Microsoft

Find Azure Virtual Machines with Ultra SSD disks and other capabilities

Prepared by

Data SQL Ninja Engineering Team (datasqlninja@microsoft.com)

Disclaimer

The High-Level Architecture, Migration Dispositions and guidelines in this document is developed in consultation and collaboration with Microsoft Corporation technical architects. Because Microsoft must respond to changing market conditions, this document should not be interpreted as an invitation to contract or a commitment on the part of Microsoft.

Microsoft has provided generic high-level guidance in this document with the understanding that MICROSOFT MAKES NO WARRANTIES, EXPRESS OR IMPLIED, WITH RESPECT TO THE INFORMATION CONTAINED HEREIN.

This document is provided "as-is". Information and views expressed in this document, including URL and other Internet Web site references, may change without notice.

Some examples depicted herein are provided for illustration only and are fictitious. No real association or connection is intended or should be inferred.

This document does not provide you with any legal rights to any intellectual property in any Microsoft product. You may copy and use this document for your internal, reference purposes.

Note: The detail provided in this document has been harvested as part of a customer engagement sponsored through the Data SQL Ninja Engineering.

Table of Contents

Introduction

Microsoft Azure Ultra Disk Storage is a Managed Disks offering that delivers unprecedented and extremely scalable performance with sub-millisecond latency for the most demanding Azure Virtual Machines and container workloads.

Ultra-SSD Disks are designed to work with all Premium SSD enabled Azure Virtual Machines. However, not all Regions and Virtual Machine Sizes have Ultra SSD disks available.

References:

Announcing Ultra SSD – the next generation of Azure Disks technology

https://azure.microsoft.com/en-us/blog/announcing-ultra-ssd-the-next-generation-of-azure-disks-technology-preview/

Announcing the general availability of Azure Ultra Disk Storage

https://azure.microsoft.com/en-us/blog/announcing-the-general-availability-of-azure-ultra-disk-storage/

Azure Ultra Disk Storage: Microsoft's service for your most I/O demanding workloads

https://azure.microsoft.com/en-us/blog/azure-ultra-disk-storage-microsoft-s-service-for-your-most-i-o-demanding-workloads/

Although we have a list of available regions and supported VM series online, there is not practical way to cross this information in the portal. Mainly because not every VM size is available in every supported region with Ultra-SSD Disks. Customers need to try different combinations of region/VM sizes until he gets a combination that allows for Ultra SSDs

This set of PowerShell scripts below enable a customer to list all available possibilities, making it easier to go back to the portal and correct selections without losing time.

Usage

Before using the scripts below you need to connect to your subscription:

https://docs.microsoft.com/en-us/powershell/azure/authenticate-azureps?view=azps-5.1.0

List Azure VM with Ultra SSD for a single region

```
<#

PLEASE REVIEW THE GENERATED SCRIPT CAREFULLY BEFORE APPLYING IT TO A PRODUCTION SYSTEM
Warranty: This script is provided on as "AS IS" basis and there are no warranties,
express or implied, including,
but not limited to implied warranties of merchantability or fitness for a particular
purpose. USE AT YOUR OWN RISK.

#>

$region="westus"
$vmtypes = Get-AzComputeResourceSku | where {$_.Locations.Contains($region)}

foreach($vmtype in $vmtypes)
{
    #Write-Output ($vmtype)[0].Capabilities

    $caps = ($vmtype)[0].Capabilities
    foreach ($cap in $caps)
    {
        if ($cap.Name.Contains("UltraSSDAvailable"))
        {
            Write-Host $vmtype.LocationInfo[0].Location, $vmtype.Name, "Has Ultra SSD"
        }
    }
}
```

List Azure VM with Ultra SSD for all regions

```
<#

PLEASE REVIEW THE GENERATED SCRIPT CAREFULLY BEFORE APPLYING IT TO A PRODUCTION SYSTEM
Warranty: This script is provided on as "AS IS" basis and there are no warranties,
express or implied, including,
but not limited to implied warranties of merchantability or fitness for a particular
purpose. USE AT YOUR OWN RISK.

#>


$vmtypes = Get-AzComputeResourceSku

foreach($vmtype in $vmtypes)
{
    #Write-Output ($vmtype)[0].Capabilities

    $caps = ($vmtype)[0].Capabilities
    foreach ($cap in $caps)
    {
        if ($cap.Name.Contains("UltraSSDAvailable"))
        {
            Write-Host $vmtype.LocationInfo[0].Location, $vmtype.Name, "Has Ultra SSD"
```

```
        }
    }
}
```

Below is another version of the script that has more capabilities:

- Can search for VMs by any of the true/false capabilities
- Allow running the script with parameters
- With no parameters, will present a list of regions and capabilities to select from
- Brought the scripts together, with an "ALL" option for regions
- Included VM memory and core count in the output list
- Script now called FindVM_by_capability

The capabilities listed for virtual machines are

- LowPriorityCapable
- PremiumIO
- EphemeralOSDiskSupported
- EncryptionAtHostSupported
- AcceleratedNetworkingEnabled
- RdmaEnabled
- UltraSSDAvailable

Dec 2020 updated to cater for UltraSSD availability which can be either by region or by zone within the region; see https://docs.microsoft.com/en-us/azure/virtual-machines/disks-enable-ultra-ssd

```
<#
.Synopsis
   Search for Virtual Machine SKUs by capabilities
.EXAMPLE
   .\FindVM_by_capability.ps1 -region ALL -capability 'UltraSSDAvailable'
.EXAMPLE
   .\FindVM_by_capability.ps1 -region eastus -capability 'LowPriorityCapable'
.INPUTS
   Region = the Azure region to search; use "ALL" for a global search (doesn't
currently allow several regions to be selected)
   Capability = the capability to look for
.NOTES
   Run without parameter inputs, the script will query Azure for regions and
capabilities and ask you to choose from the list
   Restriction; this uses out-gridview, so assumes this is being run interactively
   Will take a while before the output appears while the get-azcomputeresourceSKU
cmdlet runs

   10th Dec 2020 updated to cater for UltraSSD availability which can be either by
region or by zone within the region; see
       https://docs.microsoft.com/en-us/azure/virtual-machines/disks-enable-ultra-ssd

PLEASE REVIEW THE GENERATED SCRIPT CAREFULLY BEFORE APPLYING IT TO A PRODUCTION SYSTEM
```

```
Warranty: This script is provided on as "AS IS" basis and there are no warranties,
express or implied, including,
but not limited to implied warranties of merchantability or fitness for a particular
purpose. USE AT YOUR OWN RISK.

#>
param ([string] $region='', [string] $capability='')

# list of the current Azure locations
$regionlist = get-azlocation | select Location, DisplayName | sort DisplayName
# initial list of all the VM types available
$vmtypelist = Get-AzComputeResourceSku | where resourcetype -eq 'virtualMachines'
# query this to get the capabilities that can be true or false
$capabilitieslist = $vmtypelist | select -ExpandProperty capabilities | where
{$_.Value -in 'True','False'} | select Name -unique

# get values for the inputs; if these haven't been passed as parameters, ask for them
interactively
$region     = if ($region -ne '')     {$region}     else {$regionlist | Out-GridView -
PassThru -Title "Pick the region to search:" | select -expandproperty location}
$capability = if ($capability -ne '') {$capability} else {$capabilitieslist | Out-
GridView -PassThru -Title "Pick the capability to search for:" | select -
ExpandProperty name}

$avail_inRegion = $vmtypelist |
    where {$region -eq 'ALL' -or $_.Locations -Contains($region)} -PipelineVariable
SKU | # filter for the region selected
    select -ExpandProperty capabilities | where {$_.Name -eq $capability -and $_.Value
-eq 'True'} | # filter when the requested capability is true
    select `
        @{E={$SKU.Name};N="Name"}, `
        @{E={$SKU.Locations};N="Locations"}, `
        @{E={$SKU.Restrictions.reasoncode | select -Unique};N="Restrictions"}, `
        @{E={$_.name};N="Capability"}, `
        @{E={$_.value};N="Available"}, `
        @{E={($SKU.capabilities | where name -eq 'MemoryGB').value};N="MemoryGB"}, `
        @{E={($SKU.capabilities | where name -eq 'vCPUs').value};N="Cores"}, `
        @{E={$null};N="ZoneSpecific"}

$avail_inzone = $vmtypelist |
    where {$region -eq 'ALL' -or $_.Locations -Contains($region)} -PipelineVariable
SKU | # filter for the region selected
    select -ExpandProperty LocationInfo -PipelineVariable Location | select -
ExpandProperty ZoneDetails -PipelineVariable Zone | select -ExpandProperty
capabilities | `
    where {$_.Name -eq $capability -and $_.Value -eq 'True'} | # filter when the
requested capability is true
    select `
        @{E={$SKU.Name};N="Name"}, `
        @{E={$SKU.Locations};N="Locations"}, `
        @{E={$SKU.Restrictions.reasoncode | select -Unique};N="Restrictions"}, `
        @{E={$_.name};N="Capability"}, `
        @{E={$_.value};N="Available"}, `
        @{E={($SKU.capabilities | where name -eq 'MemoryGB').value};N="MemoryGB"}, `
        @{E={($SKU.capabilities | where name -eq 'vCPUs').value};N="Cores"}, `
        @{E={"Available in Zones " + (($Location.ZoneDetails.Name | sort) -join
","")};N="ZoneSpecific"}

$avail_inRegion + $avail_inzone | format-table name, locations, capability, memoryGB,
Cores, ZoneSpecific, restrictions
# change the format-table above to a select if you want to pipe this out to a file,
etc
# ...sample to filter the results for a specific number of cores and memory size
# $avail_inRegion + $avail_inzone | where {$_.cores -in 2..4 -and $_.MemoryGB -in
8..16} | ft
```

Feedback and suggestions

If you have feedback or suggestions for improving this data migration asset, please contact the Data SQL Ninja Engineering Team (datasqlninja@microsoft.com). Thanks for your support!

Note: For additional information about migrating various source databases to Azure, see the https://datamigration.microsoft.com/