

# Intro to Azure Virtual Machines

26 September 2018 11:52

- [Introduction to Azure Virtual Machines](#)

## Compiling a checklist for creating an Azure Virtual Machine

Performing a migration of on-premises servers to Azure requires planning and care. You can move them all at once, or more likely, in small batches or even individually. Before you create a single VM, you should sit down and sketch out your current infrastructure model and see how it might map to the cloud.

### Required resources for IaaS Virtual Machines

Let's walk through a checklist of things to think about.

- Start with the network
- Name the VM
- Decide the location for the VM
- Determine the size of the VM
- Understanding the pricing model
- Storage for the VM
- Select an operating system

## Start with the network

The first thing you should think about isn't the virtual machine at all - it's the network.

Virtual networks (VNets) are used in Azure to provide private connectivity between Azure Virtual Machines and other Azure services. VMs and services that are part of the same virtual network can access one another. By default, services outside the virtual network cannot connect to services within the virtual network. You can, however, configure the network to allow access to the external service, including your on-premises servers.

This latter point is why you should spend some time thinking about your network configuration. Network addresses and subnets are not trivial to change once you have them set up, and if you plan to connect your private company network to the Azure services, you will want to make sure you consider the topology before putting any VMs into place.

When you set up a virtual network, you specify the available address spaces, subnets, and security. If the VNet will be connected to other VNets, you must select address ranges that are not overlapping. This is the range of private addresses that the VMs and services in your network can use. You can use unroutable IP addresses such as 10.0.0.0/8, 172.16.0.0/12, or 192.168.0.0/16, or define your own range. Azure will treat

any address range as part of the private VNet IP address space if it is only reachable within the VNet, within interconnected VNets, and from your on-premises location. If someone else is responsible for the internal networks, you should work with that person before selecting your address space to make sure there is no overlap and to let them know what space you want to use, so they don't try to use the same range of IP addresses.

## Segregate your network

After deciding the virtual network address space(s), you can create one or more subnets for your virtual network. You do this to break up your network into more manageable sections. For example, you might assign 10.1.0.0 to VMs, 10.2.0.0 to back-end services, and 10.3.0.0 to SQL Server VMs.

### Note

Azure reserves the first four addresses and the last address in each subnet for its use.

## Secure the network

By default, there is no security boundary between subnets, so services in each of these subnets can talk to one another. However, you can set up Network Security Groups (NSGs), which allow you to control the traffic flow to and from subnets and to and from VMs. NSGs act as software firewalls, applying custom rules to each inbound or outbound request at the network interface and subnet level. This allows you to fully control every network request coming in or out of the VM.

## Plan each VM deployment

Once you have mapped out your communication and network requirements, you can start thinking about the VMs you want to create. A good plan is to select a server and take an inventory:

- What does the server communicate with?
- Which ports are open?
- Which OS is used?
- How much disk space is in use?
- What kind of data does this use? Are there restrictions (legal or otherwise) with not having it on-premises?
- What sort of CPU, memory, and disk I/O load does the server have? Is there burst traffic to account for?

We can then start to answer some of the questions Azure will have for a new virtual machine.

## Name the VM

One piece of information people often don't put much thought into is the name of the VM. The VM name is used as the computer name, which is configured as part of the operating system. You can specify a name of up to 15 characters on a Windows

VM and 64 characters on a Linux VM.

This name also defines a manageable Azure resource, and it's not trivial to change later. That means you should choose names that are meaningful and consistent, so you can easily identify what the VM does. A good convention is to include the following information in the name:

Element	Example	Notes
Environment	dev, prod, QA	Identifies the environment for the resource
Location	uw (US West), ue (US East)	Identifies the region into which the resource is deployed
Instance	01, 02	For resources that have more than one named instance (web servers, etc.)
Product or Service	service	Identifies the product, application, or service that the resource supports
Role	sql, web, messaging	Identifies the role of the associated resource

For example, devusc-webvm01 might represent the first development web server hosted in the US South Central location.

What is an Azure resource?

An Azure resource is a manageable item in Azure. Just like a physical computer in your datacenter, VMs have several elements that are needed to do their job:

- The VM itself
- Storage account for the disks
- Virtual network (shared with other VMs and services)
- Network interface to communicate on the network
- Network Security Group(s) to secure the network traffic
- Public Internet address (optional)

Azure will create all of these resources if necessary, or you can supply existing ones as part of the deployment process. Each resource needs a name that will be used to identify it. If Azure creates the resource, it will use the VM name to generate a resource name - another reason to be very consistent with your VM names!

## Decide the location for the VM

Azure has datacenters all over the world filled with servers and disks. These datacenters are grouped into geographic *regions* ('West US', 'North Europe', 'Southeast Asia', etc.) to provide redundancy and availability.

When you create and deploy a virtual machine, you must select a region where you want the resources (CPU, storage, etc.) to be allocated. This lets you place your VMs as close as possible to your users to improve performance and to meet any legal, compliance, or tax requirements.

Two other things to think about regarding the location choice. First, the location can limit your available options. Each region has different hardware available and some configurations are not available in all regions. Second, there are price differences between locations. If your workload isn't bound to a specific location, it can be very cost effective to check your required configuration in multiple regions to find the lowest price.

## Determine the size of the VM

Once you have the name and location set, you need to decide on the size of your VM. Rather than specify processing power, memory, and storage capacity independently, Azure provides different *VM sizes* that offer variations of these elements in different sizes. Azure provides a wide range of VM size options allowing you to select the appropriate mix of compute, memory, and storage for what you want to do.

The best way to determine the appropriate VM size is to consider the type of workload your VM needs to run. Based on the workload, you're able to choose from a subset of available VM sizes. Workload options are classified as follows on Azure:

Option	Description
General purpose	General-purpose VMs are designed to have a balanced CPU-to-memory ratio. Ideal for testing and development, small to medium databases, and low to medium traffic web servers.
Compute optimized	Compute optimized VMs are designed to have a high CPU-to-memory ratio. Suitable for medium traffic web servers, network appliances, batch processes, and application servers.
Memory optimized	Memory optimized VMs are designed to have a high memory-to-CPU ratio. Great for relational database servers, medium to large caches, and in-memory analytics.
Storage optimized	Storage optimized VMs are designed to have high disk throughput and IO. Ideal for VMs running databases.
GPU	GPU VMs are specialized virtual machines targeted for heavy graphics rendering and video editing. These VMs are ideal options for model training and inferencing with deep learning.
High performance computes	High performance compute is the fastest and most powerful CPU virtual machines with optional high-throughput network interfaces.

You're able to filter on the workload type when you configure the VM size in the Azure. The size you choose directly affects the cost of your service. The more CPU, memory, and GPU you need, the higher the price point.

## What if my size needs change?

Azure allows you to change the VM size when the existing size no longer meets your needs. You can upgrade or downgrade the VM - as long as your current hardware configuration is allowed in the new size. This provides a fully agile and elastic approach to VM management.

The VM size can be changed while the VM is running, as long as the new size is available in the current hardware cluster the VM is running on. The Azure portal makes this obvious by only showing you available size choices. The command line tools will report an error if you attempt to resize a VM to an unavailable size. Changing a running VM size will automatically reboot the machine to complete the request.

If you stop and deallocate the VM, you can then select any size available in your region since this removes your VM from the cluster it was running on.

## Warning

Be careful about resizing production VMs - they will be rebooted automatically which can cause a temporary outage and change some configuration settings such as the IP address.

## Understanding the pricing model

There are two separate costs the subscription will be charged for every VM: compute and storage. By separating these costs, you scale them independently and only pay for what you need.

**Compute costs** - Compute expenses are priced on a per-hour basis but billed on a per-minute basis. For example, you are only charged for 55 minutes of usage if the VM is deployed for 55 minutes. You are not charged for compute capacity if you stop and deallocate the VM since this releases the hardware. The hourly price varies based on the VM size and OS you select. The cost for a VM includes the charge for the Windows operating system. Linux-based instances are cheaper because there is no operating system license charge.

### Tip

You might be able to save money by reusing existing licenses for Windows with the Azure Hybrid benefit.

**Storage costs** - You are charged separately for the storage the VM uses. The status of the VM has no relation to the storage charges that will be incurred; even if the VM is stopped/deallocated and you aren't billed for the running VM, you will be charged for the storage used by the disks.

You're able to choose from two payment options for compute costs.

Option	Description
Pay as you go	With the pay-as-you-go option, you pay for compute capacity by the second, with no long-term commitment or upfront payments. You're able to increase or decrease compute capacity on demand as well as start or stop at any time. Prefer this option if you run applications with short-term or unpredictable workloads that cannot be interrupted. For example, if you are doing a quick test, or developing an app in a VM, this would be the appropriate option.
Reserved Virtual Machine Instances	The Reserved Virtual Machine Instances (RI) option is an advance purchase of a virtual machine for one or three years in a specified region. The commitment is made up front, and in return, you get up to 72% price savings compared to pay-as-you-go pricing. RIs are flexible and can easily be exchanged or returned for an early termination fee. Prefer this option if the VM has to run continuously, or you need budget predictability, and you can commit to using the VM for at least a year.

## Storage for the VM

All Azure virtual machines will have at least two virtual hard disks (VHDs). The first disk stores the operating system, and the second is used as temporary storage. You can add additional disks to store application data; the maximum number is determined by the VM size selection (typically two per CPU). It's common to create one or more data disks, particularly since the OS disk tends to be quite small. Also, separating out the data to different VHDs allows you to manage the security,

reliability, and performance of the disk independently.

The data for each VHD is held in Azure Storage as page blobs, which allows Azure to allocate space only for the storage you use. It's also how your storage cost is measured; you pay for the storage you are consuming.

## What is Azure Storage?

Azure Storage is Microsoft's cloud-based data storage solution. It supports almost any type of data and provides security, redundancy, and scalable access to the stored data. A storage account provides access to objects in Azure Storage for a specific subscription. VMs always have one or more storage accounts to hold each attached virtual disk.

Virtual disks can be backed by either Standard or Premium Storage accounts. Azure Premium Storage leverages solid-state drives (SSDs) to enable high performance and low latency for VMs running I/O-intensive workloads. Use Azure Premium Storage for production workloads, especially those that are sensitive to performance variations or are I/O intensive. For development or testing, Standard storage is fine.

When you create disks, you will have two options for managing the relationship between the storage account and each VHD. You can choose either unmanaged disks or managed disks.

Option	Description
Unmanaged disks	With unmanaged disks, you are responsible for the storage accounts that are used to hold the VHDs that correspond to your VM disks. You pay the storage account rates for the amount of space you use. A single storage account has a fixed-rate limit of 20,000 I/O operations/sec. This means that a storage account is capable of supporting 40 standard virtual hard disks at full utilization. If you need to scale out with more disks, then you'll need more storage accounts, which can get complicated.
Managed disks	Managed disks are the newer and recommended disk storage model. They elegantly solve this complexity by putting the burden of managing the storage accounts onto Azure. You specify the size of the disk, up to 4 TB, and Azure creates and manages both the disk <i>and</i> the storage. You don't have to worry about storage account limits, which makes managed disks easier to scale out.

## Select an operating system

Azure provides a variety of OS images that you can install into the VM, including several versions of Windows and flavors of Linux. As mentioned earlier, the choice of OS will influence your hourly compute pricing as Azure bundles the cost of the OS license into the price.

If you are looking for more than just base OS images, you can search the Azure Marketplace for more sophisticated install images that include the OS and popular software tools installed for specific scenarios. For example, if you needed a new WordPress site, the standard technology stack would consist of a Linux server, Apache web server, a MySQL database, and PHP. Instead of setting up and configuring each component, you can leverage a Marketplace image and install the entire stack all at once.

Finally, if you can't find a suitable OS image, you can create your disk image with what you need, upload it to Azure storage, and use it to create an Azure VM. Keep in mind that Azure only supports 64-bit operating systems.

t)rtIYCBUKgIL)B4g6

## Information needed

Please provide the following information. Don't worry, you can always change these settings later.

**Site Title**

Whyteside Action Group

**Username**

wpcloidyg

Username can have only alphanumeric characters, spaces, underscores, hyphens, full stops, and the @ symbol.

**Password**

t)rtIYCBUKgIL)B4g6

 Hide

Strong

**Important:** You will need this password to log in. Please store it in a secure location.

**Your Email**

cloudyg79@outlook.com

Double-check your email address before continuing.

**Search Engine  
Visibility**

☐ Discourage search engines from indexing this site

It is up to search engines to honour this request.

## Describe the options available to create and manage an Azure Virtual Machine

15 minutes

The Azure portal is the easiest way to create resources such as VMs when you are getting started. However, it's not necessarily the most efficient or quickest way to work with Azure, particularly if you need to create several resources together. In our case, we will eventually be creating dozens of VMs to handle different tasks. Creating them manually in the Azure portal wouldn't be a fun task!

Let's look at some other ways to create and administer resources in Azure:

- Azure Resource Manager
- Azure PowerShell
- Azure CLI
- Azure REST API
- Azure Client SDK
- Azure VM Extensions
- Azure Automation Services

## Azure Resource Manager

Let's assume you want to create a copy of a VM with the same settings. You could create a VM image, upload it to Azure, and reference it as the basis for your new VM. This process is inefficient and time-consuming. Azure provides you with the option to create a template from which to create an exact copy of a VM.

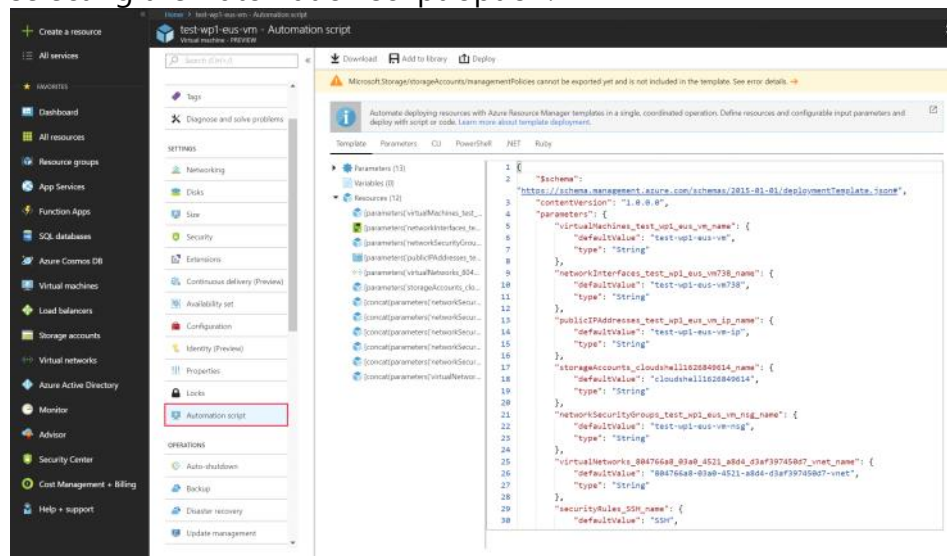
Typically, your Azure infrastructure will contain many resources, many of them related to one another in some way. For example, the VM we created has the virtual machine itself, storage, network interface, web server, and a database - all created together to run the WordPress site. Azure Resource Manager makes working with these related resources more efficient. It organizes resources into named resource groups that let you deploy, update, or delete all of the resources together. When we created the WordPress site, we identified the resource group as part of the VM creation, and Resource Manager placed the associated resources into the same group.

Resource Manager also allows you to create *templates*, which can be used to create and deploy specific configurations.

## What are Resource Manager templates?

Resource Manager templates are JSON files that define the resources you need to deploy for your solution.

You can create resource templates from the Settings section for a specific VM by selecting the Automation script option.



You have the option to save the resource template for later use or immediately deploy a new VM based on this template. For example, you might create a VM from a template in a test environment and find it doesn't quite work to replace your on-premises machine. You can delete the resource group, which deletes all of the resources, tweak the template, and try again. If you only want to make changes to the existing deployed resources, you can change the template used to create it and deploy it again. Resource Manager will change the resources to match the new template.

Once you have it working the way you want it, you can take that template and easily re-create multiple versions of your infrastructure, such as staging and production. You can parameterize fields such as the VM name, network name, storage account name, etc., and load the template repeatedly, using different parameters to customize each environment.



You can use automation scripting tools such as the Azure CLI, Azure PowerShell, or even the Azure REST APIs with your favorite programming language to process resource templates, making this a powerful tool for quickly spinning up your infrastructure.

## Azure PowerShell

Creating administration scripts is a powerful way to optimize your workflow. You can automate everyday, repetitive tasks, and once a script has been verified, it will run consistently, likely reducing errors. Azure PowerShell is ideal for one-off interactive tasks and/or the automation of repeated tasks.

Note

PowerShell is a cross-platform shell that provides services like the shell window and command parsing. Azure PowerShell is an optional add-on package that adds the Azure-specific commands (referred to as cmdlets). You can learn more about installing and using Azure PowerShell in a separate training module.

For example, you can use the `New-AzureRmVM` cmdlet to create a new Azure virtual machine.

PowerShellCopy

```
New-AzureRmVM `
  -ResourceGroupName "TestResourceGroup" `
  -Name "test-wp1-eus-vm" `
  -Location "East US" `
  -VirtualNetworkName "test-wp1-eus-network" `
  -SubnetName "default" `
  -SecurityGroupName "test-wp1-eus-nsg" `
  -PublicIpAddressName "test-wp1-eus-pubip" `
  -OpenPorts 80,3389
```

As shown here, you supply various parameters to handle the large number of VM configuration settings available. Most of the parameters have reasonable values; you only need to specify the required parameters. Learn more about creating and managing VMs with Azure PowerShell in the Automate Azure tasks using scripts with PowerShell module.

## Azure CLI

Another option for scripting and command-line Azure interaction is the Azure CLI. The Azure CLI is Microsoft's cross-platform command-line tool for managing Azure resources such as virtual machines and disks from the command line. It's available for macOS, Linux, and Windows, or in the browser using the Cloud Shell. Like Azure PowerShell, the Azure CLI is a powerful way to streamline your administrative workflow. Unlike Azure PowerShell, the Azure CLI does not need PowerShell to function.

For example, you can create an Azure VM with the `az vm create` command.

Azure CLICopy

```
az vm create \  
  --resource-group TestResourceGroup \  
  --name test-wp1-eus-vm \  
  --image win2016datacenter \  
  --admin-username jonc \  
  --admin-password aReallyGoodPasswordHere
```

The Azure CLI can be used with other scripting languages, for example, Ruby and Python. Both languages are commonly used on non-Windows-based machines where the developer might not be familiar with PowerShell.

Learn more about creating and managing VMs in the [Manage virtual machines with the Azure CLI tool module](#).

## Programmatic (APIs)

Generally speaking, both Azure PowerShell and Azure CLI are good options if you have simple scripts to run and want to stick to command-line tools. When it comes to more complex scenarios, where the creation and management of VMs form part of a larger application with complex logic, another approach is needed.

You can interact with every type of resource in Azure programmatically.

## Azure REST API

The Azure REST API provides developers with operations categorized by resource as well as the ability to create and manage VMs. Operations are exposed as URIs with corresponding HTTP methods (GET, PUT, POST, DELETE, and PATCH) and a corresponding response.

The Azure Compute APIs give you programmatic access to virtual machines and their supporting resources. With this API, you have operations to:

- Create and manage availability sets
- Add and manage virtual machine extensions
- Create and manage managed disks, snapshots, and images
- Access the platform images available in Azure
- Retrieve usage information of your resources
- Create and manage virtual machines
- Create and manage virtual machine scale sets

## Azure Client SDK

Even though the REST API is platform and language agnostic, most often developers will look toward a higher level of abstraction. The Azure Client SDK encapsulates the Azure REST API, making it much easier for developers to interact with Azure.

The Azure Client SDKs are available for a variety of languages and frameworks, including .NET-based languages such as C#, Java, Node.js, PHP, Python, Ruby, and Go.

Here's an example snippet of C# code to create an Azure VM using the Microsoft.Azure.Management.Fluent NuGet package:

C#Copy

```
var azure = Azure
    .Configure()
    .WithLogLevel(HttpLoggingDelegatingHandler.Level.Basic)
    .Authenticate(credentials)
    .WithDefaultSubscription();
// ...
var vmName = "test-wp1-eus-vm";
azure.VirtualMachines.Define(vmName)
    .WithRegion(Region.USEast)
    .WithExistingResourceGroup("TestResourceGroup")
    .WithExistingPrimaryNetworkInterface(networkInterface)
    .WithLatestWindowsImage("MicrosoftWindowsServer", "WindowsServer", "2012-
R2-Datacenter")
    .WithAdminUsername("janc")
    .WithAdminPassword("aReallyGoodPasswordHere")
    .WithComputerName(vmName)
    .WithSize(VirtualMachineSizeTypes.StandardDS1)
    .Create();
```

Here's the same snippet in Java using the Azure Java SDK:

JavaCopy

```
String vmName = "test-wp1-eus-vm";
// ...
VirtualMachine virtualMachine = azure.virtualMachines()
    .define(vmName)
    .withRegion(Region.US_EAST)
    .withExistingResourceGroup("TestResourceGroup")
    .withExistingPrimaryNetworkInterface(networkInterface)
    .withLatestWindowsImage("MicrosoftWindowsServer", "WindowsServer", "2012-
R2-Datacenter")
    .withAdminUsername("janc")
    .withAdminPassword("aReallyGoodPasswordHere")
    .withComputerName(vmName)
    .withSize("Standard_DS1")
    .create();
```

## Azure VM Extensions

Let's assume you want to configure and install additional software on your virtual machine after the initial deployment. You want this task to use a specific configuration, monitored and executed automatically.

Azure VM extensions are small applications that allow you to configure and automate tasks on Azure VMs after initial deployment. Azure VM extensions can be run with the Azure CLI, PowerShell, Azure Resource Manager templates, and the Azure portal.

You bundle extensions with a new VM deployment or run them against an existing system.

# Azure Automation Services

Saving time, reducing errors, and increasing efficiency are some of the most significant operational management challenges faced when managing remote infrastructure. If you have a lot of infrastructure services, you might want to consider using higher-level services in Azure to help you operate from a higher level. Azure Automation allows you to integrate services that allow you to automate frequent, time-consuming, and error-prone management tasks with ease. These services include process automation, configuration management, and update management.

- **Process Management.** Let's assume you have a VM that is monitored for a specific error event. You want to take action and fix the problem as soon as it's reported. Process automation allows you to set up watcher tasks that can respond to events that may occur in your datacenter.
- **Configuration Management.** Perhaps you want to track software updates that become available for the operating system that runs on your VM. There are specific updates you may want to include or exclude. Configuration management allows you to track these updates and take action as required. You use System Center Configuration Manager to manage your company's PC, servers, and mobile devices. You can extend this support to your Azure VMs with Configuration Manager.
- **Update Management.** This is used to manage updates and patches for your VMs. With this service, you're able to assess the status of available updates, schedule installation, and review deployment results to verify updates applied successfully. Update management incorporates services that provide process and configuration management. You enable update management for a VM directly from your Azure Automation account. You can also allow update management for a single virtual machine from the virtual machine blade in the portal.

As you can see, Azure provides a variety of tools to create and administer resources so that you can integrate management operations into a process *that works for you*. Let's examine some of the other Azure services to make sure your infrastructure resources are running smoothly.

#

## Manage the availability of your Azure VMs

10 minutes

Often, the success of a services company is directly related to the service level agreements (SLA) the company has with its customers. Your customers expect the services you provide always to be available and their data kept safe. This is something that Microsoft takes very seriously. Azure provides tools you can use to manage availability, data security, and monitoring, so you know your services are always available for your customers.

Administration of an Azure VM isn't limited to managing the operating system or software that runs on the VM. It helps to know which services Azure provides that ensure service availability and support automation. These services help you to plan your organization's business continuity and disaster recovery strategy.

Here, we'll cover an Azure service that helps you improve VM availability, streamlines VM management tasks, and keeps your VM data backed up and safe. Let's start by defining availability.

## What is availability?

Availability is the percentage of time a service is available for use.

Let's assume you have a website and you want your customers to be able to access information at all times. Your expectation is 100% availability concerning website access.

## Why do I need to think about availability when using Azure?

Azure VMs run on physical servers hosted within the Azure Datacenter. As with most physical devices, there's a chance that there could be a failure. If the physical server fails, the virtual machines hosted on that server will also fail. If this happens, Azure will move the VM to a healthy host server automatically. However, this self-healing migration could take several minutes, during which, the application(s) hosted on that VM will not be available.

The VMs could also be affected by periodic updates initiated by Azure itself. These maintenance events range from software updates to hardware upgrades and are required to improve platform reliability and performance. These events usually are performed without impacting any guest VMs, but sometimes the virtual machines will be rebooted to complete an update or upgrade.

### Note

Microsoft does not automatically update your VM's OS or software. You have complete control and responsibility for that. However, the underlying software host and hardware are periodically patched to ensure reliability and high performance at all times.

To ensure your services aren't interrupted and avoid a single point of failure, it's recommended to deploy at least two instances of each VM. This feature is called an *availability set*.

## What is an availability set?

An availability set is a logical feature used to ensure that a group of related VMs are deployed so that they aren't all subject to a single point of failure and not all upgraded at the same time during a host operating system upgrade in the datacenter. VMs placed in an availability set should perform an identical set of functionalities and have the same software installed.

### Tip

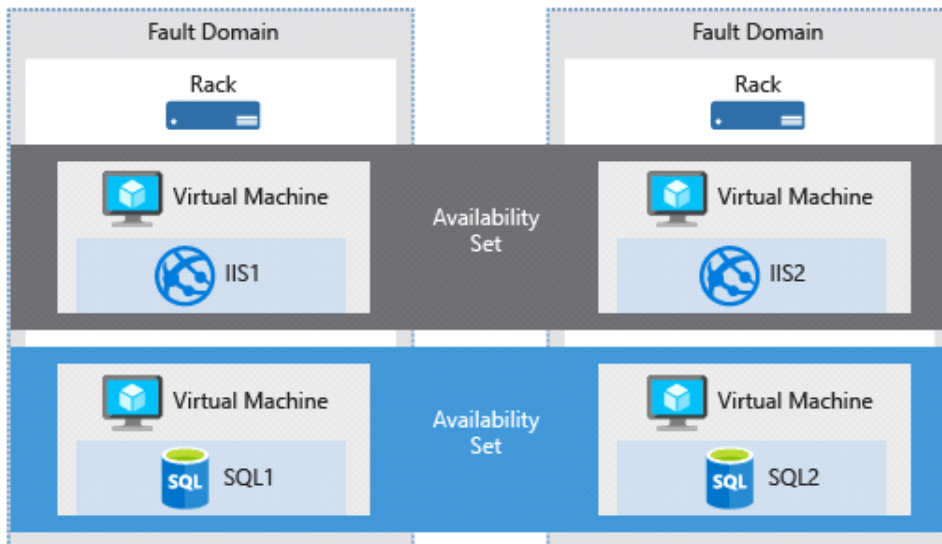
Microsoft offers a 99.95% external connectivity service level agreement (SLA) for multiple-instance VMs deployed in an availability set. That means that for the SLA to apply, there must be at least two instances of the VM deployed within an availability set.

You can create availability sets through the Azure portal in the disaster recovery

section. Also, you can build them using Resource Manager templates, or any of the scripting or API tools. When you place VMs into an availability set, Azure guarantees to spread them across Fault Domains and Update Domains.

What is a fault domain?

A fault domain is a logical group of hardware in Azure that shares a common power source and network switch. You can think of it as a rack within an on-premises datacenter. The first two VMs in an availability set will be provisioned into two different racks so that if the network or the power failed in a rack, only one VM would be affected. Fault domains are also defined for managed disks attached to VMs.



What is an update domain?

An update domain is a logical group of hardware that can undergo maintenance or be rebooted at the same time. Azure will automatically place availability sets into update domains to minimize the impact when the Azure platform introduces host operating system changes. Azure then processes each update domain one at a time. Availability sets are a powerful feature to ensure the services running in your VMs are always available to your customers. However, they aren't foolproof. What if something happens to the data or the software running on the VM itself? For that, we'll need to look at other disaster recovery and backup techniques.

## Failover across locations

You can also replicate your infrastructure across sites to handle regional failover. Azure Site Recovery replicates workloads from a primary site to a secondary location. If an outage happens at your primary site, you can fail over to a secondary location. This failover allows users to continue to access your applications without interruption. You can then fail back to the primary location once it's up and running again. Azure Site Recovery is about replication of virtual or physical machines; it keeps your workloads available in an outage.

While there are many attractive technical features to Site Recovery, there are at least two significant business advantages:

1. Site Recovery enables the use of Azure as a destination for recovery, thus eliminating the cost and complexity of maintaining a secondary physical datacenter.
2. Site Recovery makes it incredibly simple to test failovers for recovery drills without impacting production environments. This makes it easy to test your planned or unplanned failovers. After all, you don't have a good disaster recovery plan if you've never tried to failover.

The recovery plans you create with Site Recovery can be as simple or as complex as your scenario requires. They can include custom PowerShell scripts, Azure Automation runbooks, or manual intervention steps. You can leverage the recovery plans to replicate workloads to Azure, easily enabling new opportunities for migration, temporary bursts during surge periods, or development and testing of new applications.

Azure Site Recovery works with Azure resources, or Hyper-V, VMware, and physical servers in your on-premises infrastructure and can be a key part of your organization's business continuity and disaster recovery (BCDR) strategy by orchestrating the replication, failover, and recovery of workloads and applications if the primary location fails.

## Back up your virtual machines

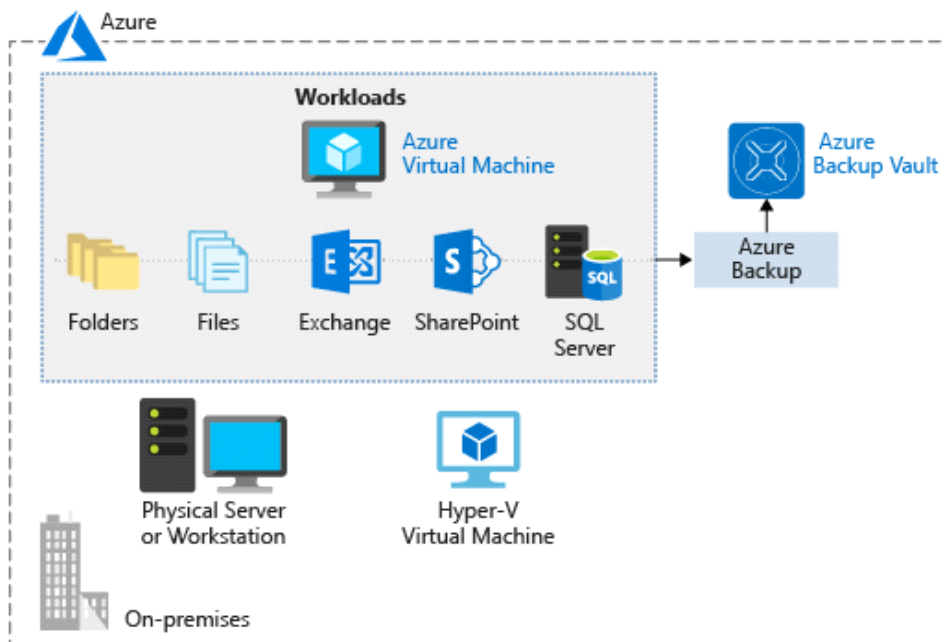
7 minutes

Data backup and recovery is a necessary piece of the planning for any good infrastructure. Perhaps it's a bug that erases some data, or maybe you need to retrieve some archived data for auditing purposes. Maintaining a good backup strategy will ensure you aren't scrambling when data or software needs to be restored.

Azure Backup is a *backup as a service* offering that protects physical or virtual machines no matter where they reside: on-premises or in the cloud.

Azure Backup can be used for a wide range of data backup scenarios, such as the following:

- Files and folders on Windows OS machines (physical or virtual, local or cloud)
- Application-aware snapshots (Volume Shadow Copy Service)
- Popular Microsoft server workloads such as Microsoft SQL Server, Microsoft SharePoint, and Microsoft Exchange
- Native support for Azure Virtual Machines, both Windows, and Linux
- Linux and Windows 10 client machines



## Advantages of using Azure Backup

Traditional backup solutions don't always take full advantage of the underlying Azure platform. The result is a solution that tends to be expensive or inefficient. The solution either offers too much or too little storage, does not offer the correct types of storage, or has cumbersome and long-winded administrative tasks. Azure Backup was designed to work in tandem with other Azure services and provides several distinct benefits.

- Automatic storage management. Azure Backup automatically allocates and manages backup storage and uses a pay-as-you-use model. You only pay for what you use.
- Unlimited scaling. Azure Backup uses the power and scalability of Azure to deliver high availability.
- Multiple storage options. Azure Backup offers locally redundant storage where all copies of the data exist within the same region and geo-redundant storage where your data is replicated to a secondary region.
- Unlimited data transfer. Azure Backup does not limit the amount of inbound or outbound data you transfer. Azure Backup also does not charge for the data that is transferred.
- Data encryption. Data encryption allows for secure transmission and storage of your data in Azure.
- Application-consistent backup. An application-consistent backup means that a recovery point has all required data to restore the backup copy. Azure Backup provides application-consistent backups.
- Long-term retention. Azure doesn't limit the length of time you keep the backup data.

## Using Azure Backup

Azure Backup utilizes several components that you download and deploy to each computer you want to back up. The component that you deploy depends on what



you want to protect.

- Azure Backup agent
- System Center Data Protection Manager
- Azure Backup Server
- Azure Backup VM extension

Azure Backup uses a Recovery Services vault for storing the backup data. A vault is backed by Azure Storage blobs, making it a very efficient and economical long-term storage medium. With the vault in place, you can select the machines to back up and define a backup policy (when snapshots are taken and for how long they're stored).