# Lenovo ThinkAgile MX GPU-P Configuration Guide

Initial Release: April 2025

**Provides detailed steps to install and configure GPU drivers for use with Lenovo ThinkAgile MX solutions**

**Applicable to both Azure Local and Windows Server 2025 Storage Spaces Direct (S2D)**

**Covers the process to assign host GPU partitions to virtual machines**

**Discusses GPU-P configuration using the Windows Admin Center**

**Dave Feisthammel**

**David Ye**

LENOVO PRESS

# Table of Contents

# 1 Introduction

Graphics Processing Unit (GPU) virtualization technologies enable GPU acceleration in a virtualized environment, typically within virtual machines. If a workload is virtualized with Hyper-V, graphics virtualization can be employed in order to provide GPU acceleration from the physical GPU to the virtualized apps or services. GPUs can be included in an Azure Local instance to provide GPU acceleration to workloads running in clustered virtual machines.

GPU acceleration can be provided using Discrete Device Assignment (DDA), also known as GPU pass-through, which allows you to dedicate one or more physical GPUs to a virtual machine. Clustered virtual machines can take advantage of GPU acceleration and clustering capabilities such as high availability via failover.

In addition to the DDA option of assigning an entire host GPU to a VM, GPU Partitioning (GPU-P) allows you to share a physical GPU device with multiple VMs. With GPU-P, each VM gets a dedicated fraction of the host GPU instead of the entire GPU.

This document provides instructions and examples to configure GPU-P for use by an Azure Local instance or S2D cluster in Windows Server 2025 Datacenter Edition. We assume that the cluster has already been created. We include information for installing GPU device drivers on hosts and VMs, creating GPU partitions, assigning these partitions to VMs, and configuring Windows Admin Center to view and manage GPU partitions. The entire process is quite straight forward and involves the following activities:

- Install GPU device driver on host
- Create GPU partitions
- Assign GPU partitions to virtual machines
- Install GPU device driver on virtual machines
- Manage GPUs using WAC

For details regarding cluster deployment, refer to one or more of the following guides:

Microsoft Learn article **About Azure Local deployment**: Provides details of the supported methods to deploy an Azure Local instance, including requirements and step-by-step instructions.

https://learn.microsoft.com/en-us/azure/azure-local/deploy/deployment-introduction

**Lenovo Storage Spaces Direct (S2D) Deployment Guide**: Scenario-based deployment instructions that use PowerShell commands to deploy S2D running in Windows Server Datacenter Edition.

https://lenovopress.com/lp0064

The examples in this document are taken from a Lenovo ThinkAgile MX cluster that contains an AMD Radeon PRO V710 GPU installed in each node. The general process discussed can be used regardless of the specific GPU model used. If you are interested in conducting a Proof-of-Concept deployment using this GPU, contact your local Lenovo sales team.

The AMD Radeon PRO V710 GPU is an enterprise-grade GPU designed for cloud computing workloads such as Desktop-as-a-Service, Workstation-as-a-Service, cloud gaming, and AI/machine learning applications. This GPU is also public Azure certified and is available in Microsoft Azure public cloud. For more information about the AMD Radeon PRO V710 GPU, visit the following site:

https://www.amd.com/en/products/accelerators/radeon-pro/amd-radeon-pro-v710.html

AMD ROCm™ is an open software stack including drivers, development tools, and APIs that enable GPU programming from low-level kernel to end-user applications. ROCm is optimized for Generative AI and HPC applications, and is easy to migrate existing code into. More information is available about AMD ROCm software at the following site:

https://www.amd.com/en/products/software/rocm.html

The process covered in this document was developed using the following Microsoft articles:

https://learn.microsoft.com/en-us/windows-server/virtualization/hyper-v/gpu-partitioning

https://learn.microsoft.com/en-us/windows-server/virtualization/hyper-v/partition-assign-vm-gpu

Note that Azure Local 24H2 or Windows Server 2025 Datacenter are required to support Live Migration of VMs that have a GPU partition assigned to them.

# 2 GPU driver installation

Before GPUs can be used in a system, the appropriate device driver must be installed. This includes both host and guest systems. The driver is installed on the host first in order to assign GPU partitions to a virtual machine but must also be installed in any virtual machines that will use a GPU partition.

## 2.1 Install GPU device driver on host

Once downloaded, the driver installation file can be copied to each node that contains one or more GPUs and installed. Alternatively, if the cluster is already configured, the installer can be copied to a CSV where it can be accessed by all nodes in the cluster. To install the driver from a CSV, follow these steps on each cluster node:

1. Copy the device driver package to each node and install the GPU device driver on the host using the following PowerShell/CLI command:

```
pnputil.exe /add-driver *.inf /install
```

```
    Directory: C:\Users\Administrator\Documents\V710_Host_Driver-240904a-407245E


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----         2/21/2025  12:02 PM                B407239
-a----          9/4/2024   4:03 PM          35367 u2407245.cat
-a----          9/4/2024   6:46 PM          68737 u2407245.inf


PS C:\Users\Administrator\Documents\V710_Host_Driver-240904a-407245E> pnputil.exe /add-driver *.inf /install
Microsoft PnP Utility

Adding driver package:  u2407245.inf
Driver package added successfully.
Published Name:         oem10.inf
Driver package installed on device: PCI\VEN_1002&DEV_7460&SUBSYS_0E341002&REV_00\6&151eb934&0&00000009

Total driver packages:  1
Added driver packages:  1
PS C:\Users\Administrator\Documents\V710_Host_Driver-240904a-407245E>
```

2. Verify that the driver was installed properly using the following command:

```
Get-WmiObject Win32_PnPSignedDriver | Select-Object DeviceName, Manufacturer, DriverVersion
| ? DeviceName -Like *Radeon*
```

```
PS C:\> Get-WmiObject Win32_PnPSignedDriver | Select-Object DeviceName, Manufacturer, DriverVersion
| ? DeviceName -Like *Radeon*

DeviceName          Manufacturer                DriverVersion
----------          ------------                -------------
AMD Radeon PRO V710 Advanced Micro Devices, Inc. 32.0.11018.3051
```

3. Use the following command to verify that all device drivers are properly installed and there are no unknown devices. There should be no output returned from this command.

```
Get-WmiObject Win32_PNPEntity | Where-Object {$_.ConfigManagerErrorCode -ne 0} | Select
Name, DeviceID
```

```
PS C:\> Get-WmiObject Win32_PNPEntity | Where-Object {$_.ConfigManagerErrorCode -ne 0} | Select Name, DeviceID
PS C:\>
```

After the host GPU driver has been installed on all nodes that contain one or more GPUs, a special GPU guest driver for virtual machines must be installed on each VM that will consume a GPU partition. Proceed with the next section to install this driver on all VMs.

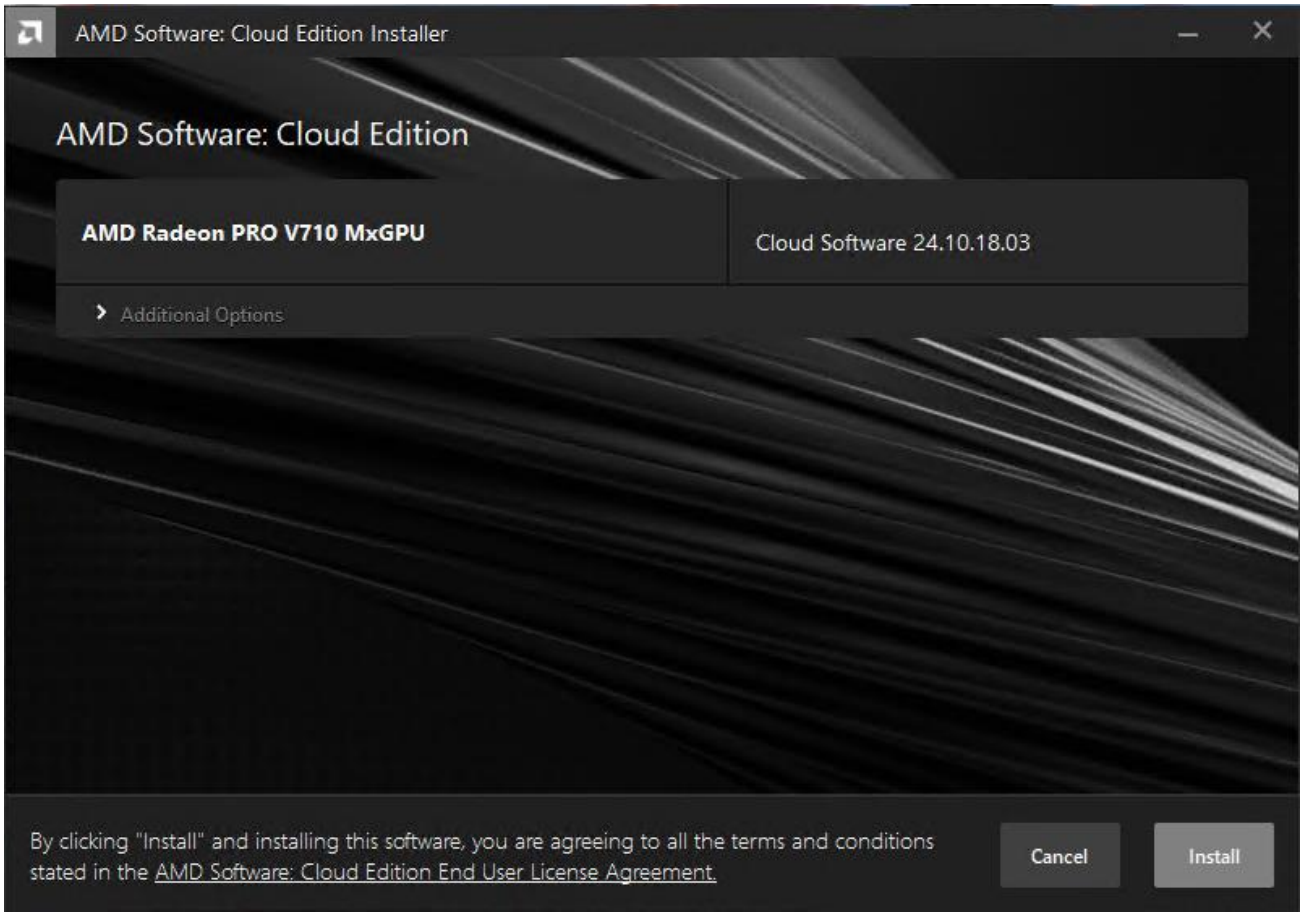
## 2.2 Install GPU device driver on virtual machines

For a VM to be able to use a GPU partition effectively, a device driver must be installed. However, the driver cannot be installed until a GPU partition is assigned to the VM (while it is shutdown). Once the VM is booted, it will see the partition as a GPU and allow the device driver to be installed.

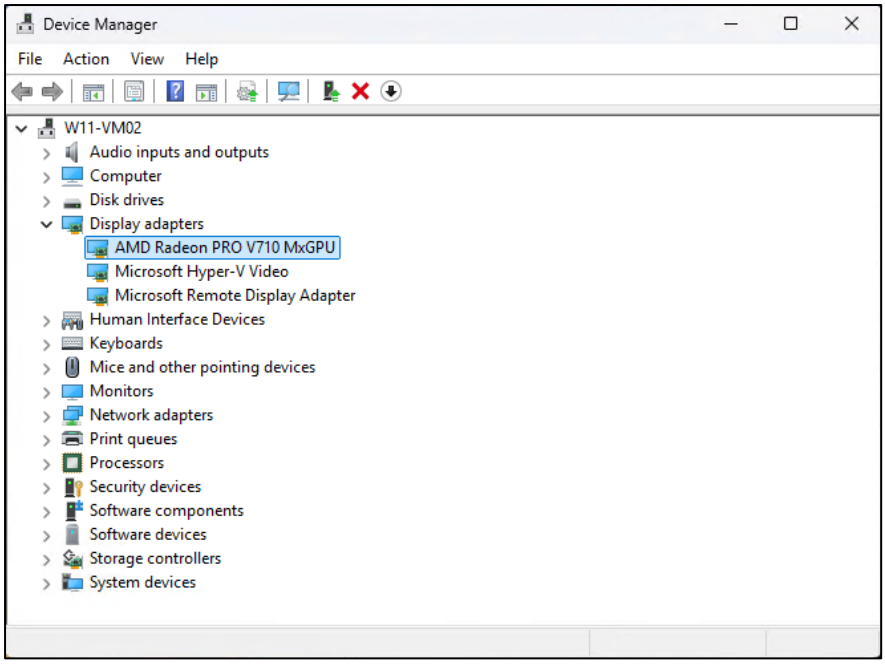
Refer to *Section 3.2 Assign GPU partitions to virtual machines* below to assign GPU partitions while the VMs are shut down. Then return here to install the GPU device driver in the VMs that have been assigned a GPU partition.

To install the GPU driver in a virtual machine, follow these steps:

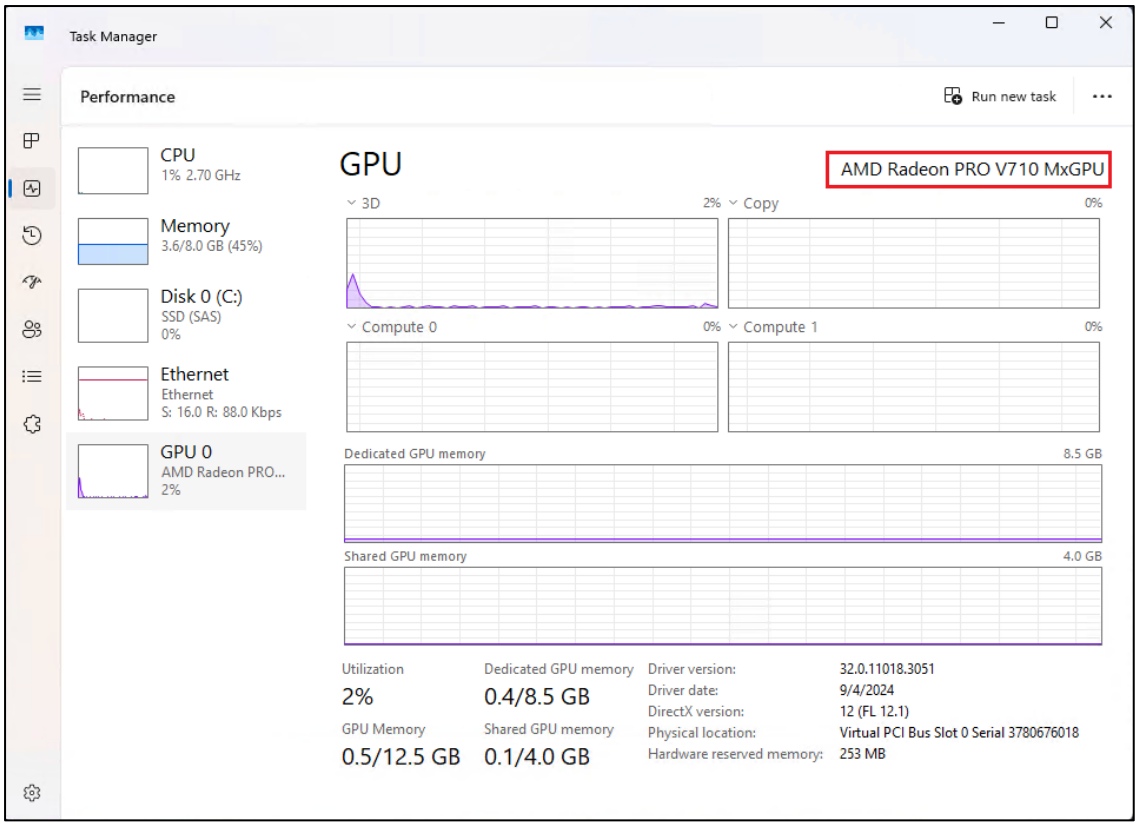
1. On each running VM that has been assigned a GPU partition, copy the AMD Radeon PRO V710 GPU guest VM driver installer to an easily accessible directory (such as **`C:\Downloads`**) and then run the installer.



2. After the device driver is installed, launch Device Manager to verify that the driver has been installed and is functioning properly, as shown in the following screenshot.




3. For additional verification, launch Task Manager and use the Performance tab on the left edge of the interface to verify that GPU performance is being shown, as shown in the following screenshot.



# 3 Create and assign GPU partitions

At this point, host GPUs can be partitioned for use by VMs. The high-level process involves creating the GPU partitions, assigning them to VMs, and then installing the GPU driver in the VMs so they can make use of their assigned partition.

To begin this process, the number of partitions required needs to be determined. It is not a good idea to assign all GPU partitions available on a host to VMs since this can cause issues with failover and VM Live Migration (i.e. there will be no partitions available on the destination host when a VM attempts to Live Migrate).

| High Density, High Performance VDI | | | |
|---|---|---|---|
| **Persona** | **Application Types** | **GPU Partitions** | **vRAM per Instance** |
| Knowledge Worker | Office, web-browsing, video calling | 8-12 | 3GB or 2GB |
| Professional Worker | Office 365, video conferencing, Autodesk Revit, Adobe | 6 | 4GB |
| Workstation Professional | Mainstream CAD and AEC | 3 | 8GB |
| High End Workstation Professional | CAD and AEC, advanced design visualization | 1-2 | 24GB or 12GB |
| Enthusiastic Gamer | AAA Games | 1x 4K60 | 24GB |
| Gamer | AA / AAA Games | Up to 4 | 6GB |
| Casual Gamer | Indy / Mobile Games | Up to 12 | 2GB |
| Renderer | Game Engine Rendering and Ray-Tracing | 1 | 24GB |
| Machine Learning | Single GPU Inference | 1 | 24GB |

## 3.1 Create GPU partitions

GPU partitions are created in the cluster node(s) in order to assign these partitions to VMs that they host. To create GPU partitions in the nodes, follow these steps:

1. Run the following commands to verify valid partitioning counts of the GPU(s) installed in each host:

```
$FormatEnumerationLimit=-1
Get-VMHostPartitionableGpu | FL Name,ValidPartitionCounts
```

   For the AMD Radeon PRO V710 GPU, there should be six valid partitioning counts, as shown in the following example.

```
PS C:\> $FormatEnumerationLimit=-1
PS C:\> Get-VMHostPartitionableGpu | fl Name, ValidPartitionCounts


Name                : \\?\PCI#VEN_1002&DEV_7460&SUBSYS_0E341002&REV_00#6&151eb934&0&00000009#{064092b3
                      -625e-43bf-9eb5-dc845897dd59}
ValidPartitionCounts : {12, 8, 6, 3, 2, 1}
```

2. Run the following command on each cluster node to create GPU partitions on all GPUs installed in the node, where "*<PartitionCount>*" is the number of partitions to create on each GPU in the host:

```
Set-VMHostPartitionableGpu -PartitionCount <PartitionCount>
```

The following example creates 6 partitions on each GPU installed in the host:

```
Get-VMHostPartitionableGpu | Set-VMHostPartitionableGpu  -PartitionCount 6
```

3. There is no output from the command above. Run the following command to verify the partition count on each GPU installed in the node:

```
Get-VMHostPartitionableGpu | FL Name,ValidPartitionCounts,PartitionCount
```

```
PS C:\> Set-VMHostPartitionableGpu -PartitionCount 6
PS C:\> Get-VMHostPartitionableGpu | fl Name,ValidPartitionCounts,PartitionCount


Name                : \\?\PCI#VEN_1002&DEV_7460&SUBSYS_0E341002&REV_00#6&151eb934&0&00000009#{064092b3
                      -625e-43bf-9eb5-dc845897dd59}
ValidPartitionCounts : {12, 8, 6, 3, 2, 1}
PartitionCount      : 6
```

Typically, the same number of partitions are created on each GPU installed in a host to balance the GPU workload. However, if you prefer to create different numbers of partitions on the GPUs installed in a host, use the **-Name** parameter in the command to define the number of partitions to create on each of the GPUs installed in the host (see the following example).

```
Set-VMHostPartitionableGpu -Name <GPUName> -PartitionCount <PartitionCount>
```

Note that the GPU name is from PCIe enumeration, which is a long and unfriendly string. Ensure that it is accurately captured and placed into the command. For reference, the GPU name shown in the partition verification command above is:

\\?\PCI#VEN_1002&DEV_7460&SUBSYS_0E341002&REV_00#6&151eb934&0&00000009#{064092b3-625e-43bf-9eb5-dc845897dd59}

Once GPU partitions are created on a host GPU, you can easily modify the number of partitions by rerunning this command with a different partition count value. However, all VMs currently using a partition on the host must be shutdown before changing the partition count.

## 3.2 Assign GPU partitions to virtual machines

Once GPU partitions have been created, these partitions can be assigned to virtual machines. GPU partition assignment is done while the VM is powered off. Once a partition is assigned, the VM will recognize the partition as if it had a physical GPU installed. This, in turn, will allow installation of a GPU device driver in the VM to be able to use the partition.

To assign a GPU partition to a virtual machine, follow these steps:

1. While the VM is powered off, run the following PowerShell commands to assign a GPU partition to it, where *<VMName>* is the name of the VM to which a GPU partition will be assigned:

```
Add-VMGpuPartitionAdapter -VMName <VMName>
```

```
SET-VM -VMName <VMName> -GuestControlledCacheType 1 -LowMemoryMappedIoSpace 1Gb -
HighMemoryMappedIoSpace 64Gb
```

Example:

```
Add-VMGpuPartitionAdapter -VMName "W11-VM01"
PS C:\> SET-VM -VMName "W11-VM01" -GuestControlledCacheType 1 -LowMemoryMappedIoSpace 1Gb -HighMemoryMappedIoSpace 64Gb
PS C:\>
```

2. As you can see in the example above, there is no output from the command. Start the target VM and once the VM is running, verify that a GPU partition is assigned to the VM using the following command, where <*VMName*> is the name of the VM to which a GPU partition was assigned:

```
Get-VMGpuPartitionAdapter -VMName <VMName> | fl InstancePath,PartitionId,PartitionVfLuid
```

Example:

```
PS C:\> Get-VMGpuPartitionAdapter -VMName "W11-VM01" | fl InstancePath,PartitionId,PartitionVfLuid


InstancePath    : \\?\PCI#VEN_1002&DEV_7460&SUBSYS_0E341002&REV_00#6&151eb934&0&00000009#{064092b3-625e-43bf-9eb5-dc845897dd59}
PartitionId     : 0
PartitionVfLuid : 02589211
```

The virtual machine now has full access and use of the GPU partition created for it. The last step is to install the GPU device driver in the VM. For these instructions, return to *Section 2.2 Install GPU device driver on virtual* machines above.

# 4 Manage GPUs using WAC

If Windows Admin Center (WAC) is used to manage Azure Local/S2D clusters, GPUs and GPU partitions can be viewed in WAC. For additional information about using WAC to manage servers, clusters, and their components, refer to the following Microsoft Learn article:

https://learn.microsoft.com/en-us/windows-server/manage/windows-admin-center/overview

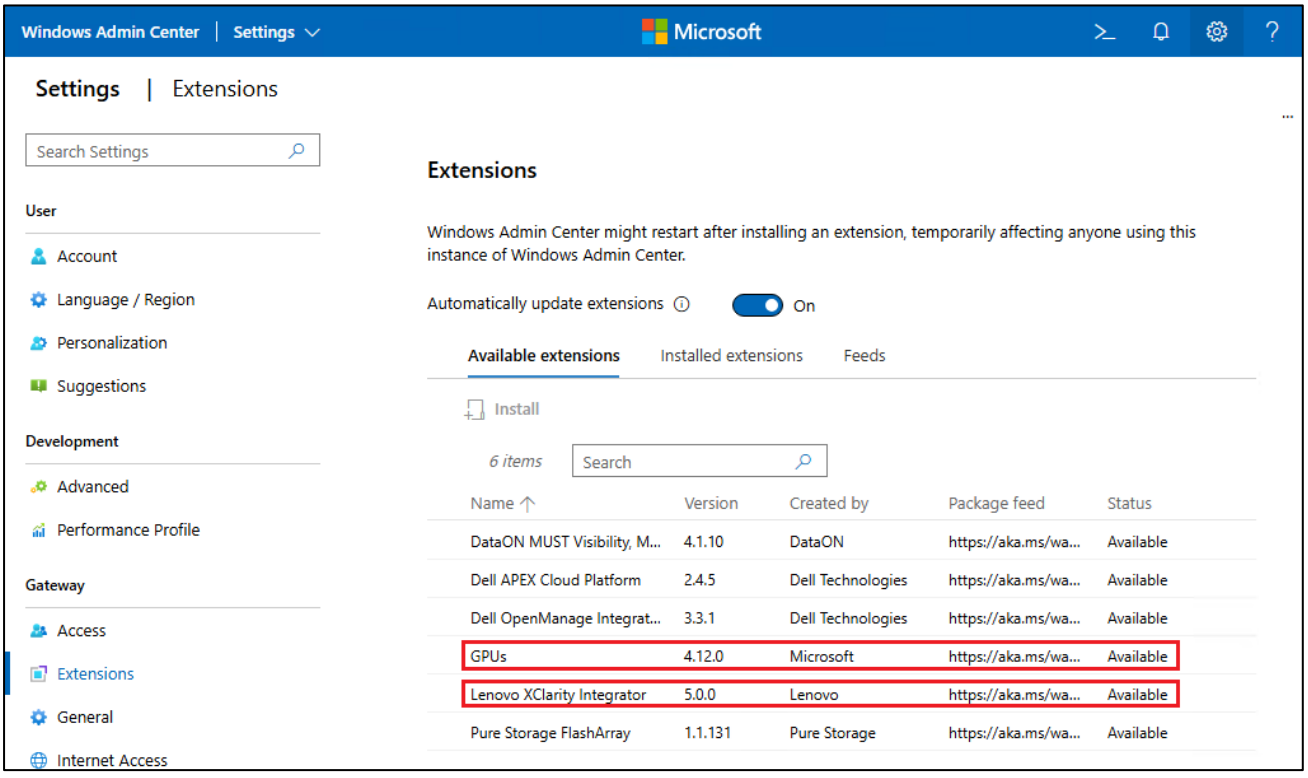
Note that although you can install and configure WAC in Azure, the Azure version of WAC is in Preview and currently does not support the GPUs extension. Further details related to running WAC in Azure can be found in the following Microsoft Learn article:

https://learn.microsoft.com/en-us/windows-server/manage/windows-admin-center/azure/manage-hci-clusters

## 4.1 Install desired WAC extensions

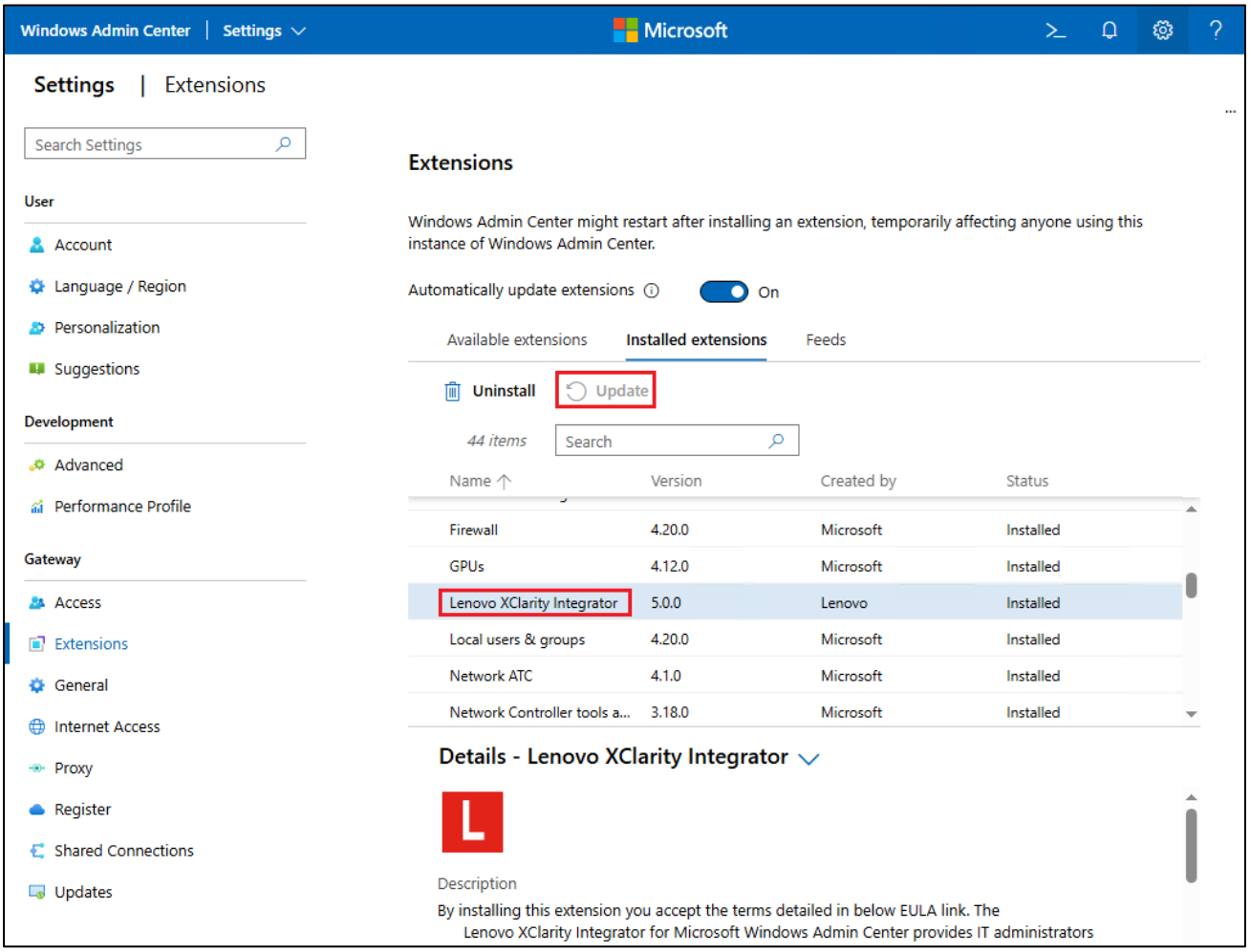
You will need to install the GPUs extension in WAC if not already done. In addition, we highly recommend installing the Lenovo XClarity Integrator extension for WAC. This extension provides significantly more system detail and management capabilities when viewing Lenovo servers. Any other desired WAC extensions can be installed at this time as well. To do this, follow these steps:

1. In WAC, navigate to Settings > Extensions.
2. If the GPUs and/or Lenovo XClarity Integrator extensions are shown in the list of Available extensions, select them one at a time and then click Install.




3. If the GPUs and/or Lenovo XClarity Integrator extensions are not shown in the list of Available

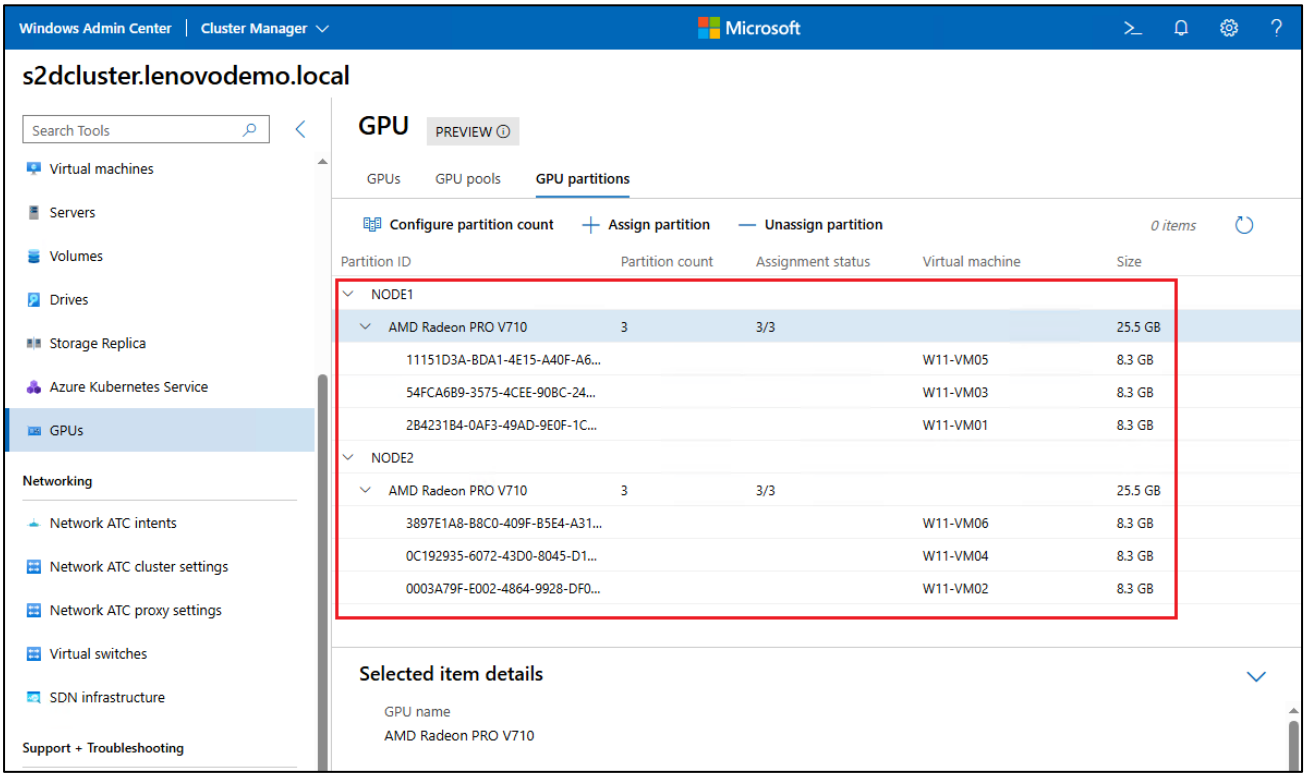
extensions, click the Installed extensions heading. Check to make sure the latest version of all extensions is installed. With the extension selected, the Update button should be grayed out.




## 4.2 Manage GPUs and partitions in WAC

Once the current version of the GPUs extension has been installed in WAC, the GPUs installed in the HCI cluster nodes can be managed and configured using WAC. This includes some basic GPU management functions, such as the ability to mount and unmount GPUs from the host, and to create GPU pools.

To verify that all GPUs have been identified in WAC, connect to the cluster in WAC and then use the left navigation pane to select the GPUs extension. Each node should be shown, including installed GPUs by name. The following screenshot shows our 2-node cluster with an AMD Radeon PRO V710 GPU installed in each of the nodes.




The environment is now ready for GPU partitions to be used for workloads running on the virtual machines in the HCI cluster.

# 5 Summary

GPU virtualization technologies enable GPU acceleration in a virtualized environment, typically within virtual machines. If a workload is virtualized with Hyper-V, then graphics virtualization can be employed in order to provide GPU acceleration from the physical GPU to the virtualized apps or services. In order for a virtual machine to use a GPU installed in its Hyper-V host, several tasks must be accomplished. This document has provided the steps used to perform the following tasks:

- Install the GPU device driver in the host
- Create GPU partitions using PowerShell
- Assign GPU partitions to VMs using PowerShell
- Install GPU device driver in VMs that have been assigned a GPU partition
- Use the WAC GPUs extension to manage GPUs and their partitions

# 6 Authors

This paper was produced by the following specialists:

**Dave Feisthammel** is a Senior Solutions Architect working at the Lenovo Center for Microsoft Technologies in Bellevue, Washington. He has over 30 years of experience in the IT field, including four years as an IBM client and over 24 years working for IBM and Lenovo. His areas of expertise include Windows Server and systems management, as well as virtualization, storage, and cloud technologies. He is currently a key contributor to Lenovo solutions related to Microsoft Azure Local and Azure Stack Hub.

**David Ye** is a Principal Solutions Architect at Lenovo with over 25 years of experience in the IT field. He started his career at IBM as a Worldwide Windows Level 3 Support Engineer. In this role, he helped customers solve complex problems and critical issues. He is now working in the Lenovo Infrastructure Solutions Group, where he works with customers on Proof-of-Concept designs, solution sizing and reviews, and performance optimization. His areas of expertise are Windows Server, SAN Storage, Virtualization and Cloud, and Microsoft Exchange Server. He is currently leading the effort in Microsoft Azure Local and Azure Stack Hub solutions development.

# 7 Additional resources

The following resources might be useful in working with Lenovo ThinkAgile MX solutions.

Resources for Lenovo ThinkAgile MX Series solutions
https://lenovopress.com/servers/thinkagile/mx-series

Lenovo Press document: **Microsoft Storage Spaces Direct (S2D) Deployment Guide**
https://lenovopress.com/lp0064

Lenovo Press document: **Lenovo Certified Configurations for Microsoft Azure Local – V1 Servers**
https://lenovopress.com/lp0866

Lenovo Press document: **Lenovo Certified Configurations for Microsoft Azure Local – V2 Servers**
https://lenovopress.com/lp1520

Lenovo Press document: **Lenovo Certified Configurations for Microsoft Azure Local – V3 Servers**
https://lenovopress.com/lp1741

Lenovo Press document: **Lenovo Certified Configurations for Microsoft Azure Local – Edge Servers**
https://lenovopress.com/lp1984

Lenovo ThinkAgile MX Best Recipe landing page
https://datacentersupport.lenovo.com/us/en/solutions/HT507406

Lenovo ThinkAgile MX Series
https://www.lenovo.com/au/en/data-center/software-defined-infrastructure/ThinkAgile-ThinkAgile MX-Certified-Node/p/WMD00000377

Microsoft article: *GPU partitioning*
https://learn.microsoft.com/en-us/windows-server/virtualization/hyper-v/gpu-partitioning

Microsoft article: *Plan for GPU acceleration in Windows Server*
https://docs.microsoft.com/en-us/windows-server/virtualization/hyper-v/plan/plan-for-gpu-acceleration-in-windows-server

Microsoft article: *Use GPUs with clustered VMs*
https://learn.microsoft.com/en-us/windows-server/virtualization/hyper-v/deploy/use-gpu-with-clustered-vm

Microsoft article: *Partition and assign GPUs to a virtual machine*
https://learn.microsoft.com/en-us/windows-server/virtualization/hyper-v/partition-assign-vm-gpu?tabs=powershell

# 8 Trademarks and special notices



References in this document to Lenovo products or services do not imply that Lenovo intends to make them available in every country.

Lenovo, the Lenovo logo, ThinkSystem, ThinkCentre, ThinkVision, ThinkVantage, ThinkPlus and Rescue and Recovery are trademarks of Lenovo.

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Intel, Intel Inside (logos), and Pentium are trademarks of Intel Corporation in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.

Information is provided "AS IS" without warranty of any kind.

All customer examples described are presented as illustrations of how those customers have used Lenovo products and the results they may have achieved. Actual environmental costs and performance characteristics may vary by customer.

Information concerning non-Lenovo products was obtained from a supplier of these products, published announcement material, or other publicly available sources and does not constitute an endorsement of such products by Lenovo. Sources for non-Lenovo list prices and performance numbers are taken from publicly available information, including vendor announcements and vendor worldwide homepages. Lenovo has not tested these products and cannot confirm the accuracy of performance, capability, or any other claims related to non-Lenovo products. Questions on the capability of non-Lenovo products should be addressed to the supplier of those products.

All statements regarding Lenovo future direction and intent are subject to change or withdrawal without notice and represent goals and objectives only. Contact your local Lenovo office or Lenovo authorized reseller for the full text of the specific Statement of Direction.

Some information addresses anticipated future capabilities. Such information is not intended as a definitive statement of a commitment to specific levels of performance, function or delivery schedules with respect to any future products. Such commitments are only made in Lenovo product announcements. The information is presented here to communicate Lenovo’s current investment and development activities as a good faith effort to help with our customers' future planning.

Performance is based on measurements and projections using standard Lenovo benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual

user will achieve throughput or performance improvements equivalent to the ratios stated here.

Photographs shown are of engineering prototypes. Changes may be incorporated in production models.

Any references in this information to non-Lenovo websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this Lenovo product and use of those websites is at your own risk.