Unit 1 of 7 ∨

Next >

✓ 100 XP ▶

Introduction

3 minutes

You manage your corporate database infrastructure of SQL Server VMs running in Azure. Times are good, and you need to scale up your operation, while still controlling costs. Some database operations involve many reads of existing data. The regular invoice and reporting runs are write-heavy operations. You'd like to find a way to optimize your infrastructure to handle all operation types. Before investing in infrastructure improvements, you decide to explore VM disk caching options first.

Caching is a common approach to speeding up computing resources. Azure supports a range of caching technologies to help optimize data access across the Azure landscape, including specific cache options for the Azure storage and disks used by Azure Virtual Machines (VMs).

We're going to explore the available disk caching options in Azure and learn how to manage disk caching with the portal and PowerShell.

Learning objectives

In this module, you will:

- Describe the key considerations around disk performance in Azure
- Describe the effects of caching on disk performance in Azure
- Enable and manage cache settings with the Azure portal
- Enable and manage cache settings with PowerShell

Prerequisites

None

Next unit: Effect of caching on disk performance in Azure

Continue >

✓ 100 XP ▶

Effect of caching on disk performance in Azure

8 minutes

Much like your local computers, virtual machine performance can often be tied directly to how quickly it can read and write data. In order to understand how to improve that performance, we first have to understand how performance is measured and the settings and choices that affect it.

We are looking specifically at the underlying disks and storage used for VMs. When you are looking at performance, keep in mind that you will also have to consider the application layer. For example, if you are running a database in a VM, you will want to look at the performance settings specific to the database to ensure it is optimized for the VM and storage you are running it on.

Let's start by defining a few terms and the guarantees Azure makes about them.

I/O operations per second

The storage type you select (standard or premium) will decide how fast your disks are. We measure this performance in I/O operations per second, or IOPS (pronounced "eye-ops").

IOPS is number of requests that can be processed by the disk in one second. A single request is a read or write operation. This measurement is applied directly to storage. For example, if you have a disk that can do a **5000 IOPS**, it means that it is theoretically capable of processing 5,000 read and or write operations per second.

IOPS directly affects your application performance. Some applications, such as retail websites, need high IOPS to handle all the small and random I/O requests that must be processed quickly to keep the site responsive.

IOPS in Azure

When you attach a premium storage disk to your high scale VM, Azure provisions a guaranteed number of IOPS as per the disk specification. For example, a **P50** disk provisions **7500 IOPS**. Each high scale VM size also has a specific IOPS limit that it can sustain. For example, a **Standard GS5** VM has an **80,000 IOPS** limit.

IOPS is a measurement of the storage disks, however it's a *theoretical* limit - two other factors can affect the actual application performance: **throughput** and **latency**.

What is throughput?

Throughput (also called "bandwidth") is the amount of data that your application is sending to the storage disks in a specified interval (typically per second). If your application is performing I/O with large blocks of data, it requires high throughput.

Azure provisions throughput in premium storage disks based on that disks' specification. For example, a **P50** disk provisions **250 MB per second** disk throughput. Each high scale VM size also has as specific *throughput limit* that it can sustain. For example, **Standard GS5** VM has a maximum throughput of **2,000 MB per second**.

IOPS vs. throughput

Throughput and IOPS have a direct relationship, changing one will have a direct impact on the other. To get a theoretical limit of throughput you can use the formula: `IOPS x I/O size = throughput`. It's important to consider both of these values when planning your application.

What is latency?

Reading and writing data takes time. This is where *latency* comes in. Latency is the time it takes your app to send a request to the disk and get a response. Essentially, latency tells us how long it takes to *process* a single read or write I/O request.

Latency puts a limit on IOPS. For example, if our disk can handle 5000 IOPS but each operation takes 10 ms to process, then our app will be capped to 100 operation per second due to the processing time. This is a simple example, most of the time latency will be much lower. Ultimately, latency and throughput will determine how fast your app can process data from storage.

Premium Storage provides consistent low latencies and you can achieve even better latency when necessary through *caching*.

Testing your disk performance

You can adjust and balance the IOPS, throughput, and latency of your VM disks by selecting the right VM size and storage type. Typically, the larger or more expensive VM sizes will have higher guarantees for max IOPS and throughput. Add into that equation Standard vs. Premium storage and HDD vs. SSD choices and you have several parameters to play with.

Selecting the right combination involves understanding what your application requirements are. High-I/O applications, such as database servers or online transactional processing systems will require higher IOPS, whereas more computational based applications might get by with much lower requirements. In addition, the *types* of operations the applications do will affect your throughput. High random access I/O tends to be slower than long sequential reads.

Once you select your configuration, you can use tools such as Iometer to test your disk performance on Linux and Windows VMs. This will give you a more real-world sense of what kind of performance to expect. It can also help you to identify ways to improve your app's usage of storage. For example, an application that does single threaded I/O is likely to suffer reduced I/O performance because of latency.

Let's look at some other things we can do to improve our disk performance.

Next unit: Enable and configure Azure VM disk cache with the Azure portal

Continue ›

Enable and configure Azure VM disk cache with the Azure portal

5 minutes

We've seen settings and properties you can select to predict your disk performance, now let's look at ways to improve that through *caching*.

Disk caching

A cache is a specialized component that stores data, typically in memory so that it can be accessed more quickly. The data in a cache is often data that has been read previously or data that resulted from an earlier calculation. The goal is to access data faster than getting it from the disk.

Caching uses specialized, and sometimes expensive, temporary storage that has faster read and write performance than permanent storage. Because cache storage is often limited, decisions need to be made as to what data operations will benefit most from caching. But even where the cache can be made widely available, such as in Azure, it's still important to know the workload patterns of each disk before deciding which caching type to use.

Read caching tries to speed up data *retrieval*. Instead of reading from permanent storage, the data is read from the faster cache. Data reads hit the cache under the following conditions:

- The data has been read before and exists in the cache.
- The cache is large enough to hold all the data.

It's important to note that read caching helps when there is some *predictability* to the read queue, such as a set of sequential reads. For random I/O, where the data you're accessing is scattered across storage, caching will be of little or no benefit and can even reduce disk performance.

Write caching tries to speed up *writing data* to persistent storage. By using a write cache, the app can consider the data to be saved. In reality, the data is queued in a cache, waiting to be written to a disk. As you can imagine, this mechanism can be a potential point of failure, such as when a system shuts down before the cached data is written. Some systems, such as SQL Server, handle writing cached data to persistent disk storage themselves.

Azure disk caching

There are two types of disk caching that concern disk storage:

- Azure storage caching
- Azure virtual machine (VM) disk caching

Azure storage caching provides cache services for Azure Blob storage, Azure Files, and other content in Azure. Configuration of these types of cache is beyond the scope of this module.

Azure virtual machine disk caching is about optimizing read and write access to the virtual hard disk (VHD) files attached to Azure VMs. We'll focus on disk caching in this module.

Azure virtual machine disk types

There are three types of disks used with Azure VMs:

- **OS disk**: When you create an Azure VM, Azure automatically attaches a VHD for the operating system (OS).

- **Temporary disk**: When you create an Azure VM, Azure also automatically adds a temporary disk. This disk is used for data, such as page and swap files. The data on this disk may be lost during maintenance or a VM redeploy. Don't use it for storing permanent data, such as database files or transaction logs.

- **Data disks**: A data disk is a VHD that's attached to a virtual machine to store application data or other data you need to keep.

OS disks and data disks take advantage of Azure VM disk caching. The cache size for a VM disk depends on the VM instance size and the number of disks mounted on the VM. Caching can be enabled for only up to 4 TiB sized disks.

Cache options for Azure VMs

There are three common options for VM disk caching:

- **Read/write** – Write-back cache. Use this option only if your application properly handles writing cached data to persistent disks when needed.
- **Read-only** - Reads are done from the cache.
- **None** - No cache. Select this option for write-only and write-heavy disks. Log files are a good candidate because they're write-heavy operations.

Not every caching option is available for each type of disk. The following table shows you the caching options for each disk type:

	Read-only	Read/write	None
OS disk	yes	yes (default)	yes
Data disk	yes (default)	yes	yes
Temp disk	no	no	no

ⓘ Note

Disk caching options can't be changed for **L-Series** and **B-series** virtual machines.

Performance considerations for Azure VM disk caching

So, how can your cache settings affect the performance of your workloads running on Azure VMs?

OS disk

For a VM OS disk, the default behavior is to use the cache in read/write mode. If you have applications that store data files on the OS disk and the apps do lots of random read/write operations to data files, consider moving those files to a data disk that has the caching turned off. Why is that? Well, if the read queue does not contain sequential reads, caching will be of little or no benefit. The overhead of maintaining the cache, as if the data was sequential, can reduce disk performance.

Data disks

For performance-sensitive applications, you should use data disks rather than the OS disk. Using separate disks allows you to configure the appropriate cache settings for each one.

For example, on Azure VMs running SQL Server, enabling **Read-only** caching on the data disks (for regular and TempDB data) can result in significant performance improvements. Log files, on the other hand, are good candidates for data disks with no caching.

> ⚠ **Warning**
>
> Changing the cache setting of an Azure disk detaches and then reattaches the target disk. If it's the operating system disk, the VM is restarted. Stop all applications/services that might be affected by this disruption before changing the disk cache setting.

You can configure virtual machine disk cache settings with any of the following tools:

- Azure portal
- Azure CLI
- Azure PowerShell
- Resource Manager templates

Using the Azure portal to configure caching

When you provision a new VM using the Azure portal, you can't change the default caching configuration for the OS disk from read/write until the VM is deployed.

When you add a data disk to an existing VM, you can configure the cache option before the disk is deployed to the VM.

Changing the cache setting of an Azure disk detaches and reattaches the target disk. If it's the operating system disk, the VM is restarted. Stop all applications/services that might be affected by this disruption before changing the disk cache setting.

Let's create a VM and change the cache settings using the Azure portal.

Next unit: Exercise - Enable and configure Azure VM disk cache with the Azure portal

Continue >

✓ 100 XP

Exercise - Enable and configure Azure VM disk cache with the Azure portal

10 minutes

This module requires a sandbox to complete. You have used 2 of 10 sandboxes for today. More sandboxes will be available tomorrow.

Activate sandbox

Suppose you run a photo sharing site with data stored on Azure virtual machines (VMs) running SQL Server and custom applications. You want to make the following adjustments:

- You need to change the disk cache settings on a VM.
- You want to add a new data disk to the VM with caching enabled.

You've decided to make these changes through the Azure portal.

In this exercise, we'll walk through making the changes to a VM that we described above. First, let's sign in to the portal and create a VM.

Create a virtual machine

In this step, we're going to create a VM with the following properties:

Property	Value
Image	**Windows Server 2016 Datacenter**

Property	Value
Name	**fotoshareVM**
Resource group	**[sandbox resource group name]**
Location	See below.

1. Sign into the Azure portal ↗ using the same account you activated the sandbox with.

2. On the Azure portal menu or from the **Home** page, select **Create a resource**.

3. *Windows Server 2016 VM* should be in the list of **Popular** Marketplace elements. If not, try searching for "Windows Server 2016 DataCenter" using the search box on the top.

4. Select the Windows VM and click **Create** to start the VM creation process.

5. In the **Basics** panel, verify the selected **Subscription** is *Concierge Subscription*.

6. Under **Resource Group**, select **Use Existing** and choose *[sandbox resource group name]*.

7. In the **Virtual machine name** box, enter *fotoshareVM*.

8. In the **Location** drop-down list, select the closest region to you from the following list.

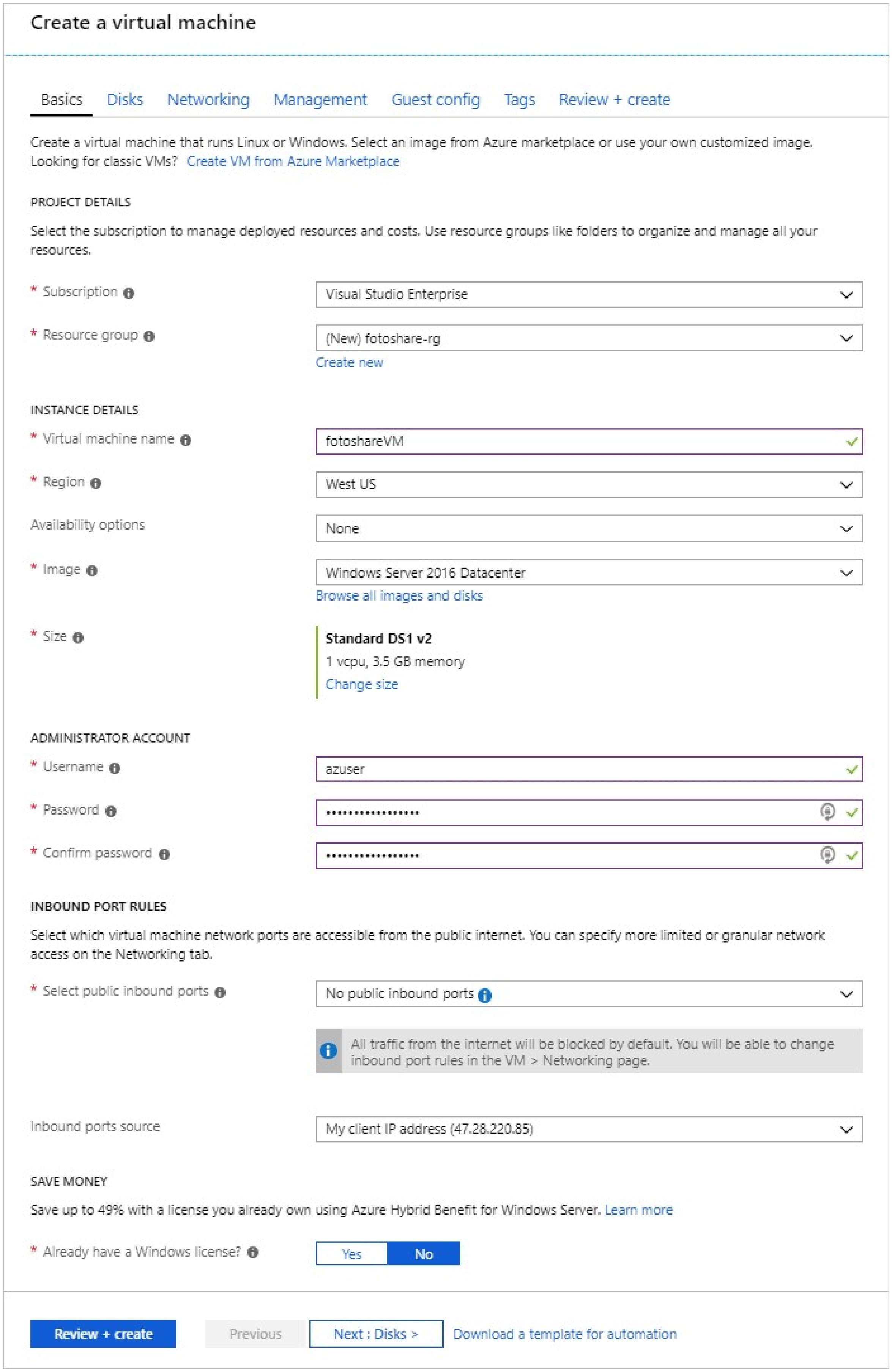
   The free sandbox allows you to create resources in a subset of the Azure global regions. Select a region from the following list when you create resources:

   - West US 2
   - South Central US
   - Central US
   - East US
   - West Europe
   - Southeast Asia
   - Japan East
   - Brazil South

9. For the VM **Size**, the default is **DS1 v2** which gives you a single CPU and 3.5 GB of memory. That's fine for this example.

10. In **ADMINISTRATOR ACCOUNT** section, enter a **Username** and **Password**/**Confirm password** for an administrator account on the new VM.

11. The following image is an example of what the **Basics** configuration looks like when filled out. Leave the defaults for the remaining tabs and fields and click **Review + create**.



Create a virtual machine

Create a virtual machine that runs Linux or Windows. Select an image from Azure marketplace or use your own customized image.
Looking for classic VMs? Create VM from Azure Marketplace

PROJECT DETAILS

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

* Subscription	Visual Studio Enterprise
* Resource group	(New) fotoshare-rg Create new

INSTANCE DETAILS

* Virtual machine name	fotoshareVM ✓
* Region	West US
Availability options	None
* Image	Windows Server 2016 Datacenter Browse all images and disks
* Size	**Standard DS1 v2** 1 vcpu, 3.5 GB memory Change size

ADMINISTRATOR ACCOUNT

* Username	azuser ✓
* Password	••••••••••••••••• ✓
* Confirm password	••••••••••••••••• ✓

INBOUND PORT RULES

Select which virtual machine network ports are accessible from the public internet. You can specify more limited or granular network access on the Networking tab.

* Select public inbound ports	No public inbound ports

All traffic from the internet will be blocked by default. You will be able to change inbound port rules in the VM > Networking page.

Inbound ports source	My client IP address (47.28.220.85)

SAVE MONEY

Save up to 49% with a license you already own using Azure Hybrid Benefit for Windows Server. Learn more

* Already have a Windows license? Yes [✓ No]

12. After reviewing your new VM settings, click **Create** to start the deploying your new VM.

VM creation can take a few minutes as it creates all the various resources (storage, network interface, etc.) to support the virtual machine. Wait until the VM has deployed before continuing with the exercise.

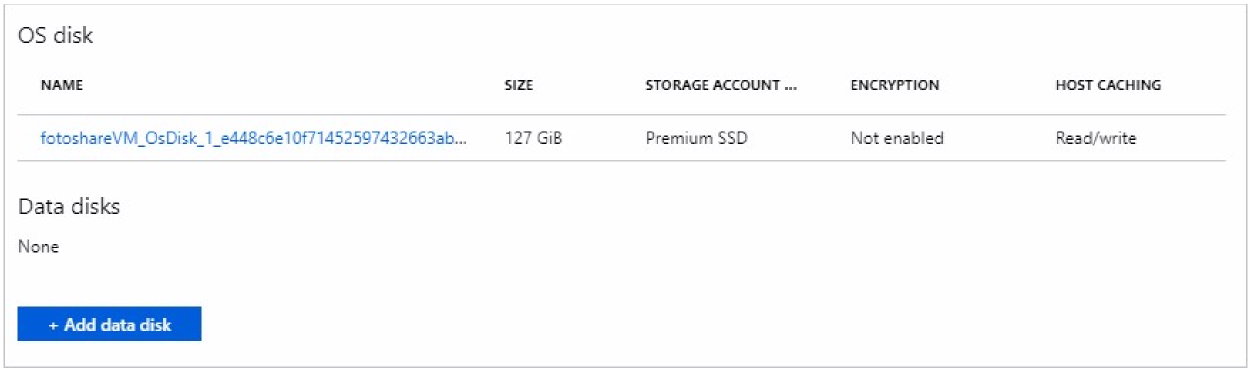
View OS disk cache status in the portal

Once our VM is deployed, we can confirm the caching status of the OS disk using the following steps:

1. Select the **fotoshareVM** resource to open the VM details in the portal. Alternatively, you can click **All resources** in the left sidebar and then select your VM, **fotoshareVM**.

2. Under **Settings**, select **Disks**.

3. On the **Disks** pane, the VM has one disk, the OS disk. Its cache type is currently set to the default value of **Read/write**.



OS disk

NAME	SIZE	STORAGE ACCOUNT ...	ENCRYPTION	HOST CACHING
fotoshareVM_OsDisk_1_e448c6e10f71452597432663ab...	127 GiB	Premium SSD	Not enabled	Read/write

Data disks

None

Change the cache settings of the OS disk in the portal

1. On the **Disks** pane, select **Edit** in the upper left of the screen.

2. Change the **HOST CACHING** value for the OS disk to **Read-only** using the drop-down list, and then select **Save** in the upper left of the screen.

3. This update can take some time. The reason is that changing the cache setting of an Azure disk detaches and reattaches the target disk. If it's the operating system disk, the VM is also restarted. When the operation completes, you'll get a notification saying the VM disks have been updated.

4. Once complete, the OS disk cache type is set to **Read-only**.

Let's move on to data disk cache configuration. To configure a disk, we'll need first to create one.

Add a data disk to the VM and set caching type

1. Back on the **Disks** view of our VM in the portal, go ahead and click **Add data disk**. An error immediately appears in the **Name** field, telling us that the field can't be empty. We don't have a data disk yet, so let's create one.

2. Click in the **Name** list, and then click **Create disk**.

3. In the **Create managed disk** pane, in the **Name** box, type **fotoshareVM-data**.

4. Under **Resource Group**, select **Use existing**, and select *[sandbox resource group name]*.

5. Note the defaults for the remaining fields:

   - Premium SSD
   - 1023 GB in size
   - In the same location as the VM (not changeable).
   - IOPS limit - 5000
   - Throughput limit (MB/s) - 200

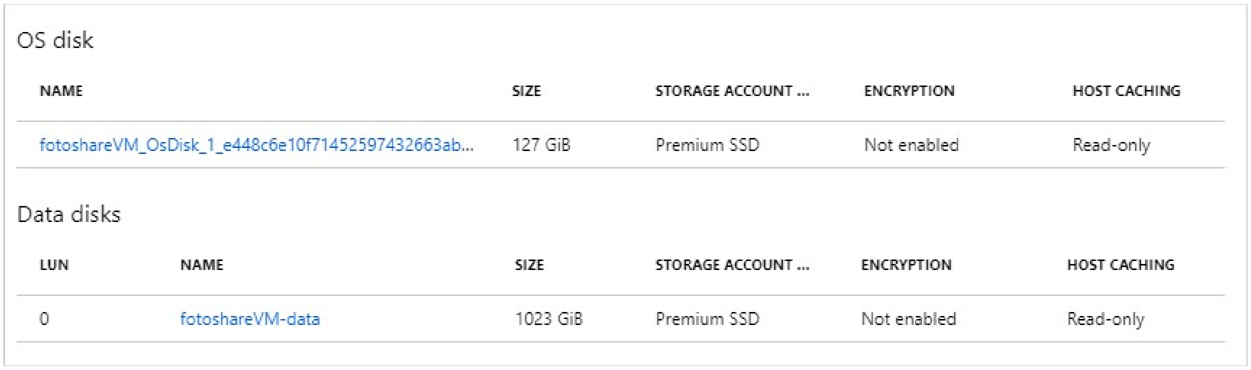
6. Click **Create** at the bottom of the screen.

   Wait until the disk has been created before continuing.

7. Change the **HOST CACHING** value for our new data disk to **Read-only** using the drop-down list (it might be set already), and then click **Save** in the upper left of the screen.

Wait for the VM to finish updating the new data disk. Once complete, you will have a new data disk on your virtual machine.

In this exercise, we used the Azure portal to configure caching on a new VM, change cache settings on an existing disk, and configure caching on a new data disk. The following screenshot shows the final configuration:



OS disk

NAME	SIZE	STORAGE ACCOUNT ...	ENCRYPTION	HOST CACHING
fotoshareVM_OsDisk_1_e448c6e10f71452597432663ab...	127 GiB	Premium SSD	Not enabled	Read-only

Data disks

LUN	NAME	SIZE	STORAGE ACCOUNT ...	ENCRYPTION	HOST CACHING
0	fotoshareVM-data	1023 GiB	Premium SSD	Not enabled	Read-only

Next unit: Manage cache settings with PowerShell

Continue >

‹ Previous Unit 5 of 7 ˅ Next ›

✓ 100 XP ▶

Manage cache settings with PowerShell

5 minutes

Creating administration scripts is a powerful way to optimize your work flow. You can automate common, repetitive tasks. Once a script has been verified, it will run consistently, which will likely reduce errors. In the previous exercise, we created a VM, added a data disk, and changed cache settings, all through the Azure portal. What if we needed to repeat these tasks across many VMs, in many regions? We can do that with Azure PowerShell.

> **Tip**
>
> We cover Azure PowerShell in detail in the **Automate Azure Tasks with PowerShell** module. Make sure to check that module out for more details on installing, configuring, and using PowerShell.

What is Azure PowerShell?

Azure PowerShell is a cross-platform command-line tool to connect to your Azure subscription and manage resources. It's combination of two things: **PowerShell**, which provides the command-line tool support; and the **Az** PowerShell module, which provides the commands (referred to as "cmdlets") to work with Azure.

Azure PowerShell has cmdlets to manipulate most aspects of Azure resources. You can work with resource groups, storage, virtual machines, Azure Active Directory, containers, machine learning, and so on. We cover all these details in other training modules.

PowerShell cmdlets for managing Azure disk caching

Azure PowerShell has specific cmdlets to help manage VMs and disks.

Command	Description
`Get-AzVM`	Gets the properties of a virtual machine.
`Update-AzVM`	Updates the state of an Azure virtual machine.
`New-AzDiskConfig`	Creates a configurable disk object.
`Add-AzVMDataDisk`	Adds a data disk to a virtual machine.

With these, we can do all the tasks we did in the Azure portal. Let's try it out on our VM.

Next unit: Exercise - Manage cache settings with PowerShell

Continue >

Summary

3 minutes

In this module, you learned about Azure disk caching and how it potentially improves performance. We used the Azure portal and Azure PowerShell to manage disk caching for our VM.

Once you have an Azure VM disk caching strategy in place, you can then quickly and easily deploy new VMs and disks with the optimum disk cache settings by using scripts and templates.

Clean up

The sandbox automatically cleans up your resources when you're finished with this module.

When you're working in your own subscription, it's a good idea at the end of a project to identify whether you still need the resources you created. Resources left running can cost you money. You can delete resources individually or delete the resource group to delete the entire set of resources.

Additional resources

- Azure Premium Storage: Design for High Performance
- Get started with Azure PowerShell
- Azure Computer Cmdlets Reference

Check your knowledge

1. Which caching option is a good choice for write-heavy operations such as storing log files?

○ Read-only

○ Read/write

● None ✓

Write-heavy operations generally do not benefit from caching. 'None' is probably the best choice for a disk dedicated to log files.

2. For which type of disk does Azure restart the VM in order to change caching type?

○ Data

● Operating system (OS) ✓

Changing the cache setting of the OS disk requires a VM restart.

○ Temporary

○ Zone-redundant storage (ZRS)

3. Suppose you are using Azure PowerShell to manage a VM. You have a local object that represents the VM and you've made several updates to that local object. Which PowerShell cmdlet would you use to apply those local changes to the actual VM?

● Update-AzVM ✓

This cmdlet updates the state of an Azure virtual machine to the state of a virtual machine object.

○ Set-AzVM

○ New-AzVM

Module complete:

Unlock achievement