	200%
B4ms	4	16	32	90%	400%
B8ms	8	32	64	135%	800%

Provisioning to the Cloud

Getting Started

Management Portal(s)

Scripting
(Windows, Linux and Mac)

Azure Resource Manager
(ARM)

REST API

Select Image and VM Size

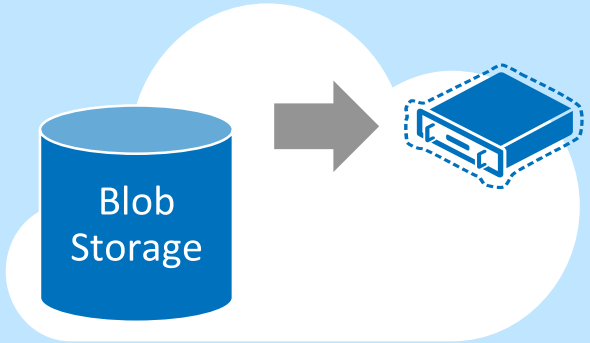
Windows Server

Linux

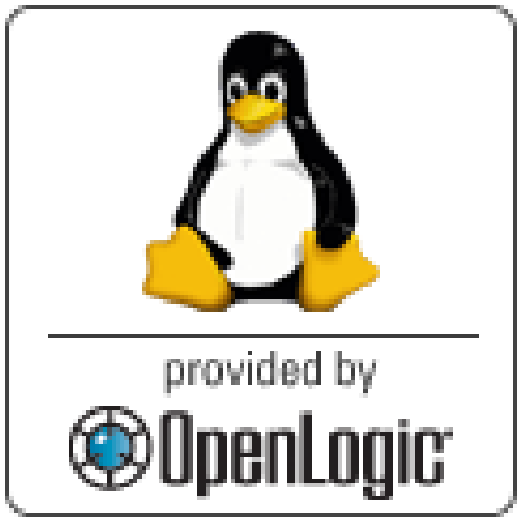
- A0 – A11
- D1 – D4/D11 – D14
- D1_v2 – D15_v2
- DS1 – DS4/DS11 – DS14
- DS1_v2 – DS15_v2
- G1 – G5
- GS1 – GS5

New Disk Persisted in Storage

Boot VM from New Disk



Cloud



Linux on Microsoft Azure

- Supported Versions:
  - SUSE SLES 11 Service Pack 3+ (SP3), SLES 12+
  - openSUSE 13.1+
  - CentOS 6.3+,7.0+ by OpenLogic*
  - Ubuntu Server 12.04.1+,14.04, 15.10 and 16.04
  - Oracle Linux 6.4+, 7.0+
  - Red Hat Enterprise Linux RHEL 6.7+, 7.1+
  - CoreOS 494.4.0+
- Specific versions are endorsed:
  - Integration Components
  - Testing and validation by partners
  - Bring other variants at your own risk**

- *Image provided by OpenLogic based on CentOS 6.5 – 7.1
- **Interoperation work will be Required
- *** Only Linux VMs in the gallery are supported

VM availability

- "Announced Single Instance Maintenance" means periods of Downtime related to network, hardware, or Service maintenance or upgrades impacting Single Instances. We will publish notice or notify you at least five (5) days prior to the commencement of such Downtime.

- "Availability Set" refers to two or more Virtual Machines deployed across different Fault Domains to avoid a single point of failure.

- "Data Disk" is a persistent virtual hard disk, attached to a Virtual Machine, used to store application data.

- "Fault Domain" is a collection of servers that share common resources such as power and network connectivity.

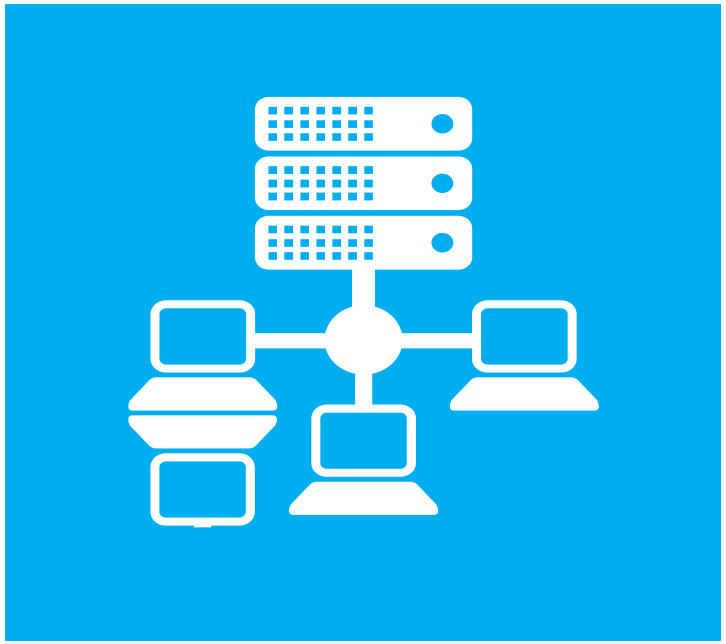
- "Operating System Disk" is a persistent virtual hard disk, attached to a Virtual Machine, used to store the Virtual Machine's operating system.

- "Single Instance" is defined as any single Microsoft Azure Virtual Machine that either is not deployed in an Availability Set or has only one instance deployed in an Availability Set.

- "Virtual Machine" refers to persistent instance types that can be deployed individually or as part of an Availability Set.

Service Level Agreements (SLA)

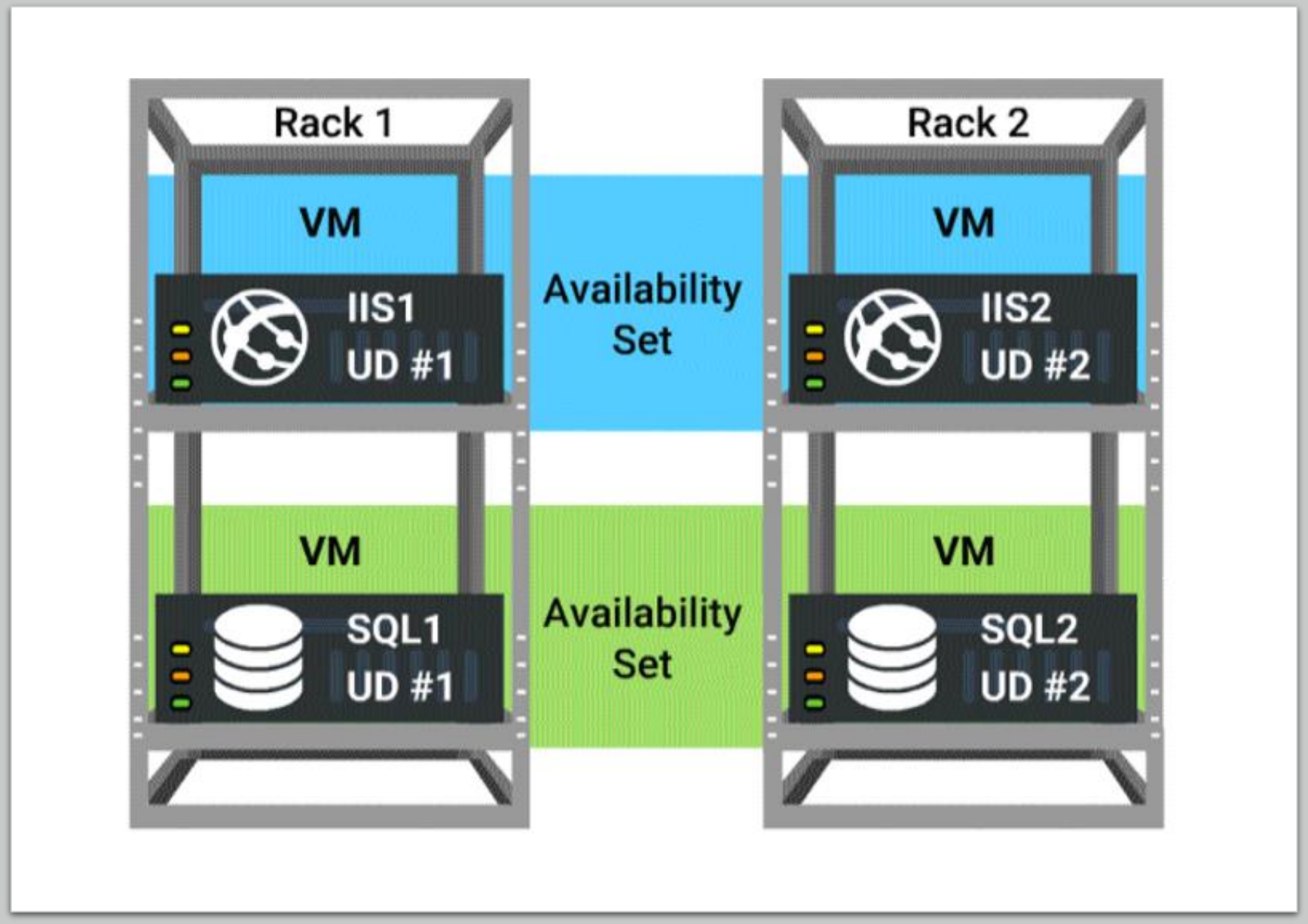
- For Cloud Services, we guarantee that when you deploy two or more role instances in different fault and upgrade domains, your Internet facing roles will have external connectivity at least 99.95% of the time.
- For all Internet facing Virtual Machines that have two or more instances deployed in the same Availability Set, we guarantee you will have external connectivity at least 99.95% of the time.
- For Virtual Network, we guarantee a 99.9% Virtual Network Gateway availability.
- "NO SLA" under the single instance



VM Availability

- To **ensure high availability** of an application, Azure places VMs into a logical grouping called a **Availability Set**.
- When deployed with a service, Azure ensures that the VMs in the Availability set are arranged across Fault Domains on different Racks. In case of a maintenance event or failure of one fault domain, at least one VM keeps running.
- Along with Load balancers, availability sets can provide up to 99.99% SLA for VMs.

VM Availability - Fault Domain: A fault domain is a set of hardware components (rack of resources like servers, power, etc.) that share a single point of failure. Web, worker and Virtual Machines are arranged in this hardware.
Azure deploys an application or service across multiple fault domains

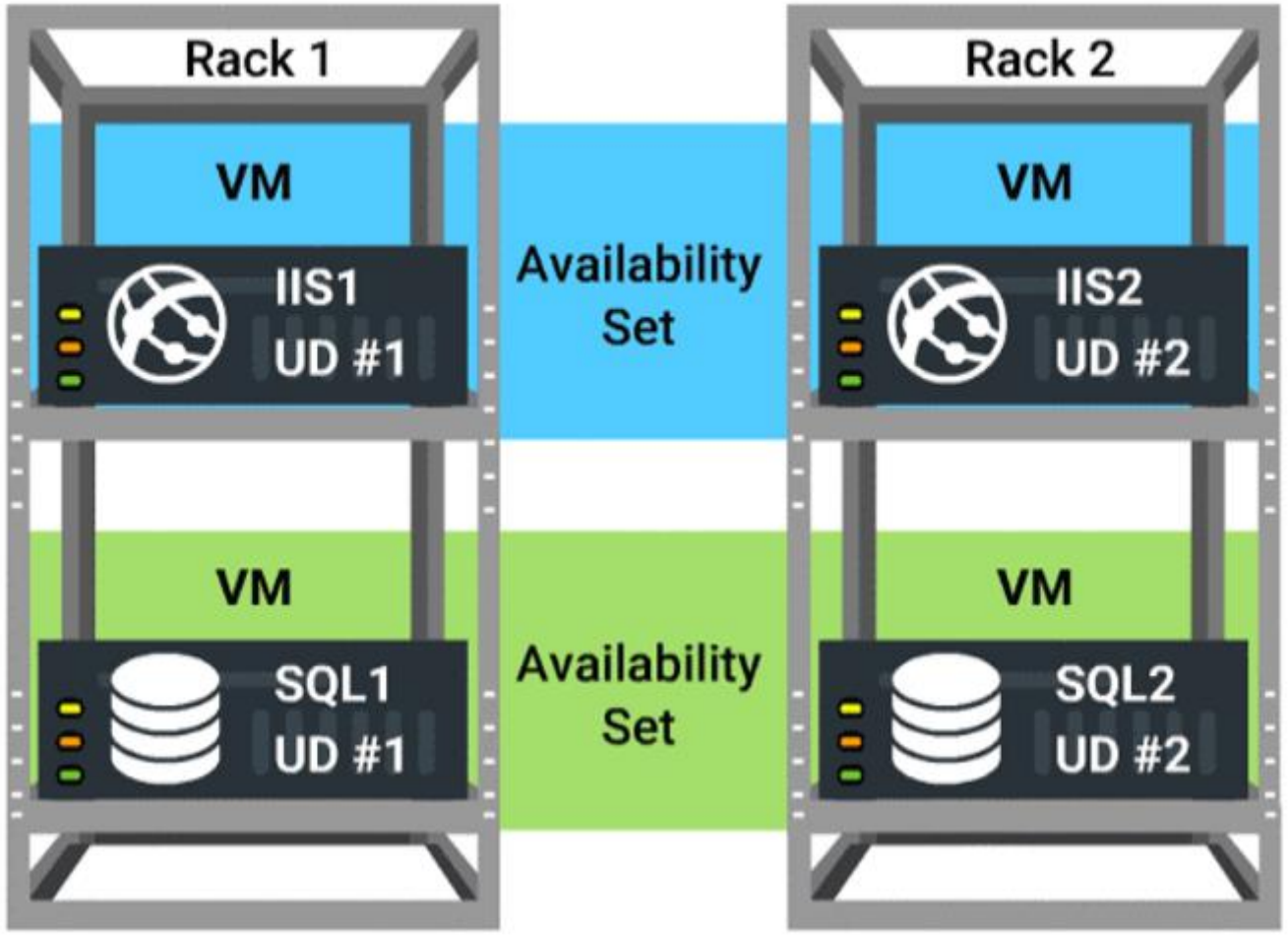


VM Availability - Fault Domain

- When you put VMs in to an availability set, Azure guarantees to spread them across Fault Domains and Update Domains. A Fault Domain (FD) is essentially a rack of servers. It consumes subsystems like network, power, cooling etc. So 2 VMs in the same availability set means Azure will provision them in to 2 different racks so that if say, the network or the power failed, only one rack would be affected.
- there are always only 3 fault domains: FD0 ,FD1 and FD2. It makes it seem like your VMs only get spread across 3 racks but that's not the case. They can be spread across more racks if you've got lots of VMs. But as far as your availability set is concerned FD0 and FD1 are a way of saying "This bit of infrastructure (FD0) is different to this bit (FD1). As you boot VMs in to an availability set, they get allocated like this – FD0, FD1, FD2, FD0, FD1, FD2 and so on. The pattern never changes. You've probably seen this diagram hundreds of times

VM Availability – Update Domain : Update domain in Azure means, that all physical servers in one update domain will get host updates like firmware, drivers and OS updates at the same time.

- In the Illustration UD#1 is getting updated but the user can access the content from UD#2.
- It provides Web or Worker role (within rack) instances with high availability by ensuring that only one of the Instances is down for an update at one time.



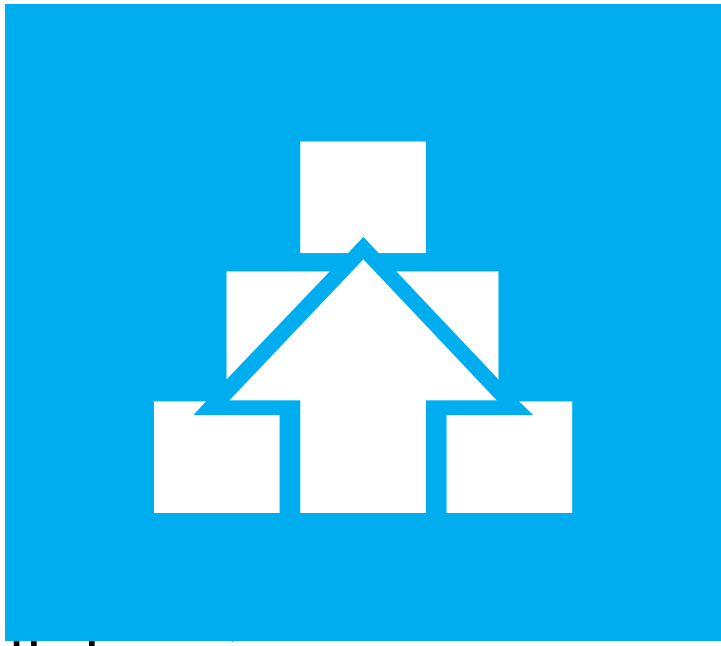
Update Domains

- Sometimes you need to update your app, or Microsoft needs to update the host on which your VM(s) are running. Note that with IaaS VMs, Microsoft does not automatically update your VMs. You have complete control (and responsibility) over that. But say if a serious security vulnerability is identified and a patch created. It's in Microsoft's interest to get that applied to the host underneath your VM as soon as possible. So how is that done without taking your service offline? Update Domains. It's similar to the FD methods, only this time, instead of an accidental failure, there is a purposeful move to take down one (or more) of your servers. So to make sure your service doesn't go offline because of an update, it will walk through your update domains one after the other.

VM	Fault Domain	Update Domain
VM0	0	0
VM1	1	1
VM2	2	2
VM3	0	3
VM4	1	4
VM5	2	0

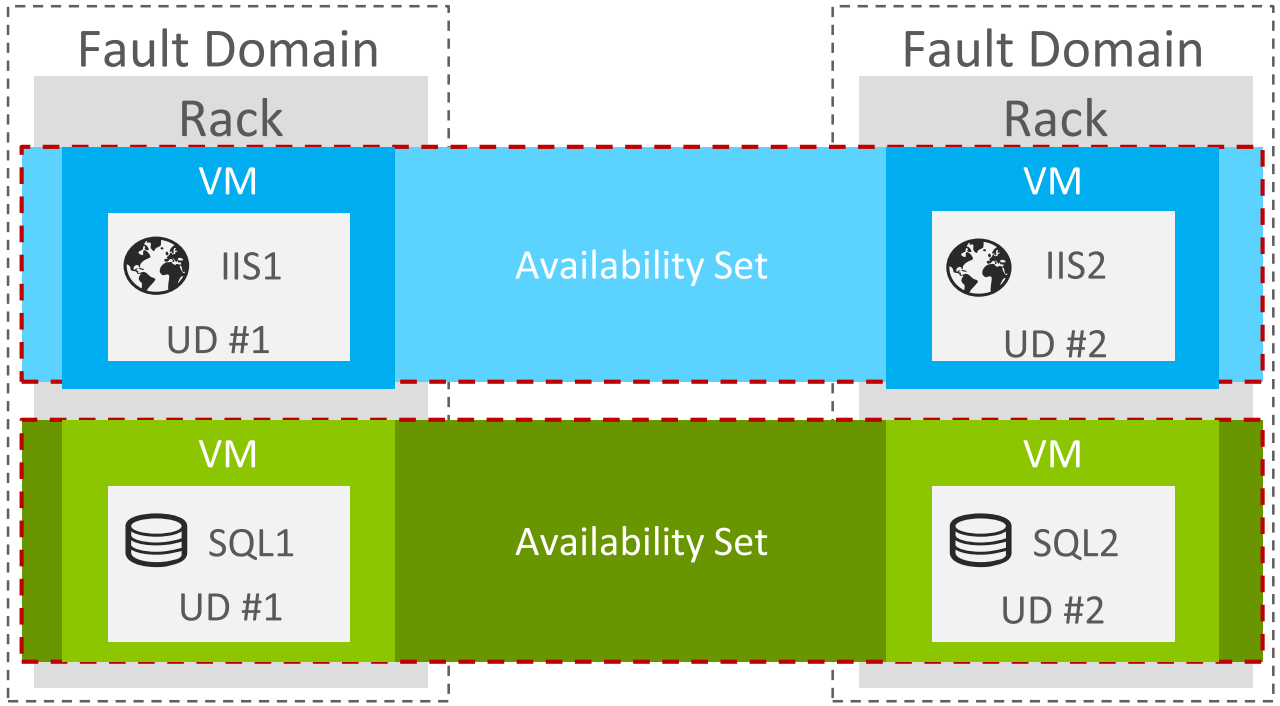
Fault and Update Domains

- Fault domains:
  - Represent groups of resources anticipated to fail together, i.e. same rack, same server
  - Fabric spreads instances across fault at least two fault domains
  - The number of fault domains is controlled by the Azure Fabric
  - Anticipated to fail together: share power source and network switch
  - 3 fault domains by default
- Update domains:
  - Represents groups of resources that will be updated together
  - Host OS updates honour service update domains
  - Specified in service definition
  - Default of five (up to 5)
  - More than 5 update domains allowed i.e. up to 20
- Fabric spreads role instances across update domains and fault domains

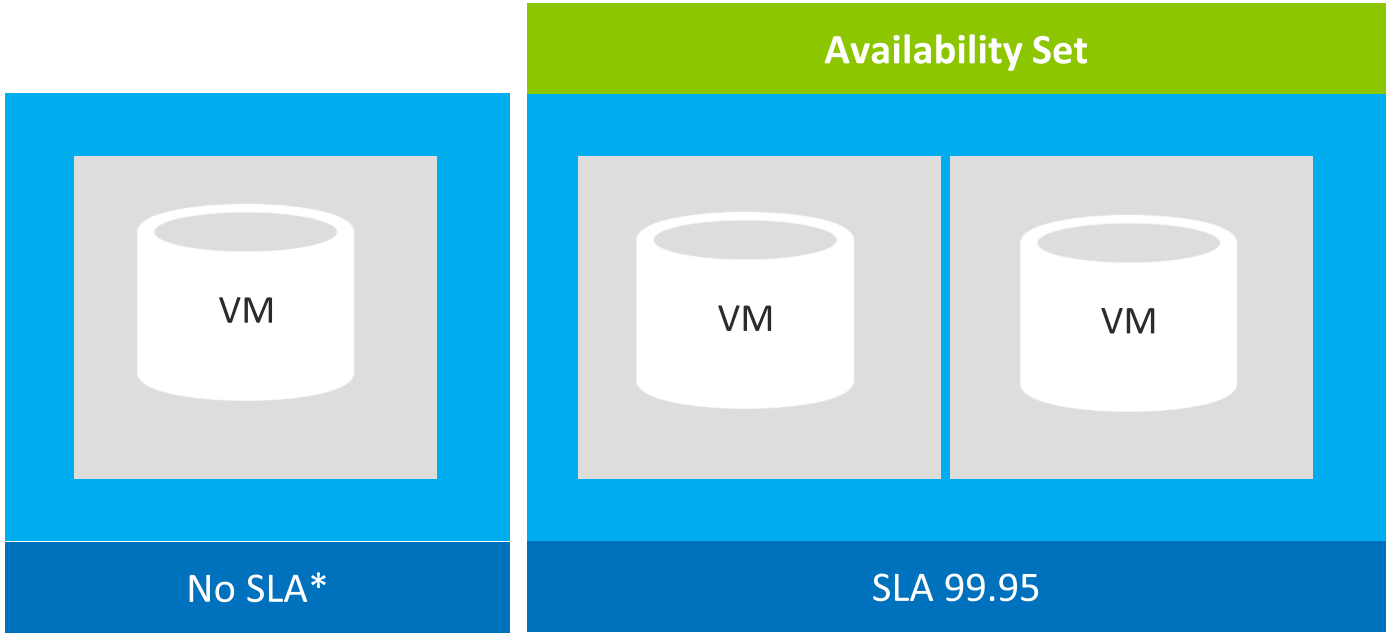


VM Availability Sets

- Update domains are honored by host OS updates

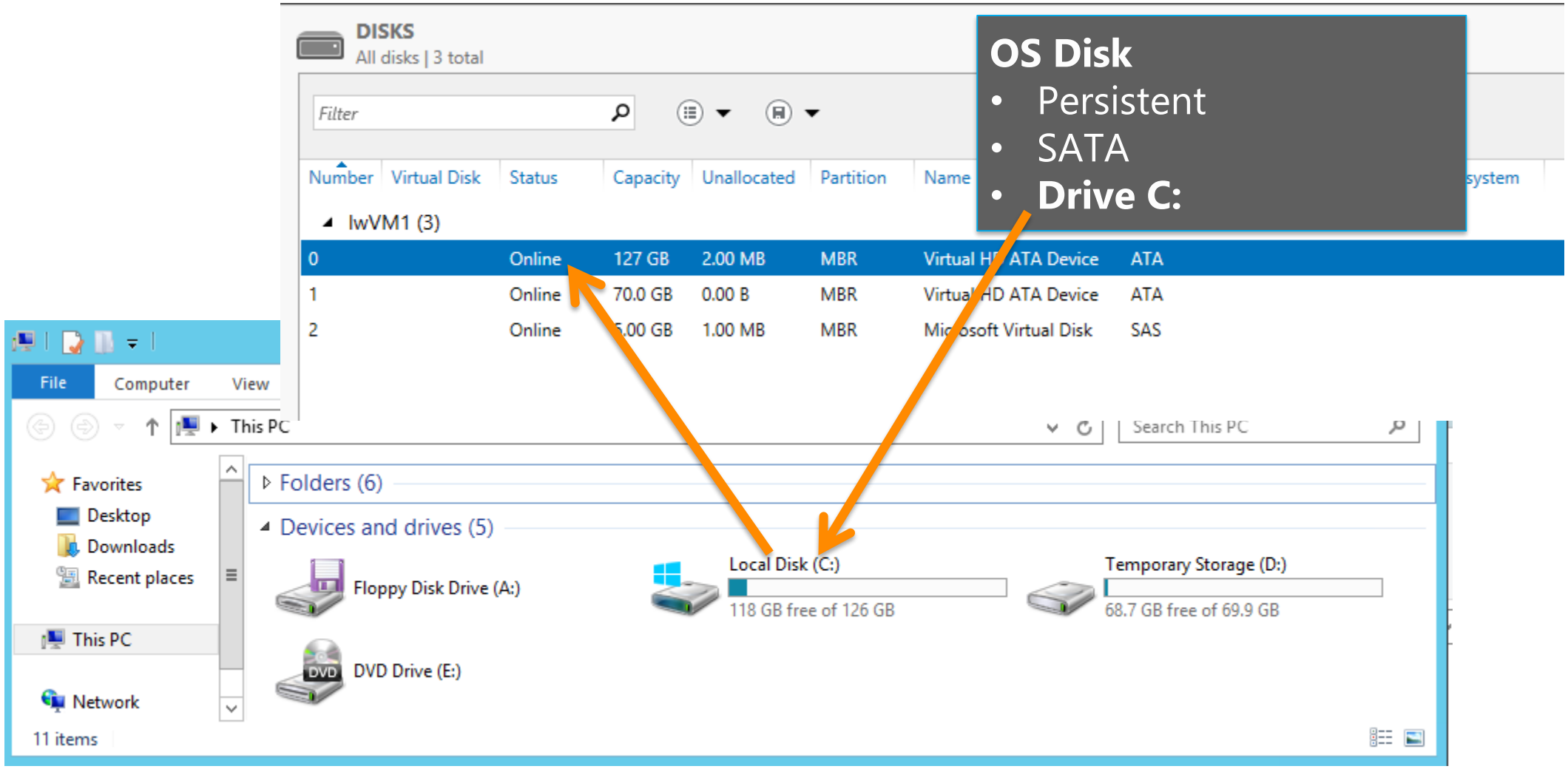


How Does this Relate to the SLA?



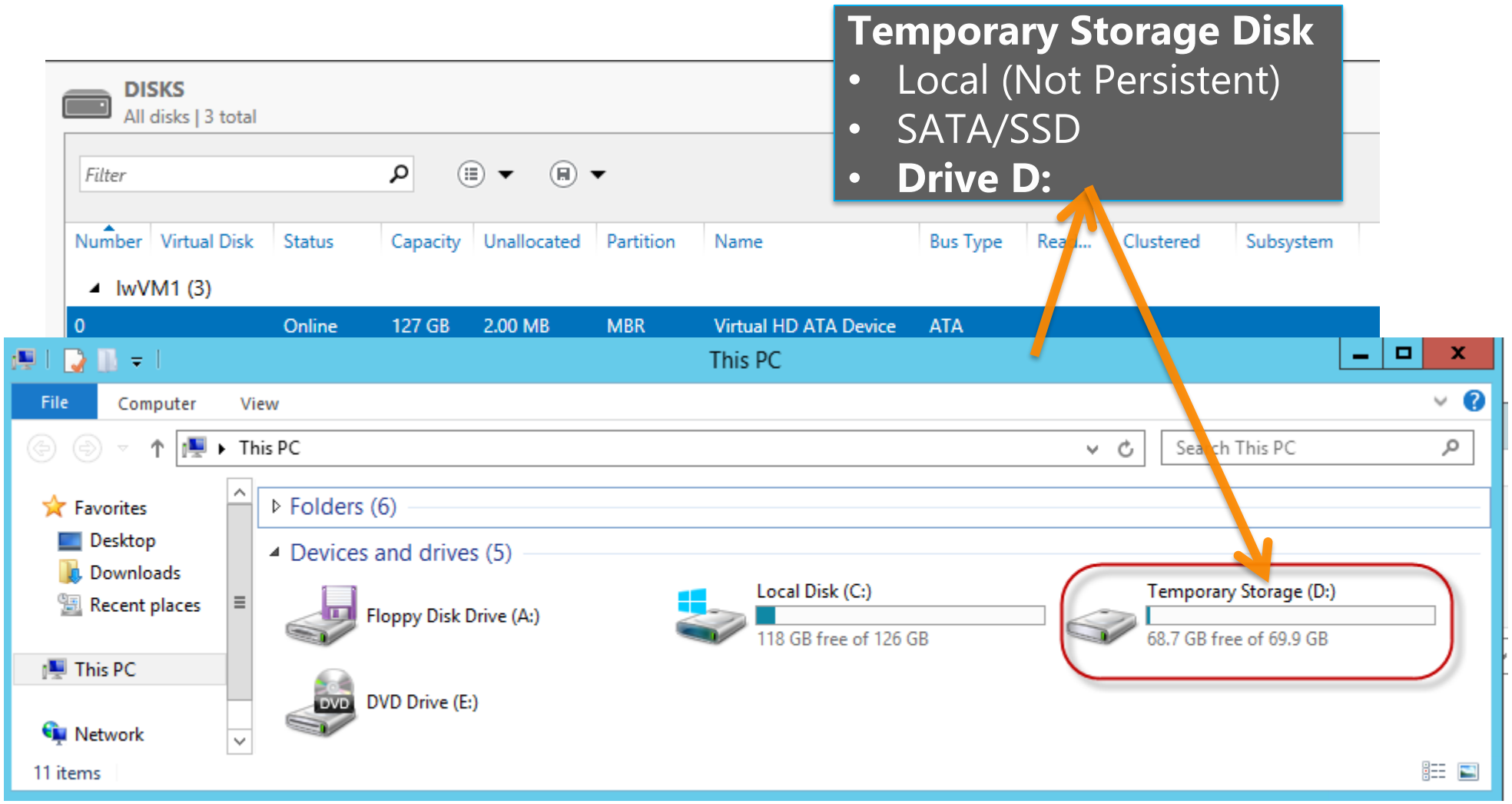
* No guaranteed SLA for single VM instance

VM Disk Layout – Windows OS

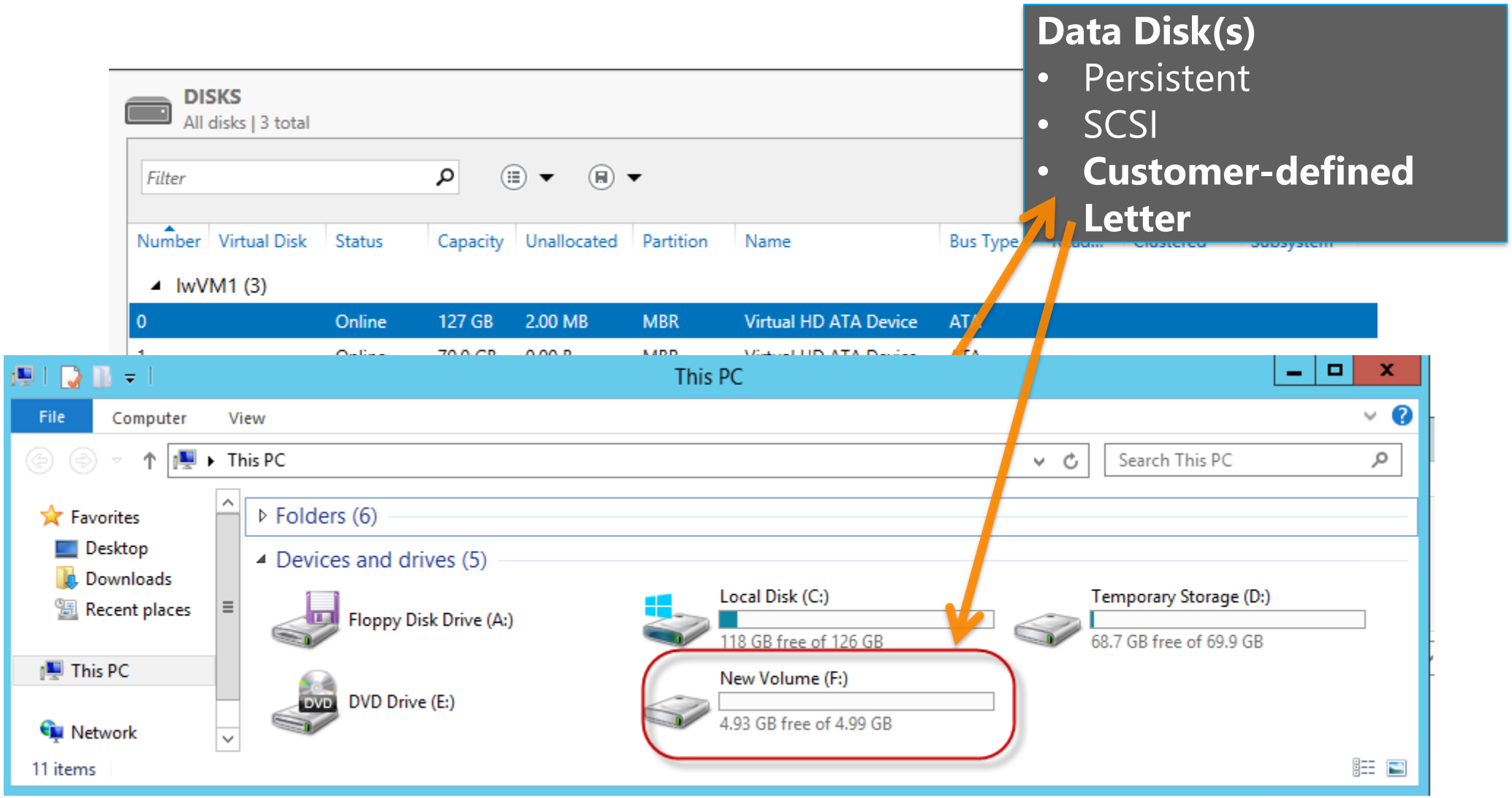


* Max. size of C:\ drive – 1,023GB

VM Disk Layout – Windows OS (continued)



VM Disk Layout – Windows OS(continued)



Persistent Disk Management – Windows OS

- *C:*\ = OS Disk
- *D:*\ = Non-Persistent Cache Disk
- *E:*\, *F:*\. *G:*\ and all subsequent Data Disks—you will need to attach and format them

Capability	OS Disk	Data Disk
Host Cache Default	ReadWrite	None
Max Capacity	1023 GB	1 TB
Imaging Capable	Yes	No
Hot Update	Cache Setting requires a reboot	Change Cache without reboot, Add/Remove without reboot

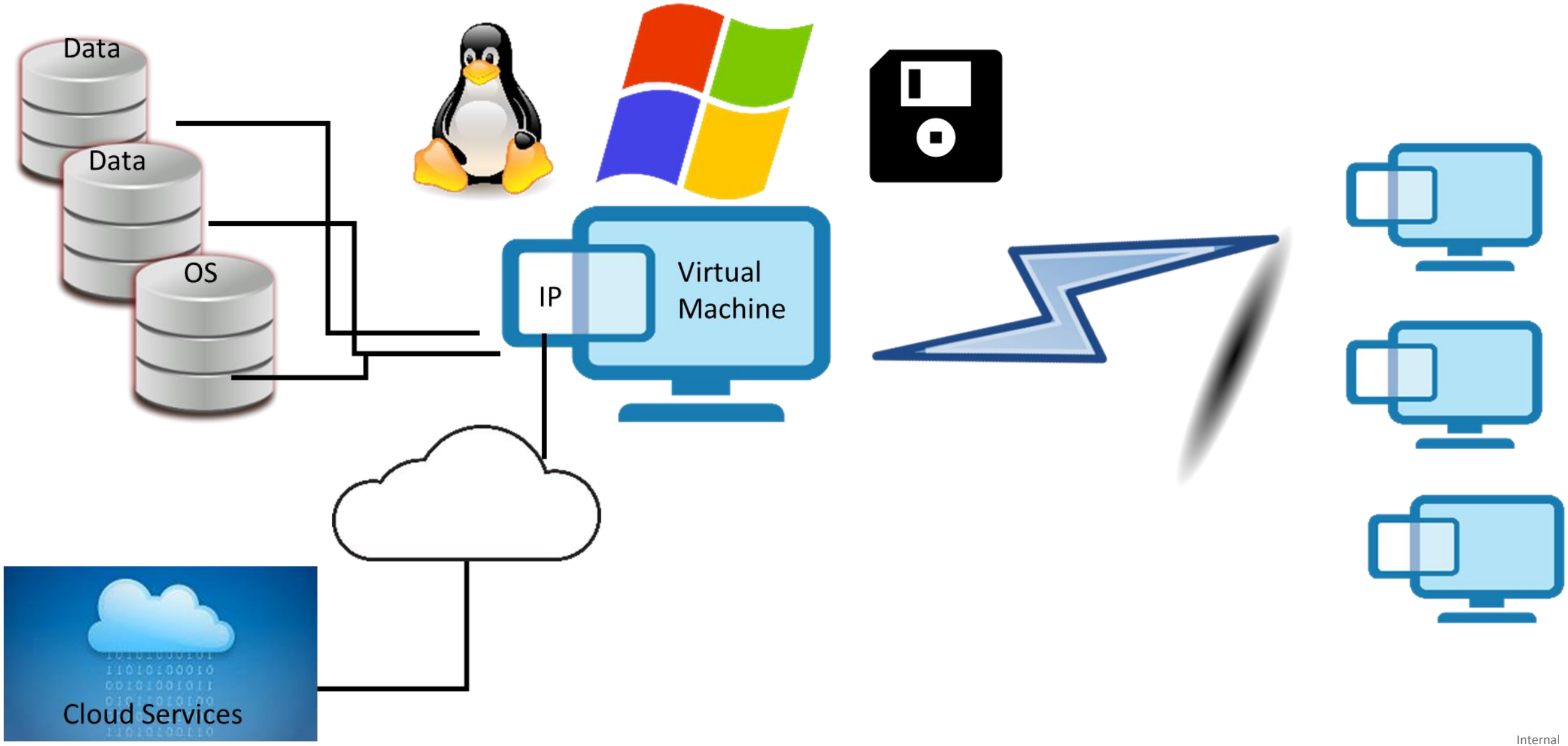
Disk Caching – Windows OS

- Modify using the **Set-AzureOSDisk** or the **Set-AzureDataDisk** cmdlets

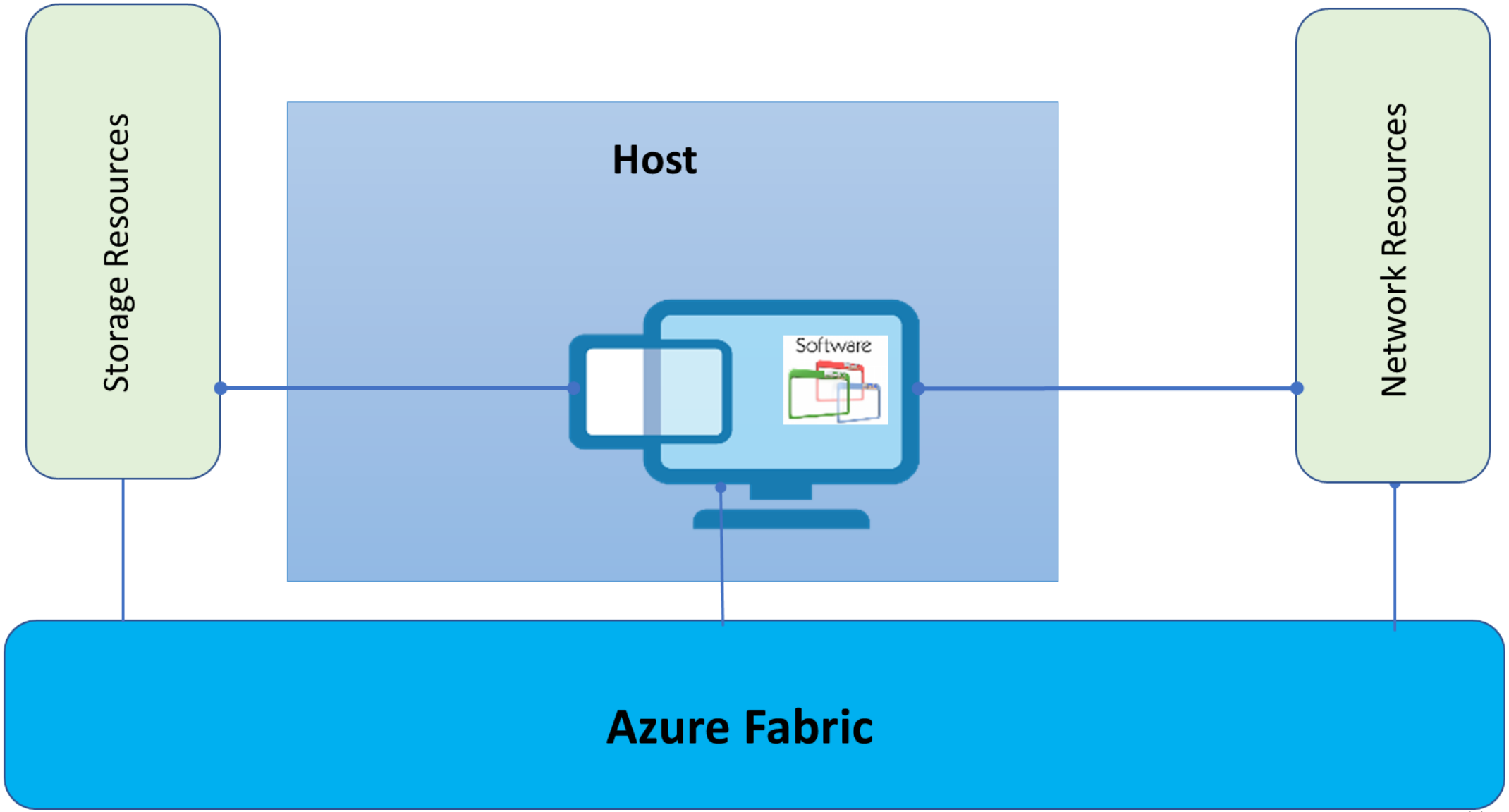
Supported Cache Modes:

Disk Type	Read Only	Read Write	None
OS Disk	Supported	Default	Not Supported
Data Disks	Supported	Supported	Default
Temporary Disk	Not stored in Microsoft Azure Storage Blob Service		

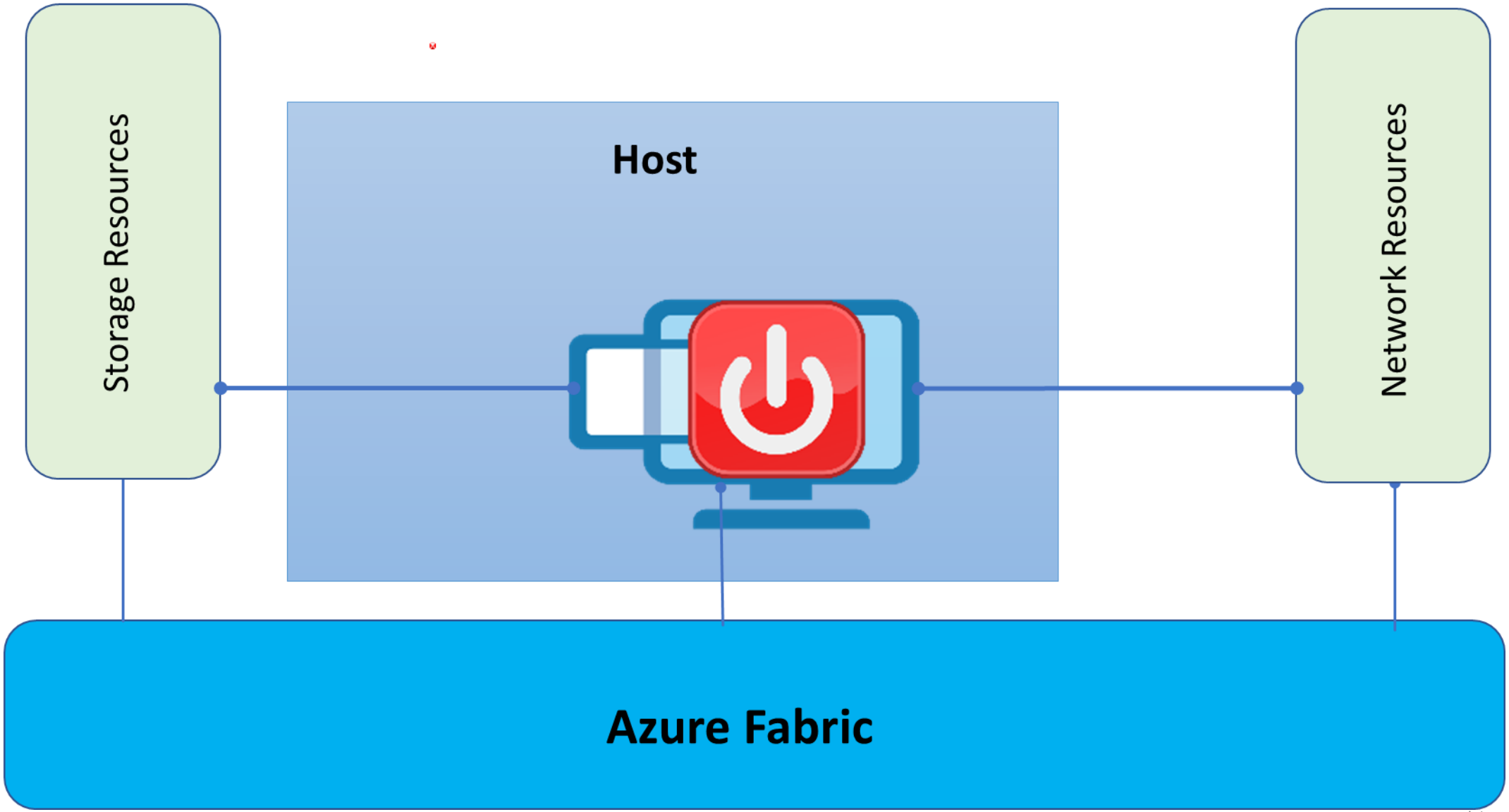
What Do you pay for?



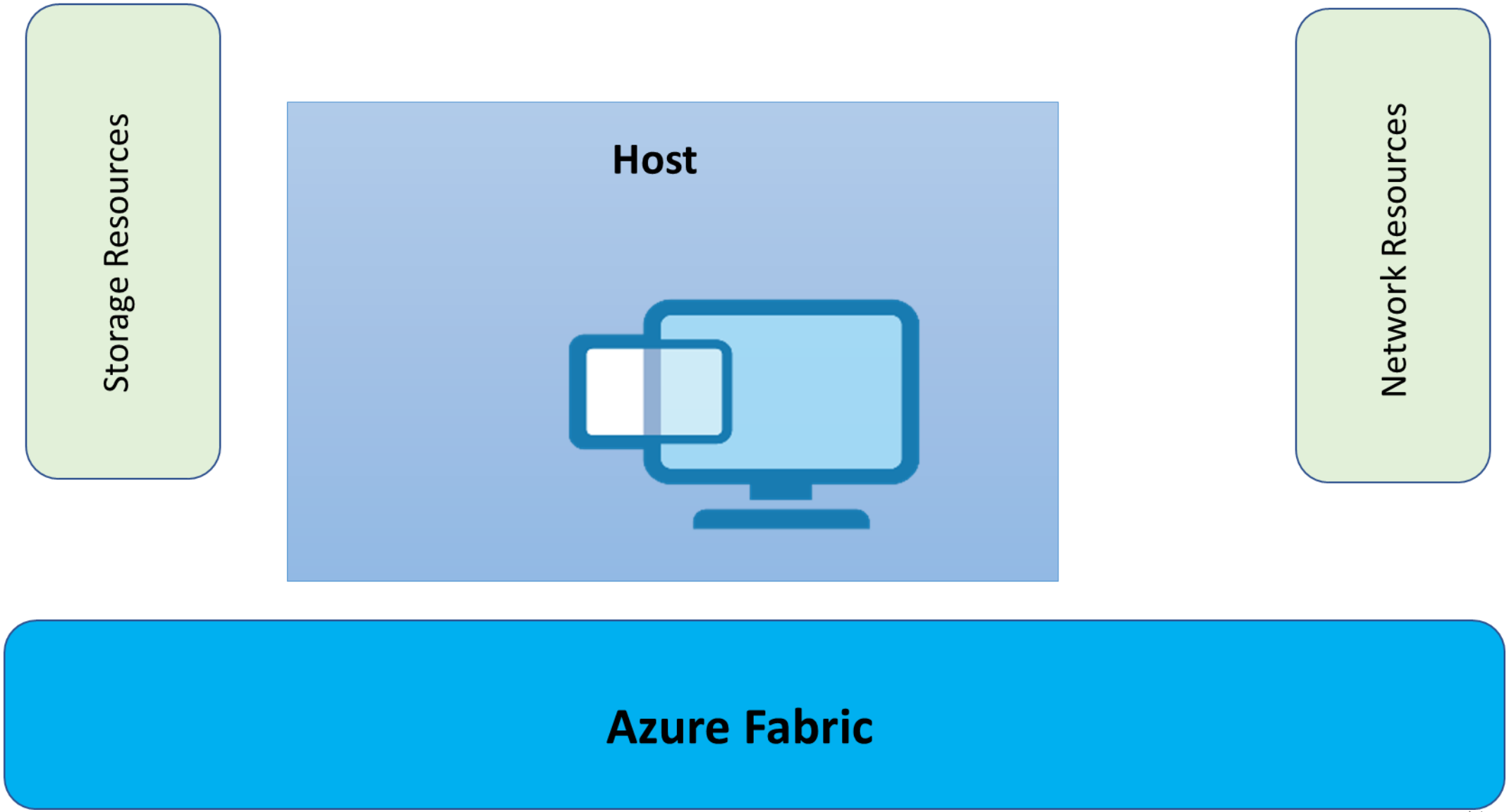
VM States: Running



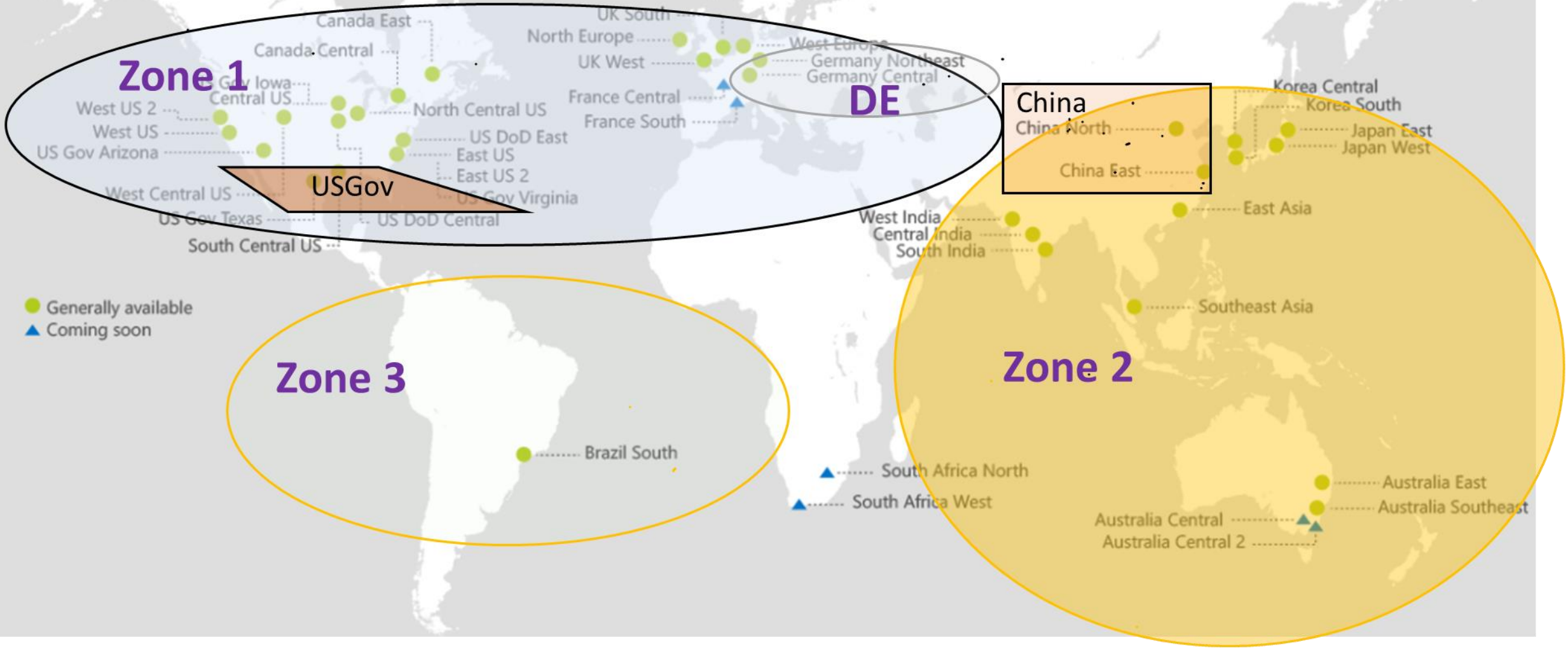
Stopped



Stopped / Deallocated



Azure Regions



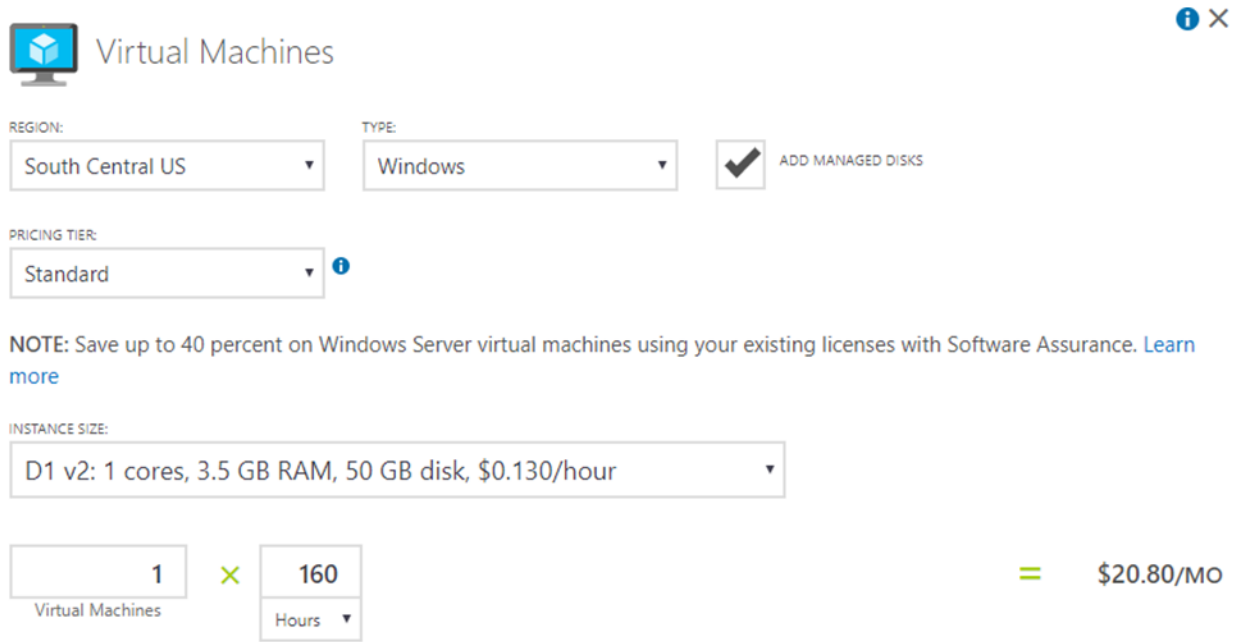
Azure Regions

Prices are Geo specific

East US2, East US and West US2

Data Transfer pricing

- Zone dependent



OUTBOUND DATA TRANSFERS	ZONE 1*	ZONE 2*	ZONE 3*	DE (TRUSTEE)
First 5 GB /Month [1]	Free	Free	Free	Free
5 GB - 10 TB [2] /Month	$0.087 per GB	$0.138 per GB	$0.181 per GB	$0.10 per GB
Next 40 TB (10 - 50 TB) /Month	$0.083 per GB	$0.135 per GB	$0.175 per GB	$0.095 per GB
Next 100 TB (50 - 150 TB) /Month	$0.07 per GB	$0.13 per GB	$0.17 per GB	$0.08 per GB
Next 350 TB (150 - 500 TB) /Month	$0.05 per GB	$0.12 per GB	$0.16 per GB	$0.057 per GB
Over 500 TB /Month	Contact us	Contact us	Contact us	Contact us

Availability Zones (AZ) 1/2

Zones vs. Availability Zones

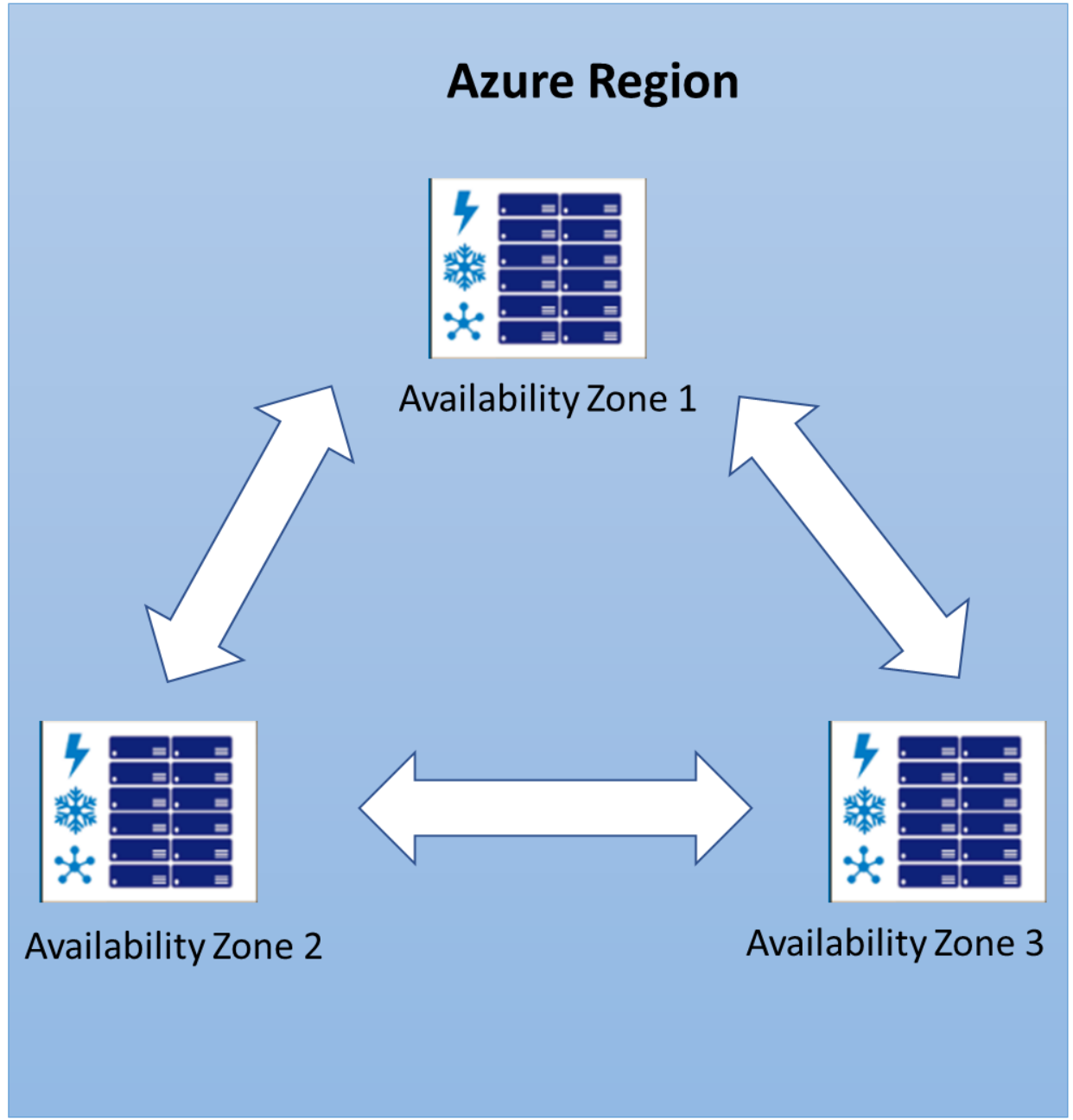
Availability Zones are now in preview
East US 2 and West Europe

With More regions coming

Availability Zones (AZ) are physically separated locations within an Azure region

AZs are fault-isolated locations within an Azure region

Each zone has independent power, network, and cooling.

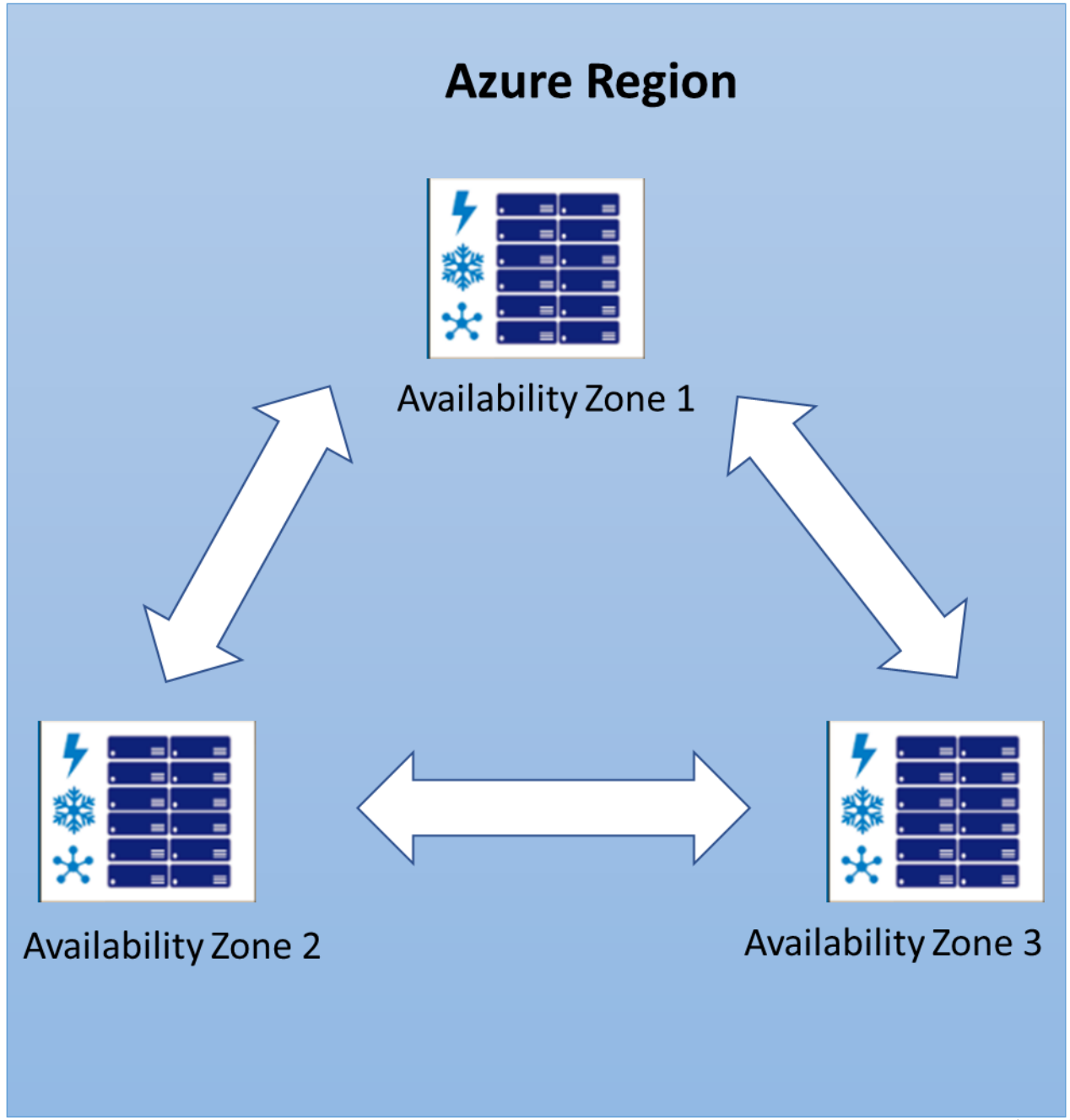


Availability Zones (AZ) 2/2

Each AZ has or multiple DCs. No DC shared by two zones.

Intra-AZ charges are free

Inter-AZ charges are free in Preview



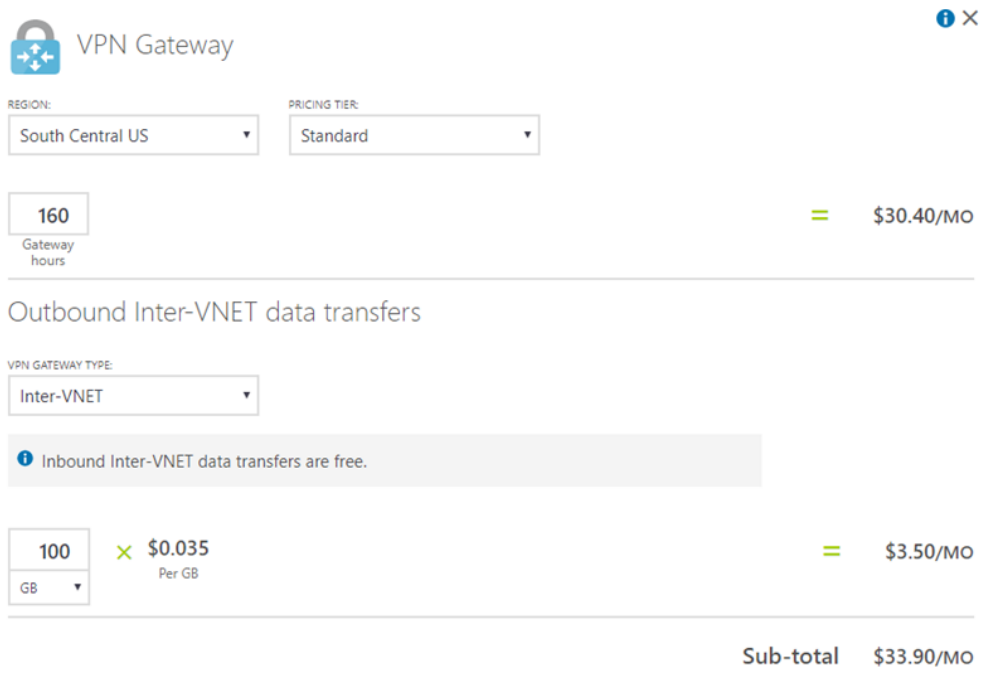
Networking Cost

Inbound data transfers
(i.e. data going into Azure data centers): Free

Outbound data transfers
(i.e. data going out of Azure data centers)
Outbound traffic is free within a region

Dynamic public IP Address: while VM is running

Static public IP Address: First 5 free



Meter Name	Meter Region
"Data Transfer In (GB)"	"Zone 1"
"Data Transfer Out (GB)"	"Zone 1"
"IP Address Hours"	

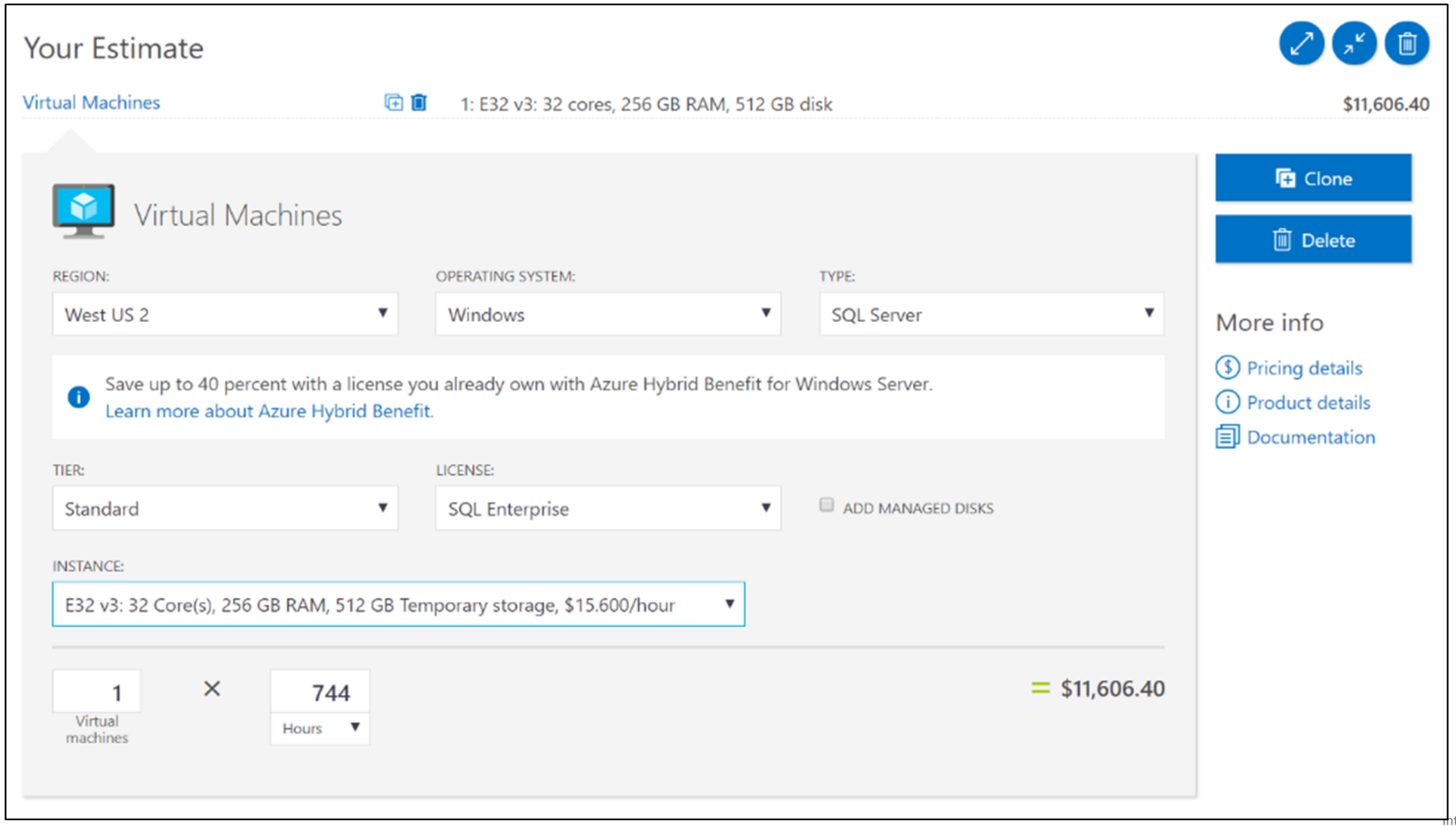
Billing Period	Meter Categor	Meter Sub-category	Meter Name	Meter I	SKU	Unit	Consumed Quantity	Included Quantity	Within	Overag	Curren	Overag	Commitment Rat	Rate	Value
201710(9/18/2017 - 10/17/2017)	"Networking"		"Data Transfer In (GB)"	"Zone 1"		"GB"	0.024745	0	0	0.024745	USD	0	0	0	$0.00 USD
201710(9/18/2017 - 10/17/2017)	"Networking"		"Data Transfer Out (GB)"	"Zone 1"	7TD-0000	"GB"	0.005945	0	0	0.005945	USD	0	0	0	$0.00 USD
201710(9/18/2017 - 10/17/2017)	"Networking"	"Public IP Addresses"	"IP Address Hours"		"7UD-0000	"Hours"	3.3	0	0	3.3	USD	0	0	0.00303	$0.01 USD

Storage Cost

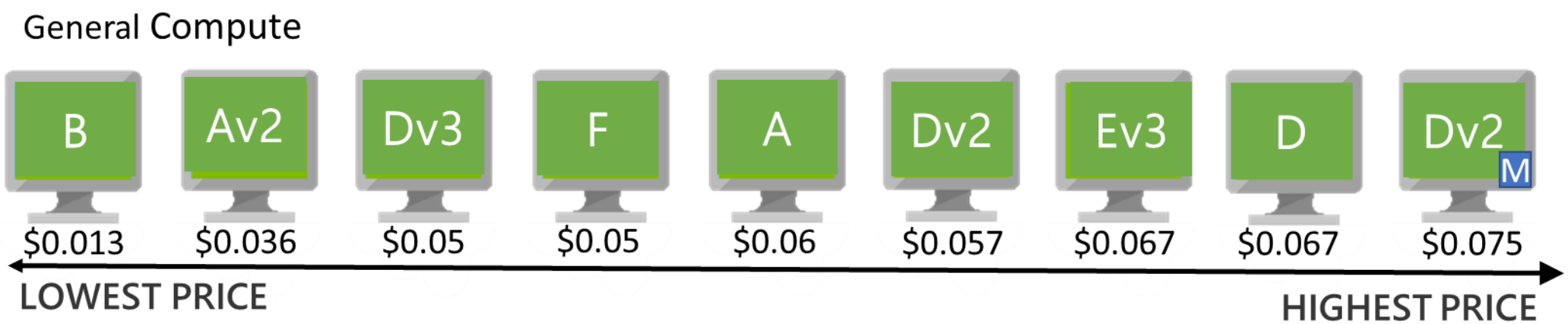
Managed or Unmanaged Disks
Managed disks are charged for provisioned capacity instead of used
Standard vs. Premium
Disk IOPS
Throughput
Data Redundancy options: LRS, GRS and RA-GRS
Storage Capacity
Start with small OS disks
Add small Data disks

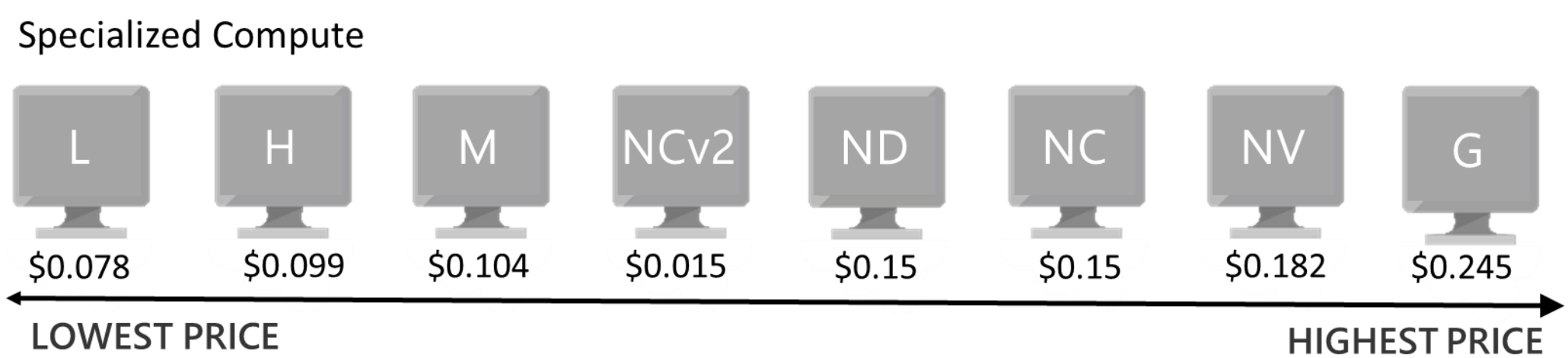
Statement													
Billing Period	Meter Category	Meter Sub-category	Meter Name	Meter Region	SKU	Unit	Consumed Quantity	Included Quantity	Within Commitment	Overage Quantity	Currency	Rate	Value
201710(9/18/2017 - 10/17/2017)	"Storage"	"Geo Redundant"	"Standard IO - Page Blob/Disk (GB)"		"7UD-00001"	"GB"	5.808318	0	0	5.808318	USD	0.094692	$0.55 USD
201710(9/18/2017 - 10/17/2017)	"Storage"	"Read-Access Geo Redundant"	"Standard IO - Page Blob/Disk (GB)"		"7UD-00001"	"GB"	46.109756	0	0	46.109756	USD	0.119931	$5.53 USD
201710(9/18/2017 - 10/17/2017)	"Storage"	"Locally Redundant"	"Standard IO - Block Blob (GB)"		"7UD-00001"	"GB"	0.748427	0	0	0.748427	USD	0.026723	$0.02 USD
201710(9/18/2017 - 10/17/2017)	"Storage"	"Locally Redundant"	"Standard IO - Page Blob/Disk (GB)"		"7UD-00001"	"GB"	28.490419	0	0	28.490419	USD	0.049841	$1.42 USD
201710(9/18/2017 - 10/17/2017)	"Storage"	"Locally Redundant"	"Standard IO - Files (GB)"		"7UD-00001"	"GB"	1.03488	0	0	1.03488	USD	0.077304	$0.08 USD
201710(9/18/2017 - 10/17/2017)	"Storage"	"Geo Redundant"	"Standard IO - Table (GB)"		"7UD-00001"	"GB"	0.028757	0	0	0.028757	USD	0.000	$0.00 USD
201710(9/18/2017 - 10/17/2017)	"Storage"	"Locally Redundant"	"Standard IO - Table (GB)"		"7UD-00001"	"GB"	0.37033	0	0	0.37033	USD	0.081009	$0.03 USD
201710(9/18/2017 - 10/17/2017)	"Storage"	"Read-Access Geo Redundant"	"Standard IO - Table (GB)"		"7UD-00001"	"GB"	0.00077	0	0	0.00077	USD	0.000	$0.00 USD
201710(9/18/2017 - 10/17/2017)	"Storage"	"Locally Redundant"	"Standard Managed Disk/S10 (Units)"	"US Central"	"7UD-00001"	"Units"	0.111552	0	0	0.111552	USD	2.958262	$0.33 USD
201710(9/18/2017 - 10/17/2017)	"Storage"	"Locally Redundant"	"Standard Managed Disk/S6 (Units)"	"US West"	"7UD-00001"	"Units"	0.205632	0	0	0.205632	USD	1.507547	$0.31 USD
201710(9/18/2017 - 10/17/2017)	"Storage"	"Locally Redundant"	"Standard Managed Disk/S30 (Units)"	"US West"	"7UD-00001"	"Units"	0.205632	0	0	0.205632	USD	20.47347	$4.21 USD
201710(9/18/2017 - 10/17/2017)	"Storage"	"Locally Redundant"	"Standard Managed Disk/S10 (Units)"	"US West"	"7UD-00001"	"Units"	0.205632	0	0	0.205632	USD	2.966464	$0.61 USD
201710(9/18/2017 - 10/17/2017)	"Storage"	"Locally Redundant"	"Standard IO - Table (GB)"	"UK South"	"7UD-00001"	"GB"	0.000457	0	0	0.000457	USD	0.000	$0.00 USD
201710(9/18/2017 - 10/17/2017)	"Storage"	"Locally Redundant"	"Premium Storage - Page Blob/P10 (Units)"	"UK South"	"7UD-00001"	"Units"	0.206976	0	0	0.206976	USD	23.86756	$4.94 USD

Compute Cost

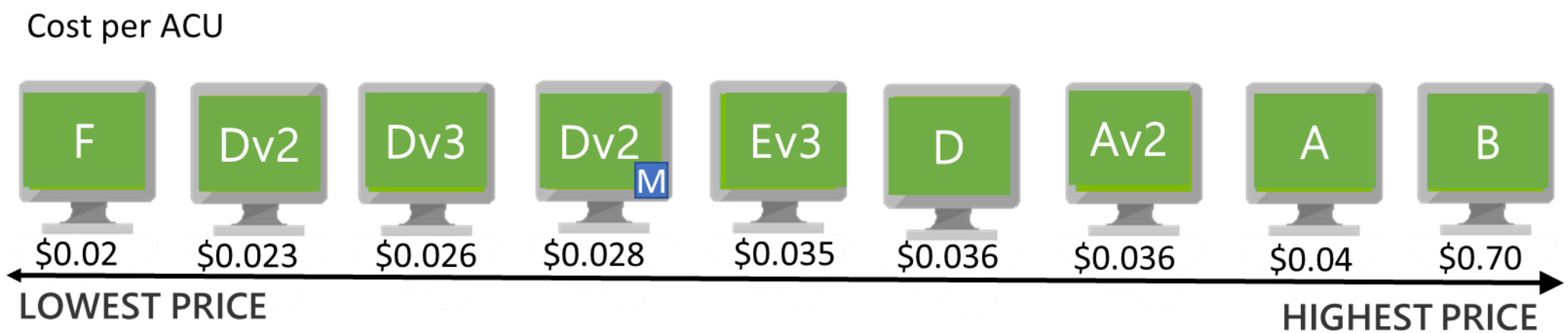


VM SKUs – Cost per vCPU





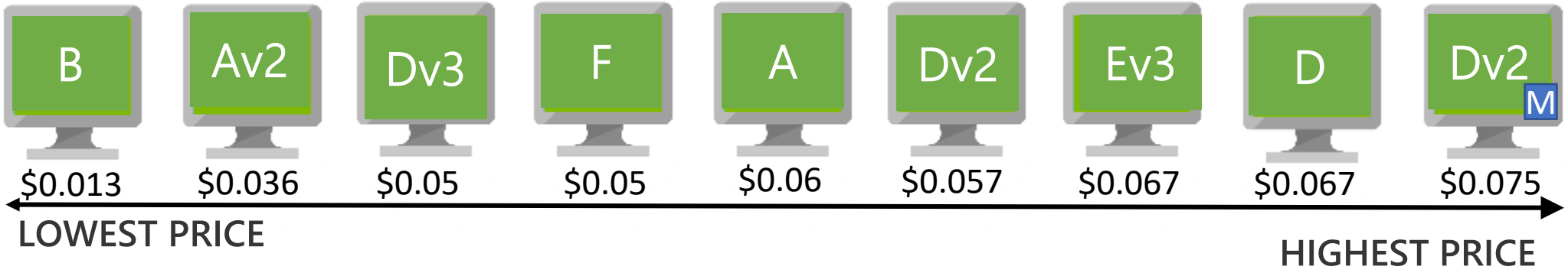
VM SKUs – Cost per ACU



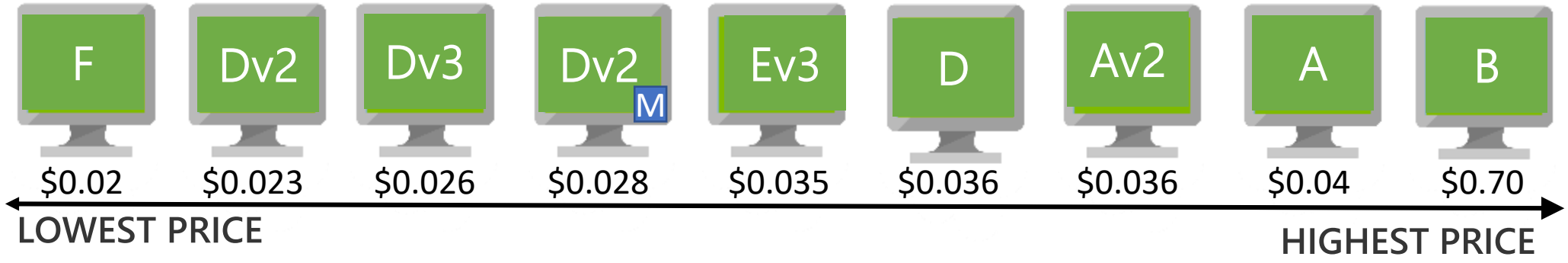
VM SKUs – Cost per ACU

Insider's Secret: Dv2 Promo

Cost per vCPU



Cost per ACU



Azure Reserved Virtual Machine Instances | Overview

Virtual Machine Reservation	Select VM size, region, term (one or three years), and quantity. There is no need to select OS.
Integration with A Hybrid Benefit	You can use Azure Hybrid Benefit for Windows Server or pay hourly based on usage.
Flexible across OS choice	Reservations are for the “base VM” and can be applied to either Windows Server and/or non-Windows Server VMs.

Azure Reserved Virtual Machine Instances | Benefits

Predictable price

1- and 3-year term options

Prioritized compute capacity

Global datacenters

Ease of purchase

Select region, VM size, and term

Assignment control

Scope reservation either to enrollment or subscription

Post-purchase flexibility

Exchange or cancel reservations

Automate, Automate, Automate

- Auto shutdown
  - Minimizing unused resources and control cost
- Auto scale
  - Help ease management overhead instead of continually monitoring system performance and resources
- Resize VM
  - Consider Region and VM SKU

Use your own License

BYOL (Bring Your Own License) vs. PAYG (Pay-as-you-go)

Windows Server: Azure Hybrid Benefits for Windows Server

SQL Server License Mobility

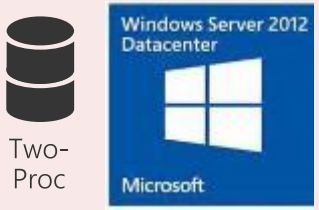
Azure Hybrid Benefit

What is Azure Hybrid Benefit?

An Azure benefit that enables customers with Windows Server Software Assurance licenses to **pay the less expensive non-Windows compute pricing** when they upload and run their self-built Windows Server images on Azure

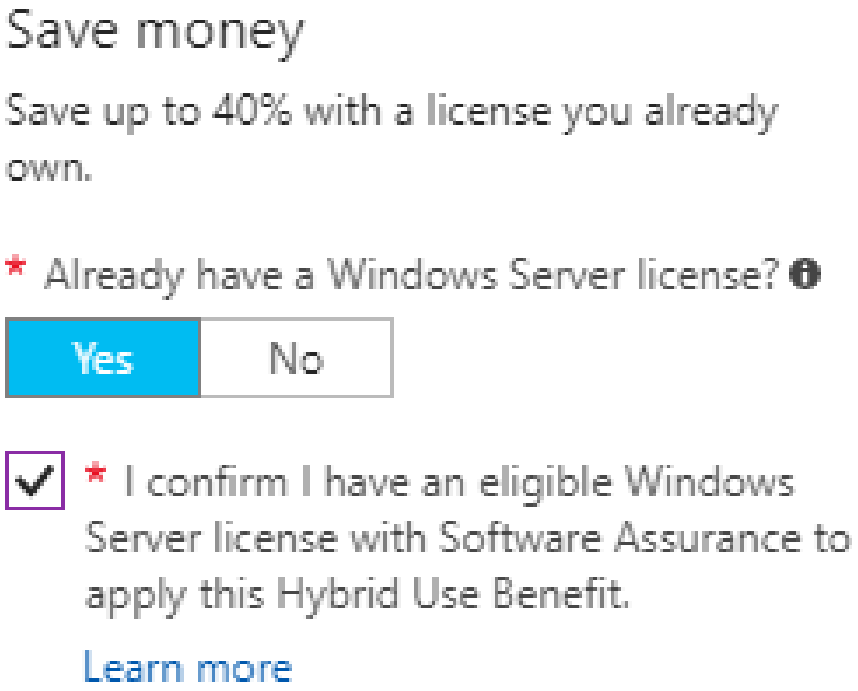
What is the Customer Value Proposition?

- Customer benefits from existing investments in Windows Server when moving to Azure
- Customer receives additional value to their Windows Server Software Assurance investment
- Azure Hybrid Benefit adds additional flexibility and value to Windows Server Standard and Datacenter

CUSTOMER'S ON-PREMISE LICENSE	LICENSE IMPACT FOR CUSTOMER	WINDOWS SERVER AZURE ENABLEMENT
	No licensing concurrency: a Window Server license **cannot** be assigned to other hardware while Azure Hybrid Benefit is being used.	Customer with Windows Server Software Assurance are entitled to: • Two instances of 1 to 8 vCPUs or • One instance of up to 16 vCPUs • Stack licenses for VMs larger than 16 vCPUs
	Licensing concurrency: a Windows Server license **can** continue to be assigned both on premise and in an Azure environment at the same time.	

Azure Hybrid Benefit for Windows Server

Provides ability to use on-premises Windows Server licenses with software assurance on Azure to save on the cost of Windows licensing on Azure virtual machines

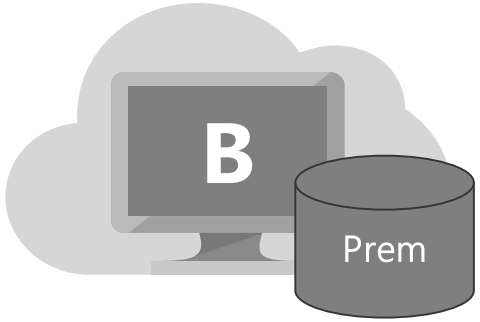


Optimizing VMs

Selecting the right VM size

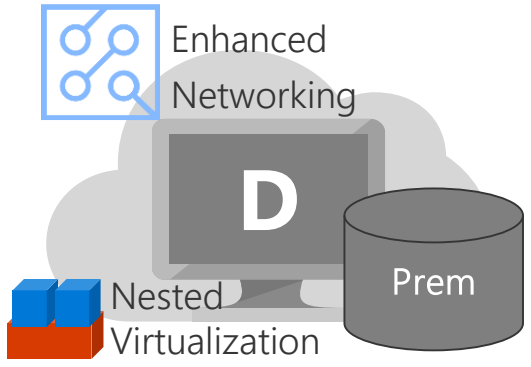
Entry Level

100 ACU/vCPU
$0.36/kACU



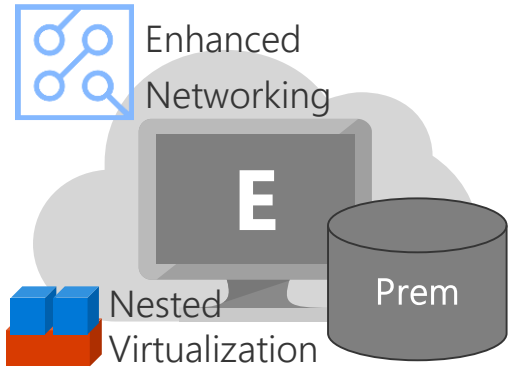
Burst VMs

21 ACU/vCPU
$0.57/kACU



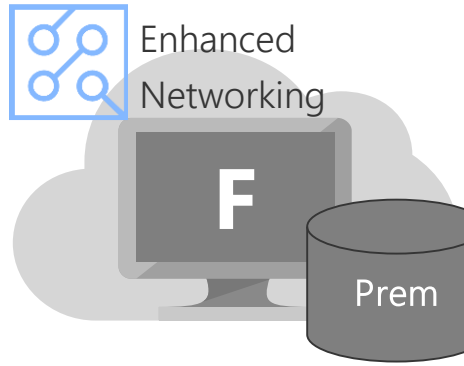
General Purpose (4GB)

160 ACU/vCPU
$0.31/kACU



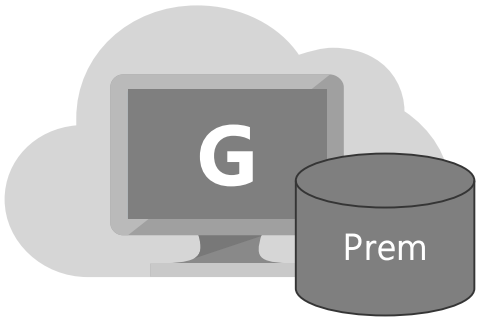
General Purpose (8GB)

160 ACU/vCPU
$0.41/kACU



High CPU

210 ACU/vCPU
$0.24/kACU

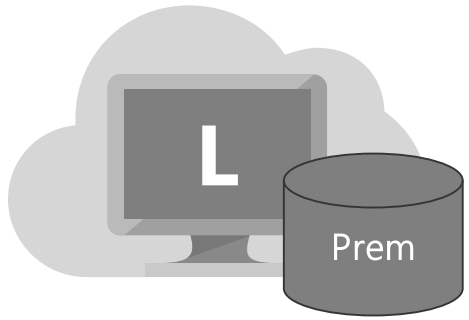


Large Memory (v1)

180 ACU/vCPU
$1.36/kACU

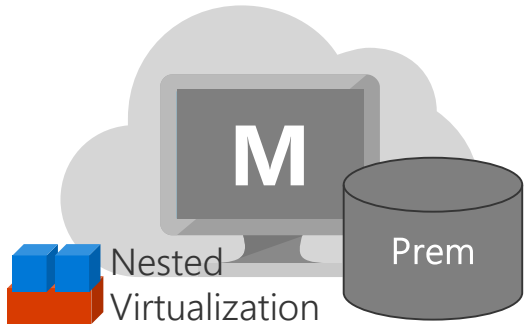
High CPU w/ RDMA

290 ACU/vCPU
$0.34/kACU



Large Local SSD

180 ACU/vCPU
$0.49/kACU



Large Memory (v2)

160 ACU/vCPU
$0.65/kACU

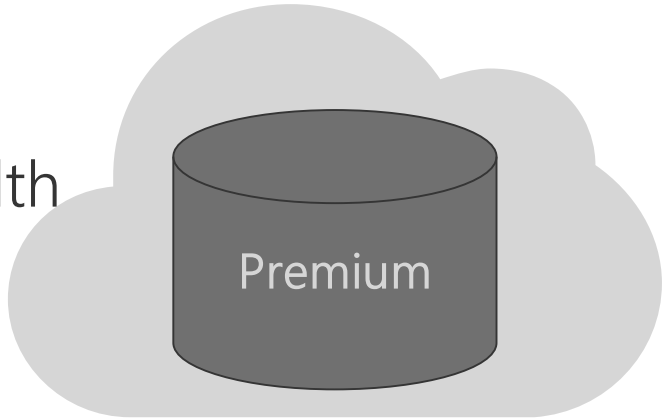
GPU-enabled VMs

Varies

Storage Performance Options

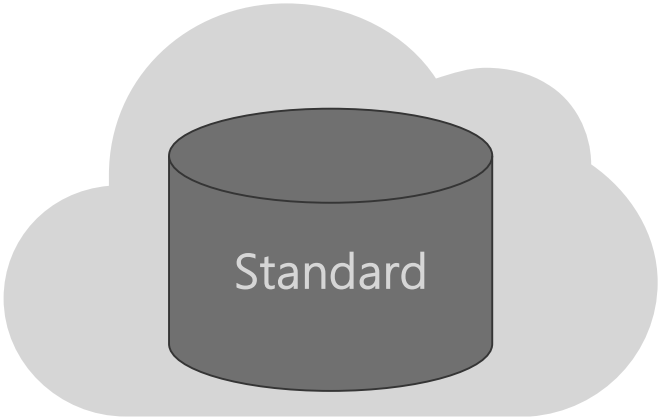
Premium storage

SSD based storage optimized for low latency or high bandwidth
Up to 7,500 IOPS or 250MB/s per disk
Up to 80,000 IOPS for large VMs
Required for single instance SLA



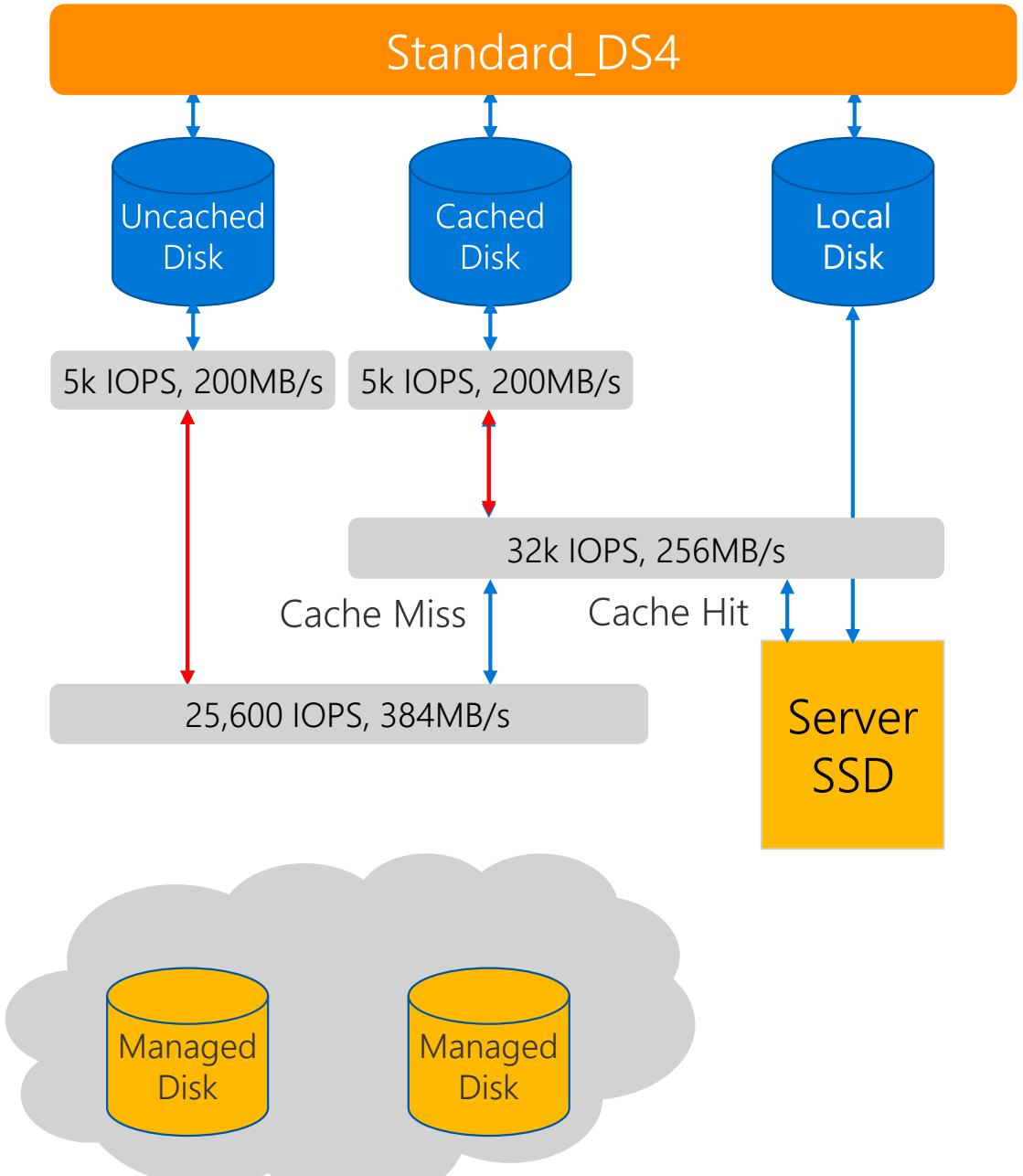
Standard storage

HDD based storage optimized for lighter workloads
Up to 500 IOPS and 60 MB/s per disk



Premium Storage Caching

Standard_DS4_v2 with 2 P30 Disks



Creating Azure VMs by using the Azure portal, Azure PowerShell

Azure VM Deploy Through PowerShell

```
# Variables
## Global
$ResourceGroupName = "GobindRG"
$Location = "centralus"

## Storage
$StorageName = "gobindstorage"
$StorageType = "Standard_GRS"

## Network
$InterfaceName = "ServerInterface09"
$Subnet1Name = "FrontEnd"
$VNetName = "VNet09"
$VNetAddressPrefix = "10.20.0.0/16"
$VNetSubnetAddressPrefix = "10.20.0.0/24"
```

```
## Compute
$VMName = "GobindVM"
$ComputerName = "Server22"
$VMSize = "Standard_A2"
$OSDiskName = $VMName + "OSDisk"

# Resource Group
New-AzureRmResourceGroup -Name $ResourceGroupName -Location $Location

# Storage
$StorageAccount = New-AzureRmStorageAccount -ResourceGroupName $ResourceGroupName -Name $StorageName
-Type $StorageType -Location $Location

# Network
$PIp = New-AzureRmPublicIpAddress -Name $InterfaceName -ResourceGroupName $ResourceGroupName -Location
$Location -AllocationMethod Dynamic
$SubnetConfig = New-AzureRmVirtualNetworkSubnetConfig -Name $Subnet1Name -AddressPrefix
$VNetSubnetAddressPrefix
$VNet = New-AzureRmVirtualNetwork -Name $VNetName -ResourceGroupName $ResourceGroupName -Location
$Location -AddressPrefix $VNetAddressPrefix -Subnet $SubnetConfig
$Interface = New-AzureRmNetworkInterface -Name $InterfaceName -ResourceGroupName $ResourceGroupName -
Location $Location -SubnetId $VNet.Subnets[0].Id -PublicIpAddressId $PIp.Id
```

Compute

Setup local VM object

```
$Credential = Get-Credential
$VirtualMachine = New-AzureRmVMConfig -VMName $VMName -VMSize $VMSize
$VirtualMachine = Set-AzureRmVMOperatingSystem -VM $VirtualMachine -Windows -ComputerName
$ComputerName -Credential $Credential -ProvisionVMAgent -EnableAutoUpdate
$VirtualMachine = Set-AzureRmVMSourceImage -VM $VirtualMachine -PublisherName MicrosoftWindowsServer -
Offer WindowsServer -Skus 2012-R2-Datacenter -Version "latest"
$VirtualMachine = Add-AzureRmVMNetworkInterface -VM $VirtualMachine -Id $Interface.Id
$OSDiskUri = $StorageAccount.PrimaryEndpoints.Blob.ToString() + "vhds/" + $OSDiskName + ".vhd"
$VirtualMachine = Set-AzureRmVMOSDisk -VM $VirtualMachine -Name $OSDiskName -VhdUri $OSDiskUri -
CreateOption FromImage
```

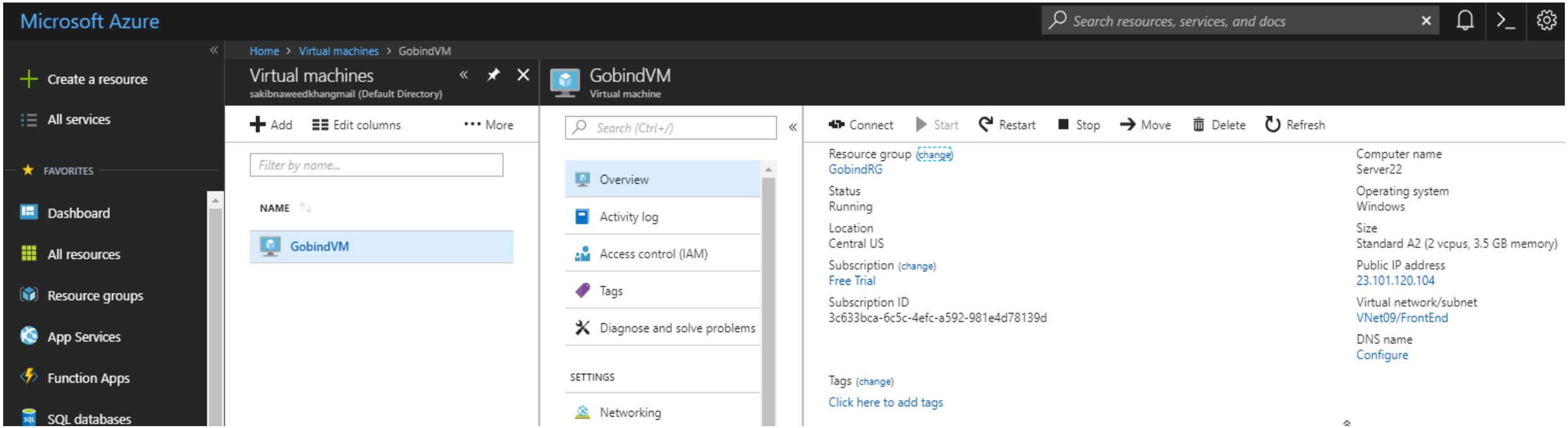
Create the VM in Azure

```
New-AzureRmVM -ResourceGroupName $ResourceGroupName -Location $Location -VM $VirtualMachine
```

```
PS C:\WINDOWS\system32> # Variables
## Global
$ResourceGroupName = "GobindRG"
$Location = "centralus"

## Storage
$StorageName = "gobindstorage"
$StorageType = "Standard_GRS"

## Network
$InterfaceName = "ServerInterface09"
$Subnet1Name = "FrontEnd"
$VNetName = "VNet09"
$VNetAddressPrefix = "10.20.0.0/16"
$VNetSubnetAddressPrefix = "10.20.0.0/24"
## Compute
$VMName = "GobindVM"
$ComputerName = "Server22"
$VMSize = "Standard_A2"
$OSDiskName = $VMName + "OSDisk"

PS C:\WINDOWS\system32> # Resource Group
New-AzureRmResourceGroup -Name $ResourceGroupName -Location $Location


ResourceGroupName : GobindRG
Location          : centralus
ProvisioningState : Succeeded
Tags              :
ResourceId        : /subscriptions/3c633bca-6c5c-4efc-a592-981e4d78139d/resourceGroups/GobindRG




PS C:\WINDOWS\system32> # Storage
$StorageAccount = New-AzureRmStorageAccount -ResourceGroupName $ResourceGroupName -Name $StorageName -Type $StorageType -Location $Location

PS C:\WINDOWS\system32> # Network
$PIp = New-AzureRmPublicIpAddress -Name $InterfaceName -ResourceGroupName $ResourceGroupName -Location $Location -AllocationMethod Dynamic
$SubnetConfig = New-AzureRmVirtualNetworkSubnetConfig -Name $Subnet1Name -AddressPrefix $VNetSubnetAddressPrefix
$VNet = New-AzureRmVirtualNetwork -Name $VNetName -ResourceGroupName $ResourceGroupName -Location $Location -AddressPrefix $VNetAddressPrefix -Subnet $SubnetConfig
$Interface = New-AzureRmNetworkInterface -Name $InterfaceName -ResourceGroupName $ResourceGroupName -Location $Location -SubnetId $VNet.Subnets[0].Id -PublicIpAddressId $PIp.Id
WARNING: The output object type of this cmdlet will be modified in a future release.
WARNING: The output object type of this cmdlet will be modified in a future release.
WARNING: The output object type of this cmdlet will be modified in a future release.
```

```
PS C:\WINDOWS\system32> # Compute

## Setup local VM object
$Credential = Get-Credential
$VirtualMachine = New-AzureRmVMConfig -VMName $VMName -VMSize $VMSize
$VirtualMachine = Set-AzureRmVMOperatingSystem -VM $VirtualMachine -Windows -ComputerName $ComputerName -Credential $Credential -ProvisionVMAgent -EnableAutoUpdate
$VirtualMachine = Set-AzureRmVMSourceImage -VM $VirtualMachine -PublisherName MicrosoftWindowsServer -Offer WindowsServer -Skus 2012-R2-Datacenter -Version "latest"
$VirtualMachine = Add-AzureRmVMNetworkInterface -VM $VirtualMachine -Id $Interface.Id
$OSDiskUri = $StorageAccount.PrimaryEndpoints.Blob.ToString() + "vhds/" + $OSDiskName + ".vhd"
$VirtualMachine = Set-AzureRmVMOSDisk -VM $VirtualMachine -Name $OSDiskName -VhdUri $OSDiskUri -CreateOption FromImage

cmdlet Get-Credential at command pipeline position 1
Supply values for the following parameters:

PS C:\WINDOWS\system32> ## Create the VM in Azure
New-AzureRmVM -ResourceGroupName $ResourceGroupName -Location $Location -VM $VirtualMachine

RequestId IsSuccessStatusCode StatusCode ReasonPhrase
--------- ------------------- ---------- ------------
                         True         OK OK
```

Check the PowerShell deployment on Azure Portal



Module 5 Azure storage

Module Overview

Microsoft Azure storage
Azure Storage replication
Azure storage types
Azure services and tools
Premium storage

Microsoft Azure Storage

Azure Storage or Storage Account is a service from Azure, which provides storage service for various use cases.

Azure Storage is the cloud storage solution for modern applications that rely on durability, availability, and scalability to meet the needs of their customers.

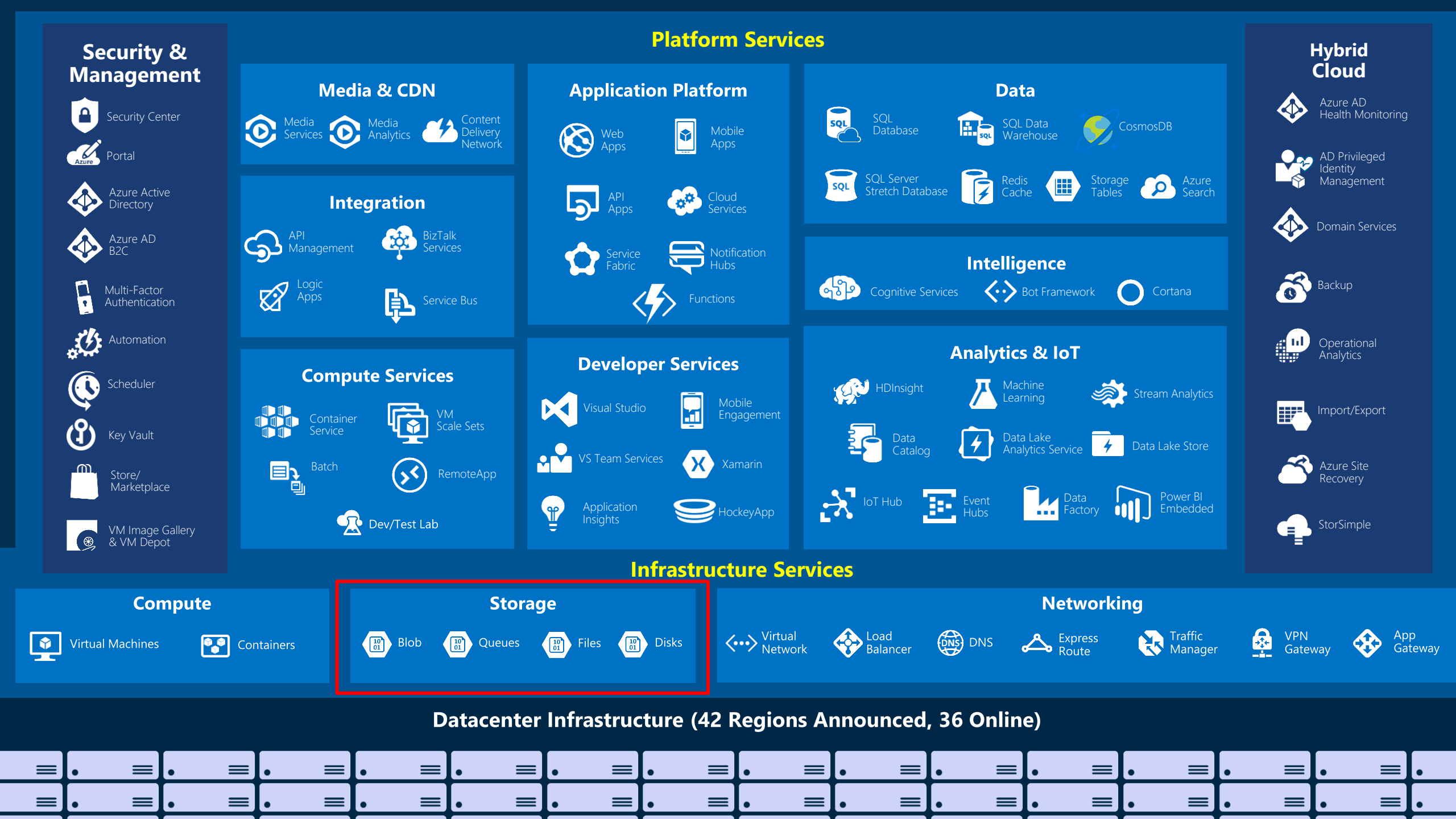
- Cloud storage - anywhere and anytime access

  Blobs, Tables, Queues and Files
- Highly durable, available and massively scalable

  Easily build “Internet scale” applications

  More than 25 trillion stored objects

  2.5+ Million requests/sec on average
- Pay for what you use
- Exposed via easy and open REST APIs, cross-platform client libraries and tools


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
CosmosDB
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 (42 Regions Announced, 36 Online)

Azure Storage

Foundational Building Block of Azure

Azure Services: SQL Data Warehouse, HDInsight, Data Lake Store, Event Hubs, IoT Hubs…

Microsoft Services: Office 365, OneDrive, XBox, Skype…

Hyper Scale	>30 million transactions per second, trillions of objects
Durable	Never lose your data. Multiple redundancy options. Automatic data checks
Secure	Encryption at Rest. Client side Encryption. Integration with KeyVault
Highly Available	Fault tolerance to hardware/software issues. Automatic load balancing
Open	REST API, Open sourced Client Libraries – .NET, Java, C++, Python, Node.js, iOS, Android, Xamarin…
Hybrid	Extensive partner ecosystem. Azure Stack for private/hosted clouds.

Azure Storage Services

IaaS

Storage

Virtual machines

Networking

PaaS

Existing frameworks

Web and mobile

Microservices

Serverless Compute

Disks

Persistent disks for Azure IaaS VMs

Standard Storage Disks: Magnetic disk based, low IOPS, moderate latency

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

Managed Disks

Files

Fully Managed File Shares in the Cloud

SMB and REST access

"Lift and shift" legacy apps

Blobs

Highly scalable, REST based cloud object store

Block Blobs: Sequential I/O, Hot, Cool and Archive Tiers

Page Blobs: Random-write pattern data

Append Blobs

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

Built on a unified Distributed Storage System

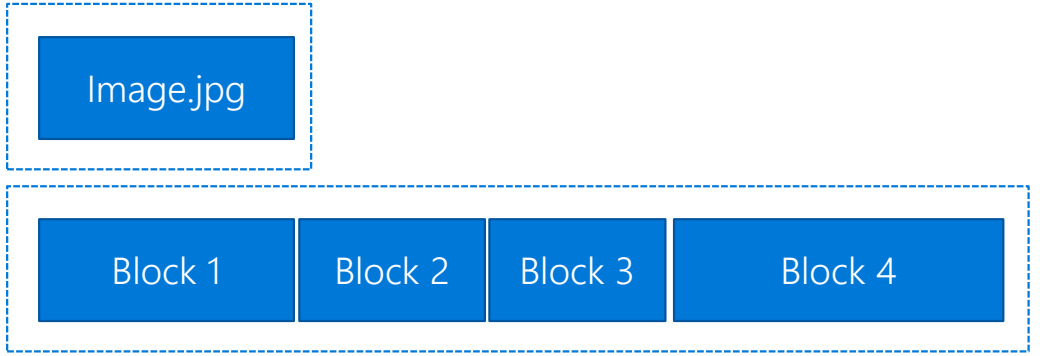
Durability, Encryption at Rest, Strongly Consistent Replication, Fault Tolerance, Auto Load-Balancing

Azure Scale

36 GA, 6 coming soon – Storage is available in every region

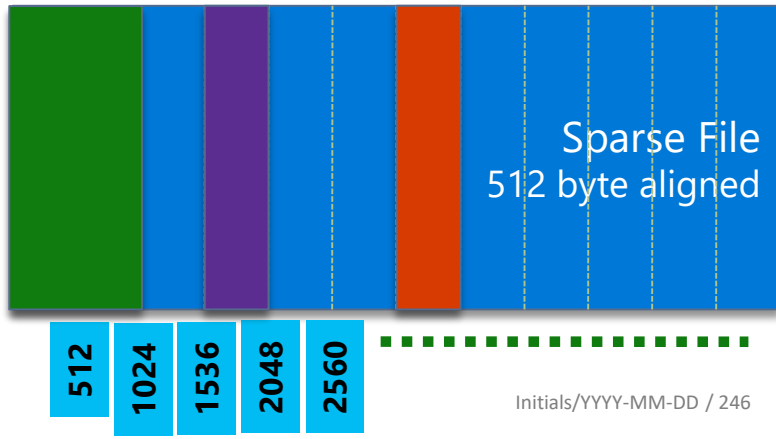
What is the Blob Storage Service?

Azure Blob Storage is a service that stores unstructured data in the cloud as Objects/blobs. Blob storage can store any type of text or binary data, such as a document, media file, or application installer. It is also referred to as Object Storage.



Types of Blobs

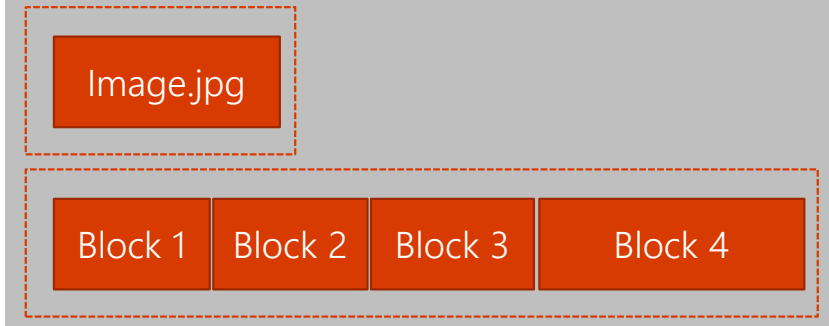
- Block Blobs - Most object storage scenarios
- Append Blobs - Multi-writer append only scenarios
- Page Blobs - Page aligned random reads and writes



Blob Storage Service

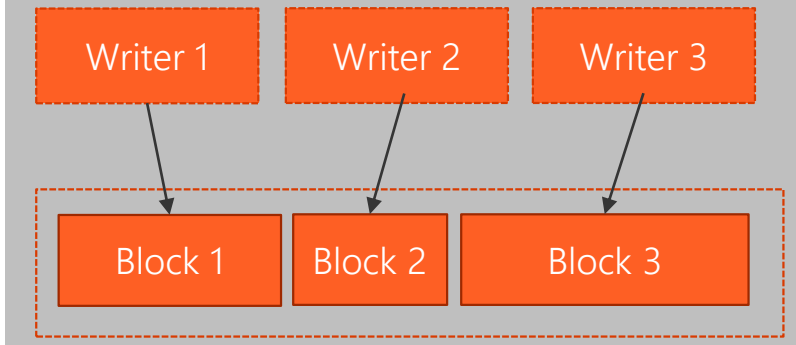
Block Blobs

Most object storage scenarios



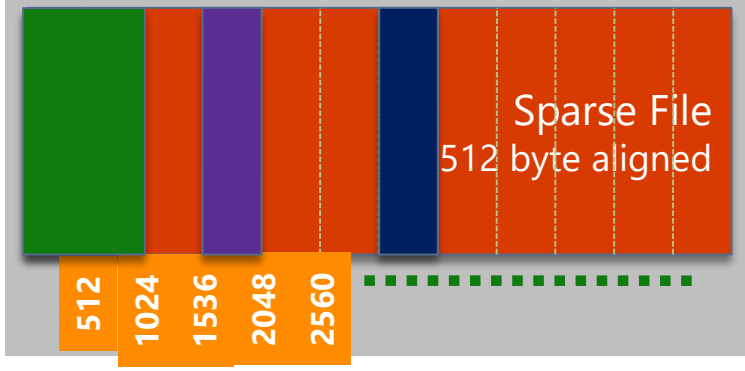
Append Blobs

Multi-writer append only scenarios



Page Blobs

Page aligned random reads &writes



1. Block Blobs : Ideal for working with large files. Suppose we have taken a backup of a VM which is a backup operation of huge large binary size files and we can upload that storage account as block blob.

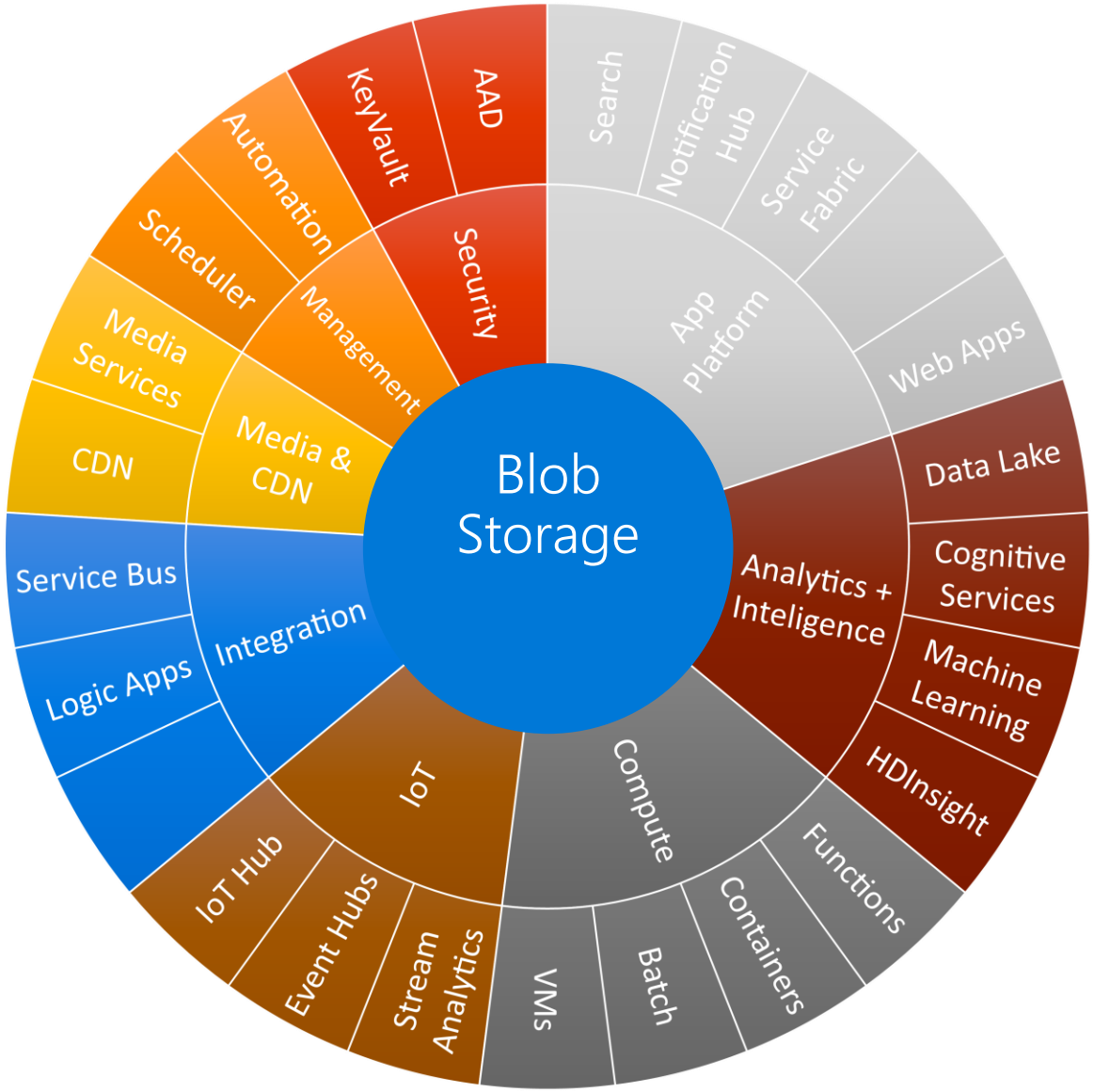
Block sizes are dynamic and it can be upto a maximum of 100MB for single block and then a block blob can contain upto 50000 block which is maximum upto 4.75 TB. This is optimised for large blocks.

2. Page Blobs : There is smaller limits on this and so page blobs can have maximum upto 1 TB. This is optimised for randomised Read and Write Operations.Thats why VHD files are idela for page blobs.

It is going to write anything we save in a VM as VHD. it wll write on individual page on blobs with maximum of 1 TB.

3. Append Blobs: This is optimised for Append operations.That means when we append anything that we add to the end of the blob. It is similar to like logging or something like systemware adding at the end of operations.

Azure Ecosystem and Blob Storage



Broad integration for Blobs across Azure services

Enables many scenarios

What is the Table Storage Service?

It provides a NoSQL Key-Value store for massive scale structured data. You can use the Table service API to create tables for structured storage, and to insert, update, delete, and query data.

Azure Table storage is a service that stores structured NoSQL data in the cloud, providing a key/attribute store with a schemaless design. Because Table storage is schemaless,

it's easy to adapt your data as the needs of your application evolve. Access to Table storage data is fast and cost-effective for many types of applications,

and is typically lower in cost than traditional SQL for similar volumes of data.

You can use Table storage to store flexible datasets like user data for web applications, address books, device information, or other types of metadata your service requires.

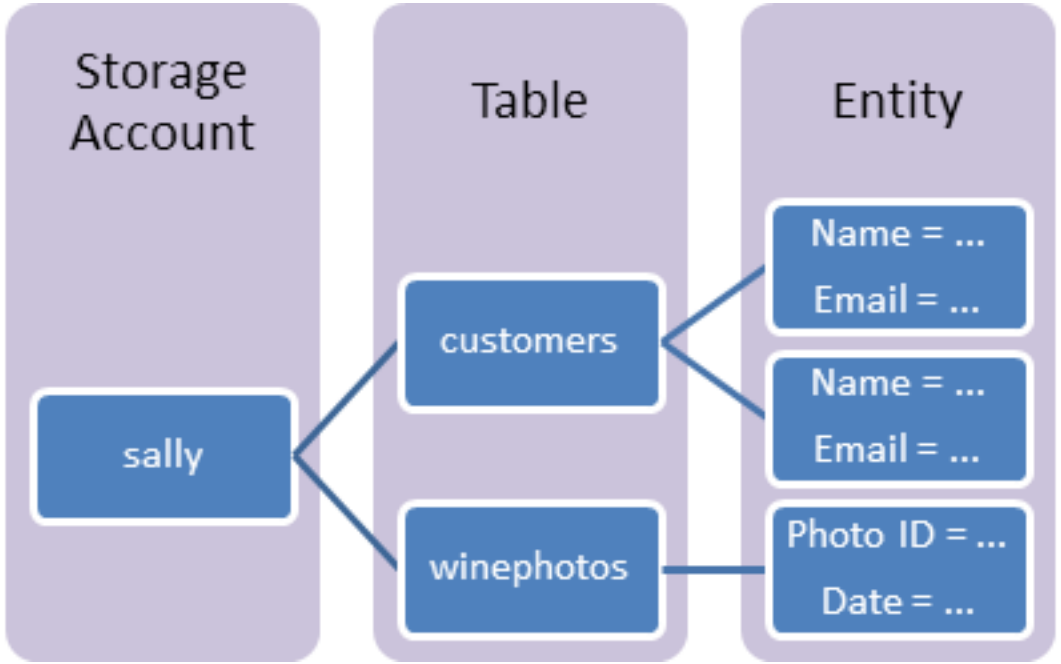
You can store any number of entities in a table, and a storage account may contain any number of tables, up to the capacity limit of the storage account.

Azure Table storage stores large amounts of structured data. The service is a NoSQL datastore which accepts authenticated calls from inside and outside the Azure cloud.

Azure tables are ideal for storing structured, non-relational data.

You can use Table storage to store and query huge sets of structured, non-relational data, and your tables will scale as demand increases.

Table storage contains the following components:

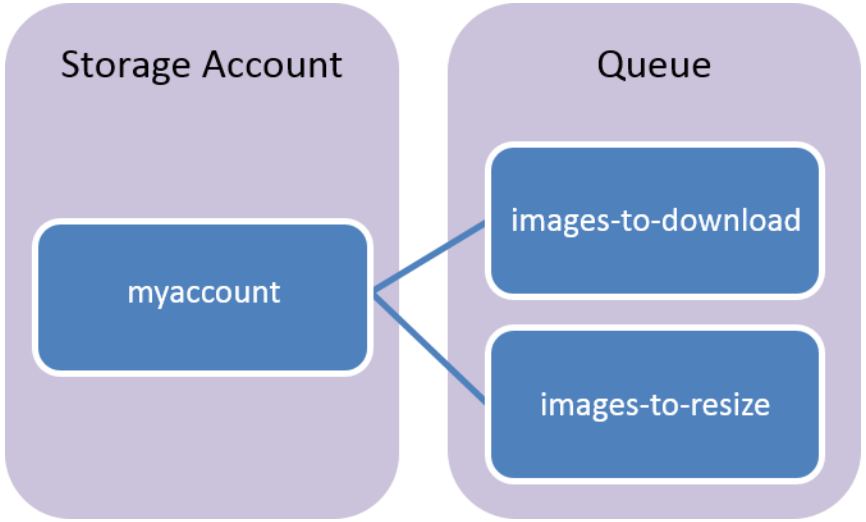


URL format: Azure Table Storage accounts use this format:
http://<storage account>.table.core.windows.net/<table>

What is the Queue Storage Service?

Azure Queue storage is a service for storing large numbers of messages that can be accessed from anywhere in the world via authenticated calls using HTTP or HTTPS.

A single queue message can be up to 64 KB in size, and a queue can contain millions of messages, up to the total capacity limit of a storage account.



Common uses of Queue storage include:

1. Creating a backlog of work to process asynchronously
2. Passing messages from an Azure web role to an Azure worker role

Concept : Storage Account - myaccount

Queue - images-to-download and images-to-resize

Queue: A queue contains a set of messages. All messages must be in a queue. Note that the queue name must be all lowercase. For information on naming queues.

Message: A message, in any format, of up to 64 KB. The maximum time that a message can remain in the queue is seven days.

Azure Queue storage is a service for storing lagre numbers of messages that can be accessed from anywhere in the world via authenticated calls using HTTP or HTTPS.

A Single queue message can be up to 64 KB in size, and a queue can contain millions of messages, upto to total capacity limit of a storage account.

URL format: Queues are addressable using the following URL format:
http://<storage account>.queue.core.windows.net/<queue>

What is the File Storage Service?

Fully managed file shares in the cloud, accessible via standard Server Message Block (SMB) protocol. Enables sharing files between applications using Windows APIs or REST API.

Simple, secure and fully managed cloud file shares

1.Extend your servers to Azure with SyncPreview for on-premises performance and capability
2.Secure data at rest and in-transit using SMB 3.0 and HTTPS
3.Simplify cloud file share management using familiar tools
4.Azure File storage offers network file shares in the cloud
5.Azure Files can be used to completely replace or supplement traditional on-premises file servers or NAS devices.
6.Azure Files makes it easy to "lift and shift" applications to the cloud that expect a file share to store file application or user data.

Blobs

Storage for any type of data, analogous to files in a file system, with individual blobs storing up to 1 TB of data

Queues

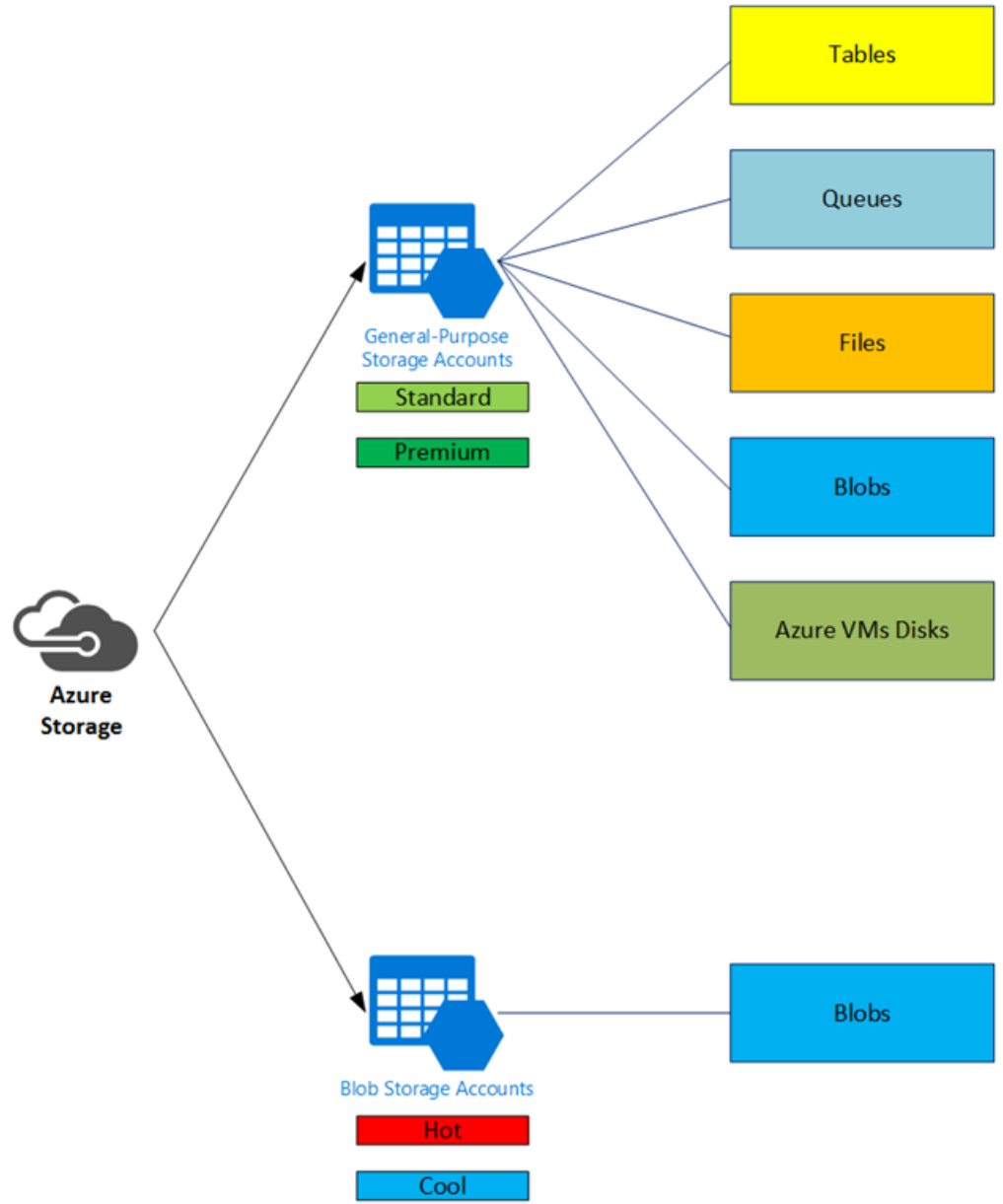
Reliable messaging for workflow processing and for communication between applications or application components

Tables

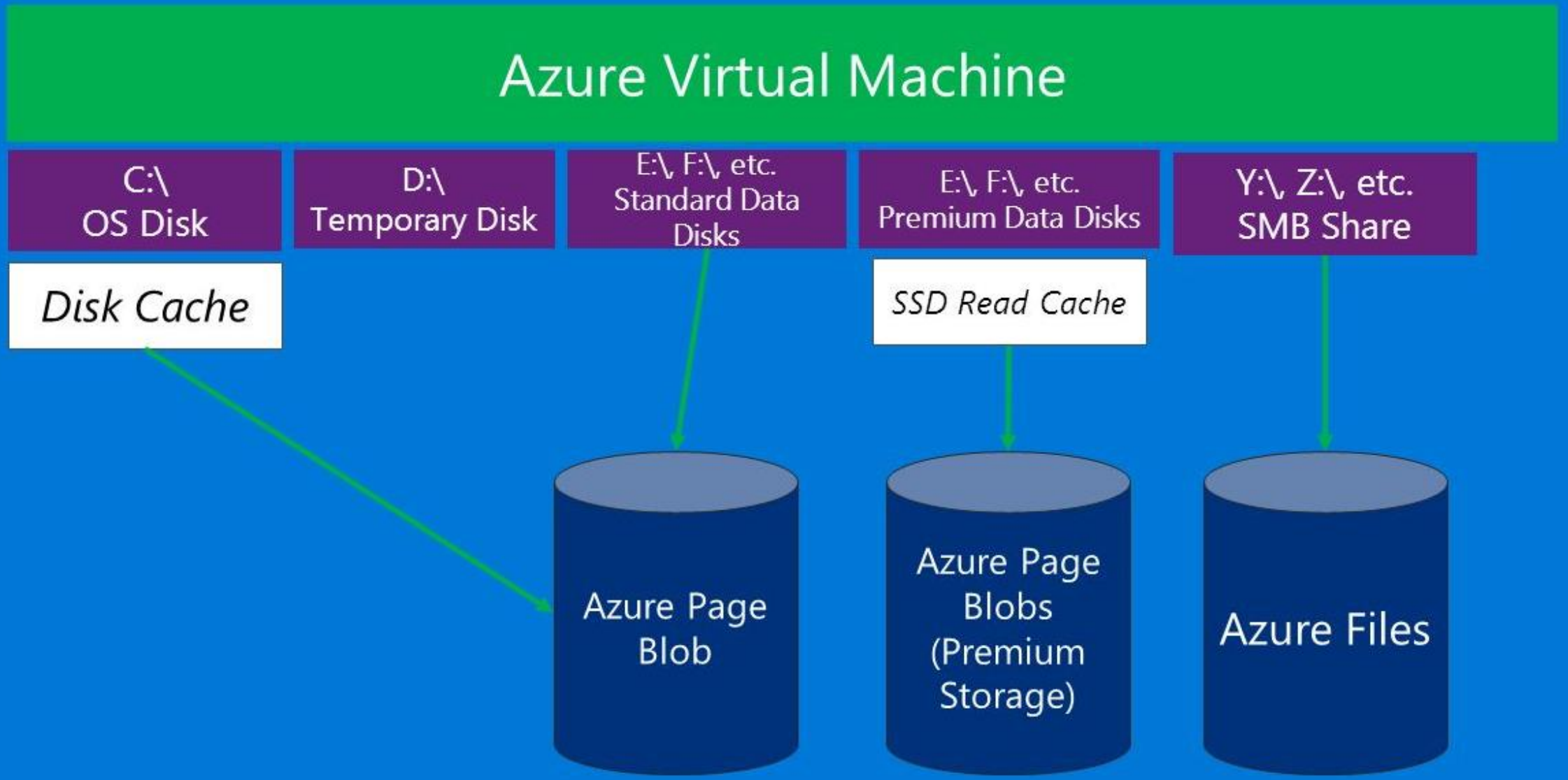
NoSQL storage of semi-structured data for rapid development and fast access to large quantities of data

Files

Shared storage for VMs and cloud services using Server Message Block (SMB) protocol



Summary - Virtual Machine Storage Architecture



An Azure virtual machine supports attaching a number of data disks. This article describes scalability and performance targets for a VM's data disks. Use these targets to help decide the number and type of disk that you need to meet your performance and capacity requirements

Azure Managed Disks:

The disk limit for managed disks is per region and per disk type. The maximum limit, and also the default limit, is 10,000 managed disks per region and per disk type for a subscription. For example, you can create up to 10,000 standard managed disks and also 10,000 premium managed disks in a region, per subscription. Managed snapshots and images count against the managed disks limit.

Standard storage accounts:

A standard storage account has a maximum total request rate of 20,000 IOPS. The total IOPS across all of your virtual machine disks in a standard storage account should not exceed this limit.

You can roughly calculate the number of highly utilized disks supported by a single standard storage account based on the request rate limit. For example, for a Basic Tier VM, the maximum number of highly utilized disks is about 66 (20,000/300 IOPS per disk), and for a Standard Tier VM, it is about 40 (20,000/500IOPS per disk).

Premium storage accounts:

A premium storage account has a maximum total throughput rate of 50 Gbps. The total throughput across all of your VM disks should not exceed this limit.

Managed virtual machine disks

Standard managed virtual machine disks

STANDARD DISK TYPE	S4	S6	S10	S20	S30	S40	S50
Disk size	32 GB	64 GB	128 GB	512 GB	1024 GB (1 TB)	2048 GB (2TB)	4095 GB (4 TB)
IOPS per disk	500	500	500	500	500	500	500
Throughput per disk	60 MB/sec	60 MB/sec	60 MB/sec	60 MB/sec	60 MB/sec	60 MB/sec	60 MB/sec

Premium managed virtual machine disks: per disk limits

PREMIUM DISKS TYPE	P4	P6	P10	P20	P30	P40	P50
Disk size	32 GB	64 GB	128 GB	512 GB	1024 GB (1 TB)	2048 GB (2 TB)	4095 GB (4 TB)
IOPS per disk	120	240	500	2300	5000	7500	7500
Throughput per disk	25 MB/sec	50 MB/sec	100 MB/sec	150 MB/sec	200 MB/sec	250 MB/sec	250 MB/sec

Premium managed virtual machine disks: per VM limits

RESOURCE	DEFAULT LIMIT
Max IOPS Per VM	80,000 IOPS with GS5 VM
Max throughput per VM	2,000 MB/s with GS5 VM

Unmanaged virtual machine disks

Standard unmanaged virtual machine disks: per disk limits

VM TIER	BASIC TIER VM	STANDARD TIER VM
Disk size	4095 GB	4095 GB
Max 8 KB IOPS per persistent disk	300	500
Max number of disks performing max IOPS	66	40

Premium unmanaged virtual machine disks: per account limits

RESOURCE	DEFAULT LIMIT
Total disk capacity per account	35 TB
Total snapshot capacity per account	10 TB
Max bandwidth per account (ingress + egress[1])	<=50 Gbps

[1]*Ingress* refers to all data (requests) being sent to a storage account. *Egress* refers to all data (responses) being received from a storage account.

Premium unmanaged virtual machine disks: per disk limits

PREMIUM STORAGE DISK TYPE	P10	P20	P30	P40	P50
Disk size	128 GiB	512 GiB	1024 GiB (1 TB)	2048 GiB (2 TB)	4095 GiB (4 TB)
Max IOPS per disk	500	2300	5000	7500	7500
Max throughput per disk	100 MB/s	150 MB/s	200 MB/s	250 MB/s	250 MB/s
Max number of disks per storage account	280	70	35	17	8

Premium unmanaged virtual machine disks: per VM limits

RESOURCE	DEFAULT LIMIT
Max IOPS Per VM	80,000 IOPS with GS5 VM
Max throughput per VM	2,000 MB/s with GS5 VM

Premium Storage

High Bandwidth with Low Latency

Up to **64 TB** of storage per VM

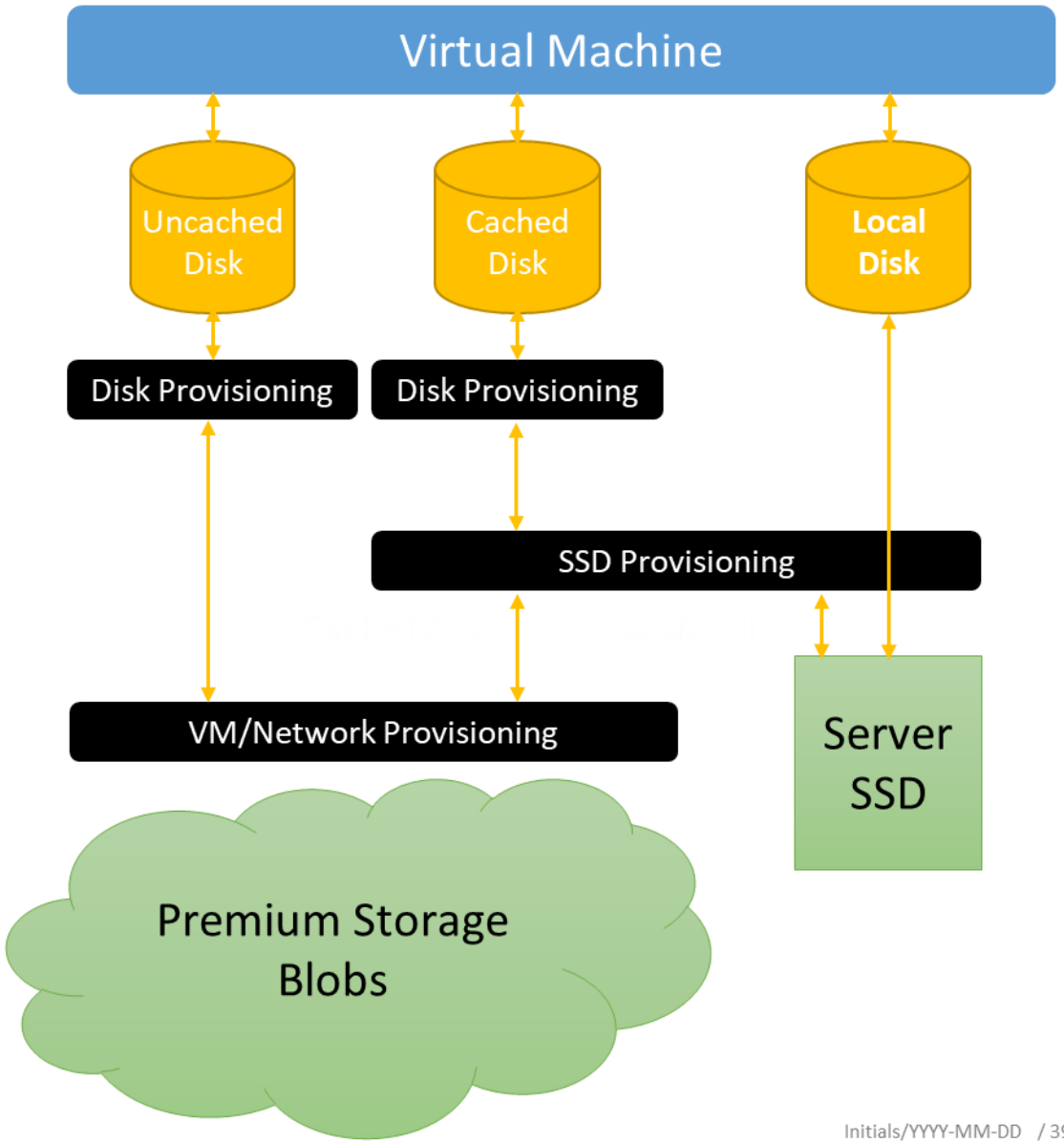
80,000 IOPS per VM

5,000 IOPS per disk

~5 ms read/write (no cache)

Less than 1ms read latency (cache)

Supports only Azure VM



Types of Storage Accounts:

Azure Storage provides three distinct account options, with different pricing and features supported. The three different storage account options are:

- General-purpose v1 (GPv1) accounts
- General-purpose v2 (GPv2) accounts
- Blob storage accounts

General-purpose v1

General-purpose v1 (GPv1) accounts provide access to all Azure Storage services, but may not have the latest features or the lowest per gigabyte pricing. For example, cool storage and archive storage are not supported in GPv1. Pricing is lower for GPv1 transactions, so workloads with high churn or high read rates may benefit from this account type. General-purpose v1 (GPv1) storage accounts are the oldest type of storage account, and the only kind that can be used with the classic deployment model.

General-purpose v2

General-purpose v2 (GPv2) accounts are storage accounts that support all of the latest features for blobs, files, queues, and tables. GPv2 accounts support all APIs and features supported in GPv1 and Blob storage accounts. They also support the same durability, availability, scalability, and performance features in those account types. Pricing for GPv2 accounts has been designed to deliver the lowest per gigabyte prices, and industry competitive transaction prices. You can upgrade your GPv1 account to a GPv2 account using Azure portal, PowerShell, or Azure CLI.

For block blobs in a GPv2 storage account, you can choose between hot and cool storage tiers at the account level, or hot, cool, and archive tiers at the blob level based on access patterns. Store frequently, infrequently, and rarely accessed data in the hot, cool, and archive storage tiers respectively to optimize costs.
GPv2 storage accounts expose the **Access Tier** attribute at the account level, which specifies the default storage account tier as **Hot** or **Cool.** The default storage account tier is applied to any blob that does not have an explicit tier set at the blob level. If there is a change in the usage pattern of your data, you can also switch between these storage tiers at any time. The **archive tier** can only be applied at the blob level.

Now we have the v2 General Storage account which takes the features of the blob storage accounts and combines then with the general storage account, plus tiering.

Blob storage accounts

Blob storage accounts could be deployed as hot or cool tiers. Hot tier offered the cheaper access rate, but per-GB capacity billing was priced between General Purpose V1 storage accounts and cool blob storage accounts. Cool blob storage accounts offered the cheapest per GB capacity, but had the highest charge for accessing blobs. Blob storage accounts can only store blobs.

Tiering means that we can move a blob between 3 tiers within the same storage account (not automatic tiering today):

- **Hot:** Lowest access rates, most expensive per GB capacity.
- **Cool:** Still low latency, but cheap per GB capacity at higher access rate.
- **Archive:** The cheapest per GB capacity (~$2.05 per TB per month!), but it takes up to 15 hours to move a blob back to cool/hot where it can be accessed again.

	Hot storage tier	Cool storage tier	Archive storage tier
Availability	99.9%	99%	N/A
Availability (RA-GRS reads)	99.99%	99.9%	N/A
Usage charges	Higher storage costs, lower access and transaction costs	Lower storage costs, higher access and transaction costs	Lowest storage costs, highest access and transaction costs
Minimum object size	N/A	N/A	N/A
Minimum storage duration	N/A	30 days (GPv2 only)	180 days
Latency (Time to first byte)	milliseconds	milliseconds	< 15 hrs
Scalability and performance targets	Same as general-purpose storage accounts	Same as general-purpose storage accounts	Same as general-purpose storage accounts

Azure

Storage

Azure Premium Storage delivers high-performance, low-latency disk support for virtual machines running I/O-intensive workloads. Virtual machine (VM) disks that use Premium Storage store data on solid state drives (SSDs). You can migrate your application's VM disks to Azure Premium Storage to take advantage of the speed and performance of these disks.

Premium storage accounts

Virtual machine disks: per account limits

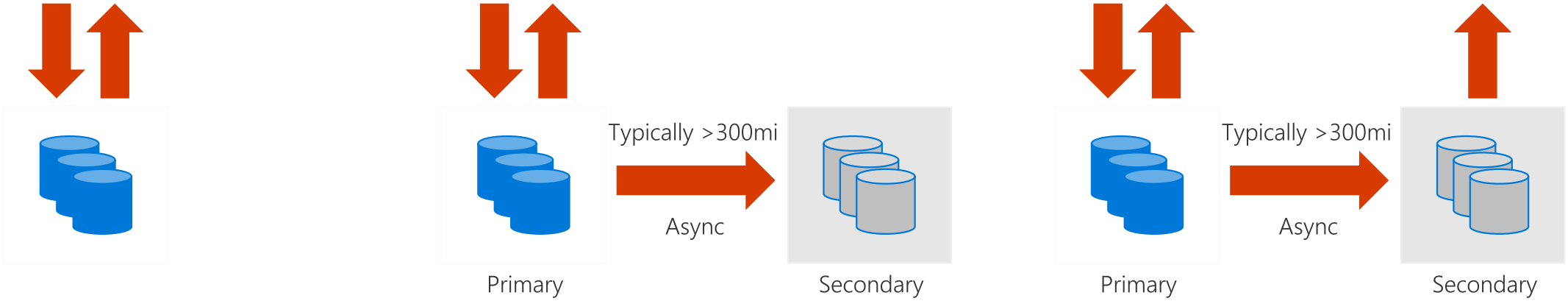
Resource	Default Limit
Total disk capacity per account	35 TB
Total snapshot capacity per account	10 TB
Max bandwidth per account (ingress + egress[1])	<=50 Gbps

[1] *Ingress* refers to all data (requests) being sent to a storage account. *Egress* refers to all data (responses) being received from a storage account.

Virtual machine disks: per disk limits

Premium Storage Disk Type	P10	P20	P30
Disk size	128 GiB	512 GiB	1024 GiB (1 TB)
Max IOPS per disk	500	2300	5000
Max throughput per disk	100 MB per second	150 MB per second	200 MB per second
Max number of disks per storage account	280	70	35

Azure Storage Durability



LRS

3 replicas, 1 region
Protect against disk, node, rack failures
Write is ack'd when all replicas are committed
Superior to dual-parity RAID

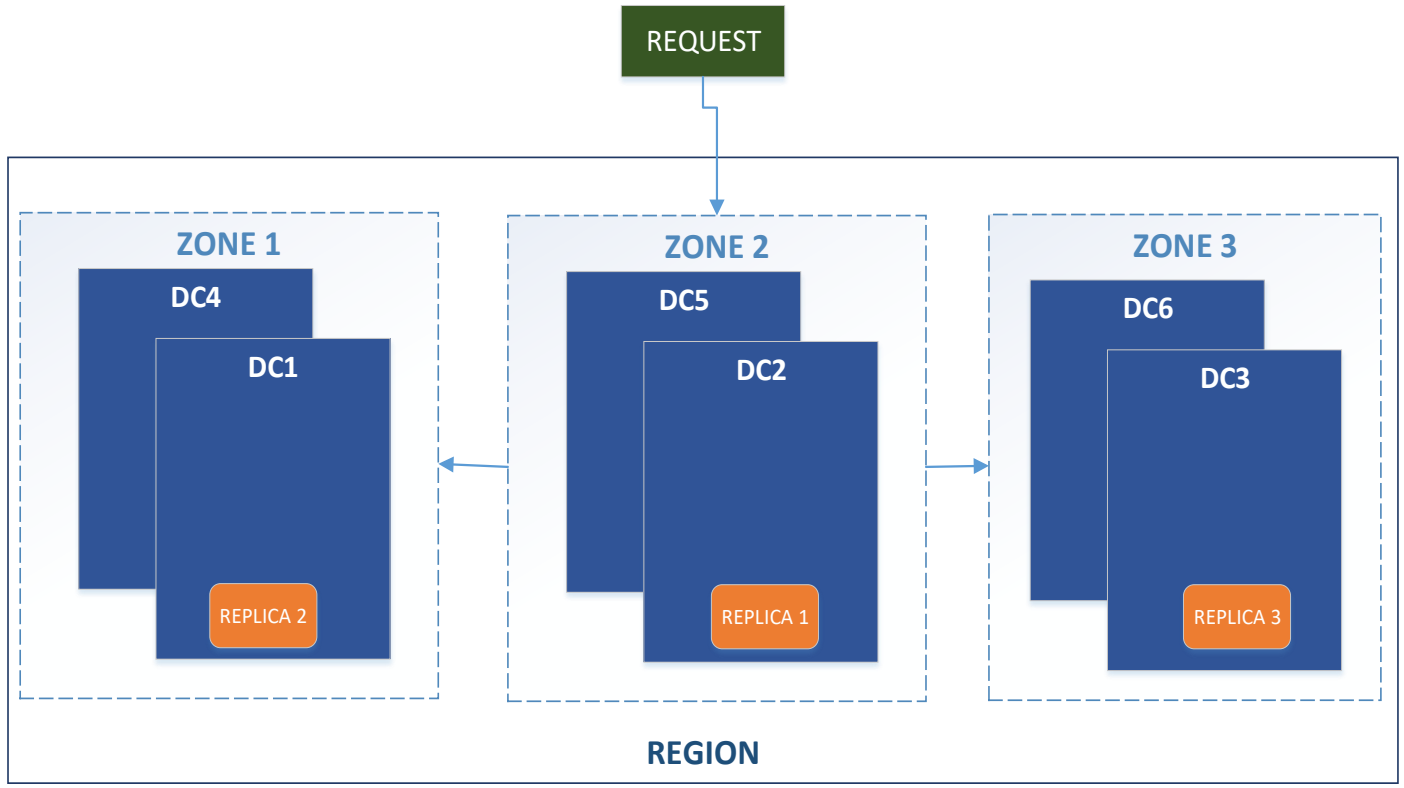
GRS

6 replicas, 2 regions (3/region)
Protects against major regional disasters
Asynchronous to secondary

RA-GRS

GRS + Read access to secondary
Separate secondary endpoint
RPO delay to secondary can be queried

NEW - Azure Zone Redundant Storage (ZRS)



Read / write resilience against single cluster / DC unavailability

Support for Blob, Table, File, Queue Storage

Public Preview in Q4 CY 2017 in multiple regions.
GA in H1 CY 2018

Synchronous data replication across Azure Availability Zones within region

LRS	ZRS	GRS	RA-GRS
Resilient to disk/node/rack failures	Resilient to single cluster / datacenter outage	Resilient to regional outage	Resilient to regional outage Read access to second region

Microsoft Azure Storage: Redundancy

	LOCALLY REDUNDANT STORAGE (LRS)	ZONE REDUNDANT STORAGE (ZRS)	GEOGRAPHICALLY REDUNDANT STORAGE (GRS)	READ-ACCESS GEOGRAPHICALLY REDUNDANT STORAGE (RA-GRS)
How it works	Makes multiple synchronous copies of your data within a single datacenter	Stores three copies of data across multiple datacenters within or across regions. For block blobs only.	Same as LRS, plus multiple asynchronous copies to a second datacenter hundreds of miles away	Same as GRS, plus read access to the secondary datacenter
Total copies	3	3	6	6
Why use it	For economical local storage or data governance compliance	An economical, higher durability option for block blob storage	For protection against a major datacenter outage or disaster	Provides read access to data during an outage, for maximum data availability and durability
Availability SLA	99.9% read/write	99.9% read/write	99.9% read/write	99.9% write 99.99% read

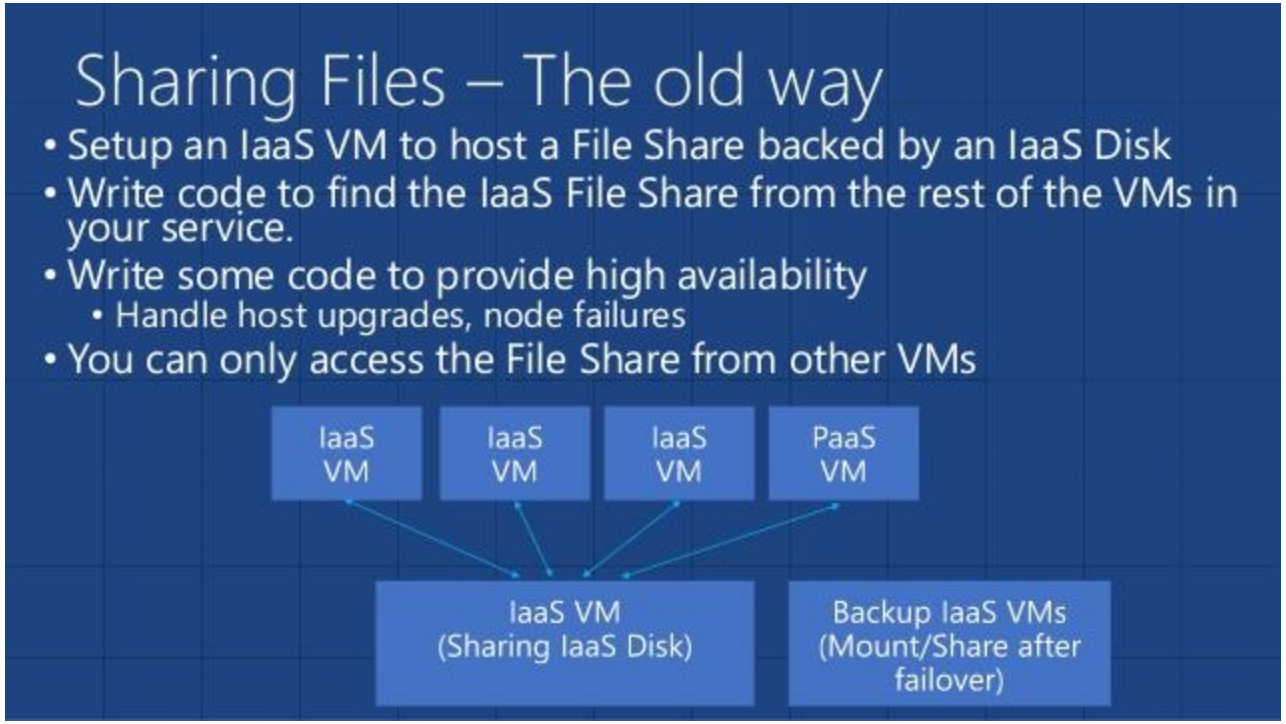
- **Locally redundant storage (LRS)** replicates your data three times within a storage scale unit, which is hosted in a datacenter in the region in which you created your storage account. A write request returns successfully only once it has been written to all three replicas. These three replicas each reside in separate fault domains and upgrade domains within one storage scale unit.

- **Zone-redundant storage (ZRS)** replicates your data asynchronously across datacenters within one or two regions in addition to storing three replicas similar to LRS, thus providing higher durability than LRS. Data stored in ZRS is durable even if the primary datacenter is unavailable or unrecoverable.

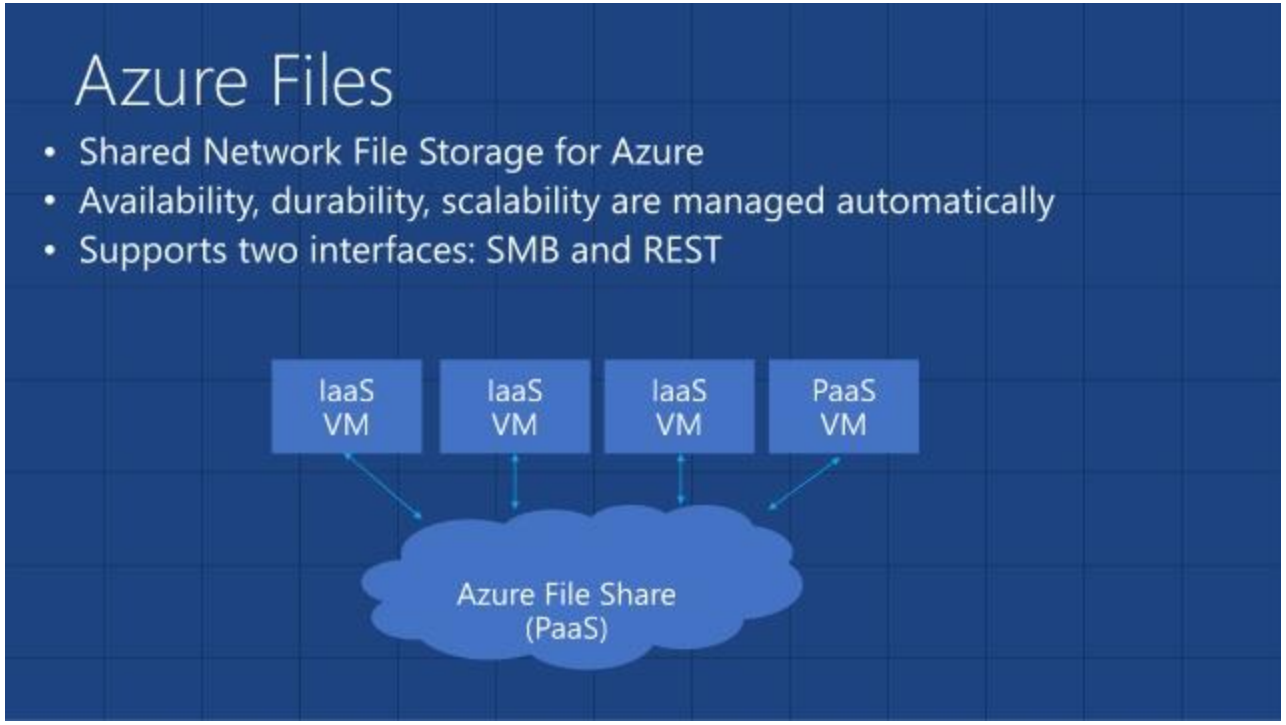
- **Geo-redundant storage (GRS)** replicates your data to a secondary region that is hundreds of miles away from the primary region. If your storage account has GRS enabled, then your data is durable even in the case of a complete regional outage or a disaster in which the primary region is not recoverable.

- **Read-access geo-redundant storage (RA-GRS)** maximizes availability for your storage account, by providing read-only access to the data in the secondary location, in addition to the replication across two regions provided by GRS

File sharing the old way in Azure

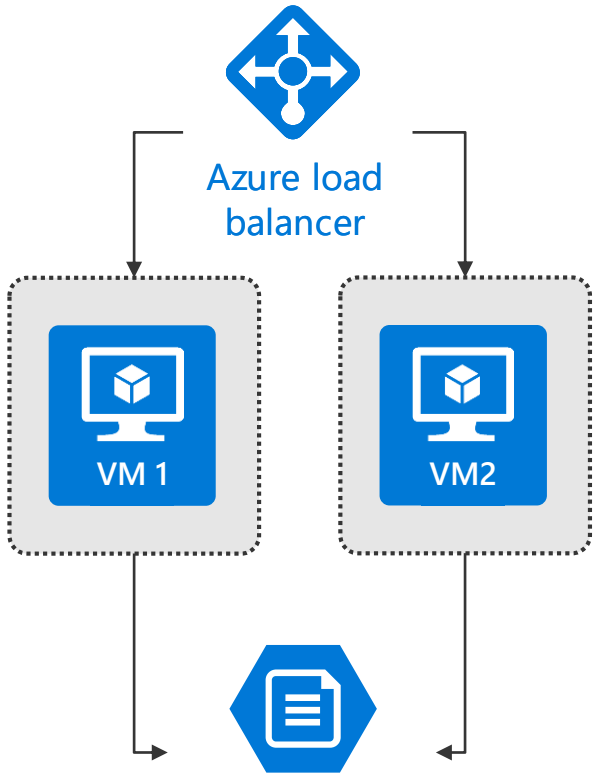


File sharing nowadays in Azure



Top Use Cases: Highly Available FTP Server

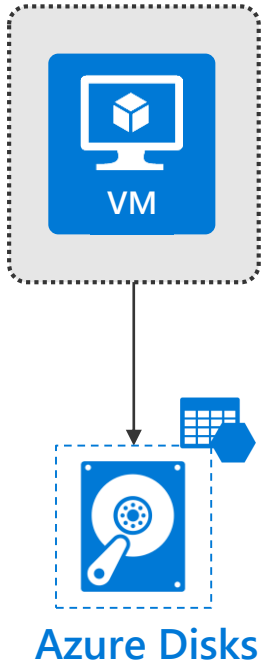
FTP server using Files



Increased availability through multiple VM instances

vs

FTP server using Disks



Single VM instance is prone to availability loss

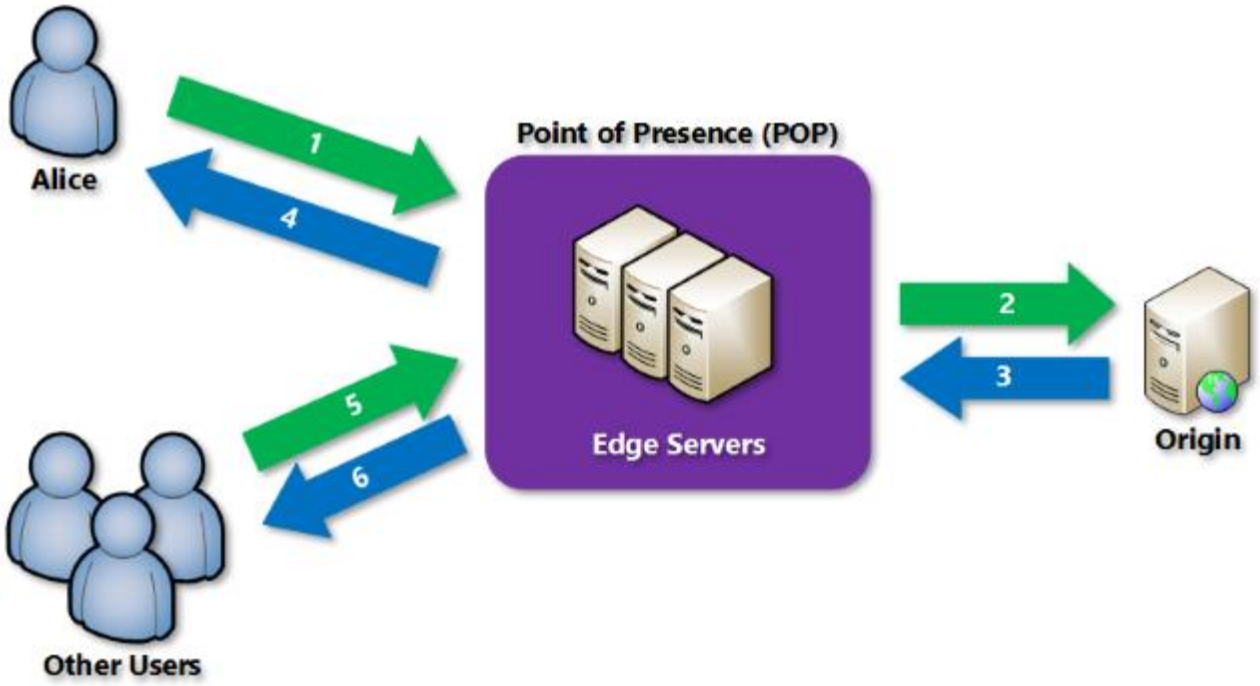
Content Delivery Network(CDN)

The Azure Content Delivery Network (CDN) caches static web content at strategically placed locations to provide maximum throughput for delivering content to users. The CDN offers developers a global solution for delivering high-bandwidth content by caching the content at physical nodes across the world.

It is designed to send audio, video, images, and other files faster and more reliably to customers using servers that are closest to the users. This dramatically increases speed and availability, resulting in significant user experience improvements.

Benefits

- Better performance and user experience for end users,
- Large scaling
- By distributing user requests and serving content from edge servers, less traffic is sent to the origin.



There are three Azure CDN products:

- Azure CDN Standard from Akamai,
- Azure CDN Standard from Verizon,
- Azure CDN Premium from Verizon

You can Integrate an Azure storage account with Azure CDN

Azure CDN POP Locations

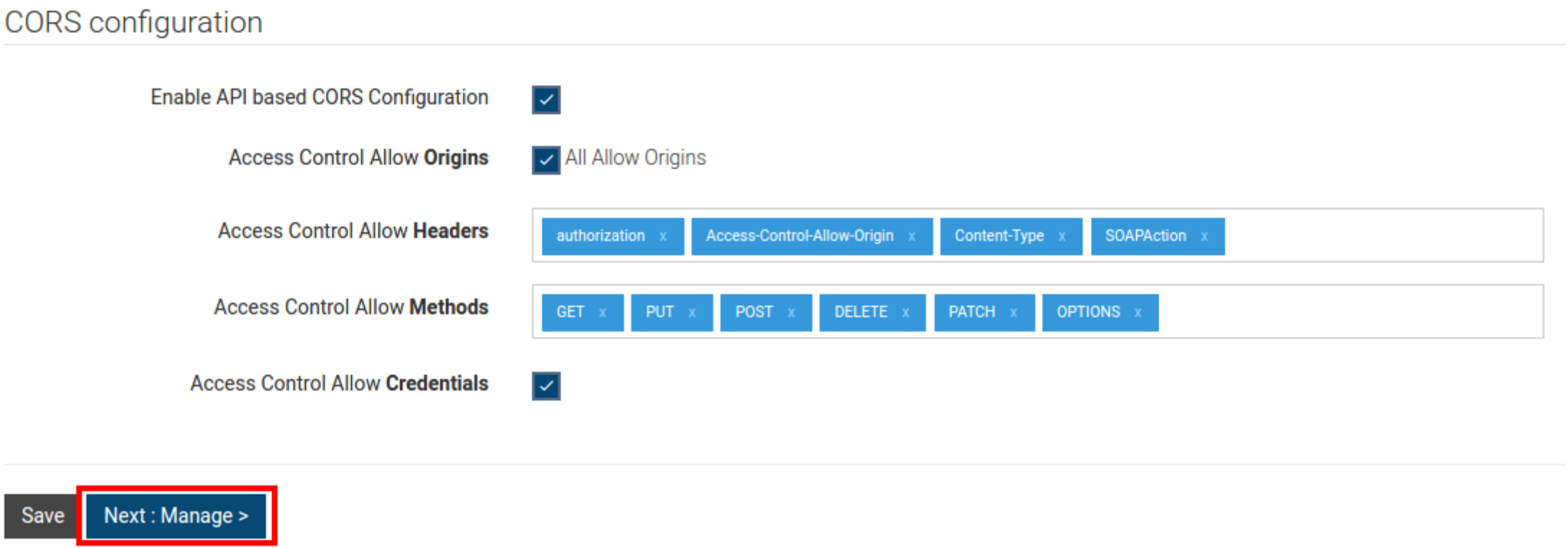
Region	Verizon	Akamai
North America	Atlanta, GA, Boston, MA , Chicago, IL , Dallas, TX, Denver, CO ,Los Angeles, CA Miami, FL ,New York, NY ,Philadelphia, PA, San Jose, CA ,Seattle, WA , Washington DC	Canada Mexico United States
South America	Buenos Aires, Argentina , Rio de Janeiro, Brazil ,São Paulo, Brazil ,Valparaíso, Chile Barranquilla, Colombia ,Medellin, Colombia , Quito, Ecuador ,Lima, Peru	Argentina ,Brazil ,Chile ,Colombia ,Ecuador ,Peru ,Uruguay
Northern and Eastern Europe	Copenhagen, Denmark, Helsinki, Finland ,Warsaw, Poland Stockholm, Sweden	Bulgaria , Denmark ,Finland ,Norway ,Poland ,Sweden
Western Europe	Vienna, Austria ,Paris, France ,Frankfurt, Germany , Amsterdam, Netherlands, London, UK	Austria , Belgium ,France ,Germany ,Ireland ,Netherlands ,Switzerland United Kingdom
Southern Europe	Milan, Italy Madrid, Spain	Greece ,Italy , Portugal ,Spain
East Asia	Hong Kong ,Batam, Indonesia ,Jakarta, Indonesia, Osaka, Japan, Tokyo, Japan ,Singapore Seoul, South Korea, Kaohsiung, Taiwan	Hong Kong, Indonesia , Japan, Macau, Malaysia, Philippines, Singapore, ,South Korea ,Taiwan ,Thailand
South and Central Asia	Bangalore, India , Chennai, India , Delhi, India ,Mumbai, India ,New Delhi, India	India Sri Lanka
Middle East/West Asia	Muscat, Oman Fujirah, United Arab Emirates	Israel, Kuwait, Qatar ,Turkey, United Arab Emirates
Africa		Egypt, South Africa
Australia and New Zealand	Melbourne, Australia, Sydney, Australia, Auckland, New Zealand	Australia, New Zealand

Cross-Origin Resource Sharing(CORS)

CORS is an HTTP feature that enables a web application running under one domain to access resources in another domain. Web browsers implement a security restriction known as same-origin policy that prevents a web page from calling APIs in a different domain. CORS provides a secure way to allow one domain (the origin domain) to call APIs in another domain.

We can set CORS rules individually for each of the storage services(i.e. blob, file, queue, table).Once we set the CORS rues for the service then a properly authenticated request made against the service from different domain will be evaluated to determine whether it is allowed according to the rules we have specified.

CORS rules allow clients to access blobs from any web domains you authorize.



Cross origin resource sharing is required when you are dealing with multiple domains and all of them need to be able to make calls to specific sub-domain or the API layer. Many times we even need to allow the Partner networks to have access to such API sub-domains.
One can do this on backend servers but it gets complicated quickly and every change needs to be replicated on multiple backend servers in the setup. Doing the same through the load balancing setup is a much simpler way to get there.

Process flow for CORS requests