

**Cloud Solutions: The Business Rationale**

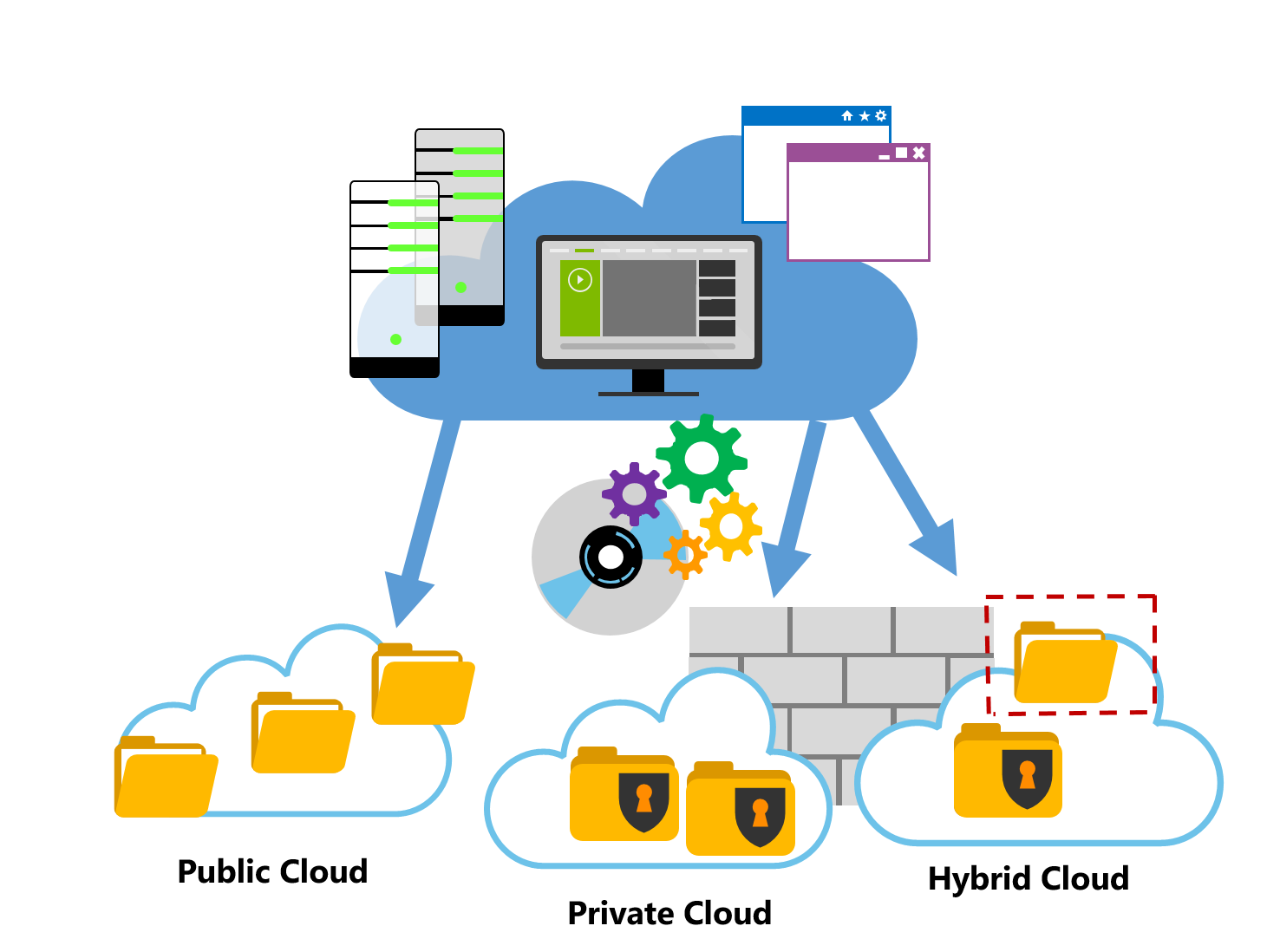
Business and market opportunities can emerge and change very quickly. A viable business opportunity today may no longer be relevant tomorrow. Modern IT operations needs to be agile and able to respond quickly and easily to business changes, enabling a business to remain competitive. Cloud computing allows them to do just that.

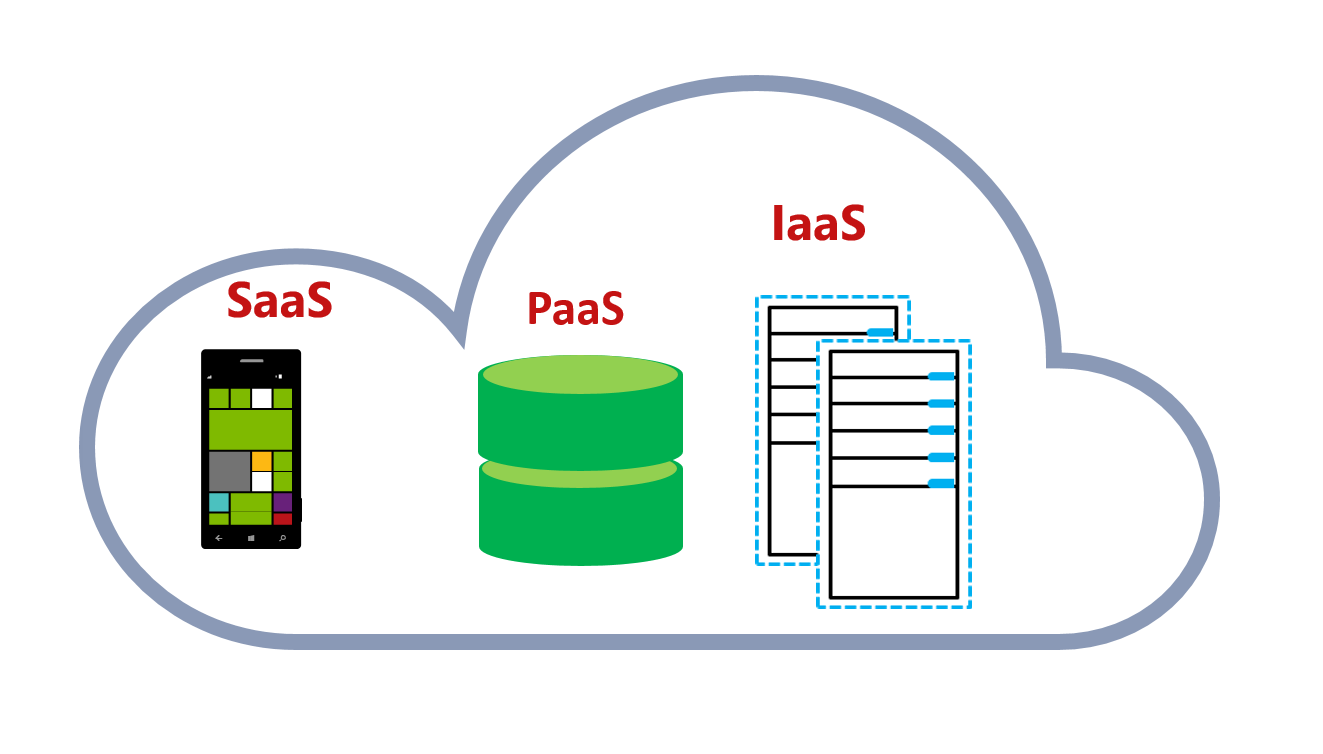
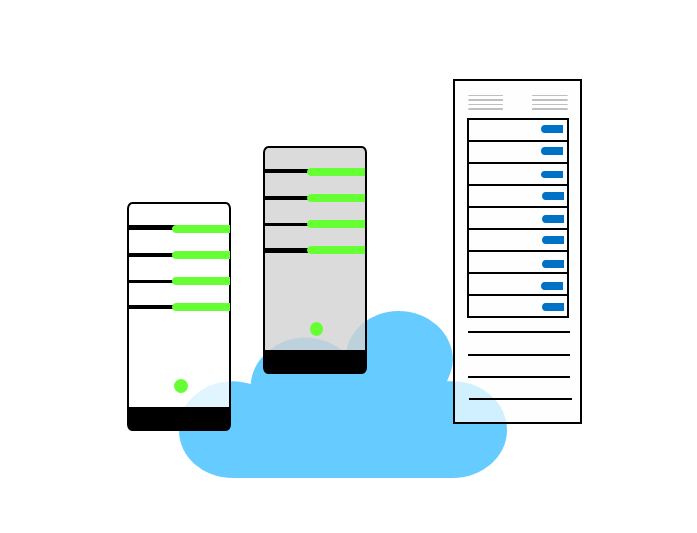
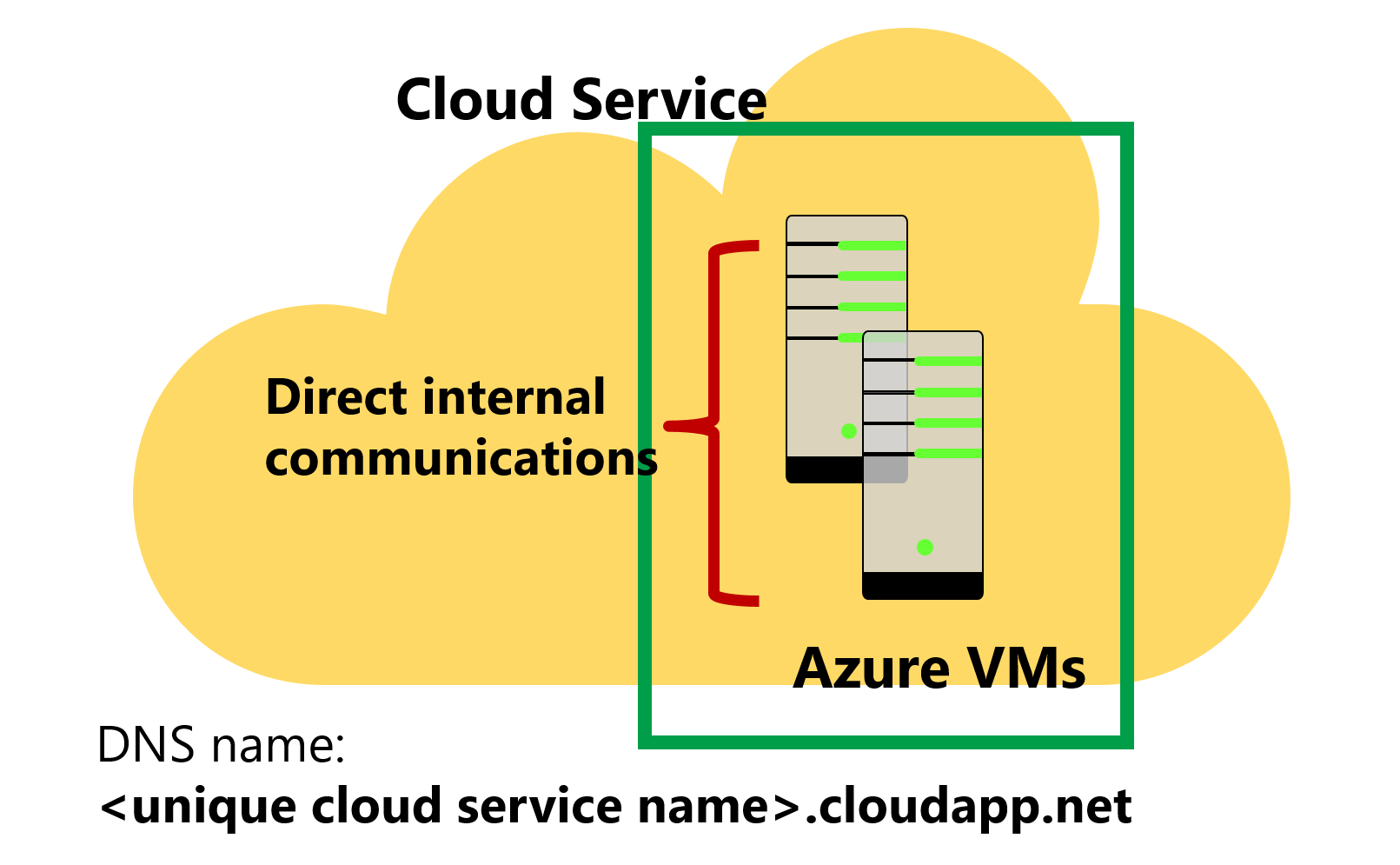
Economics are also a major driver for the cloud. In many instances, it is simply much cheaper to use a cloud service rather than building, managing, and maintaining the services yourself. This leads to a more efficient use of resources and clear and concise reporting, which helps reduce overall costs.

**Cloud Solutions: Essential Characteristics**

Regardless of the specific technologies that organizations use to implement cloud computing solutions, the National Institute of Standards and Technology (NIST), the official US based standards and technology definitions body, has identified five essential characteristics that are part of a cloud computing solution.

* On-demand self-service. The ability to allocate required resources by the end user themselves as needed without involvement from the cloud service provider.
* Broad network access. Being accessible via standard network access mechanisms, without the need for any specialized infrastructure.
* Resource pooling. The pooling of the various resources to be allocated from and returned to as needed.
* Rapid elasticity. The ability to scale up and scale down as required, whether automatically or manually, without lead times being required.
* Measured service. The ability to measure exactly what resources are being used, to monitor and control those services and to be able to present that data to the service provider or end user.
* **Public Cloud**
* You can configure and access these different service types through three different implementation models. These are Public Cloud, Private Cloud, or a mixture of both using a Hybrid Cloud model.



* **Public Cloud**
* A public cloud is owned by the cloud services provider (also known as a hosting provider). The cloud services provider provides cloud resources for an organization, which the end user connects to via a secure network connection, typically over the internet. The cloud service provider may share its resources with multiple organizations, or with the public. The distinguishing feature of a public cloud is that the resources that the organization uses, such as storage, processing power, various web-based applications, and other components, do not belong to the organization that is utilizing the resources, but rather to the cloud services provider. There are many cloud service providers and business can use multiple companies of varying scale.
* **Private Cloud**
* A private cloud operates only within one organization on a private network and is highly secure. It provides cloud functionality to external customers or specific internal departments, such as Accounting or Human Resources. By creating a private cloud, you can provide a pool of resources for the infrastructure and the applications to be shared to each end user as a tenant with the respective resources that they need.
* **Hybrid Cloud**
* A hybrid cloud is a combination of private and public deployment models. In a hybrid cloud, specific resources are run or used in a public cloud, and others are run or used in a private cloud. A hybrid cloud offers benefits from both private and public cloud models and may be preferable when you want to control and manage some of your workloads locally but also still want to leverage some of the benefits of cost, efficiency, and scale available with a public cloud model.
* There are different models for cloud services depending on how a service is actually being used or provided. In terms of service types, these effectively can be broken into three main categories. As services evolve, these categories can also contain subsets, or higher level services, such as Database as a Service (DBaaS).
* 
* **Software as a Service (SaaS)**
* SaaS offerings consist of fully-formed software applications that are delivered as cloud-based services. Users can subscribe to the service and use the application, normally through a web browser or by installing a client-side app. Examples of Microsoft SaaS services include Microsoft Office 365™, Skype®, and Microsoft Dynamics CRM Online. The primary advantage of SaaS services is that they enable users to easily access applications without the need to install and maintain them. Typically, users do not have to worry about issues such as updating applications and maintaining compliance because the service provider handles them.
* **Infrastructure as a Service (IaaS)**
* IaaS offerings provide virtualized server, network and storage infrastructure components that can be easily provisioned and decommissioned as required. Typically, IaaS facilities are managed in a similar way to on-premises infrastructure, and provide an easy migration path for moving existing applications to the cloud.
* A key point to note is that an infrastructure service might be a single IT resource—such as a virtual server that has a default installation of Windows Server and Microsoft SQL Server or it might be a completely preconfigured infrastructure environment for a specific application or business process. For example, a retail organization might empower departments to provision their own database servers to use as data stores for custom applications. Alternatively, the organization might define a set of virtual machine and network templates that can be provisioned as a single unit to implement a complete, preconfigured infrastructure solution for a branch or store, including all the required applications and settings.
* **Platform as a Service (PaaS)**
* PaaS offerings consist of cloud-based services that provide resources on which developers can build their own solutions. Typically, PaaS encapsulates fundamental operating system (OS) capabilities, including storage and compute, in addition to functional services for custom applications. Usually, PaaS offerings provide application programming interfaces (APIs), in addition to configuration and management user interfaces. Azure provides PaaS services that simplify the creation of solutions such as web and mobile applications. PaaS enables developers and organizations to create highly scalable custom applications without having to provision and maintain hardware and operating system resources. Examples of PaaS include Azure Websites and Azure Cloud Services, which can run a web application that your developer team creates.
* For more information, you can see:
* Cloud Computing: What is Infrastructure as a Service - <https://technet.microsoft.com/en-us/library/hh509051.aspx>
* There are several options for running applications in the cloud. One such way, is virtual machines. An Azure virtual machine is a server that runs in the Azure cloud. It makes use of a range of Azure services, such as storage, virtual networks, cloud services, and directories.
* Azure virtual machines provide you with all the flexibility of virtualization, but without requiring the capital expense of buying and maintaining your own host datacenter. With an Azure virtual machine, all the hardware and infrastructure management tasks are performed by Microsoft.
* 
* Virtual machines are part of the Azure IaaS offering. They are often used together with virtual networks (VNets). Azure virtual machines run within an IaaS cloud service, which provides a public endpoint IP address. Although like Platform as a Service (PaaS) cloud services, IaaS cloud services have different features and capabilities, and are configured and managed separately. Azure virtual machines consume Azure storage, and require a storage account to store virtual hard disk (VHD) files.
* For more information, you can see:
* Virtual Machines - <https://aka.ms/edx-azure215x-az>
* An IaaS cloud services is the network container for hosting Azure virtual machines. Any virtual machine in a cloud service can communicate directly with all other virtual machines in that cloud service, and by using Azure communications; all communications within a cloud service are internal to that cloud service only, and virtual machines are not using the Internet to communicate with each other.
* 
* A built-in Azure DNS server provides name resolution for all virtual machines within the same cloud service. Cloud services have an assigned publicly-reachable DNS name, in the form . A cloud service has at least one Virtual Internet Protocol (VIP) address assigned, and the cloud service VIP enables inbound connections to Azure virtual machines from the Internet.
* ✔️ You cannot create an Azure virtual machine without first specifying a cloud service to use.
* For more information, you can see:
* Cloud Services - <https://aka.ms/edx-azure215x-az>
* Azure App Service is an integrated service that lets you create web and mobile apps for any platform or device, easily integrate with SaaS solutions, like Office 365 or Dynamics CRM, using new capabilities along with built in connectors for building logic workflows. In addition to Office 365 and Dynamics CRM these capabilities will work with many SaaS and on-premises applications currently in the market today.



**What is Azure?**

**Bookmark this page**

Microsoft Azure is Microsoft’s public cloud offering that enables individuals and organizations to create, deploy, and operate cloud-based applications and infrastructure services. The Microsoft Azure public cloud platform offers IaaS, PaaS, and SaaS services to organizations worldwide.

Microsoft Azure can also be used in conjunction with various Microsoft solutions, such as Microsoft System Center, and can be leveraged together to extend an organization's current datacenter into a hybrid cloud that expands capacity and provides capabilities beyond what could be delivered solely from an on-premises standpoint.

**Uses for Microsoft Azure**

Microsoft Azure offers many services and resource offerings. For example, you can use the Azure Virtual Machines compute services to build a network of virtual servers to host an application, database, or custom solution, which would be an IaaS based offering. Other services can be categorized as PaaS because you can use them without maintaining the underlying operating systems. For example, when you run a website in Azure Web Apps, or a SQL database in Azure SQL Databases it is not necessary to ensure that you are using the latest version of Internet Information Services (IIS) or SQL Server and have the latest patches and updates installed, as this is the responsibility of the Microsoft Azure platform.

An example of a SaaS service on Microsoft Azure would be Operations Management Suite, which is set up and accessed via Operational Insights services. Here, you just set up your management tools and connect into the services you wish to manage all through Microsoft Azure, so no local infrastructure is required or needed to be managed. The Microsoft Azure platform is responsible for all of that and provides direct access to the Management software. Microsoft Azure provides lots of services which fall into IaaS, PaaS or SaaS contexts and these services are constantly being added to and evolving.

✔️ Microsoft Azure was formerly known as Windows Azure® and you may see legacy tools and documentation that still use this name.

For more information, you can see:

Microsoft Azure - <https://aka.ms/edx-azure215x-az2>

First, take a look at how Microsoft

implements its data centers to support

its cloud strategy. We think about, you know what Microsoft does and what it

focuses on.. the first one is obviously reliability. If you're going to take all

your customers and all their data and all their services and load that onto

your data centers, you better make sure those data centers are fault-tolerant,

highly available and reliable. In a core

tenet of the investment in the design

of Azure is around reliability. Both internally and also giving you the

option to distribute things like VM workloads, Applications and data, across

many many different data centers

in many countries or regions to make sure that

in addition to what Microsoft is doing.

you can choose how to distribute and backup your data for the maximum

reliability, security and compliance, ensuring

that is compliant with not only local

and federal government regulations, but to give you the option to enforce

corporate policies around data

and then finally environmental sustainability... data

centers obviously require a lot of power

and making sure that, you know we

focus on green data centers

and data centers that minimize the environmental

footprint is a core tenet of how these data centers are designed. Now there's a

lot of information, you can go online on you know anywhere from YouTube to

Microsoft.com, to Azure.com and see videos and documentation on how things

like that the data center design is done. It's really quite revolutionary the way

that they've designed the pod systems,

the way the infrastructure is done and

some of these statistics are actually pretty.. pretty astounding.

So for example to give you some idea.. about 1.4 million databases on Azure and

this data by the way I can guarantee you is old, because you're watching this

at least the day after I recorded it, so this is now old data. Millions of users

in Active Directory, 1.5 trillion

messages per month in Azure Internet of

Things, 777 trillion storage transactions per day.

It's been said that the

fiber-optic in North America, alone can circle the globe

I think about six times and that there's enough cable to go to the moon

and back. A hundred percent of servers,

on electronic computers are cycled after

decommissioning.. there again that kind of

goes back to that point on environmental

sustainability. We think about you know, some of the other design points..

there's a global distribution model. Microsoft manages it and it's a modular

architecture. All this the data centers are made up of individual pods that are

brought in as a unit.

So this unit of compute power that can simply be built,

manufactured and plugged in and then

that adds a certain amount of capability

to enhance your data center. Everything runs both Windows Server which

is the same Windows Server that you run in your datacenter, I was going to say at

home but in your datacenter and wherever your datacenter happens

to be. We think about locations, I'm not going to use the slides, I'm going to

jump in and just kind of take a look, this is the Azure dashboard. So this is

going to be the most current information all the time. Every green check box

represents an Azure datacenter

location and there are more datacenter

locations planned all the time. So you can see there's a pretty large cluster

around North America, there's a couple of datacenters in Europe and there's a

pretty large segment of datacenters.

all around the Asia Pacific Rim, both in

Japan, down in Singapore and Australia as well a couple in India. So wherever the

world's computing power is needed

there's Azure datacenters that are definitely close by. We look at the

services that are available in these datacenters. The categories go across

the top, so when you think about you know what can we actually do inside all these

datacenters. It's worth noting by the way that not all services are available

in all datacenters. For example some datacenters don't support the latest

generation Virtual Machine the very very

large Virtual Machines because the

hardware that's installing the data centers may be a couple of years old. So

it supports some of the smaller generation Virtual Machines. But under

the compute category we have things like

Cloud Services, Service Fabric, Virtual

Machine Scale Sets, Containers under data

and storage, we have SQL Databases, we

have StorSimple which is a file storage mechanism for uploading files

from your local network into Azure managed

storage. Under analytics we have Machine

Learning and something called Data Lake from massive quantities of data for

Internet of things, this is things like your Internet-connected fridge there's

that are called notification hubs that facilitate

communication, so your fridge

has the ability to report when you need more milk. Under networking we have

things like DNS, you can create managed DNS servers in Azure and then you can

implement load balancers and traffic

managers. We have Media distribution. For

hybrid, we have something called Azure site recovery, that allows you to have

virtual machines and infrastructure

in your datacenter. But if your datacenter

goes offline, automatically fail to an Azure instance of your virtual

machines. So keeps it in sync. Keeps it up-to-date if you go offline, it fails

over directly into Azure and then finally for Identity as we move to the

world of bring-your-own-device to work to where, you know the device that

you have, is when you pick up in your local electronic store

it's not something that's issued and

managed by your company, it's not something

that's subject to for example group policy.

How do you manage user

identity across this world of online applications,

multiple applications, multiple companies

and the devices that you don't maintain

or manage..? Azure active directory and Azure identity allow you to do that

and we'll see some examples of this in other courses and how we set up things

like Azure Active Directory and how we

federate with external applications. For

developers we have Azure Visual Studio Team Services. In the past developers

would collaborate, but they required what was essentially connectivity in the

same LAN.. so you'd VPN in the work,

upload your code, checkout your code. With Azure, Team Services in Visual

Studio Team Service on Azure that can now be centralized. So as long as you

have an Internet connection, you can connect to and collaborate with other

developers. There's also Azure Dev Test Labs that allow you to automate the

deployment of its full environments. For example I have an application

that requires 10 servers to fully test.

Azure Dev Test Labs gives you the ability do a template and deploy a

test lab to run it and test on your application.. And then finally for

management things like Automation, so we can do

scripting, Schedulers, Log Analytics

to analyze event logs. You can have Servers, upload their event logs to

Azure. Azure log analytics will scan those event logs and bring to your

attention things that look like they should be checked out.

You know errors that maybe need some additional attention.

Now we mentioned

that not everything is available everywhere, again this is you know by all

means not something that I would expect you to read,

however it gives you a sense

that depending on the datacenter, depending on the location, the services

may vary. Now there's a website,

you can go to Azure.microsoft.com/regions that will

let you break down in incredible detail exactly what's available in each region.

so as you plan your implementation, as you figure out you know what is it you

need you can visit that website and see where it's available to determine if

it's available close to you or if maybe you've got to go somewhere else.

The good thing that you can kind of, if

you look at this and you know in any amount of

depth, in every major geographic region of the world you can get

everything. It maybe not you know East, it may be available in East US. But not

East US2 for example. But that's going to be available in North America. So

you're going to be generally close to

whatever component of Azure it is that

you want to use. So that gives you a sense of, you know what is Azure

implementation of cloud and just kind of the scale.

You know obviously there's a lot there.

We're focused in this course on, the role of IT professionals and how

we administer and manage Azure. But as you begin to think about it you can see

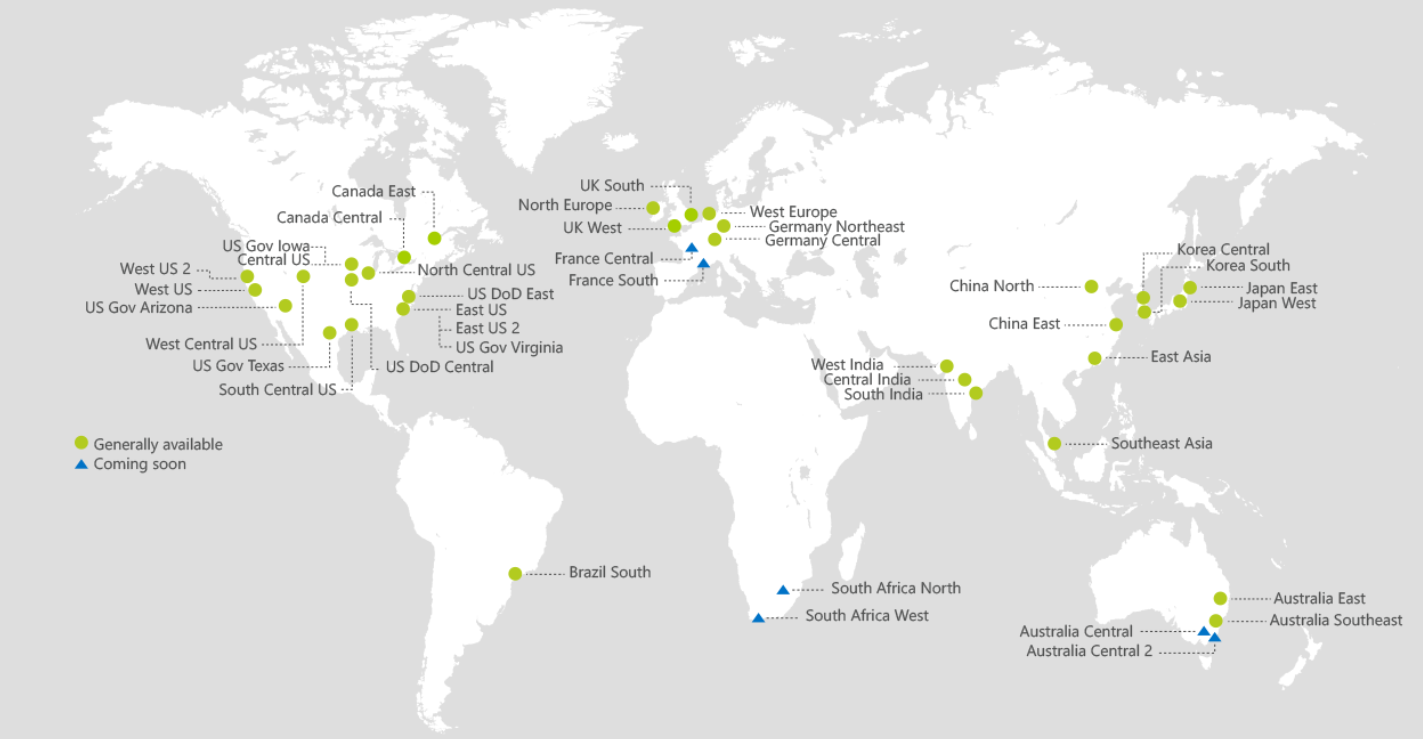
how you know, for developers

there's a component for Azure, for administrators there's component for

Azure, for business users and data scientists, there's components in Azure.

So it's a very comprehensive, very large platform.

Azure services are hosted in physical Microsoft-managed data centers throughout the world. The data centers are in multiple geographic areas, with a pair of regional data centers in each geographic region. At the time of this writing, Azure is generally available in 42 regions around the world, with plans announced for 6 additional regions.



Not all Azure services are available from every region.

For more information, you can see:

Azure Regions - <https://azure.microsoft.com/en-us/regions/>

Annotate

Microsoft Azure provides cloud services for accomplishing various tasks and functions across the IT spectrum and those services can be organized in several broad categories. There are services for different usage scenarios and a wide range of services that can be used as building blocks to create custom cloud solutions, for the application developer as well as the IT Pro.

In the following two topics, we will list some of the most commonly used categories and the services contained in them that may be of interest to you as an IT Pro. Not all categories and services have been included, and some services are listed in more than one place. For a complete list of services, see <https://azure.microsoft.com/en-us/features/azure-portal/> and click Products.



**Compute and Networking Services**

* **Virtual Machines**. Create Windows® and Linux virtual machines from pre-defined templates, or deploy your own custom server images in the cloud.
* **Virtual Machine Scale Sets**. Deploy Virtual Machine Scale Sets using Azure Resource Manager templates.
* **Virtual Networks**. Provision networks to connect your virtual machines, PaaS cloud services, and on-premises infrastructure.
* **Cloud Services**. Define multi-tier PaaS cloud services that you can deploy and manage on Microsoft Azure.
* **Load Balancer**. Quickly create highly-available and scalable applications, with support for the most common networking protocols.
* **VPN Gateway**. Connect on-premises networks to Azure through Site-to-Site VPNs using secure protocols like IPSec and IKE.
* **Azure DNS**. Use Azure DNS to host your Domain Name System (DNS) domains in Azure.
* **ExpressRoute**. Create a dedicated high-speed connection from your on-premises data center to Azure.
* **Traffic Manager**. Implement load-balancing for high scalability and availability.
* **Network Watcher**. Monitor and diagnose networking issues without logging in to your virtual machines (VMs) using Network Watcher.

**Storage and Backup Services**

* **Azure Storage**. Store data in files, binary large objects (BLOBs), tables, and queues.
* **Data Lake Store**. Use as a hyper scale repository for big data analytics workloads.
* **StorSimple**. Consolidate storage infrastructure, automate data management across the enterprise, accelerate disaster recovery, and improve compliance.
* **Backup**. Use Azure as a backup destination for your on-premises servers.
* **Azure Site Recovery**. Manage complete site failover for on-premises and Azure private cloud infrastructures.

**Security and Identity Services**

* **Security Center**. Use Azure Security Center to get a central view of the security state of all of your Azure resources.
* **Key Vault**. Create and import encryption keys, reduce latency with cloud scale and global redundancy, and simplify and automate tasks for SSL/TLS certificates.
* **Azure Active Directory**. Integrate your corporate directory with cloud services for a single sign on (SSO) solution.
* **Azure Active Directory Domain Services**. Join Azure virtual machines to a domain without domain controllers.
* **Azure Multi-Factor Authentication**. Implement additional security measures in your applications to verify user identity.

**Web and Mobile Services** (including media and content delivery)

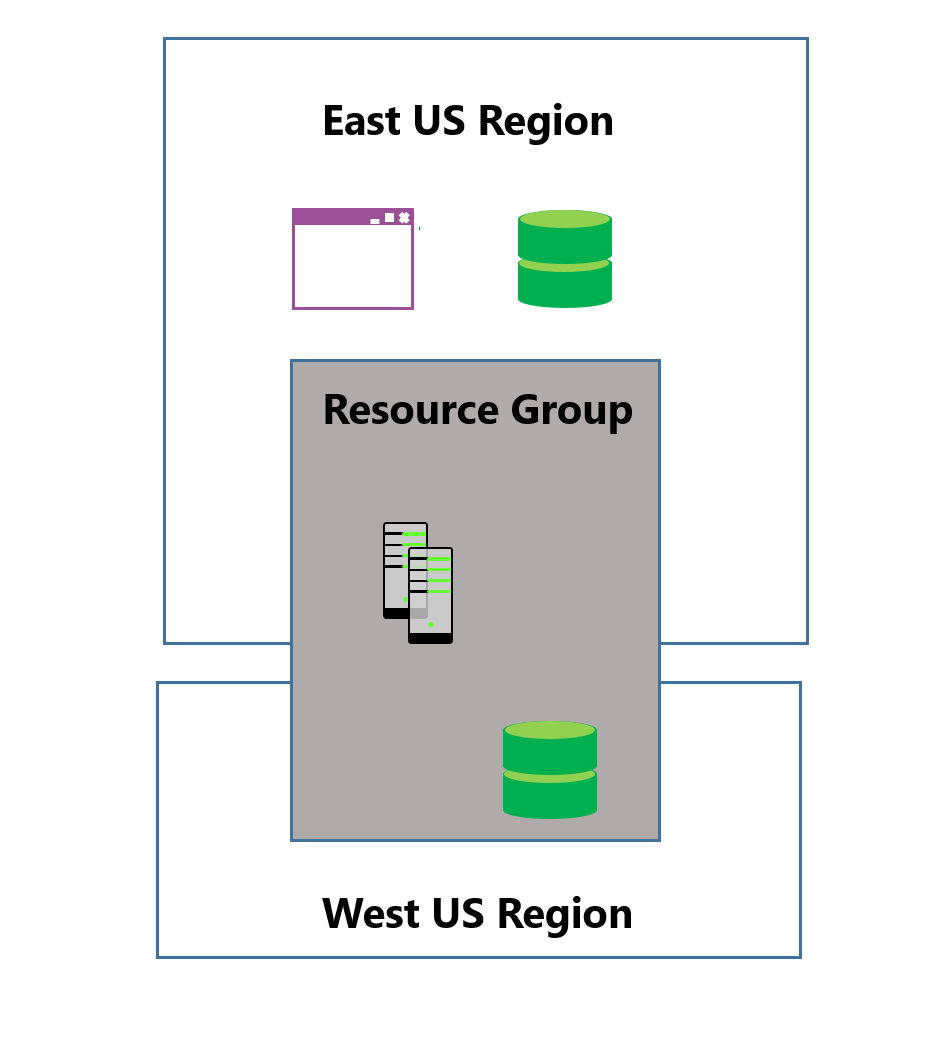
* **App Service**. Create scalable cloud apps for web and mobile without the need to manage the underlying web server configuration.
* **Web Apps**. Quickly create and deploy mission critical Web apps at scale.
* **Mobile Apps**. Implement a hosted back-end service for mobile applications that run on multiple mobile platforms.
* **API Apps**. Publish your service APIs securely.
* **Logic Apps**. Automate the access and use of data across clouds without writing code.
* **Content Delivery Network**. Ensure secure, reliable content delivery with broad global reach.
* **Media Services**. Encode, store, and stream video and audio at scale.
* **Azure Search**. Provide a fully managed search service.

**Databases, Data and Analytics Services**

* **SQL Database**. Implement relational databases for your applications without the need to provision and manage a database server.
* **SQL Data Warehouse**. Learn how to use SQL Data Warehouse, which combines the SQL Server relational database with massively parallel processing.
* **Azure Cosmos DB**. Implement an Azure Cosmos DB service that functions as a globally distributed database using one of the multi-model APIs.
* **HDInsight**. Use Apache Hadoop to perform big data processing and analysis.
* **Redis Cache**. Implement high-performance caching solutions for your applications.
* **Machine Learning**. Apply statistical models to your data and perform predictive analytics.

**Monitoring and Management Services**

* **Microsoft Azure Portal**. Build, manage, and monitor all Azure products in a single, unified console.
* **Azure Resource Manager**. Use Azure Resource Manager to deploy, manage, and monitor the infrastructure components and resources for applications and services.
* **Log Analytics**. Centralize log data from multiple systems in a single data store, gaining deeper insight into your hybrid IT environment.
* **Automation**. Simplify cloud management with process automation.
* **Scheduler**. Use Scheduler to schedule and monitor jobs such as recurring application actions and routine maintenance.



**Grouping Related Services**

When provisioning Azure services, you can group related services that exist in multiple regions to more easily manage those services. Resource groups are logical groups and can therefore span multiple regions.

For example, you might take a cloud application that consists of a database and a website. As an administrator, you can combine these into a logical named resource group, making it easier to identify the specific individual instances required to support the application and manage those instances as a unit.

**Colocating Services by Using Regions**

Although resource groups provide a logical grouping of services, they do not reflect the geographical location of the data centers in which those services are deployed. You can specify the region in which you want to host those services. This is known as colocating the services and it is a best practice to colocate interdependent Azure services in the same region. In some cases, Azure will actually enforce the colocation of services where a resource in that same region would be required.

Azure subscriptions

**Set up Access to Azure**

Azure is a subscription based service. You need to sign up for the service. There are several different available options to purchase Azure services, and because subscriptions are on a pay as you go basis, this means that you only pay for the actual resources consumed, such as compute, storage and bandwidth. Therefore, it is advisable to analyze your business needs and plan your spend accordingly. Some organizations have an enterprise agreement through which they manage their infrastructure spend and access to cloud services, but Azure subscriptions are available at various levels to meet the needs of smaller business and individuals.

* [Microsoft Open License Program](https://www.microsoft.com/en-us/licensing/licensing-programs/open-license.aspx): Typically used by small to medium sized organizations
* [Enterprise Agreements](https://azure.microsoft.com/en-us/pricing/enterprise-agreement/): Typically used by larger enterprises

There is also the ability to get an individual subscription to Azure. This is covered in the next topic.

For more information, you can see:

Microsoft Account - <https://aka.ms/edx-azure215x-msa>

Azure Free Account - <https://azure.microsoft.com/en-us/free/free-account-faq/>

How to buy Azure - <https://aka.ms/edx-azure215x-az3>

Getting started with Azure is now even easier and the benefits have been recently updated. You can try Azure for free and we’ll add a $200 credit for you, which allows you to experiment with any combination of Azure services for 30 days.

When you sign up, you’ll also get 12-months of free compute, storage, network, and database services, and over 30 services that are continuously free, to learn and build your next ideas into prototypes.

Get the details, activate your free account, and get to work developing with Azure today - [https://azure.microsoft.com/en-us/free/?OCID=AID624663\_OLA\_205658197\_93454499.](https://azure.microsoft.com/en-us/free/?OCID=AID624663_OLA_205658197_93454499)

0

00:00:01,350 --> 00:00:04,030

You are watching a quick guide on how to understand my bill.

1

00:00:04,030 --> 00:00:07,259

This short video will walk your through the recommended approach, the important terms

2

00:00:07,259 --> 00:00:10,360

to know, and next steps for managing Azure costs.

3

00:00:10,360 --> 00:00:12,900

The first step is to download your invoice.

4

00:00:12,900 --> 00:00:15,170

To do this, you'll first sign into the Azure portal.

5

00:00:15,170 --> 00:00:17,600

You'll then select the billing icon on the left.

6

00:00:17,600 --> 00:00:21,790

You'll select invoices in the blade that pops up, and then you'll select the subscription

7

00:00:21,790 --> 00:00:23,869

that you're interested in in the drop down menu.

8

00:00:23,869 --> 00:00:28,320

Finally, you'll click on download invoice to open a PDF copy of the selected period

9

00:00:28,320 --> 00:00:29,400

statement.

10

00:00:29,400 --> 00:00:32,320

The second step is to download your detailed usage file.

11

00:00:32,320 --> 00:00:35,450

To do this, you'll first sign into the Azure account center.

12

00:00:35,450 --> 00:00:38,940

Note that only the account administrator can access the Azure account center.

13

00:00:38,940 --> 00:00:44,050

Other billing admins, such as the owner, can get usage information using billing APIs.

14

00:00:44,050 --> 00:00:49,839

Next, you'll select the same subscription that you selected when downloading your invoice.

15

00:00:49,839 --> 00:00:51,829

Then select billing history.

16

00:00:51,829 --> 00:00:57,260

Finally, click download usage to open a CSV file containing the details of each resource's

17

00:00:57,260 --> 00:00:59,870

daily usage and charges.

18

00:00:59,870 --> 00:01:03,459

Now that invoice and your usage report downloaded you're ready to understand your bill.

19

00:01:03,459 --> 00:01:06,211

The biggest reason you may not understand your bill is because you have a charge you

20

00:01:06,211 --> 00:01:07,950

don't recognize or you didn't expect.

21

00:01:07,950 --> 00:01:11,890

So first you want to find the charge you're interested in, in the usage charges section

22

00:01:11,890 --> 00:01:12,890

of the invoice.

23

00:01:12,890 --> 00:01:14,619

In this example, we have selected scheduler.

24

00:01:14,619 --> 00:01:19,580

Second, you'll find the scheduler charge in the statements section of the usage report.

25

00:01:19,580 --> 00:01:23,649

By comparing charges, you can make sure they match in both reports and get further detail

26

00:01:23,649 --> 00:01:25,050

in the usage report.

27

00:01:25,050 --> 00:01:29,399

To see a breakdown of this charge on a daily basis, you can go to the daily usage section

28

00:01:29,399 --> 00:01:30,869

of the CSV.

29

00:01:30,869 --> 00:01:33,200

Here's some of the daily detailed usage highlighted.

30

00:01:33,200 --> 00:01:37,480

In total, all the daily detailed usages should add up to the consumed quantity in the statements

31

00:01:37,480 --> 00:01:40,719

section of the CSV, and in the line item of the invoice.

32

00:01:40,719 --> 00:01:44,719

If you still have questions about a particular charge, you can open a support ticket through

33

00:01:44,719 --> 00:01:45,719

the Azure portal.

34

00:01:45,719 --> 00:01:49,450

It's important to note that many of the terms have the same meanings between the invoice

35

00:01:49,450 --> 00:01:52,240

and the detailed usage file, but with different names.

36

00:01:52,240 --> 00:01:56,749

So, for example, "resource" is equivalent to "meter name," in the detailed usage file.

37

00:01:56,749 --> 00:02:01,500

You can see here that as well, "billable" on the invoice is the same as "overage quantity"

38

00:02:01,500 --> 00:02:04,850

on the usage report.

39

00:02:04,850 --> 00:02:08,690

Here's the full list of terms you can compare.

40

00:02:08,690 --> 00:02:12,910

Now that you've compared charges and terms, you may want to begin managing costs in Azure.

41

00:02:12,910 --> 00:02:16,420

You can do so by visiting the cost analysis tab on the Azure portal.

42

00:02:16,420 --> 00:02:20,520

Here you can find charts like "Current cost by resource or burn rate," which is the current

43

00:02:20,520 --> 00:02:22,980

and forecasted cost of this subscription.

44

00:02:22,980 --> 00:02:25,930

these can be helpful for seeing where you're spending the most and adjusting your usage

45

00:02:25,930 --> 00:02:27,450

to save money.

46

00:02:27,450 --> 00:02:31,030

As well, if you're interested in more information, you can learn more and prevent unexpected

47

00:02:31,030 --> 00:02:34,000

costs with Azure.

48

00:02:34,000 --> 00:02:36,100

There are four other resources you should be aware of.

49

00:02:36,100 --> 00:02:39,670

The first is Microsoft's documentation on understanding your bill.

50

00:02:39,670 --> 00:02:42,210

This builds on the content in this video.

51

00:02:42,210 --> 00:02:46,110

The second is the documentation on the descriptions of different terms you'll encounter in the

52

00:02:46,110 --> 00:02:48,950

Azure usage report.

53

00:02:48,950 --> 00:02:53,840

And third, is document on how to use tags to organize Azure resources.

54

00:02:53,840 --> 00:02:57,970

To recap, to understand your Azure bill, you'll want to compare charges on your invoice with

55

00:02:57,970 --> 00:02:59,310

the detailed usage file.

56

00:02:59,310 --> 00:03:03,580

To start managing costs for different Azure resources, you can get started at the Cost

57

00:03:03,580 --> 00:03:06,720

Analysis blade in the Subscription section of the Azure portal.

58

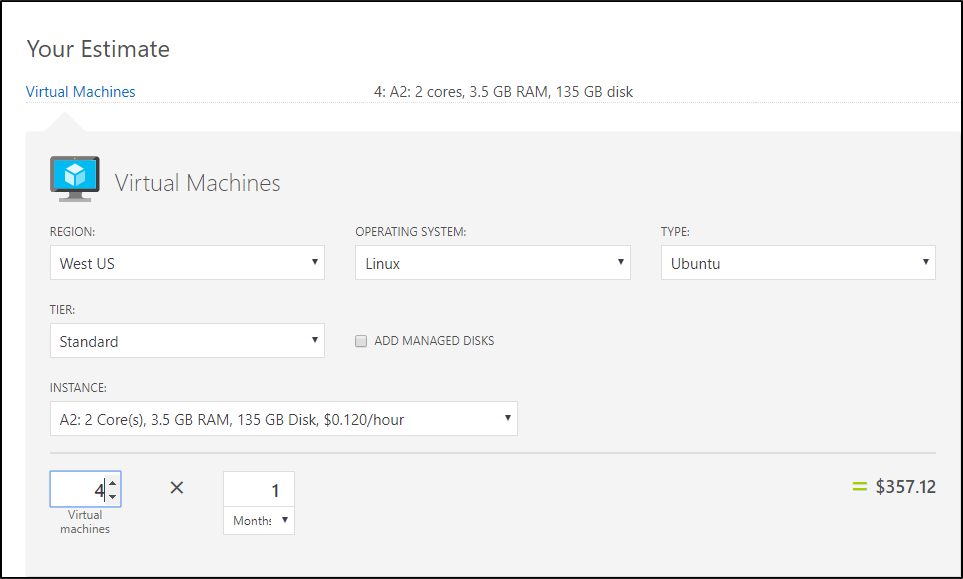
00:03:06,720 --> 00:03:08,150

Thank you for watching our quick guide.

Estimating the Cost of Paid subscription

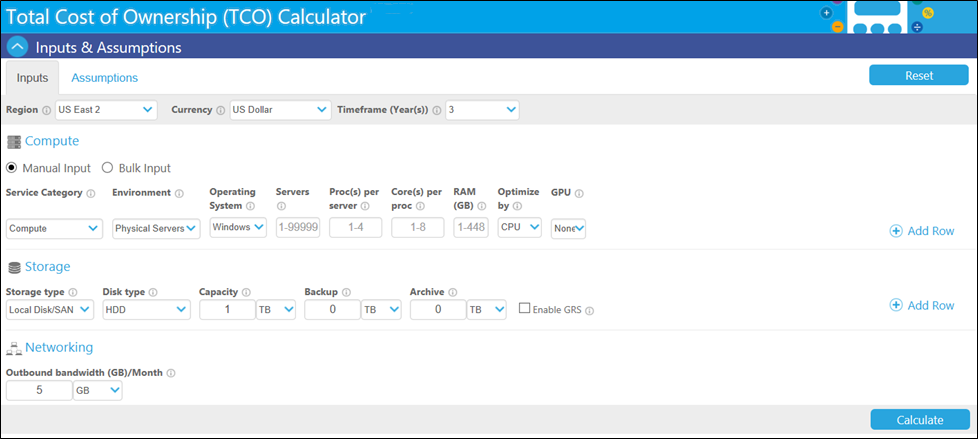
If you are considering obtaining an Azure subscription for individual use, or you are part of a small business and need to manage your cloud infrastructure, there are tools to help you estimate costs.

**Pricing Calculator**



The Pricing Calculator is used to estimate costs based on individual scenarios. The following illustration shows an example of a sample calculation for the cost of running four Linux virtual machines for a month at the specified pricing tier. You can likewise estimate costs for the number of databases, amount and type of storage, and other services and the tool will provide a total of estimated costs in whichever currency you require.

**Total Cost of Ownership (TCO) Calculator**

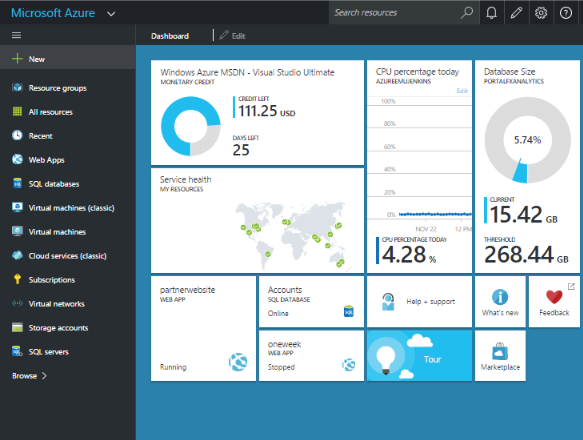


The TCO calculator quickly compares the TCO of your on-premises infrastructure with a comparable Azure deployment. With this tool, you can estimate the savings you can realize by moving workloads to Azure. For example, you can input information about your on-premises servers, storage, and networking infrastructure and calculate savings in different situations.

For more information, you can see:

Azure Pricing - <https://azure.microsoft.com/en-us/pricing/>

New Azure Portal



**Simplified and Unified Administration**

The New Azure portal brings together all the different cloud resources into a customizable console called a Hub, from where you can manage and control all your resources. This includes web applications, databases, virtual machines, virtual networks, storage, and more. For software development teams, it provides a repository where they can manage the entire DevOps lifecycle: code check-in, test, build, and deployment of an application or service to the cloud.

**Role-based Access Control**

Role-based Access is natively integrated into the management platform and you can use it to assign explicit management and access rights to individuals and groups. You can set this up at the subscription, service, and operations levels, giving you a precise degree of control over who can access or manage what.

**Used to Create and Manage Resource Manager Resources**

From the Azure portal, you can manage your application components, such as a virtual machine, a storage account, virtual network or database as resource groups. Resource groups are containers and are part of the Azure Resource Manager deployment model. Using templates, you can deploy, update or delete all of your resources in a single, coordinated action or set of actions.

Powershell

PowerShell is a cross-platform (Windows, Linux, and macOS) automation and configuration tool/framework that works well with your existing tools and is optimized for dealing with structured data (e.g. JSON, CSV, XML, etc.), REST APIs, and object models. It includes a command-line shell, an associated scripting language and a framework for processing cmdlets.

* If you are running Windows you do not need to install PowerShell. It is available natively within Windows.
* If you are running Ubuntu, CentOS and macOS you can [download the product](https://github.com/PowerShell/PowerShell/blob/master/docs/installation/linux.md) from GitHub.

**PowerShell Editors**

To use PowerShell, you will need a scripting editor. If you do not have a preferred editor here are a couple of options.

* Windows PowerShell Integrated Scripting Environment (ISE): Available natively on Windows platform. Currently, it is not supported on Linux or macOS.
* Visual Studio Code (VS Code): Available on Windows, Linux and macOS.

✔️ If you need help getting started PowerShell, there is a [Windows PowerShell Basics](https://courses.edx.org/courses/course-v1:Microsoft+INF210x+1T2018a/course/) course on edx.org.

For more information, you can see:

PowerShell Documentation - [https://docs.microsoft.com/en-us/powershell/#pivot=main&panel=getstarted](https://docs.microsoft.com/en-us/powershell/)

Installing PowerShell (Learning) - <https://github.com/PowerShell/PowerShell/tree/master/docs/learning-powershell>

Installing PowerShell (non-Windows) - <https://github.com/PowerShell/PowerShell/blob/master/docs/installation/linux.md>

PowerShell on Linux and Open Source - <https://channel9.msdn.com/Blogs/hybrid-it-management/PowerShell-on-Linux-and-Open-Source>

Azure Powershll

Azure PowerShell is a set of modules within PowerShell that provide cmdlets to manage Azure. You can use the cmdlets to create, test, deploy, and manage solutions and services delivered through the Azure platform. In most cases, the cmdlets can be used for the same tasks as the Azure Portal, such as creating and configuring cloud services, virtual machines, virtual networks, and web apps. Here are two ways to install Azure PowerShell

1. Use the Microsoft Web Platform Installer (Web PI). Web PI is a free tool that makes getting the latest components of the Microsoft Web Platform, including PowerShell, very easy.
2. You can use an elevated Windows PowerShell console or the PowerShell ISE. Simply run the following commands.

**Install the Azure Service Management modules**

Install-Module Azure

**Install the Azure Resource Manager modules**

Install-Module AzureRM

✏️ There is a practical exercise to introduce you to Azure PowerShell.

For more information, you can see:

Web Platform Installer - <https://aka.ms/edx-azure215x-wpi>

Install and configure Azure PowerShell - <https://docs.microsoft.com/en-us/powershell/azure/install-azurerm-ps?view=azurermps-4.3.1>

Azure Powershell

1. PowerShell interface and in PowerShell ISE. Now we'll move on to the Azure
2. Resource Manager deployment model. So these cmdlets, again very similar
3. set of cmdlets, but now we're using the Azure Resource Manager tools. So this
4. is the new portal. The difference if you look at these cmdlets, right away, is
5. the... is the addition of the RM letters... so instead of Login-AzureAccount, it's
6. Login-AzureRMAccount. You've got to remember this difference because
7. if you, for example, Login-AzureAccount, but then try to create an AzureRMVm
8. it's gonna fail. It's gonna tell you that you're not logged in. You're gonna sit there
9. going "but I am..." You have to use the right set of cmdlets for the deployment
10. model that you're going to use. So continuing down... get subscription, create
11. a resource group... and then, if we have a template that represents the objects we
12. want to deploy, we can use the New-AzureRMResourceGroup deployment cmdlet...
13. and simply reference the template, and it will actually install or deploy
14. all the resources in the template into the resource group specified in that
15. **cmdlet.**

Azure Powershell cmdlets

There are too many Azure cmdlets to discuss all of them in detail, but some key points are:

* The default Azure module (Service Management mode) includes cmdlets for managing Azure services as individual resources.
* The AzureResourceManager module (Resource Manager mode) includes cmdlets that enable you to manage related services as a single unit. All cmdlets have RM as part of the command.
* You can use standard PowerShell commands like [Get-Command](https://msdn.microsoft.com/powershell/reference/5.1/microsoft.powershell.core/Get-Command) and [Get-Help](https://msdn.microsoft.com/powershell/reference/5.1/microsoft.powershell.core/Get-Help) to see more information about the available Azure cmdlets in each module.

**Service Management Mode**

The Azure module contains a comprehensive set of cmdlets, which you can use to view, create, and manage individual Azure services in your subscription. For example, you can use the [New-AzureWebsite](https://docs.microsoft.com/en-us/powershell/servicemanagement/azure.compute/v1.6.1/New-AzureWebsite?redirectedfrom=msdn) cmdlet to create an Azure website, or use the [Get-AzureStorageAccount](https://msdn.microsoft.com/en-us/library/azure/dn495134.aspx) cmdlet to get a reference to an existing storage account.

#Get a list of cmdlets in the Azure module

Get-Command -Module Azure | Get-Help | Format-Table Name, Synopsis

**Resource Manager Mode**

In Resource Manager mode, you can use PowerShell to create and manage Azure resources in resource groups. This approach makes it easier to manage related sets of resources as a unit. For example, you could use the [Get-AzureResourceGroup](https://msdn.microsoft.com/en-us/library/dn757694.aspx) cmdlet to get a reference to an existing resource group, or use the [Remove-AzureResourceGroup](https://msdn.microsoft.com/en-us/library/dn757684.aspx) cmdlet to remove a resource group and all of the resources it contains.

#Get a list of cmdlets in the Resource Manager module

Get-Command -Module AzureRM | Get-Help | Format-Table Name, Synopsis

For more information, you can see:

Azure Resource Manager Cmdlets - <https://msdn.microsoft.com/en-us/library/mt786812(v=azure.300).aspx>

Azure Cross Platform Interface CLI

The Azure CLI is an open-source, cross-platform, shell-based command line interface for scripting and automating the creation and management of resources in Azure. Azure CLI installs and runs on Windows, Linux or Mac. Programs that you create in one platform will run on the other platforms.

**Installing the Azure CLI**

There are several ways you can install the Azure CLI

* **Microsoft Web Platform Installer (Web PI)**. Web PI is a free tool that makes getting the latest components of the Microsoft Web Platform, including  
  Azure CLI.
* **Node Package Module (npm)**. Run npm (the package for JavaScript) to install the latest Azure CLI package on your Linux distribution. Requires node.js and npm on your computer.
* **Installer**. Download an installer for simple installation on Mac or Windows.
* **Docker container**. Start using the latest CLI in a ready-to-run Docker container. Requires Docker host on your computer.

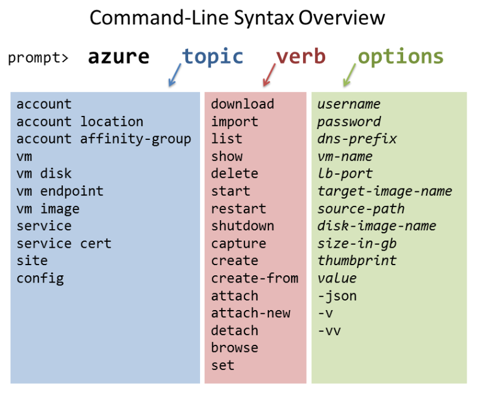
For more information, you can see:

Microsoft Azure Cross Platform Command Line 2.0 (GitHub) - <https://github.com/azure/azure-xplat-cli>

Install Azure CLI 2.0 - <https://docs.microsoft.com/en-us/cli/azure/install-azure-cli>

Azure CLI syntax usage

The CLI syntax is easy to understand. The syntax uses a standard format: azure (az) topic (noun) verb and options. For example, az vm create can be used to create a virtual machine. Another example is az group create. This command could be used to create a resource group.



For more information, you can see:

Azure CLI Samples for Linux virtual machines -https://docs.microsoft.com/en-us/azure/virtual-machines/linux/cli-samples

Logging in

1. And so to get started I'm going to go ahead and log in.
2. **Now the login process is a little bit different in the sense that you've not got to go to**
3. a browser to complete the logon process.
4. So we'll just open up a browser instance and go to aka.ms/devicelogin... and we'll just
5. complete the logon process by pasting in this code they give us in the console.
6. It identifies that this comes from the Cross-Platform Command Line and we'll go ahead and actually
7. just complete the logon.
8. So what can we do with a resource group?
9. Type azure group --help.
10. This will give us help specific to what we can do with a resource group.
11. If I scroll up a little bit, you can see that we can create a resource group.
12. And we'll export the TravelApp resource group.
13. That's going to download the entire contents of that resource group, which contains storage,
14. and which contains... virtual machines and networks, and all sorts of settings... as
15. an ARM template that we can then bring into our favorite editor and modify.
16. And so we can see if we look in Windows Explorer, here's the JSON template that was downloaded,
17. and if I open that in Visual Studio, or any text editor... here's the template right here.
18. And we can then take this template as the basis for an additional deployment.
19. One additional thing that I want to show you real quick, is we go log on to our portal...
20. so let's go to portal.azure.com, this client becomes part of the automation that Azure
21. produces automatically for any object that you create.
22. So if the unified client is your preferred method of scripting against Azure, any object
23. that you create, you can get the script that's appropriate for the cross-platform client
24. just by getting it straight out of the portal.

Explore the Azure Portal

In this exercise, you will sign in to the new Azure Portal and explore the layout, navigation, and basic functionality.

1. Navigate to the new Azure Portal and sign in – **portal.azure.com**
2. Notice the menu of services on the left side. Practice ordering the items in the list, by selecting and dragging the items.
3. Click **More Services** on the bottom of the menu. Notice there are many more services than are displayed in the left menu.
4. Filter the list by **SQL**. Notice the star to the right of the services. This lets you add the item to your favorites (left menu).
5. Return to the **Dashboard**. Practice moving tiles around and adding new elements to your dashboard.
6. Click the **Settings** icon in the menu bar (top right). Review the available portal settings such as: theme, language, and regional format.
7. Click the **Notifications** (bell) icon in the menu bar (top right). This is where information about deployments is shown.
8. Notice the **Search textbox** on the menu bar (top). This makes it easy to locate both general and specific resources.
9. As you have time, **explore** other areas of the portal.

Install Azure Powershell

Install Azure PowerShell

From your computer, open an elevated PowerShell prompt.

**Cmdlets for Service Manager (Classic)** – also includes basic cmdlets such as subscription management

1. Run the **Install-Module Azure** command. This will install the Azure module which represents service management.
2. If you are notified that the repository is untrusted, confirm that you want to install the modules by typing **Y** and then pressing the Enter key.
3. Once the download and installation is finished, run the **Import-Module Azure** command.

**Cmdlets for Resource Manager**

1. Run the **Install-Module AzureRM** command. This will install the AzureRM module which represents resource management.
2. If you get prompted to install and import the NuGet provider, Type **Y** and then press the Enter key.
3. If you are notified that the repository is untrusted, confirm that you want to install the modules by typing **Y** and then pressing the Enter key.
4. The installation process will take several minutes as packages are downloaded and installed.
5. After the download and installation is finished, run the **Import-Module AzureRM** command.
6. Note: If you receive a message about running scripts on your computer has been disabled, temporarily change the execution policy:

**Set-ExecutionPolicy Unrestricted**

1. After the import command has completed, return the execution policy to restricted.

**Set-ExecutionPolicy Restricted**

**Explore cmdlets and update the Help pages**

1. To view storage cmdlets, run:

Get-Command \*rmstorage\*

1. to view syntax, description, and usage informatio for a command, run:

**Get-Help New-AzureStorageAccount -full**

If you have trouble installing the PowerShell modules from the PowerShell gallery, you can try the WebPI method instead. Visit <http://aka.ms/webpi-azps> to download and install the modules.

Connect to Azure using Powershell

Connect to Azure using PowerShell

In this exercise, you will use PowerShell to connect to your Azure subscription and retrieve information about your subscription.

1. Open a PowerShell prompt.
2. Run the **Login-AzureRmAccount** command.
3. If you are prompted to enable data collection, type **N**.
4. If you receive an error message that you can’t be signed in because cookies are blocked, perform Step 5. Otherwise, continue with Step 6.
5. Run Internet Explorer. If this is the first time running it, enable the default settings. Otherwise, configure Internet Explorer to accept cookies by going to the settings, going to the Privacy tab, and configuring the Advanced settings.
6. In the sign-in window, type the email address or phone number of your Azure account. If you are prompted to specify a work/school account or a personal account, choose the one that corresponds to your Azure subscription. Then, type your password and click the Sign in button. If you have multi-factor authentication, complete the authentication process.
7. If the sign in succeeds and the account is an administrator for an Azure subscription, then your Azure account information will be listed. Now, you can explore.
8. Run the **Get-AzureRmSubscription** command to view the subscriptions associated with the account.

Explore Azure Resources

In this exercise, you will explore different Azure resources. These resources will help you during the following courses. You should bookmark these favorites.

1. Visit the [Azure Documentation](https://azure.microsoft.com/en-us/documentation/) center for the latest information about Azure.
2. The [Microsoft Azure Blog](https://azure.microsoft.com/en-us/blog/) is a great place to get the latest information.
3. The [Azure Resources](https://azure.microsoft.com/en-us/resources/) page has links to events, white papers, and downloads.
4. The [Azure Fridays](https://azure.microsoft.com/en-us/resources/videos/index/?series=azure-friday) series provides short deep dive videos from the experts.
5. [Microsoft Virtual Academy](https://mva.microsoft.com/search/SearchResults.aspx) offers online training specific to IT professionals working with Azure.
6. The [Microsoft Press Store](https://www.microsoftpressstore.com/search/index.aspx?page=1&query=Azure&showResults=Store&searchagain=Search+again&sortBy=Relevance) offers a large variety of Azure ebooks and books.