

Describe Azure compute services

1. Introduction

<https://learn.microsoft.com/en-us/training/modules/describe-azure-compute-networking-services/1-introduction>

Introduction

Completed

This module introduces you to Azure compute services and related innovation options. You learn about core compute choices such as virtual machines, containers, and Azure Functions. You also review application hosting options with Azure App Service and fundamentals of Azure AI, machine learning, and IoT/Edge services.

Learning objectives

After completing this module, you'll be able to:

- Compare compute types, including container instances, virtual machines, and functions.
- Describe virtual machine options, including virtual machines (VMs), virtual machine scale sets, and virtual machine availability sets.
- Describe resources required for virtual machines.
- Describe application hosting options, including Azure Web Apps, containers, and virtual machines.
- Describe AI, machine learning, and IoT/Edge service categories in Azure.

2. Describe Azure virtual machines

<https://learn.microsoft.com/en-us/training/modules/describe-azure-compute-networking-services/2-virtual-machines>

Describe Azure virtual machines

Completed

With Azure Virtual Machines (VMs), you can run virtualized servers in Azure as infrastructure as a service (IaaS). Like a physical server, you control the operating system and installed software. VMs are a good fit when you need:

- Total control over the operating system (OS).
- The ability to run custom software.
- To use custom hosting configurations.

Azure VMs remove the need to buy and maintain physical server hardware. As an IaaS service, you still manage patching, updates, and configuration inside the VM.

You can deploy VMs quickly from prebuilt images. An image is a template that already includes an operating system and tools such as web hosting components.

Examples of when to use VMs

Common VM use cases include:

- **Testing and development.** Create different OS and app configurations quickly, then remove the VM when testing is complete.
- **Cloud application hosting.** Run applications in Azure and scale capacity up or down as demand changes.
- **Datacenter extension.** Extend an on-premises network into Azure and host workloads in a connected virtual network.
- **Disaster recovery.** Keep failover capacity in Azure and run critical workloads there if your primary site is unavailable.
- **Lift and shift migration.** Move existing server workloads with minimal application redesign.

VM resources and sizing

When you provision a VM, you choose resources such as:

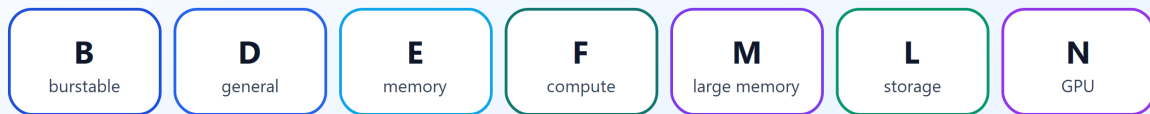
- Size (purpose, number of processor cores, and amount of RAM)
- Storage disks (hard disk drives, solid state drives, etc.)
- Networking (virtual network, public IP address, and port configuration)

Understand VM size families and names

Azure VM sizes are grouped into families so you can quickly choose a size based on your workload needs.

VM size families

Choose the family that best matches workload behavior.



The following table highlights common use cases for the Azure VM families.

Family	Typical focus	Example use
B-series	Burstable, cost-efficient	Dev/test workloads with occasional CPU spikes
D-series	General purpose	Web servers, small-to-medium app servers
E-series	Memory optimized	In-memory databases, analytics workloads
F-series	Compute optimized	CPU-intensive application tiers
M-series	Large memory footprint	Large enterprise databases
L-series	Storage optimized	High-throughput storage and data processing
N-series	GPU enabled	AI training/inference and graphics workloads

Each VM also has options that you can customize based on your needs. You can adjust the number of virtual CPUs (vCPUs), the amount of RAM, and the storage disk configuration.

Virtual machine options

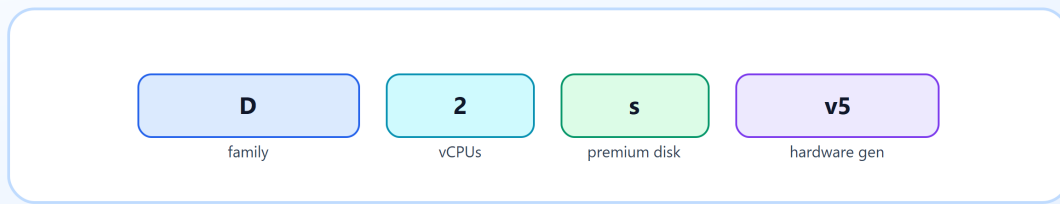


Some of the adjustments you can make are:

- **vCPU count:** affects compute capacity for concurrent and CPU-bound workloads.
- **RAM:** affects how much working data the VM can keep in memory.
- **Disk configuration:** affects storage capacity, IOPS, and throughput.
- **Network throughput:** affects data transfer performance in and out of the VM.
- **Premium SSD support:** indicates whether the size supports premium managed disks.
- **Hardware generation:** indicates platform generation and can affect baseline performance.

VM names also carry useful sizing information. The base name indicates the family (or purpose), the number of vCPUs, and the hardware generation. It may also indicate features such as premium storage support.

How to read Standard_D2s_v5



- **D** : the VM family (general purpose in this case)
- **2** : the number of vCPUs in this size
- **s** : supports Premium SSD storage
- **v5** : hardware generation for that family

At a fundamentals level, start by selecting a family that matches workload behavior, then choose the size that meets performance needs and scale as demand grows.

Scale and resiliency options for VMs

You can run single VMs for testing, development, or minor tasks. Or you can group VMs together to provide high availability, scalability, and redundancy. Azure can manage these groupings with features such as scale sets and availability sets.

Virtual machine scale sets

Virtual machine scale sets let you create and manage groups of identical, load-balanced VMs. Without scale sets, you must manually keep VM configuration consistent, monitor utilization, and adjust instance counts. Scale sets centralize configuration and can automatically scale out or in based on demand or schedules. They also integrate with load balancing so traffic is distributed efficiently.

Virtual machine availability sets

Virtual machine availability sets improve VM resiliency inside a region. They reduce the chance that all VMs are affected by one maintenance event or hardware failure.

Availability sets group VMs by:

- **Update domain:** VMs that can be rebooted together during planned maintenance.
- **Fault domain:** VMs that share a potential power or network failure point.

Use availability sets for VM-level redundancy. In regions that support Availability Zones, zone-based designs are often preferred because they provide broader failure isolation. Availability sets themselves don't add cost; you pay for the VM instances.

3. Describe Azure virtual desktop

Describe Azure virtual desktop

Completed

Azure Virtual Desktop is a desktop and application virtualization service in Azure. It lets users securely access Windows desktops and apps from many device types and locations.

At a fundamentals level, Azure Virtual Desktop is a managed option for remote desktop access where desktops and apps stay in the cloud instead of on local devices.

The following video gives you an overview of Azure Virtual Desktop:

When to use Azure Virtual Desktop

Use Azure Virtual Desktop when a team needs centralized desktop and app access across distributed users, contractors, or hybrid workers. For example, a support team can use standardized cloud-hosted desktops so each shift has the same tools, access policies, and security controls.

If you need a full desktop experience for many users with centralized management, Azure Virtual Desktop is often easier to operate than building separate VM-based desktop environments for each user group.



Key points

- It centralizes desktop and app delivery in Azure and integrates with Microsoft Entra ID for identity and access controls.
- It helps reduce data exposure on local devices because apps and data can remain in Azure-hosted sessions.
- It supports single-session and multi-session Windows experiences, depending on user and workload needs.

4. Describe Azure containers

<https://learn.microsoft.com/en-us/training/modules/describe-azure-compute-networking-services/5-containers>

Describe Azure containers

Completed

Virtual machines reduce costs compared to physical hardware, but they're still limited to a single operating system per VM. If you want to run multiple instances of an application on a single host machine, containers are an excellent choice.

What are containers?

Containers are a virtualization environment. Much like running multiple virtual machines on a single physical host, you can run multiple containers on a single physical or virtual host. Unlike virtual machines, you don't manage the operating system for a container. Each virtual machine runs its own operating system that you can connect to and manage. Containers are lightweight and designed to be created, scaled out, and stopped dynamically. You can create and deploy virtual machines as application demand increases, but containers are a lighter-weight, more agile method. Containers help you respond to changes on demand and restart quickly after a crash or hardware interruption. One of the most popular container engines is Docker, and Azure supports Docker.

Compare virtual machines to containers

The following video highlights several of the important differences between virtual machines and containers:

Azure Container Instances

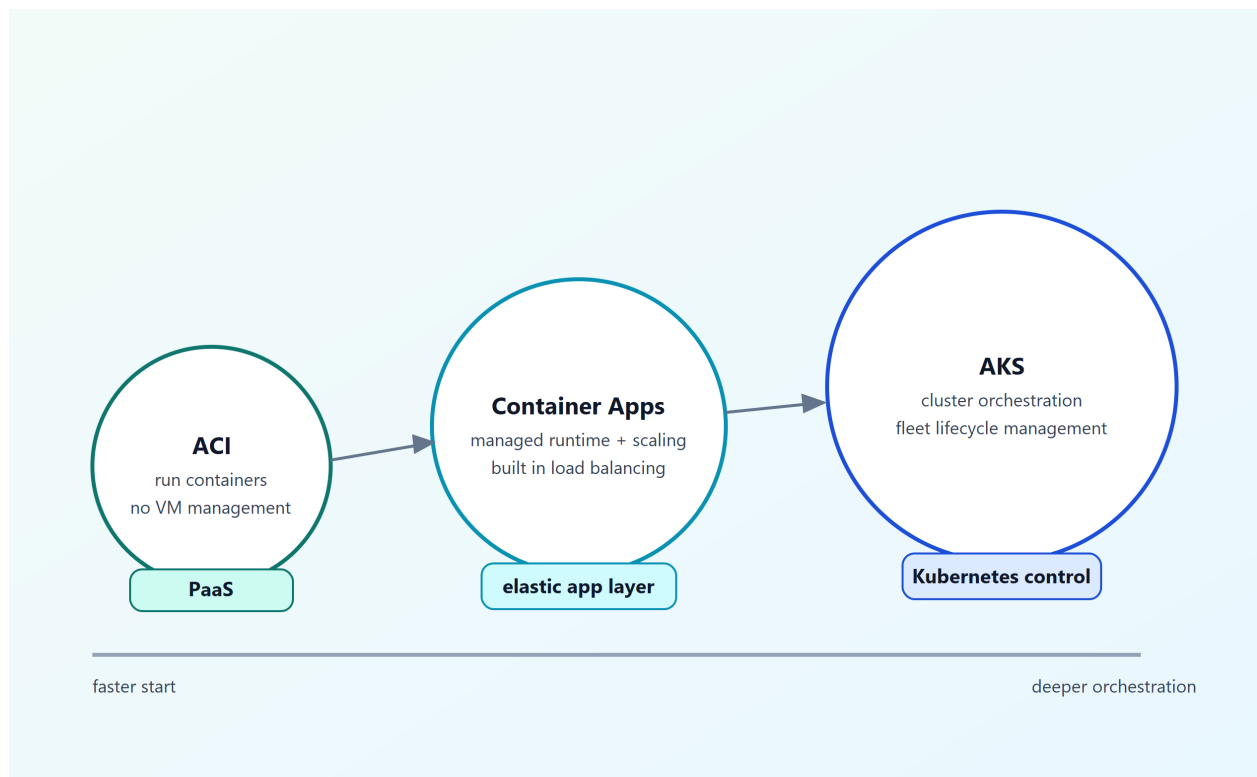
Azure Container Instances offer the fastest and simplest way to run a container in Azure, without managing any virtual machines or adopting extra services. Azure Container Instances are a platform as a service (PaaS) offering. You upload your containers and the service runs them for you.

Azure Container Apps

Azure Container Apps are similar in many ways to a container instance. They let you get up and running right away, they remove the container management overhead, and they're a PaaS offering. Container Apps also include built-in load balancing and scaling, so your design can adapt to changing demand.

Azure Kubernetes Service

Azure Kubernetes Service (AKS) is a container orchestration service. An orchestration service manages the lifecycle of containers. When you're deploying a fleet of containers, AKS can make fleet management simpler and more efficient.



Use containers in your solutions

Containers are often used to create solutions that use a microservice architecture. In this architecture, you break solutions into smaller, independent pieces. For example, you might split a website into a container hosting your front end, another hosting your back end, and a third for storage. This split lets you maintain, scale, or update each part of your app independently.

Imagine your website back end reaches capacity, but the front end and storage aren't stressed. With containers, you can scale the back end separately to improve performance. You can also change the storage service or modify the front end without affecting the other components.

5. Describe Azure functions

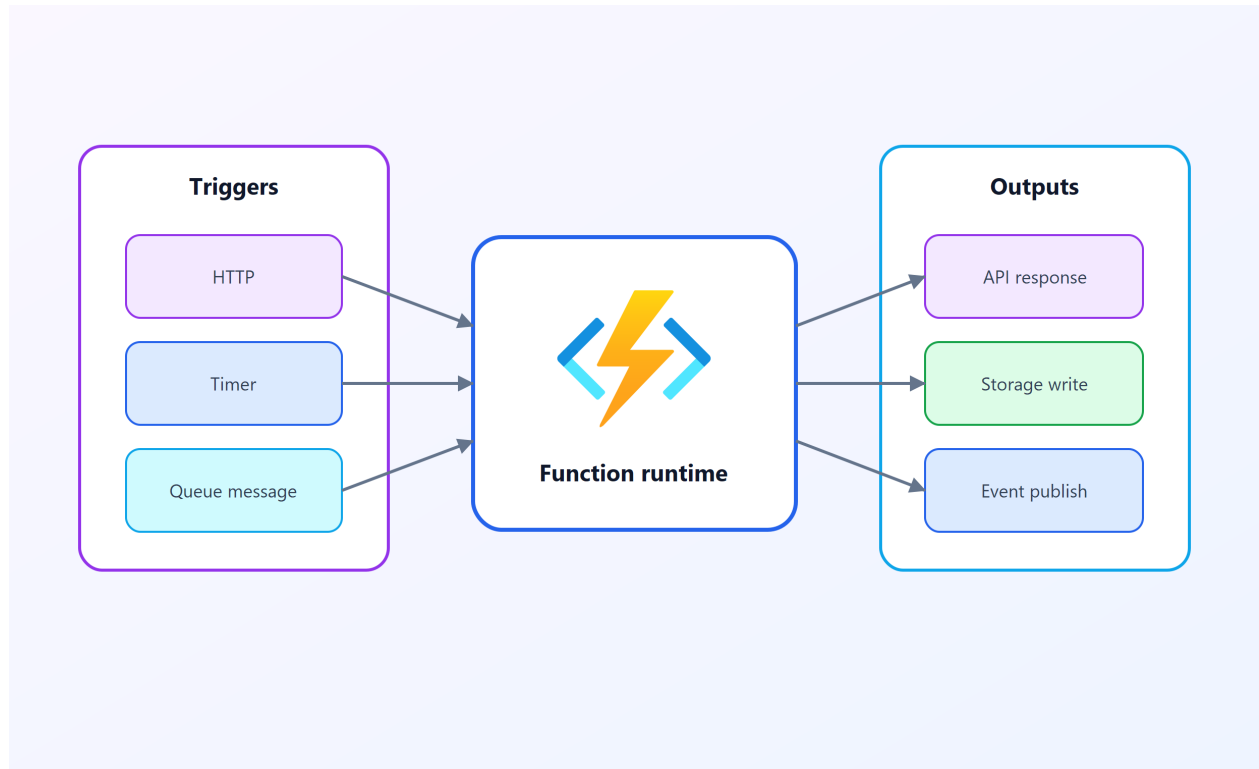
<https://learn.microsoft.com/en-us/training/modules/describe-azure-compute-networking-services/6-functions>

Describe Azure functions

Completed

Azure Functions is an event-driven, serverless compute option that doesn't require maintaining virtual machines or containers. If you build an app using VMs or containers, those resources have to

be “running” in order for your app to function. With Azure Functions, an event wakes the function, alleviating the need to keep resources provisioned when there are no events.



Serverless computing in Azure

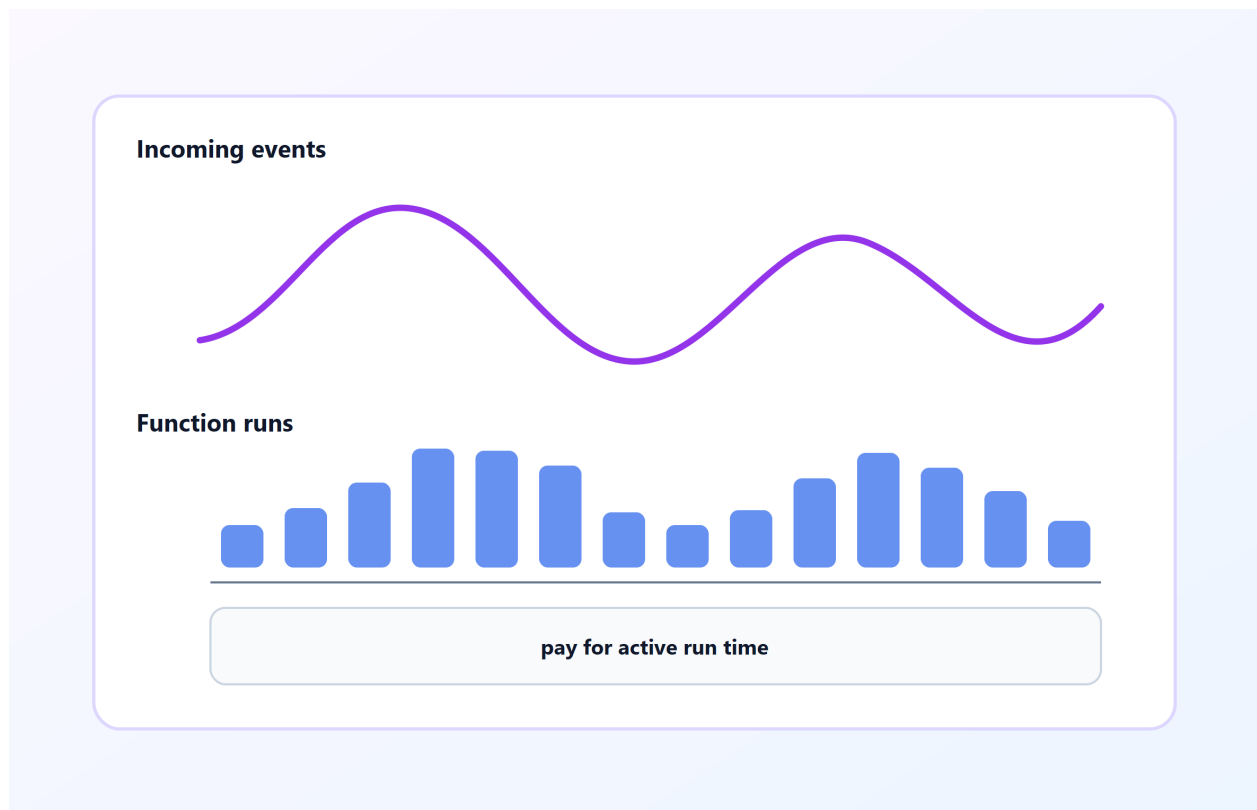
Benefits of Azure Functions

Using Azure Functions is ideal when you're only concerned about the code running your service and not about the underlying platform or infrastructure. Functions are commonly used when you need to perform work in response to an event (often via a REST request), timer, or message from another Azure service, and when that work can be completed quickly, within seconds or less.

Functions scale automatically based on demand, so they may be a good choice when demand is variable.

Azure Functions runs your code when it's triggered and automatically deallocates resources when the function finishes. In this model, Azure charges you only for the CPU time used while your function

runs.



Functions can be either stateless or stateful. When they're stateless (the default), they behave as if they restart every time they respond to an event. When they're stateful (called Durable Functions), the runtime passes a context through the function to track prior activity.

Functions are a key component of serverless computing. They're also a general compute platform for running any type of code. If the needs of your app change, you can deploy the project in an environment that isn't serverless. This flexibility lets you manage scaling, run on virtual networks, and even completely isolate the functions.

6. Describe application hosting options

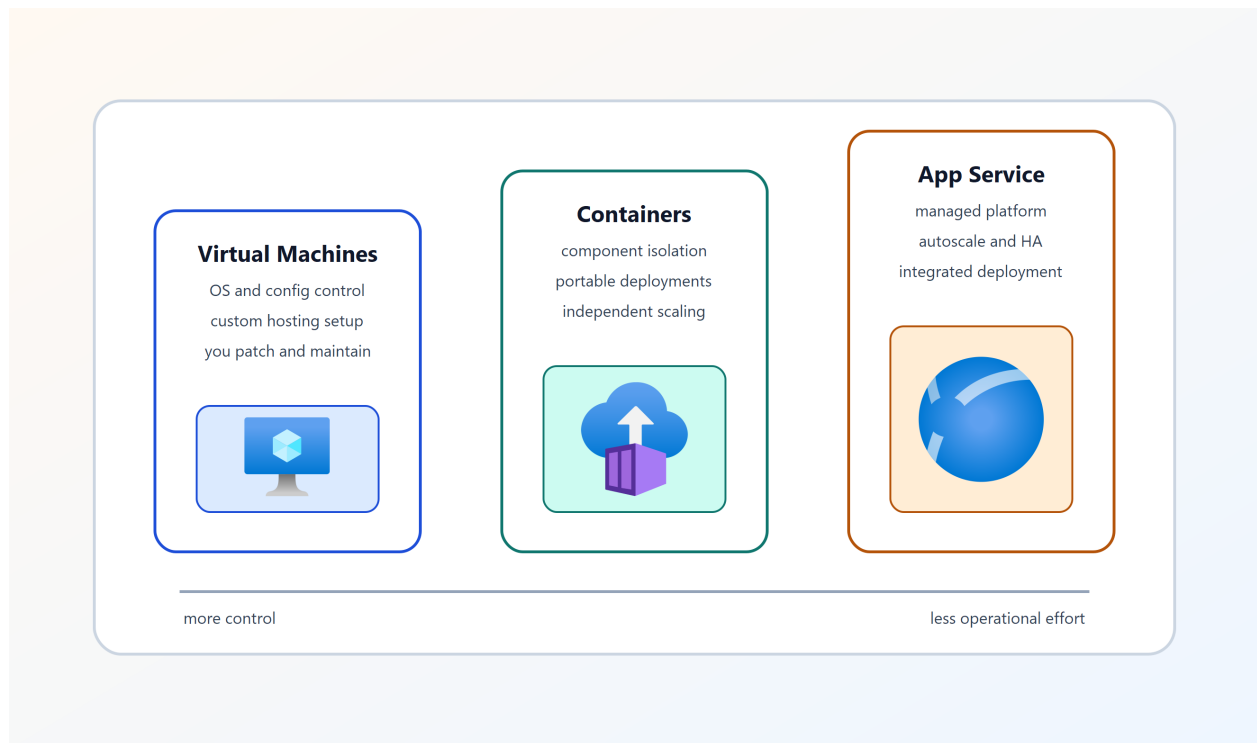
<https://learn.microsoft.com/en-us/training/modules/describe-azure-compute-networking-services/7-describe-application-hosting-options>

Describe application hosting options

Completed

If you need to host your application on Azure, you might initially turn to a virtual machine (VM) or containers. Both VMs and containers provide excellent hosting solutions. VMs give you maximum control of the hosting environment and allow you to configure it exactly how you want. VMs also may be the most familiar hosting method if you're new to the cloud. Containers, with the ability to isolate and individually manage different aspects of the hosting solution, can also be a robust and compelling option.

There are other hosting options that you can use with Azure, including Azure App Service.



Azure App Service

App Service lets you build and host web apps, background jobs, mobile back-ends, and RESTful APIs in the programming language of your choice without managing infrastructure. It offers automatic scaling and high availability. App Service supports Windows and Linux and supports automated deployments from GitHub, Azure DevOps, or any Git repo for continuous deployment.

Azure App Service is an HTTP-based service for hosting web applications, REST APIs, and mobile back ends. It lets you focus on building and maintaining your app while Azure keeps the environment up and running. App Service supports multiple languages and frameworks, including .NET, .NET Core, Java, PHP, Python, and Node.js.

Types of app services

With App Service, you can host most common app styles:

- Web apps
- API apps

- WebJobs
- Mobile apps

App Service handles most of the infrastructure decisions you deal with in hosting web-accessible apps:

- Deployment and management are integrated into the platform.
- Endpoints can be secured.
- Sites can be scaled quickly to handle high traffic loads.
- The built-in load balancing and traffic manager provide high availability.

All of these app styles are hosted in the same infrastructure and share these benefits. This flexibility makes App Service the ideal choice to host web-oriented applications.

Web apps

App Service includes full support for hosting web apps by using ASP.NET, ASP.NET Core, Java, Ruby, Node.js, PHP, or Python. You can choose either Windows or Linux as the host operating system.

API apps

Much like hosting a website, you can build REST-based web APIs by using your choice of language and framework. App Service provides full Swagger support and the ability to package and publish your API in Azure Marketplace. The published APIs can be consumed from any HTTP- or HTTPS-based client.

WebJobs

You can use the WebJobs feature to run a program (.exe, Java, PHP, Python, or Node.js) or script (.cmd, .bat, PowerShell, or Bash) in the same context as a web app, API app, or mobile app. They can be scheduled or run by a trigger. WebJobs are often used to run background tasks as part of your application logic.

Mobile apps

Use the Mobile Apps feature of App Service to quickly build a back end for iOS and Android apps. With just a few actions in the Azure portal, you can:

- Store mobile app data in a cloud-based SQL database.
- Authenticate customers against common social providers, such as MSA, Google, X, and Facebook.
- Send push notifications.
- Execute custom back-end logic in C# or Node.js.

On the mobile app side, there's SDK support for native iOS and Android, Xamarin, and React Native apps.

7. Describe AI, machine learning, and IoT/Edge services in Azure

<https://learn.microsoft.com/en-us/training/modules/describe-azure-compute-networking-services/12a-describe-ai-ml-iot-edge-services>

Describe AI, machine learning, and IoT/Edge services in Azure

Completed

Azure includes service categories that help you build intelligent and connected solutions without starting from scratch.

Azure AI services

Azure AI services provides prebuilt capabilities for common AI scenarios, such as language, speech, vision, and document processing. These services are useful when you want to add intelligent features through APIs instead of training your own model first.

Azure OpenAI Service is another AI option in Azure that supports generative AI scenarios, such as chat and content generation, with built-in security and governance controls.

Agentic AI patterns

Agentic applications combine an AI model with instructions, context, and tool use to complete multistep goals. In Azure, you usually build these patterns by combining Azure AI services and Azure OpenAI Service with your own application logic.

At a fundamentals level, treat agentic AI as an application pattern built from Azure AI capabilities, not as a separate compute service category.

Azure Machine Learning

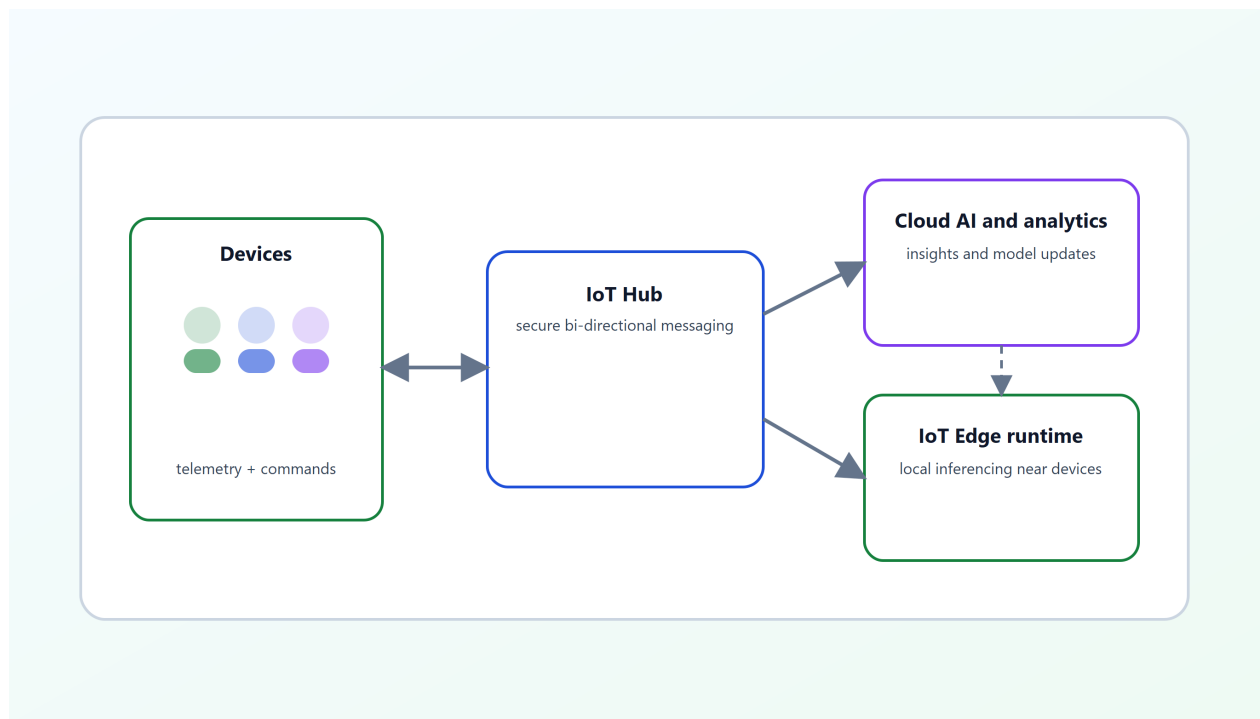
Use Azure Machine Learning when you need to build, train, and manage custom machine learning models. This option is a better fit when your scenario requires model development, experimentation, and lifecycle management.

IoT and Edge services

Azure IoT services help you connect, monitor, and manage devices.

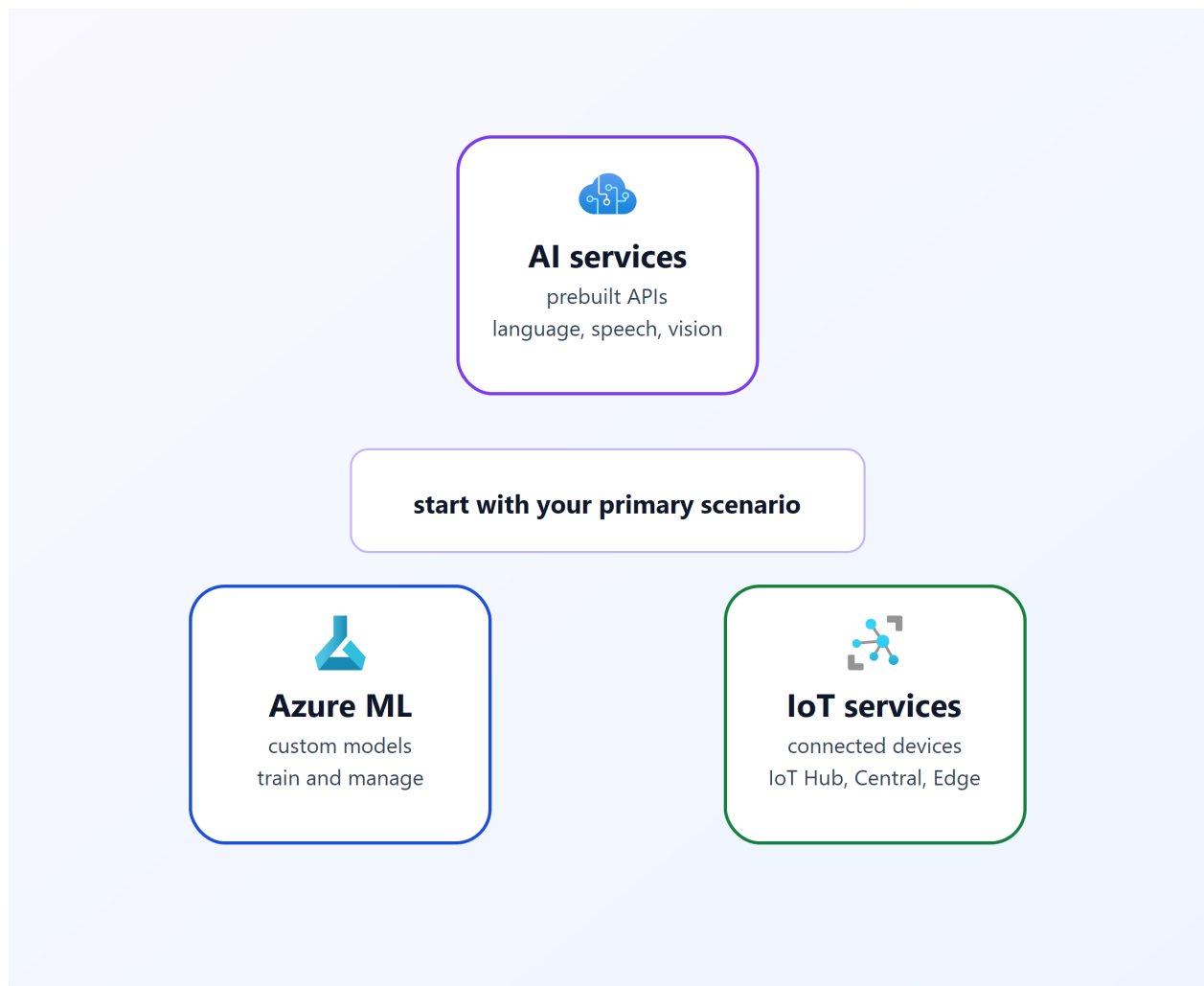
- Azure IoT Hub enables secure, bi-directional communication between cloud services and IoT devices.
- Azure IoT Central provides a simplified software as a service (SaaS) IoT platform for solution builders.
- Azure IoT Edge extends cloud capabilities to edge devices so some workloads can run closer to where data is generated.

This flow commonly starts with devices sending telemetry through IoT Hub, then uses cloud analytics to generate insights and model updates that IoT Edge runtime applies near the devices. IoT Central can simplify solution management for these connected-device scenarios.



Choosing the right option

At a fundamentals level, use this decision pattern:



- Use Azure AI services when you need prebuilt AI features exposed through APIs.
- Use Azure Machine Learning when you need custom model development and management.
- Use Azure IoT services when your solution centers on connected devices and telemetry.

8. Module assessment

<https://learn.microsoft.com/en-us/training/modules/describe-azure-compute-networking-services/13-knowledge-check>

Module assessment

Completed

9. Summary

<https://learn.microsoft.com/en-us/training/modules/describe-azure-compute-networking-services/14-summary>

Summary

Completed

In this module, you learned about core Azure compute services and when to use them. You reviewed virtual machines and options such as virtual machine scale sets and virtual machine availability sets. You explored containers, Azure Functions, application hosting options, and Azure AI, machine learning, and IoT/Edge service categories.

You learned that agentic AI is an application pattern that can be built by combining Azure AI capabilities with application logic when a solution needs multistep reasoning and tool use.

Learning objectives

You should now be able to:

- Compare compute types, including container instances, virtual machines, and functions.
- Describe virtual machine options, including virtual machines (VMs), virtual machine scale sets, and virtual machine availability sets.
- Describe resources required for virtual machines.
- Describe application hosting options, including Azure Web Apps, containers, and virtual machines.
- Describe AI, machine learning, and IoT/Edge service categories in Azure.
- Recognize agentic AI as an application pattern built from Azure AI capabilities.

Additional resources

The following additional resources are intended to provide more information on topics in this module or on additional topics related to this module.

- [Host a web application with Azure App Service](#) is a Microsoft Learn module that explores the process of hosting a web application in Azure.

Explore with Copilot

Tip

Try one of these prompts in Copilot Chat:

- "Build a decision matrix for when to use virtual machines, containers, Azure Functions, and Azure App Service across three realistic workloads."
- "Explain scale sets, availability sets, and Availability Zones with one failure scenario that shows how each option improves resiliency."
- "Create a short architecture walkthrough that combines compute hosting with one AI or IoT Edge service and explains why each component was chosen."