

Azure for AWS Professionals

09/19/2018 • 15 minutes to read • Contributors  [all](#)

In this article

[Overview](#)

[Accounts and subscriptions](#)

[Resource management](#)

[Regions and zones \(high availability\)](#)

[Services](#)

[Next steps](#)

This article helps Amazon Web Services (AWS) experts understand the basics of Microsoft Azure accounts, platform, and services. It also covers key similarities and differences between the AWS and Azure platforms.

You'll learn:

- How accounts and resources are organized in Azure.
- How available solutions are structured in Azure.
- How the major Azure services differ from AWS services.

Azure and AWS built their capabilities independently over time so that each has important implementation and design differences.

Overview

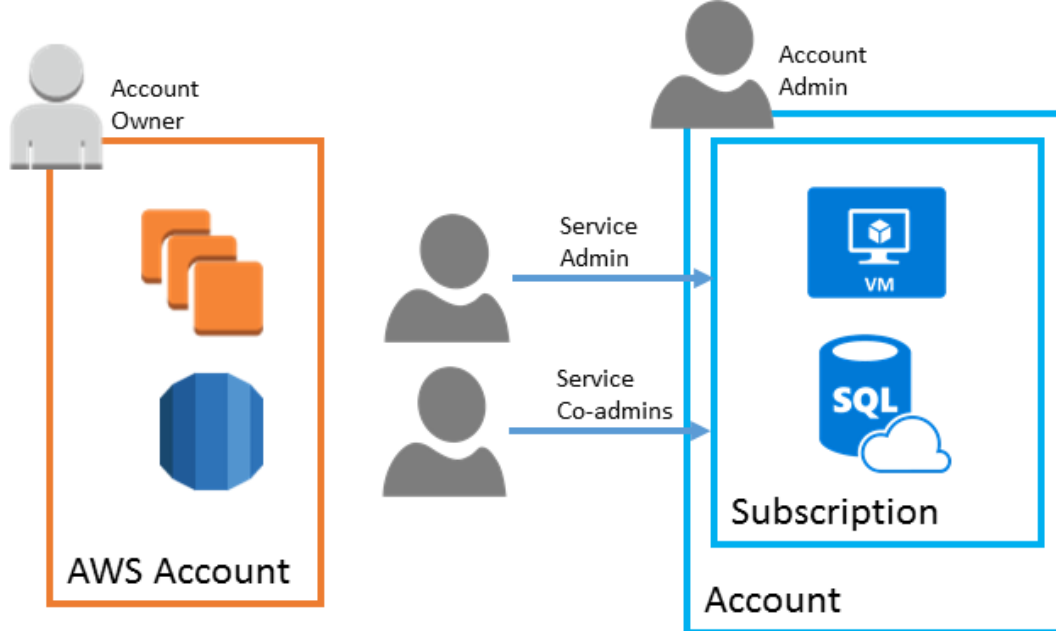
Like AWS, Microsoft Azure is built around a core set of compute, storage, database, and networking services. In many cases, both platforms offer a basic equivalence between the products and services they offer. Both AWS and Azure allow you to build highly available solutions based on Windows or Linux hosts. So, if you're used to development using Linux and OSS technology, both platforms can do the job.

While the capabilities of both platforms are similar, the resources that provide those capabilities are often organized differently. Exact one-to-one relationships between the services required to build a solution are not always clear. There are also cases where a particular service might be offered on one platform, but not the other. See [charts of comparable Azure and AWS services](#).

Accounts and subscriptions

Azure services can be purchased using several pricing options, depending on your organization's size and needs. See the [pricing overview](#) page for details.

[Azure subscriptions](#) are a grouping of resources with an assigned owner responsible for billing and permissions management. Unlike AWS, where any resources created under the AWS account are tied to that account, subscriptions exist independently of their owner accounts, and can be reassigned to new owners as needed.



Comparison of structure and ownership of AWS accounts and Azure subscriptions

Subscriptions are assigned three types of administrator accounts:

- **Account Administrator.** The subscription owner and the account billed for the resources used in the subscription. The account administrator can only be changed by transferring ownership of the subscription.
- **Service Administrator.** This account has rights to create and manage resources in the subscription, but is not responsible for billing. By default, the account administrator and service administrator are assigned to the same account. The account administrator can assign a separate user to the service administrator account for managing the technical and operational aspects of a subscription. There is only one service administrator per subscription.
- **Co-administrator.** There can be multiple co-administrator accounts assigned to a subscription. Co-administrators cannot change the service administrator, but otherwise have full control over subscription resources and users.

Below the subscription level user roles and individual permissions can also be assigned to specific resources, similarly to how permissions are granted to IAM users and groups in AWS. In Azure, all user accounts are associated with either a Microsoft Account or Organizational Account (an account managed through an Azure Active Directory).

Like AWS accounts, subscriptions have default service quotas and limits. For a full list of these limits, see [Azure subscription and service limits, quotas, and constraints](#). These limits can be increased up to the maximum by [filing a support request in the management portal](#).

See also

- [How to add or change Azure administrator roles](#)
- [How to download your Azure billing invoice and daily usage data](#)

Resource management

The term "resource" in Azure is used in the same way as in AWS, meaning any compute instance, storage object, networking device, or other entity you can create or configure within the platform.

Azure resources are deployed and managed using one of two models: [Azure Resource Manager](#), or the older Azure [classic deployment model](#). Any new resources are created using the Resource Manager model.

Resource groups

Both Azure and AWS have entities called "resource groups" that organize resources such as VMs, storage, and virtual networking devices. However, [Azure resource groups](#) are not directly comparable to AWS resource groups.

While AWS allows a resource to be tagged into multiple resource groups, an Azure resource is always associated with one resource group. A resource created in one resource group can be moved to another group, but can only be in one resource group at a time. Resource groups are the fundamental grouping used by Azure Resource Manager.

Resources can also be organized using [tags](#). Tags are key-value pairs that allow you to group resources across your subscription irrespective of resource group membership.

Management interfaces

Azure offers several ways to manage your resources:

- [Web interface](#). Like the AWS Dashboard, the Azure portal provides a full web-based management interface for Azure resources.
- [REST API](#). The Azure Resource Manager REST API provides programmatic access to most of the features available in the Azure portal.
- [Command Line](#). The Azure CLI 2.0 tool provides a command-line interface capable of creating and managing Azure resources. Azure CLI is available for [Windows, Linux, and Mac OS](#).
- [PowerShell](#). The Azure modules for PowerShell allow you to execute automated management tasks using a script. PowerShell is available for [Windows, Linux, and Mac OS](#).
- [Templates](#). Azure Resource Manager templates provide similar JSON template-based resource management capabilities to the AWS CloudFormation service.

In each of these interfaces, the resource group is central to how Azure resources get created, deployed, or modified. This is similar to the role a "stack" plays in grouping AWS resources during CloudFormation deployments.

The syntax and structure of these interfaces are different from their AWS equivalents, but they provide comparable capabilities. In addition, many third-party management tools used on AWS, like [Hashicorp's Terraform](#) and [Netflix Spinnaker](#), are also available on Azure.

See also

- [Azure resource group guidelines](#)

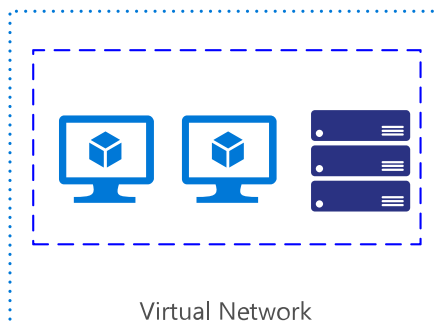
Regions and zones (high availability)

Failures can vary in the scope of their impact. Some hardware failures, such as a failed disk, may affect a single host machine. A failed network switch could affect a whole server rack. Less common are failures that disrupt a whole data center, such as loss of power in a data center. Rarely, an entire region could become unavailable.

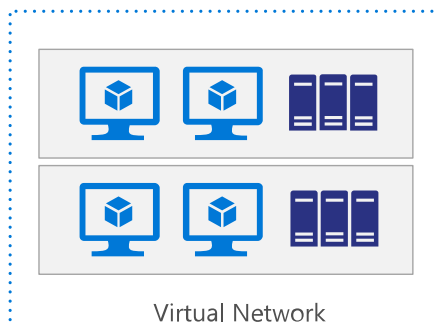
One of the main ways to make an application resilient is through redundancy. But you need to plan for this redundancy when you design the application. Also, the level of redundancy that you need depends on your business requirements—not every application needs redundancy across regions to guard against a regional outage. In general, there is a tradeoff between greater redundancy and reliability versus higher cost and complexity.

In AWS, a region is divided into two or more Availability Zones. An Availability Zone corresponds with a physically isolated datacenter in the geographic region. Azure has a number of features to make an application redundant at every level of failure, including **availability sets**, **availability zones**, and **paired regions**.

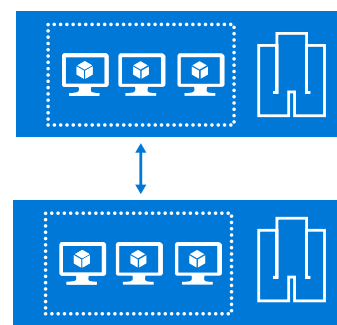
Availability Set



Availability Zones



Paired Regions



The following table summarizes each option.

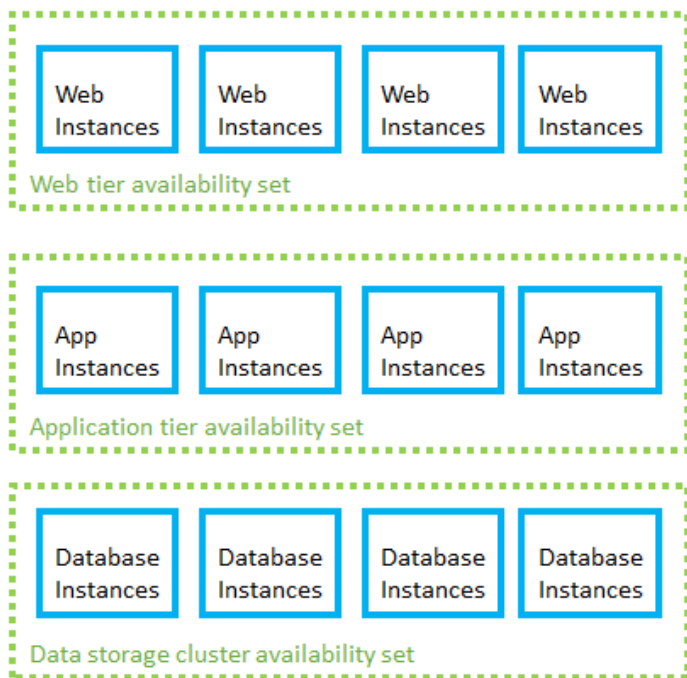
	Availability Set	Availability Zone	Paired region
Scope of failure	Rack	Datacenter	Region
Request routing	Load Balancer	Cross-zone Load Balancer	Traffic Manager
Network latency	Very low	Low	Mid to high
Virtual networking	VNet	VNet	Cross-region VNet peering

Availability sets

To protect against localized hardware failures, such as a disk or network switch failing, deploy two or more VMs in an availability set. An availability set consists of two or more *fault domains* that share a common power source and network switch. VMs in an availability set are distributed across the fault domains, so if a hardware failure affects one fault domain, network traffic can still be routed the VMs in the other fault domains. For more information about Availability Sets, see [Manage the availability of Windows virtual machines in Azure](#).

When VM instances are added to availability sets, they are also assigned an [update domain](#). An update domain is a group of VMs that are set for planned maintenance events at the same time. Distributing VMs across multiple update domains ensures that planned update and patching events affect only a subset of these VMs at any given time.

Availability sets should be organized by the instance's role in your application to ensure one instance in each role is operational. For example, in a three-tier web application, create separate availability sets for the front-end, application, and data tiers.



Availability zones

An [Availability Zone](#) is a physically separate zone within an Azure region. Each Availability Zone has a distinct power source, network, and cooling. Deploying VMs across availability zones helps to protect an application against datacenter-wide failures.

Paired regions

To protect an application against a regional outage, you can deploy the application across multiple regions, using [Azure Traffic Manager](#) to distribute internet traffic to the different regions. Each Azure region is paired with another region. Together, these form a [regional pair](#). With the exception of Brazil South, regional pairs are located within the same geography in order to meet data residency requirements for tax and law enforcement jurisdiction purposes.

Unlike Availability Zones, which are physically separate datacenters but may be in relatively nearby geographic areas, paired regions are typically separated by at least 300 miles. This is intended to ensure larger scale disasters only affect one of the regions in the pair. Neighboring pairs can be set to sync database and storage service data, and are configured so that platform updates are rolled out to only one region in the pair at a time.

Azure [geo-redundant storage](#) is automatically backed up to the appropriate paired region. For all other resources, creating a fully redundant solution using paired regions means creating a full copy of your solution in both regions.

See also

- [Regions and availability for virtual machines in Azure](#)
- [High availability for Azure applications](#)
- [Failure and disaster recovery for Azure applications](#)
- [Planned maintenance for Linux virtual machines in Azure](#)

Services

For a listing of how services map between platforms, see [AWS to Azure services comparison](#).

Not all Azure products and services are available in all regions. Consult the [Products by Region](#) page for details. You can find the uptime guarantees and downtime credit policies for each Azure product or service on the [Service Level Agreements](#) page.

The following sections provide a brief explanation of how commonly used features and services differ between the AWS and Azure platforms.

Compute services

EC2 Instances and Azure virtual machines

Although AWS instance types and Azure virtual machine sizes are categorized similarly, there are differences in the RAM, CPU, and storage capabilities.

- [Amazon EC2 Instance Types](#)
- [Sizes for virtual machines in Azure \(Windows\)](#).
- [Sizes for virtual machines in Azure \(Linux\)](#).

Similar to AWS' per second billing, Azure on-demand VMs are billed per second.

EBS and Azure Storage for VM disks

Durable data storage for Azure VMs is provided by [data disks](#) residing in blob storage. This is similar to how EC2 instances store disk volumes on Elastic Block Store (EBS). [Azure temporary storage](#) also provides VMs the same low-latency temporary read-write storage as EC2 Instance Storage (also called ephemeral storage).

Higher performance disk IO is supported using [Azure premium storage](#). This is similar to the Provisioned IOPS storage options provided by AWS.

Lambda, Azure Functions, Azure Web-Jobs, and Azure Logic Apps

[Azure Functions](#) is the primary equivalent of AWS Lambda in providing serverless, on-demand code. However, Lambda functionality also overlaps with other Azure services:

- [WebJobs](#) allow you to create scheduled or continuously running background tasks.
- [Logic Apps](#) provides communications, integration, and business rule management services.

Autoscaling, Azure VM scaling, and Azure App Service Autoscale

Autoscaling in Azure is handled by two services:

- [Virtual machine scale sets](#) allow you to deploy and manage an identical set of VMs. The number of instances can autoscale based on performance needs.
- [App Service Autoscale](#) provides the capability to autoscale Azure App Service solutions.

Container Service

The [Azure Kubernetes Service](#) supports Docker containers managed through Kubernetes.

Other compute services

Azure offers several compute services that do not have direct equivalents in AWS:

- [Azure Batch](#) allows you to manage compute-intensive work across a scalable collection of virtual machines.
- [Service Fabric](#) is a platform for developing and hosting scalable [microservice](#) solutions.

See also

- [Create a Linux VM on Azure using the portal](#)
- [Azure Reference Architecture: Running a Linux VM on Azure](#)
- [Get started with Node.js web apps in Azure App Service](#)
- [Azure Reference Architecture: Basic web application](#)
- [Create your first Azure Function](#)

Storage

S3/EBS/EFS and Azure Storage

In the AWS platform, cloud storage is primarily broken down into three services:

- **Simple Storage Service (S3)**. Basic object storage that makes data available through an Internet accessible API.
- **Elastic Block Storage (EBS)**. Block level storage intended for access by a single VM.
- **Elastic File System (EFS)**. File storage meant for use as shared storage for up to thousands of EC2 instances.

In Azure Storage, subscription-bound [storage accounts](#) allow you to create and manage the following storage services:

- [Blob storage](#) stores any type of text or binary data, such as a document, media file, or application installer. You can set Blob storage for private access or share contents publicly to the Internet. Blob storage serves the same purpose as both AWS S3 and EBS.
- [Table storage](#) stores structured datasets. Table storage is a NoSQL key-attribute data store that allows for rapid development and fast access to large quantities of data. Similar to AWS' SimpleDB and DynamoDB services.
- [Queue storage](#) provides messaging for workflow processing and for communication between components of cloud services.
- [File storage](#) offers shared storage for legacy applications using the standard server message block (SMB) protocol. File storage is used in a similar manner to EFS in the AWS platform.

Glacier and Azure Storage

[Azure Archive Blob Storage](#) is comparable to AWS Glacier storage service. It is intended for rarely accessed data that is stored for at least 180 days and can tolerate several hours of retrieval latency.

For data that is infrequently accessed but must be available immediately when accessed, [Azure Cool Blob Storage tier](#) provides cheaper storage than standard blob storage. This storage tier is comparable to AWS S3 - Infrequent Access storage service.

See also

- [Microsoft Azure Storage Performance and Scalability Checklist](#)
- [Azure Storage security guide](#)
- [Best practices for using content delivery networks \(CDNs\)](#)

Networking

Elastic Load Balancing, Azure Load Balancer, and Azure Application Gateway

The Azure equivalents of the two Elastic Load Balancing services are:

- [Load Balancer](#) - provides the same capabilities as the AWS Classic Load Balancer, allowing you to distribute traffic for multiple VMs at the network level. It also provides failover capability.
- [Application Gateway](#) - offers application-level rule-based routing comparable to the AWS Application Load Balancer.

Route 53, Azure DNS, and Azure Traffic Manager

In AWS, Route 53 provides both DNS name management and DNS-level traffic routing and failover services. In Azure this is handled through two services:

- [Azure DNS](#) provides domain and DNS management.
- [Traffic Manager](#) provides DNS level traffic routing, load balancing, and failover capabilities.

Direct Connect and Azure ExpressRoute

Azure provides similar site-to-site dedicated connections through its [ExpressRoute](#) service. ExpressRoute allows you to connect your local network directly to Azure resources using a dedicated private network connection. Azure also offers more conventional [site-to-site VPN connections](#) at a lower cost.

See also

- [Create a virtual network using the Azure portal](#)
- [Plan and design Azure Virtual Networks](#)
- [Azure Network Security Best Practices](#)

Database services

RDS and Azure relational database services

Azure provides several different relational database services that are the equivalent of AWS' Relational Database Service (RDS).

- [SQL Database](#)
- [Azure Database for MySQL](#)
- [Azure Database for PostgreSQL](#)

Other database engines such as [SQL Server](#), [Oracle](#), and [MySQL](#) can be deployed using Azure VM Instances.

Costs for AWS RDS are determined by the amount of hardware resources that your instance uses, like CPU, RAM, storage, and network bandwidth. In the Azure database services, cost depends on your database size, concurrent connections, and throughput levels.

See also

- [Azure SQL Database Tutorials](#)
- [Configure geo-replication for Azure SQL Database with the Azure portal](#)

- [Introduction to Cosmos DB: A NoSQL JSON Database](#)
- [How to use Azure Table storage from Node.js](#)

Security and identity

Directory service and Azure Active Directory

Azure splits up directory services into the following offerings:

- [Azure Active Directory](#) - cloud-based directory and identity management service.
- [Azure Active Directory B2B](#) - enables access to your corporate applications from partner-managed identities.
- [Azure Active Directory B2C](#) - service offering support for single sign-on and user management for consumer facing applications.
- [Azure Active Directory Domain Services](#) - hosted domain controller service, allowing Active Directory compatible domain join and user management functionality.

Web application firewall

In addition to the [Application Gateway Web Application Firewall](#), you can also use web application firewalls from third-party vendors like [Barracuda Networks](#).

See also

- [Getting started with Microsoft Azure security](#).
- [Azure Identity Management and access control security best practices](#)

Application and messaging services

Simple Email Service

AWS provides the Simple Email Service (SES) for sending notification, transactional, or marketing emails. In Azure, third-party solutions like [SendGrid](#) provide email services.

Simple Queueing Service

AWS Simple Queueing Service (SQS) provides a messaging system for connecting applications, services, and devices within the AWS platform. Azure has two services that provide similar functionality:

- [Queue storage](#): a cloud messaging service that allows communication between application components within Azure.
- [Service Bus](#): a more robust messaging system for connecting applications, services, and devices. Using the related [Service Bus relay](#), Service Bus can also connect to remotely hosted applications and services.

Device Farm

The AWS Device Farm provides cross-device testing services. In Azure, [Xamarin Test Cloud](#) provides similar cross-device front-end testing for mobile devices.

In addition to front-end testing, the [Azure DevTest Labs](#) provides back end testing resources for Linux and Windows environments.

See also

- [How to use Queue storage from Node.js](#)
- [How to use Service Bus queues](#)

Analytics and big data

[The Cortana Intelligence Suite](#) is Azure's package of products and services designed to capture, organize, analyze, and visualize large amounts of data. The Cortana suite consists of the following services:

- [HDInsight](#) - managed Apache distribution that includes Hadoop, Spark, Storm, or HBase.
- [Data Factory](#) - provides data orchestration and data pipeline functionality.
- [SQL Data Warehouse](#) - large-scale relational data storage.
- [Data Lake Store](#) - large-scale storage optimized for big data analytics workloads.
- [Machine Learning](#) - used to build and apply predictive analytics on data.
- [Stream Analytics](#) - real-time data analysis.
- [Data Lake Analytics](#) - large-scale analytics service optimized to work with Data Lake Store
- [Power BI](#) - used to power data visualization.

See also

- [Cortana Intelligence Gallery](#)
- [Understanding Microsoft big data solutions](#)
- [Azure Data Lake and Azure HDInsight Blog](#)

Internet of Things

See also

- [Get started with Azure IoT Hub](#)
- [Comparison of IoT Hub and Event Hubs](#)

Mobile services

Notifications

Notification Hubs do not support sending SMS or email messages, so third-party services are needed for those delivery types.

See also

- [Create an Android app](#)
- [Authentication and Authorization in Azure Mobile Apps](#)
- [Sending push notifications with Azure Notification Hubs](#)

Management and monitoring

See also

- [Monitoring and diagnostics guidance](#)
- [Best practices for creating Azure Resource Manager templates](#)
- [Azure Resource Manager Quickstart templates](#)

Next steps

- [Get started with Azure](#)
- [Azure solution architectures](#)
- [Azure Reference Architectures](#)