

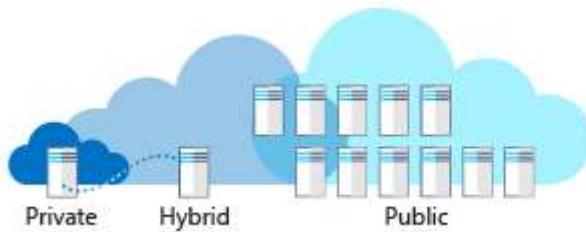
Manage resources in Azure

Align requirements with cloud types and service models in Azure

Cloud computing is the provisioning of services and applications on demand over the internet. Servers, applications, data, and other resources are provided as a service.

To the user, the details of the services are abstracted. You can quickly provision computing resources and use the service with minimal management. You shouldn't think of cloud computing as a datacenter available through the internet. Cloud computing uses virtualization, commodity hardware, and automated processes to provide a self-service user experience to customers similar to a public utility.

There are three deployment models for cloud computing: public cloud, private cloud, and hybrid cloud.



Public cloud

Public clouds are the most common way of deploying cloud computing. Services are offered over the public internet and available to anyone who wants to purchase them. The cloud resources such as servers and storage are owned and operated by a third-party cloud service provider and delivered over the internet. Services may be free or sold on demand, allowing customers to pay only per usage for the CPU cycles, storage, or bandwidth they consume. Microsoft Azure is an example of a public cloud.

Let's imagine your healthcare company needs an enrollment website. The site needs to scale and be responsive during peak enrollment various times during the year. Your customers access the site from global locations. You can use the public cloud to automatically scale up to meet demand at peak enrollment. When site traffic is low, your site can scale down to save costs. Your site is responsive at peak demand, and you only pay for more resources when needed. You can also deploy your website in multiple geographic regions to increase reliability and responsiveness.

During the development of your website, developers want to create multiple development environments to speed their development process. Developers can use the public cloud to quickly provision virtual machines for sandboxed environments to develop a solution. When the developers no longer need an environment, they can delete it.

Why public cloud?

Public clouds can be deployed faster than on-premises infrastructures and with an almost infinitely scalable platform. Every employee of a company can use the same application from any office or branch using their device of choice as long as they can access the internet.

Examples of why you would use public cloud:

- **Service consumption through on-demand or subscription model:** The on-demand or subscription model allows you to pay for the portion of CPU, storage, and other resources that you use or reserve.
- **No up-front investment of hardware:** No requirement to purchase, manage, and maintain on-premises hardware and application infrastructure. The cloud service provider is held responsible for all management and maintenance of the system.
- **Automation:** Quickly provision infrastructure resources using a web portal, scripts, or via automation.
- **Geographic dispersity:** Store data near your users, or in desired locations without having to maintain your own datacenters.
- **Reduced hardware maintenance:** The service provider is responsible for hardware maintenance.

Private cloud

A private cloud consists of computing resources used exclusively by users from one business or organization. It can be physically located at your organization's on-site datacenter, or it can be hosted by a third-party service provider. The term private cloud should not be considered a rebranding of traditional on-premises datacenters. A private cloud uses on-premises infrastructure and services to provide similar benefits of the public cloud. It uses an abstraction platform to provide cloud-like services such as Kubernetes clusters or a complete cloud environment like Azure Stack. The organization is responsible for the purchase, configuration, and maintenance of the hardware. Communication between the systems is usually on the network infrastructure that the business owns and maintains. For example, a private internal network or a dedicated fiber optic connection between buildings.

Imagine you work at healthcare company and you have an application that is in use at one of your datacenters. The operating environment can't be replicated in the public cloud. You have a new requirement to access data at another one of your datacenters. The database containing the data needs to remain at the other site because of regulatory compliance. This scenario is a private cloud. You have two datacenters your organization owns. You could use a public cloud VPN over the internet to connect the datacenters. However, the scenario would be considered a private cloud since the solution is private to the organization.

Why private cloud?

A private cloud can provide more flexibility to an organization. Your organization can customize its cloud environment to meet specific business needs. Since resources are not shared with others, high levels of control and security are possible. Also, private clouds can provide a level of scalability and efficiency.

Examples of why you would use private cloud:

- **Pre-existing environment:** An existing operating environment that can't be replicated in the public cloud. A large investment in hardware and employees with solution expertise. A large organization may choose to commoditize their computing resources.
- **Legacy applications:** Business-critical legacy applications that can't easily be physically relocated.
- **Data sovereignty and security:** Political borders and legal requirements may dictate where data can physically exist.
- **Regulatory compliance / certification:** PCI or HIPAA compliance. Certified on-premises datacenter.

Hybrid cloud

A hybrid cloud is a computing environment that combines a public cloud and a private cloud by allowing data and applications to be shared between them. When computing and processing demand fluctuates, hybrid cloud computing gives businesses the ability to seamlessly scale their on-premises infrastructure up to the public cloud to handle any overflow - without giving third-party datacenters access to the entirety of their data. Organizations gain the flexibility and computing power of the public cloud for basic and non-sensitive computing tasks, while keeping business-critical applications and data on-premises, safely behind a company firewall.

Using a hybrid cloud helps eliminate the need to make up-front capital expenditures to handle short-term spikes in demand. It also has the flexibility to manage which resources are local versus resources in the cloud. Companies pay only for resources they temporarily use instead of having to purchase, program, and maintain additional resources and equipment that could remain idle over long periods of time. Integration is generally through a secure VPN between cloud providers like Azure and on-premises datacenters.

Imagine you work at healthcare company and you have an application where customers can access their healthcare information. A regulation requires that the data needs to remain at a physical location. The customer website needs to be responsive to their many global users. As a solution, the database could be hosted in an on-premises datacenter and the website could be hosted in the public cloud. A VPN is used between the on-premises datacenter and the public cloud. This scenario would be considered a hybrid cloud.

Why hybrid cloud?

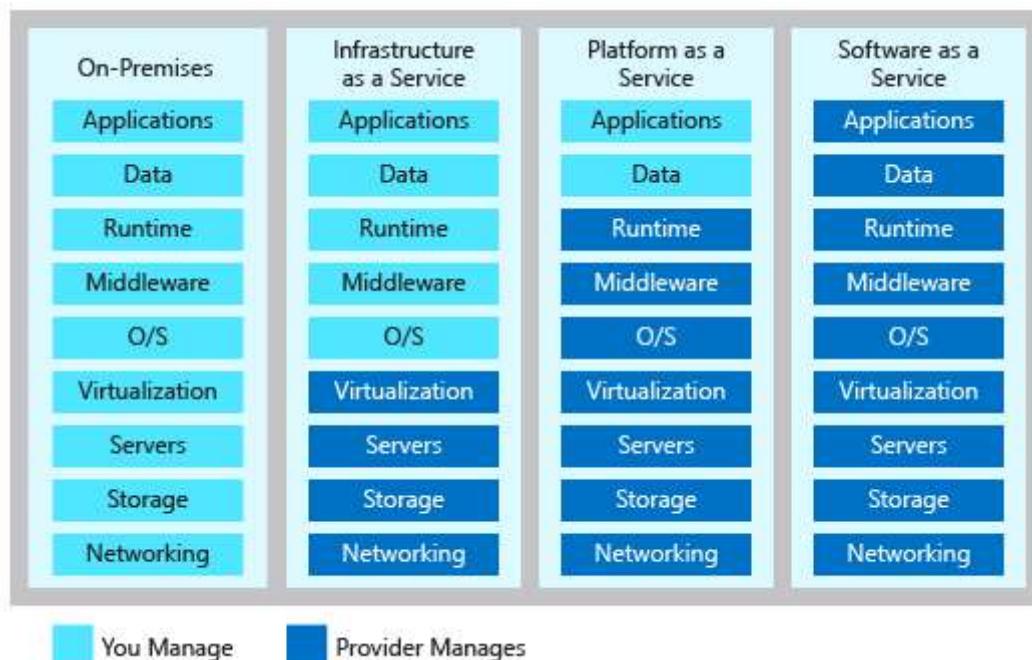
Hybrid cloud allows your organization to control and maintain a private infrastructure for sensitive assets. It also gives you the flexibility to take advantage of additional resources in the public cloud when you need them. With the ability to scale to the public cloud, you pay for extra computing power only when needed. It can also ease transitioning to the cloud. You can migrate gradually by phasing in workloads over time.

Examples of why you would use hybrid cloud:

- **Existing hardware investment:** Business reasons require that you use an existing operating environment and hardware.
- **Regulatory requirements:** Regulation requires that the data needs to remain at a physical location.
- **Unique operating environment:** Public cloud can't replicate a legacy operating environment.
- **Migration:** Move workloads to the cloud over time.

Cloud computing resources are delivered using three different service models.

- **Infrastructure-as-a-service (IaaS)** provides instant computing infrastructure that you can provision and manage over the Internet. (managing VMs)
- **Platform as a service (PaaS)** provides ready-made development and deployment environments that you can use to deliver your own cloud services. (managing Web Apps)
- **Software as a service (SaaS)** delivers applications over the Internet as a web-based service. (using o365 or Power BI)



Infrastructure as a service

Infrastructure as a service (IaaS) is an instant computing infrastructure, provisioned and managed over the Internet. IaaS enables you to quickly scale resources to meet demand and only pay for what you use. IaaS avoids the expense and complexity of buying and managing your own physical servers and other datacenter infrastructure. Each resource is offered as a separate service component, and you rent the resource as long as you need it. As a result, IaaS is flexible. You can provision common infrastructure such as VMs, storage, virtual subnets, firewalls, and VPNs to build a solution. You don't need to manage physical servers and appliances. However, you are responsible for configuring and managing the components. For example, configuring firewalls, updating VM OS's, updating DBMS's, and runtimes.

Common scenarios

Let's imagine your healthcare company has a need to run a special version of desktop software. The software is only supported on a specific version of an operating system and only one user license is required. You can create a virtual machine with the required software. The user can use a remote desktop connection to connect to the virtual machine to use the software.

Let's imagine another scenario. Your development teams need several unique development environments. Through the development cycle, they need to test various versions of the product. The developers can provision environments when needed. When an environment is no longer needed, it can be easily deleted.

Some other common scenarios include:

Website hosting: If you want more control of hosting a website, running websites using IaaS may be a better option than traditional web hosting.

Web apps: IaaS provides all the infrastructure to support web apps, including storage, web and application servers, and networking resources. Organizations can quickly deploy web apps on IaaS and easily scale infrastructure up and down when demand for the apps is unpredictable.

Storage, backup, and recovery: Storage management can be complex, requiring a large capital investment and skilled staff to manage data and meet legal and compliance requirements. IaaS can help simplify planning, management, unpredictable demand, and steadily growing storage needs.

High-performance computing: If you have a workload that requires high-performance computing, you can run the workload in the cloud avoiding the up-front cost of the hardware and only pay for the usage when needed.

Big data analysis: If you have large data sets that contain potentially valuable patterns, trends, and associations, IaaS can provide the processing power to mine data sets to locate patterns.

Advantages

Eliminates capital expense and reduces ongoing cost: IaaS sidesteps the upfront expense of setting up and managing an on-site datacenter, making it an economical option for start-ups and businesses testing new ideas. As soon as you've decided to launch a new product or initiative, the necessary computing infrastructure can be ready in minutes or hours, rather than the days or weeks—and sometimes months—it could take to set up internally.

Improves business continuity and disaster recovery: Achieving high availability, business continuity, and disaster recovery is expensive, since it requires a significant amount of technology and staff. But with the right service level agreement (SLA) in place, IaaS can reduce this cost and access applications and data as usual during a disaster or outage.

Respond quicker to shifting business conditions: IaaS enables you to quickly scale up resources to accommodate spikes in demand for your application—during the holidays, for example—then scale resources back down again when activity decreases to save money. Because you don't need to first set up the infrastructure before you can develop and deliver apps, you can get them to users faster with IaaS.

Increase stability, reliability, and supportability: With IaaS, there's no need to maintain and upgrade hardware or troubleshoot equipment problems. With the appropriate agreement in place, the service provider assures that your infrastructure is reliable and meets SLAs.

Platform as a service

Platform as a service (PaaS) is a complete development and deployment environment in the cloud. With PaaS, you can build and deploy everything from simple cloud-based apps to sophisticated, cloud-enabled enterprise applications. You purchase the resources from a cloud service provider on a pay-as-you-go basis and access them over a secure Internet connection. Like IaaS, PaaS includes infrastructure such as servers, storage, and networking. In addition, it also includes middleware, development tools, and other services. PaaS supports the complete web application lifecycle: building, testing, deploying, managing, and updating. PaaS removes the need to manage software licenses, middleware, and infrastructure of the services. You manage the applications and services you develop, and the cloud service provider typically manages everything else.

Common scenarios

Let's imagine your healthcare company needs a website to describe a product. Your developers want to use PHP. Using PaaS, your developers have the option to create a web app. The infrastructure details such as creating a virtual machine, installing a web server, and installing middleware are abstracted away. You don't need to care what operating system it runs on or what physical hardware is required. Your developers deploy the website files to the cloud and your website is available on the Internet.

Let's imagine another scenario. Your company needs a SQL database to support data analysts for a special project. You don't have infrastructure to accommodate the request. You can quickly provision a SQL Server in the cloud that meets the need of the project. The data analysts can connect to the server. The SQL Server database is provided as a service. Therefore, you don't worry about updates, security patches, or optimizing physical storage for reads and writes.

Some other common scenarios include:

Development framework: PaaS provides a framework that developers can build upon to develop or customize cloud-based applications. Similar to the way you create an Excel macro, PaaS lets developers create applications using built-in software components. Cloud features such as scalability, high-availability, and multi-tenant capability are included, reducing the amount of coding that developers must do.

Analytics or business intelligence: Analysis tools provided as a service allows you to analyze and mine data. Organizations can find insights and patterns to predict outcomes to improve forecasting, product design decisions, investment returns, and other business decisions.

Advantages

By delivering infrastructure as a service, PaaS has similar advantages as IaaS. But its additional features including middleware, development tools, and other business tools provide additional advantages:

Reduced development time: PaaS development tools can reduce development time for new applications. Developers can use pre-coded application components built into the platform, such as workflow, directory services, security features, and search. Platform as a service components can give your development team new capabilities without you needing to add staff having the required skills.

Develop for multiple platforms: Some service providers give you development options for multiple platforms, such as desktop, mobile devices, and browsers making cross-platform apps quicker and easier to develop.

Use sophisticated tools affordably: A pay-as-you-go model makes it possible for individuals or organizations to use sophisticated development software and business intelligence and analytics tools that they could not afford to purchase outright.

Support geographically distributed development teams: Because the development environment is accessed over the Internet, development teams can work together on projects even when team members are at remote locations.

Efficiently manage the application lifecycle: PaaS provides all of the capabilities that you need to support the complete web application lifecycle: building, testing, deploying, managing, and updating within the same integrated environment.

Software as a service

Software as a service (SaaS) allows users to connect to and use cloud-based apps over the Internet. Common examples are email, calendaring, and office tools such as Microsoft 365. SaaS provides a complete software solution that you purchase on a pay-as-you-go basis from a cloud service provider. You can rent the use of an application for your organization. Your users connect to the service over the Internet, usually with a web browser. All of the underlying infrastructure, middleware, app software, and app data are located in the service provider's data center. The service provider manages the hardware and software, and with the appropriate service agreement, will ensure the availability and the security of the app and your data as well. SaaS allows your organization to get quickly up and running with an app at minimal upfront cost.

If you've used a web-based email service such as Outlook, Hotmail, or Yahoo! Mail, then you've already used a form of SaaS. With these services, you log into your account over the Internet, often from a web browser. The email software is located on the service provider's network, and your messages are stored there too. You can access your email and stored messages from a web browser on any computer or Internet-connected device.

Common scenarios

Let's imagine your healthcare company requires a customer relationship management (CRM) solution for its sales team. The team is global. You can use a SaaS CRM provider to quickly implement a solution to your organization's sales team.

For organizational use, you can rent productivity apps, such as email, collaboration, and calendaring; and sophisticated business applications such as customer relationship management (CRM), enterprise resource planning (ERP), and document management. You pay for the use of these apps by subscription or according to the level of use.

Advantages

Gain access to sophisticated applications: To provide SaaS apps to users, you don't need to purchase, install, update, or maintain any hardware, middleware, or software. SaaS makes even sophisticated enterprise applications, such as ERP and CRM, affordable for organizations that lack the resources to buy, deploy, and manage the required infrastructure and software themselves. Pay only for what you use. You also save money because the SaaS service automatically scales up and down according to the level of usage.

Use free client software: Users can run most SaaS apps directly from their web browser without needing to download and install any software. You don't need to purchase or deploy client software for your users.

Access app data from anywhere: With data stored in the cloud, users can access their information from any Internet-connected computer or mobile device. And as app data is stored in the cloud, no data is lost if a user's computer or device fails.

Summary

1. A SQL server database is needed for a short-term project. The IT department does not have available hardware that meets the performance requirements or resources to deploy it. The project starts next week. Once the project is over, the database is no longer needed.

Public cloud

The need for a quick deployment and the short lifecycle of the SQL database make public cloud the best option.

PaaS

Because of the immediate need and short lifetime, PaaS is the best option. For example, you could use Azure SQL Database for the solution. IaaS is not as good of a choice because creating a virtual machine and installing SQL Server would require more time and effort than using a pre-built PaaS offering.

2. You create several virtual machines in the cloud. The VMs are networked together using a virtual network. The VMs have access to x-ray image files in cloud storage. One of the VMs is a web server that host a website exposed to the Internet for customers to access their records. There is a VPN that connects the solution to your on-premises datacenter for customer information to display with the image files.

Hybrid cloud

This scenario uses both public and private cloud with a connection between the two.

Therefore, it is a hybrid cloud deployment.

IaaS

All the resources used in this scenario can be implemented using IaaS. Only part of the solution could be implemented using PaaS, so it would not be a good choice. Since this is a custom solution, we can assume there isn't a SaaS offering that delivers this functionality.

3. You have two datacenters in your organization. One datacenter has a database that can only reside in one of the datacenters because of a regulatory requirement. You want to access the database from the other datacenter. You create a VPN using a cloud provider to connect the two datacenters.

Private cloud

The VPN is a public cloud resource; however, because the VPN is a private connection between two private datacenters, this is a private cloud deployment.

4. You work at a start-up company with a small number of employees. They need email, calendar scheduling, and somewhere to store documents. The team is technical but do not have the time or hardware to implement and manage a solution.

SaaS

A SaaS solution such as Microsoft 365 would be a good solution for the collaboration requirements.

Control Azure services with the CLI

The Azure CLI is a command-line program to connect to Azure and execute administrative commands on Azure resources. It runs on Linux, macOS, and Windows and allows administrators and developers to execute their commands through a terminal or command-line prompt (or script!) instead of a web browser. For example, to restart a virtual machine (VM), you would use a command like the following:

The Azure CLI provides cross-platform command-line tools for managing Azure resources, and can be installed locally on Linux, Mac, or Windows computers. The Azure CLI can also be used from a browser through the Azure Cloud Shell. In both cases, it can be used interactively or scripted. For interactive use, you first launch a shell such as cmd.exe on Windows or Bash on Linux or macOS and then issue the command at the shell prompt. To automate repetitive tasks, you assemble the CLI commands into a shell script using the script syntax of your chosen shell and then execute the script.

The Azure CLI lets you control nearly every aspect of every Azure resource. You can work with resource groups, storage, virtual machines, Azure Active Directory (Azure AD), containers, machine learning, and so on.

Commands in the CLI are structured in groups and subgroups. Each group represents a service provided by Azure, and the subgroups divide commands for these services into logical groupings. For example, the storage group contains subgroups including **account**, **blob**, and **queue**.

So, how do you find the particular commands you need? One way is to use `az find`, the AI robot that uses the Azure documentation to tell you more about commands, the CLI and more.

```
az find blob  
az find "az vm"  
az storage blob --help
```

When creating a new Azure resource, there are typically three steps: connect to your Azure subscription, create the resource, and verify that creation was successful. The following illustration shows a high-level overview of the process.

```
az login  
az group create --name <name> --location <location>
```

For many Azure resources, the Azure CLI provides a **list** subcommand to view resource details. For example, the Azure CLI **group list** command lists your Azure resource groups. This is useful here to verify whether creation of the resource group was successful:

```
az group list  
az group list --output table
```

```
az group list --query "[?name == '$RESOURCE_GROUP']"
```

The query is formatted using **JMESPath**, which is a standard query language for JSON requests. You can learn more about this powerful filter language at <http://jmespath.org/>. We also cover queries in more depth in the **Manage VMs with the Azure CLI** module.

Summary

1. What do you need to install on your machine to let you execute Azure CLI commands locally?

Only the Azure CLI

You only need to install the Azure CLI. You will use a shell to issue the CLI commands, but every platform has at least one built-in shell.

2. True or false: The Azure CLI can be installed on Linux, macOS, and Windows, and the CLI commands you use are the same in all platforms.

True

The CLI is cross-platform and can be installed on Linux, macOS, and Windows. After installation, the CLI commands that you run are the same everywhere. This means you can learn the commands once and use them with any local installation or in the Azure Cloud Shell.

3. Which parameter can you add to most CLI commands to get concise, formatted output?

table

The table parameter formats the output as a table. This can make things much more readable for commands that produce a large amount of output.

Automate Azure tasks using scripts with PowerShell

A PowerShell command is called a **cmdlet** (pronounced "command-let"). A cmdlet is a command that manipulates a single feature. The term **cmdlet** is intended to imply "small command". By convention, cmdlet authors are encouraged to keep cmdlets simple and single-purpose.

The base PowerShell product ships with cmdlets that work with features such as sessions and background jobs. You add modules to your PowerShell installation to get cmdlets that manipulate other features. For example, there are third-party modules to work with ftp, administer your operating system, access the file system, and so on.

Cmdlets follow a verb-noun naming convention; for example, **Get-Process**, **Format-Table**, and **Start-Service**. There is also a convention for verb choice: "get" to retrieve data, "set" to insert or update data, "format" to format data, "out" to direct output to a destination, and so on.

Cmdlet authors are encouraged to include a help file for each cmdlet. The Get-Help cmdlet displays the help file for any cmdlet. For example, we could get help on the Get-ChildItem cmdlet with the following statement:

Cmdlets are shipped in modules. A PowerShell Module is a DLL that includes the code to process each available cmdlet. You load cmdlets into PowerShell by loading the module they are contained in. You can get a list of loaded modules using the Get-Module command.

Az is the formal name for the Azure PowerShell module containing cmdlets to work with Azure features. It contains hundreds of cmdlets that let you control nearly every aspect of every Azure resource. You can work with resource groups, storage, virtual machines, Azure Active Directory, containers, machine learning, and so on. This module is an open source component available on GitHub.

Azure PowerShell provides the New-AzVm cmdlet to create a virtual machine. The cmdlet has many parameters to let it handle the large number of VM configuration settings. Most of the parameters have reasonable default values so we only need to specify five things:

- **ResourceGroupName**: The resource group into which the new VM will be placed.
- **Name**: The name of the VM in Azure.
- **Location**: Geographic location where the VM will be provisioned.
- **Credential**: An object containing the username and password for the VM admin account. We will use the Get-Credential cmdlet. This cmdlet will prompt for a username and password and package it into a credential object.
- **Image**: The operating system image to use for the VM. This is often a Linux distribution, or Windows Server.

Summary

1. True or false: The Azure portal, the Azure CLI, and Azure PowerShell offer significantly different services, so it is unlikely that all three will support the operation you need.

False

The three tools offer almost the same set of services. Generally, this is not a factor in deciding which tool is best for your tasks.

2. Suppose you are building a video-editing application that will offer online storage for user-generated video content. You will store the videos in Azure Blobs, so you need to create an Azure storage account to contain the blobs. Once the storage account is in place, it is unlikely you would remove and recreate it because this would delete all the user videos. Which tool is likely to offer the quickest and easiest way to create the storage account?

Azure portal

The portal is a good choice for one-off operations like creating a long-lived storage account. The portal gives you a GUI containing all the storage-account properties and provides tool tips to help you select the right options for your needs.

3. What needs to be installed on your machine to let you execute Azure PowerShell cmdlets locally?

The base PowerShell product and the Az module

You need both the base PowerShell product and the Az module. The base product gives you the shell itself, a few core commands, and programming constructs like loops, variables, etc. The Az modules adds the cmdlets you need to work with Azure resources.

[Predict costs and optimize spending for Azure](#)

When you provision an Azure resource, Azure creates one or more meter instances for that resource. The meters track the resources' usage, and generate a usage record that is used to calculate your bill.

For example, a single virtual machine that you provision in Azure might have the following meters tracking its usage:

- Compute Hours
- IP Address Hours
- Data Transfer In
- Data Transfer Out
- Standard Managed Disk
- Standard Managed Disk Operations
- Standard IO-Disk
- Standard IO-Block Blob Read

The meters and pricing vary per product and often have different pricing tiers based on the size or capacity of the resource. Check the documentation for specific details on what each service area costs.

At the end of each monthly billing cycle, the usage values will be charged to your payment method and the meters are reset. The key takeaway is that resources are always charged based on *usage*.

Factors affecting costs

Resource type

Costs are resource-specific, so the usage that a meter tracks and the number of meters associated with a resource depend on the resource type.

The usage that a meter tracks correlates to a number of billable units. The rate per billable unit depends on the resource type you are using. Those units are charged to your account for each billing period.

Services

Azure usage rates and billing periods can differ between Enterprise, Web Direct, and Cloud Solution Provider (CSP) customers. Some subscription types also include usage allowances, which affect costs.

The Azure team develops and offers first-party products and services, while products and services from third-party vendors are available in the Azure Marketplace . Different billing structures apply to each of these categories.

Location

Azure has datacenters all over the world. Usage costs vary between locations that offer particular Azure products, services, and resources based on popularity, demand, and local infrastructure costs.

For example, you might want to build your Azure solution by provisioning resources in locations that offer the lowest prices. This approach, though, would require transferring data between locations if any dependent resources and their users are located in different parts of the world. If there are meters tracking the volume of data moving between the resources you provision, any potential savings you make from choosing the cheapest location could be offset by the additional cost of transferring data between those resources.

Azure billing zones

Bandwidth refers to data moving in and out of Azure datacenters. Most of the time inbound data transfers (data going into Azure datacenters) are free. For outbound data transfers (data going out of Azure datacenters), the data transfer pricing is based on **Billing Zones**.

A Zone is a geographical grouping of Azure Regions for billing purposes. The following zones exist and include the listed countries (regions).

Zone	Areas
Zone 1	United States, US Government, Europe, Canada, UK, France, Switzerland
Zone 2	East Asia, Southeast Asia, Japan, Australia, India, Korea
Zone 3	Brazil, South Africa, UAE
DE Zone 1	Germany

In most zones, the first outbound 5 gigabytes (GB) per month are free. After that amount, you are billed a fixed price per GB.

Azure pricing calculator

The Azure pricing calculator is a free web-based tool that allows you to input Azure services and modify properties and options of the services. It outputs the costs per service and total cost for the full estimate.

The options that you can configure in the pricing calculator vary between products, but basic configuration options include:

Option	Description
Region	Lists the regions from which you can provision a product. Southeast Asia, central Canada, the western United States, and northern Europe are among the possible regions available for some resources.
Tier	Sets the type of tier you wish to allocate to a selected resource, such as Free Tier, Basic Tier, etc.
Billing Options	Highlights the billing options available to different types of customers and subscriptions for a chosen product.
Support Options	Allows you to pick from included or paid support pricing options for a selected product.
Programs and Offers	Allows you to choose from available price offerings according to your customer or subscription type.
Azure Dev/Test Pricing	Lists the available development and test prices for a product. Dev/Test pricing applies only when you run resources within an Azure subscription that is based on a Dev/Test offer.

Azure Advisor

Azure Advisor is a free service built into Azure that provides recommendations on high availability, security, performance, operational excellence, and cost. Advisor analyzes your deployed services and looks for ways to improve your environment across each of these areas. We'll focus on the cost recommendations, but you'll want to take some time to review the other recommendations as well.

Advisor makes cost recommendations in the following areas:

1. **Reduce costs by eliminating unprovisioned Azure ExpressRoute circuits.** This recommendation identifies ExpressRoute circuits that have been in the provider status of Not Provisioned for more than one month. Advisor recommends deleting the circuit if you aren't planning to provision the circuit with your connectivity provider.
2. **Buy reserved instances to save money over pay-as-you-go.** Advisor will review your virtual machine usage over the last 30 days and determine if you could save money in the future by purchasing reserved instances. Advisor will show you the

regions and sizes where you potentially have the most savings and will show you the estimated savings you might achieve from purchasing reserved instances.

3. **Right-size or shutdown underutilized virtual machines.** This analysis monitors your virtual machine usage for 14 days and then identifies underutilized virtual machines. Virtual machines whose average CPU utilization is 5 percent or less and network usage is 7 MB or less for four or more days are considered underutilized virtual machines. The average CPU utilization threshold is adjustable up to 20 percent. By identifying these virtual machines, you can decide to resize them to a smaller instance type, reducing your costs.

Azure Cost Management

Azure Cost Management is another free, built-in Azure tool that can be used to gain greater insights into where your cloud money is going. You can see historical breakdowns of what services you are spending your money on and how it is tracking against budgets that you have set. You can set budgets, schedule reports, and analyze your cost areas.

If you are starting to migrate to the cloud, a useful tool you can use to predict your cost savings is the **Total Cost of Ownership (TCO) calculator**. To use the TCO calculator, you need to complete four steps.

Azure Hybrid Benefit for Windows Server

Many customers have invested in Windows Server licenses and would like to repurpose this investment on Azure. The Azure Hybrid Benefit gives customers the right to use these licenses for virtual machines on Azure.

To be eligible for this benefit, your Windows licenses must be covered by Software Assurance. The following guidelines will also apply:

- Each two-processor license or each set of 16-core licenses is entitled to two instances of up to eight cores or one instance of up to 16 cores.
- Standard Edition licenses can only be used either on-premises or in Azure, but not both. That means you can't use the same license for an Azure VM and a local computer.
- Datacenter Edition benefits allow for simultaneous usage both on-premises and in Azure so that the license will cover two running Windows machines.

Use Dev/Test subscription offers

The Enterprise Dev/Test and Pay-As-You-Go (PAYG) Dev/Test offers are a benefit you can take advantage of to save costs on your non-production environments. This benefit gives you several discounts, most notably for Windows workloads, eliminating license charges and only billing you at the Linux rate for virtual machines. SQL Server and other Microsoft software covered under a Visual Studio subscription (formerly known as MSDN) are also included.

There are a few requirements for this benefit. First, it's only for non-production workloads. This benefit also requires any users of these environments (excluding testers) must be

covered under a Visual Studio subscription. In short, for non-production workloads, this benefit allows you to save money on your Windows, SQL Server, and other Microsoft virtual machine workloads.

Summary

1. Which tab of the Azure pricing calculator will you use to put together your estimate?

Products

This tab has all the Azure services listed and is where you'll add or remove services to get your estimate.

2. True or false: You can share your estimate through an Excel spreadsheet or through a URL.

True

Clicking Export at the bottom of the estimate will export an Excel spreadsheet that you can share, or you can click Share to get a URL link that you can share with your team.

3. Azure Advisor provides recommendations for _____.

High availability, security, performance, operational excellence, and cost

Azure Advisor provides recommendations on high availability, security, performance, operational excellence, and cost.

4. Azure Cost Management allows you to _____.

See historical breakdowns of what services you are spending your money on.

Cost Management analyzes where you are historically spending your money and can track it against budgets you have set.

5. Which one of these approaches is not a cost-saving solution?

Load balance your virtual machines for incoming messages.

Load balancing is used for performance optimization not cost savings.

6. True or false: PaaS is typically less expensive than IaaS.

True

IaaS requires Azure to dedicate resources while PaaS give Azure more flexibility in how services are delivered. This approach means Azure can fill and operate hardware efficiently and therefore offer PaaS services at a savings over IaaS.

7. True or false: If you already have unused Windows Server licenses, you have to pay for them again on Azure.

False

Under certain circumstances, you can utilize the hybrid benefit for Windows Server and pay only the Linux rate.

8. True or false: Azure has money-saving options for test and development servers.

True

The Azure Enterprise Dev/Test and Azure Pay-As-You-Go Dev/Test benefits give you several discounts, most notably for Windows workloads, eliminating license charges and billing you only at the Linux rate for virtual machines. This benefit also applies to SQL Server and any other Microsoft software that is covered under a Visual Studio subscription.

9. Which one of the following systems is used to determine Azure costs for each billing period?

Usage meters

Correct. Azure is billed according to your consumption based on monthly usage meters.

10. Which of the following factors affects costs?

Location

The location you place your resources will vary the price for the resource.

11. Complete the following sentence. As an Azure customer, Azure Reservations offer discounted prices if you _____

Make upfront commitments on compute capacity

Azure Reservations offer discounted prices on certain Azure products and resources. To get a discount, you reserve products and resources through an upfront commitment. You can then either prepay or pay monthly for one or three years of usage of certain Azure resources.

Control and organize Azure resources with Azure Resource Manager

What are resource groups?

Resource groups are a fundamental element of the Azure platform. A resource group is a logical container for resources deployed on Azure. These resources are anything you create in an Azure subscription like virtual machines, Application Gateways, and CosmosDB instances. All resources must be in a resource group and a resource can only be a member of a single resource group. Many resources can be moved between resource groups with some services having specific limitations or requirements to move. Resource groups can't be nested. Before any resource can be provisioned, you need a resource group for it to be placed in.

Resource groups exist to help manage and organize your Azure resources. By placing resources of similar usage, type, or location, you can provide some order and organization to resources you create in Azure. Logical grouping is the aspect that you're most interested in here, since there's a lot of disorder among our resources.



If you delete a resource group, all resources contained within are also deleted. Organizing resources by life cycle can be useful in non-production environments, where you might try an experiment, but then dispose of it when done. Resource groups make it easy to remove a set of resources at once.

Resource groups are also a scope for applying role-based access control (RBAC) permissions. By applying RBAC permissions to a resource group, you can ease administration and limit access to allow only what is needed.

Consistent naming convention

You can start with using an understandable naming convention. You named our resource group **msftlearn-core-infrastructure-rg**. You've given some indication of what it's used for (**msftlearn**), the types of resources contained within (**core-infrastructure**), and the type of resource it is itself (**rg**). This descriptive name gives us a better idea of what it is. If you had named it **my-resource-group** or **rg1**, you have no idea on a glance of what the usage may be. In this case, you can deduce that there are probably core pieces of infrastructure contained within. If you created additional virtual networks, storage accounts, or other resources the company may consider core infrastructure, you could place them here as well,

to improve the organization of our resources. Naming conventions can vary widely between and even within companies, but some planning can help.

Organizing principles

Resource groups can be organized in a number of ways, let's take a look at a few examples. You might put all resources that are *core infrastructure* into this resource group. But you could also organize them strictly by resource type. For example, put all virtual networks in one resource group, all virtual machines in another resource group, and all Azure Cosmos DB instances in yet another resource group.



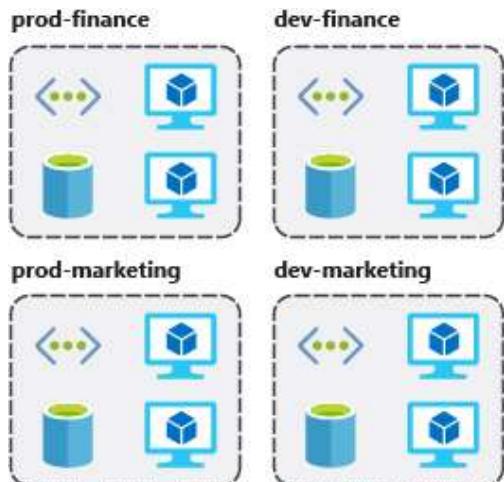
You could organize them by environment (prod, qa, dev). In this case, all production resources are in one resource group, all test resources are in another resource group, and so on.



You could organize them by department (marketing, finance, human resources). Marketing resources go in one resource group, finance in another resource group, and HR in a third resource group.



You could even use a combination of these strategies and organize by environment and department. Put production finance resources in one resource group, dev finance resources in another, and the same for the marketing resources.



There are a few factors that can play into the strategy you use to organize resources: authorization, resource life cycle, and billing.

Organizing for authorization

Since resource groups are a scope of RBAC, you can organize resources by who needs to administer them. If your database administration team is responsible for managing all of your Azure SQL Database instances, putting them in the same resource group would simplify administration. You could give them the proper permissions at the resource group level to administer the databases within the resource group. Similarly, the database administration team could be denied access to the resource group with virtual networks, so they don't inadvertently make changes to resources outside the scope of their responsibility.

Organizing for life cycle

We mentioned earlier that resource groups serve as the life cycle for the resources within it. If you delete a resource group, you delete all the resources in it. Use this to your advantage, especially in areas where resources are more disposable, like non-production environments. If you deploy 10 servers for a project that you know will only last a couple of months, you might put them all in a single resource group. One resource group is easier to clean up than 10 or more resource groups.

Organizing for billing

Lastly, placing resources in the same resource group is a way to group them for usage in billing reports. If you're trying to understand how your costs are distributed in your Azure environment, grouping them by resource group is one way to filter and sort the data to better understand where costs are allocated.

What are tags?

Tags are name/value pairs of text data that you can apply to resources and resource groups. Tags allow you to associate custom details about your resource, in addition to the standard Azure properties a resource has the following properties:

- department (like finance, marketing, and more)

- environment (prod, test, dev)
- cost center
- life cycle and automation (like shutdown and startup of virtual machines)

A resource can have up to 50 tags. The name is limited to 512 characters for all types of resources except storage accounts, which have a limit of 128 characters. The tag value is limited to 256 characters for all types of resources. Tags aren't inherited from parent resources. Not all resource types support tags, and tags can't be applied to classic resources.

Use tags for organization

You can use tags to group your billing data. For example, if you're running multiple VMs for different organizations, use the tags to group usage by cost center. You can also use tags to categorize costs by runtime environment, such as the billing usage for VMs running in the production environment. When exporting billing data or accessing it through billing APIs, tags are included in that data and can be used to further slice your data from a cost perspective.

You can retrieve all the resources in your subscription with a specific tag name or value. Tags enable you to retrieve related resources from different resource groups. This approach is helpful when you need to organize resources for billing or management.

Tagging resources can also help in monitoring to track down impacted resources. Monitoring systems could include tag data with alerts, giving you the ability to know exactly who is impacted. In our example above, you applied the **Department** tag with a value of **Finance** to the **msftlearn-vnet1** resource. If an alarm was thrown on **msftlearn-vnet1** and the alarm included the tag, you'd know that the finance department may be impacted by the condition that triggered the alarm. This contextual information can be valuable if an issue occurs.

It's also common for tags to be used in automation. If you want to automate the shutdown and startup of virtual machines in development environments during off-hours to save costs, you can use tags to assist in this automation. Add a **shutdown:6PM** and **startup:7AM** tag to the virtual machines, then create an automation job that looks for these tags, and shuts them down or starts them up based on the tag value. There are several solutions in the Azure Automation Runbooks Gallery that use tags in a similar manner to accomplish this result.

What is Azure Policy?

Azure Policy is a service you can use to create, assign, and manage policies. These policies apply and enforce rules that your resources need to follow. These policies can enforce these rules when resources are created, and can be evaluated against existing resources to give visibility into compliance.

Policies can enforce things such as only allowing specific types of resources to be created, or only allowing resources in specific Azure regions. You can enforce naming conventions

across your Azure environment. You can also enforce that specific tags are applied to resources. You'll take a look at how policies work.

Secure resources with RBAC

RBAC provides fine-grained access management for Azure resources, enabling you to grant users the specific rights they need to perform their jobs. RBAC is considered a core service and is included with all subscription levels at no cost.

Using RBAC, you can:

- Allow one user to manage VMs in a subscription, and another user to manage virtual networks.
- Allow a database administrator (DBA) group to manage SQL databases in a subscription.
- Allow a user to manage all resources in a resource group, such as VMs, websites, and virtual subnets.
- Allow an application to access all resources in a resource group.

Best Practises:

- Segregate duties within your team and grant only the amount of access to users that they need to perform their jobs. Instead of giving everybody unrestricted permissions in your Azure subscription or resources, allow only specific actions at a particular scope.
- When planning your access control strategy, grant users the lowest privilege level that they need to do their work.
- Use Resource Locks to ensure critical resources aren't modified or deleted (as you'll see in the next unit).

What are resource locks?

Resource locks are a setting that can be applied to any resource to block modification or deletion. Resource locks can set to either **Delete** or **Read-only**. Delete will allow all operations against the resource but block the ability to delete it. Read-only will only allow read activities to be performed against it, blocking any modification or deletion of the resource. Resource locks can be applied to subscriptions, resource groups, and to individual resources, and are inherited when applied at higher levels.

Use resource locks to protect those key pieces of Azure that could have a large impact if they were removed or modified. Some examples are ExpressRoute circuits, and virtual networks, critical databases, and domain controllers. Evaluate your resources, and apply locks where you'd like to have an extra layer of protection from accidental actions.

Summary

1. Tags can be applied to any type of resource on Azure

False

2. Tags applied at a resource group level are propagated to resources within the resource group.

False

3. Which of the following features does not apply to resource groups?

Resource groups can be nested.

4. Which of the following approaches might be a good usage of tags?

Using tags to associate a cost center with resources for internal chargeback

Using tags in conjunction with Azure Automation to schedule maintenance windows

Using tags to store environment and department association

All of the above are good ways to use tags

5. Which of the following approaches would be the most efficient way to ensure a naming convention was followed across your subscription?

Create a policy with your naming requirements and assign it to the scope of your subscription

6. Which of the following items would be good use of a resource lock?

An ExpressRoute circuit with connectivity back to your on-premises network

[Move Azure resources to another resource group](#)

Organize Azure resources into resource groups

You use resource groups to logically group resources together. Each resource must be part of a single resource group. A resource group can hold many different types of resources in different regions.

Generally, resource groups hold resources that are related. For example, you can have a resource group for testing and development, and a separate one for production. A production resource group might hold resources that support a live website. These resources can include several virtual machines, storage accounts, virtual networks, and Azure SQL Database instances.

Make sure that all the resources in your resource group have a single lifecycle. You want to update, delete, or deploy these resources together. If you believe a resource doesn't fit into the lifecycle of other resources, move it to another resource group.

You can control which resources are available to different types of users. In this way, resource groups help you protect your resources and control costs.

You might need to change who's billed for resources in resource groups. You can move resources from one resource group to another resource group in a different subscription. In this case, you'd need to put any dependent resources into one resource group. Then move the resources from that resource group into a new resource group in the other subscription.

Prepare to test your move

Before attempting to move a resource, you can test whether it will be successful by calling the validate move operation from the Azure REST API. This test is especially useful if you're trying to move resources by using, for example, Azure PowerShell or the Azure CLI. You can use these tools to script moves, with minimal human interaction. Testing a move doesn't affect your resources. The operation only tests whether your move operation would succeed, based on the options you provide.

If you're trying to move resources through the Azure portal, you don't need to validate the move before attempting it. The Azure portal does an automatic validation before allowing you to move resources.

Understand resource moves

When you start a move operation, the resource group holding your resources and the new destination resource group are locked. You can't do write or delete operations on the resource groups until the move operation ends. Your resources aren't affected, but you can't add, delete, or update any resources in these resource groups.

Your moved resources don't change location. For example, if you have a storage account in the East US region, and you move it to another resource group, it keeps its East US region location.

Summary

1. Which one of these resource types can't be moved between resource groups?

Azure Active Directory Domain Services

According to the move operation support list, Azure AD DS can't be moved.

2. Which one of these resource types can be moved across resource groups?

Azure virtual networks

You can move Azure virtual networks, according to the move operation support list.

3. When can you move a virtual machine?

You can move virtual machines, but you have to first make sure that you can move all of its dependent resources along with it.

Your virtual machines have dependent resources without which they can't function properly.

Make sure you move them together.

4. In which of the following situations would a validation test happen automatically?

When you use the Azure portal to move resources to a new resource group.

The Azure portal always does a move validation test before it attempts the move operation itself.

5. In which of the following situations do you need to obtain an access token before you can validate a move?

When you use custom code to call the validateMoveResources REST API.

To call any Azure REST API method from custom code, you must obtain and pass a valid access token.