

Configure DCPMM in Windows Server with AppDirect Mode

Contents

[Introduction](#)

[Prerequisites](#)

[Requirements](#)

[Components Used](#)

[Background Information](#)

[Data Center Persistent Memory Module](#)

[Modes of Operation](#)

[Memory Mode](#)

[AppDirect Mode](#)

[Combo Mode](#)

[Goal](#)

[Region](#)

[Namespace](#)

[Direct Access](#)

[Configure](#)

[Verify](#)

[Troubleshoot](#)

[Reference](#)

Introduction

This document describes the Intel® Optane™ Data Center persistent memory (PMEM) configuration in AppDirect mode for Windows Server.

Contributed by Ana Montenegro, Cisco TAC Engineer.

Prerequisites

Requirements

Cisco recommends that you have knowledge of these topics:

- Intel® Optane™ Data Center persistent memory module (DCPMM).
- Windows Server Administration.

Ensure the server has the minimum requirements before you attempt this configuration:

- Refer to the PMEM guidelines on the B200/B480 M5 [specification guide](#).
- Ensure the CPU is second generation Intel® Xeon® Scalable processors.

Components Used

The information in this document is based on these software and hardware versions:

- UCS B480 M5
- UCS Manager 4.1(2a)
- Windows Server 2019

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, ensure that you understand the potential impact of any command.

Background Information

Cisco IMC and Cisco UCS Manager Release 4.0(4) introduce support for the Intel[®] Optane[™] Data Center persistent memory modules on the UCS M5 servers that are based on the Second Generation Intel[®] Xeon[®] Scalable processors.

Data Center Persistent Memory Module

Data Center Persistent Memory Module (DCPMM) is a new technology that bridges the gap between storage and traditional memory. It achieves the best of both worlds by combining the high-speed performance of DRAM and the high capacity of traditional storage. They have high performance than SSDs and lower cost per gigabyte than system memory.

Modes of Operation

- **Memory Mode**

In Memory Mode, the DDR4 acts as a cache module for the DCPMMs. It provides a large memory capacity, although data is volatile. The operating system sees the persistent memory module capacity as the system's main memory.

- **AppDirect Mode**

All memory used as storage. Memory is byte-addressable and provides direct load/store access without any modifications to the existing applications or the file systems. App Direct Mode provides high-performance block storage, without the latency of moving data to and from the I/O bus.

- **Combo Mode**

This mode allows the usage of the module with 25% capacity used as volatile memory and 75% as non-volatile memory.

Switching between modes is possible via UCSM as well as OS tools on the host.

Goal

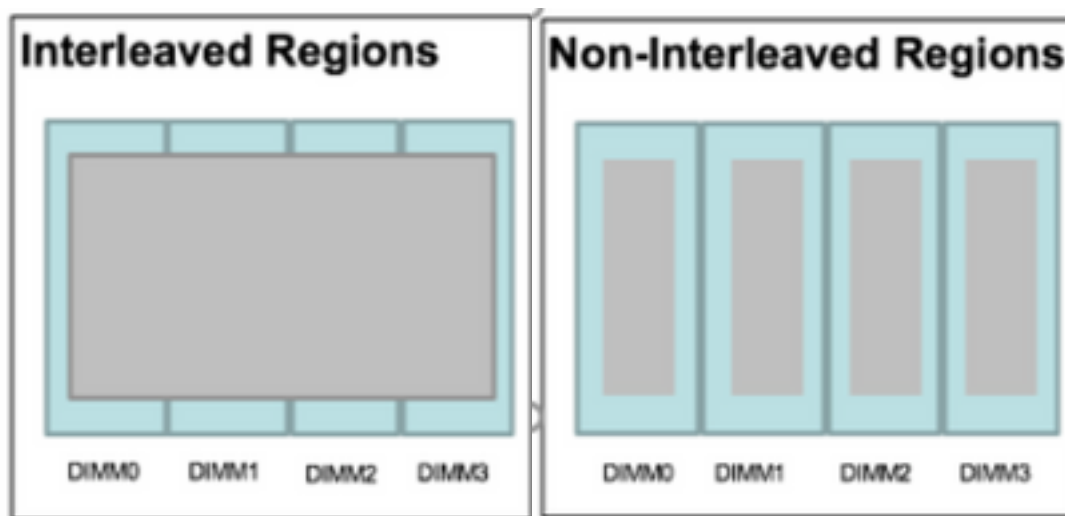
A goal is used to configure how persistent memory modules connected to a CPU socket are used.

- App Direct configures one region for all the persistent memory modules connected to a socket.
- App Direct Non interleaved configures one region for each persistent memory module.

Region

A region is a group of one or more persistent memory modules that can be divided up into one or more namespaces. A region is created based on the persistent memory type selected during goal creation.

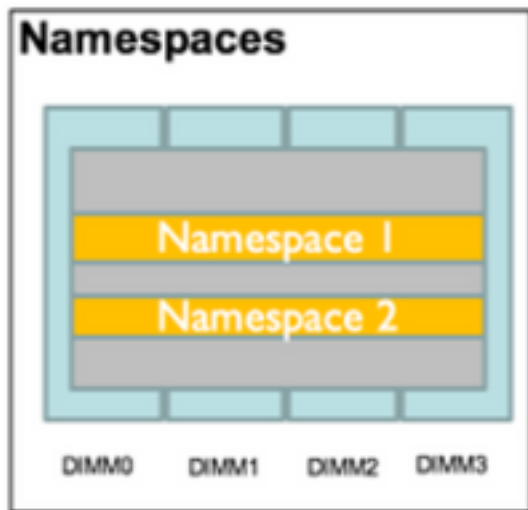
Regions can be created as either non-interleaved, meaning one region per persistent memory module, or interleaved, which creates one large region over all modules in a CPU socket. Regions cannot be created across CPU sockets.



Namespace

A namespace is a partition of a region. When using the App Direct persistent memory type, you can create namespaces on the region mapped to the socket. When using the App Direct Non Interleaved persistent memory type, you can create namespaces on the region mapped to a specific memory module on the socket.

A namespace can be created in Raw or Block mode. A namespace created in Raw mode is seen as a raw mode namespace in the host OS. A namespace created in Block mode is seen as a sector mode namespace in the host OS.



Direct Access

Direct access (DAX) is a mechanism that allows applications to directly access the persistent media from the CPU (through loads and stores), bypassing the traditional I/O stack (page cache and block layer).

Configure

1. Create a PMEM policy

Navigate to **Servers > Persistent Memory Policy** and click on **Add**.

Create a **Goal**, ensure the Memory Mode is 0%.

Create Persistent Memory Policy

Properties

Name : AppDirect_PMEM

Description :

General

Security

Goals

Advanced

Socket

All S

Config

Advanced Filter

Export

Print

Name

Socket Id

Socket Local DIMM ...

Mode

Capacity (GiB)

No data available

Create Goal

Properties

Socket ID : All Sockets

Memory Mode (%) :

Persistent Memory Type : App Direct App Direct Non Interleaved

OK

Cancel

OK

Cancel

Create Persistent Memory Policy



Properties

Name :

Description :

General

Security

Goals

Advanced Filter Export Print

Socket Id	Memory Mode (%)	Persistent Memory Type
All Sockets	0	App Direct

Add Delete Modify

Configure Namespace

Advanced Filter Export Print

Name	Socket Id	Socket Local DIMM ...	Mode	Capacity (GiB)
No data available				

OK

Cancel

Note: If you include a persistent memory policy in a service profile associated to a server, the persistent memory configuration on the server is **UCS-managed**. In the **UCS-managed** mode, you can use Cisco UCS Manager and host tools to configure and manage persistent memory modules, otherwise, the persistent memory configuration on the server is **host-managed**. In the **host-managed** mode, you can use the host tools to configure and manage persistent memory modules.

2. Assign the persistent memory policy to the service profile.

Navigate to **Service Profile > Policies > Persistent Memory Policy** and select the policy previously created

Caution: This action requires a server reboot

- [+ IPMI/Redfish Access Profile Policy](#)
- [+ Power Control Policy](#)
- [+ Scrub Policy](#)
- [+ Serial over LAN Policy](#)
- [+ Stats Policy](#)
- [+ KVM Management Policy](#)
- [+ Power Sync Policy](#)
- [+ Graphics Card Policy](#)
- [- Persistent Memory Policy](#)

Persistent Memory Policy : AppDirect_PMEM [Create Persistent Memory Policy](#)

Persistent Memory Policy Instance :

OK
Apply
Cancel
Help

3. (Optional) Verify the mode is **AppDirect**.

Navigate to **Server > Inventory > Persistent Memory > Regions**.

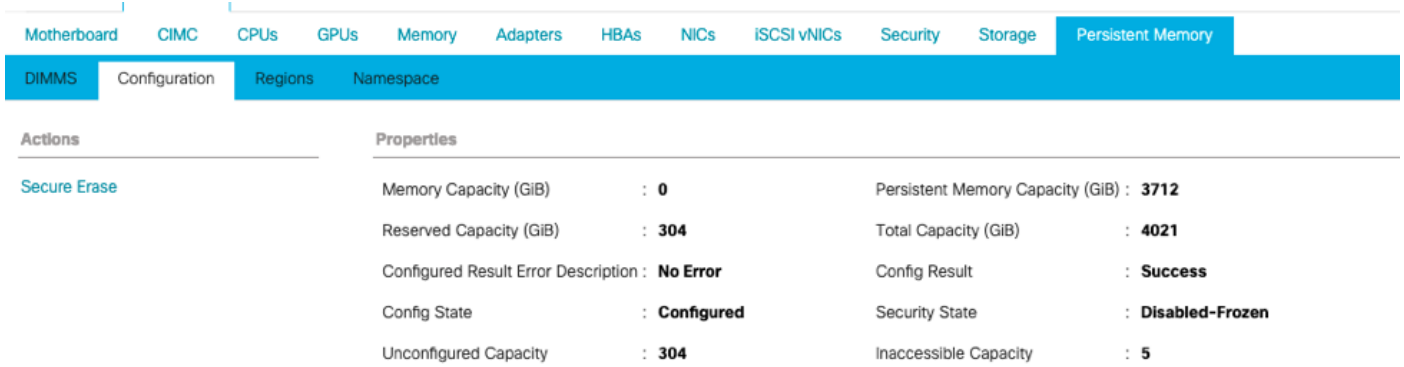
[General](#)
[Inventory](#)
[Virtual Machines](#)
[Installed Firmware](#)
[CIMC Sessions](#)
[SEL Logs](#)
[VIF Paths](#)
[Health](#)
[Diagnostics](#)
[Faults](#)
[Events](#)
[FSM](#)

[Motherboard](#)
[CIMC](#)
[CPUs](#)
[GPUs](#)
[Memory](#)
[Adapters](#)
[HBAs](#)
[NICs](#)
[iSCSI vNICs](#)
[Security](#)
[Storage](#)
[Persistent Memory](#)

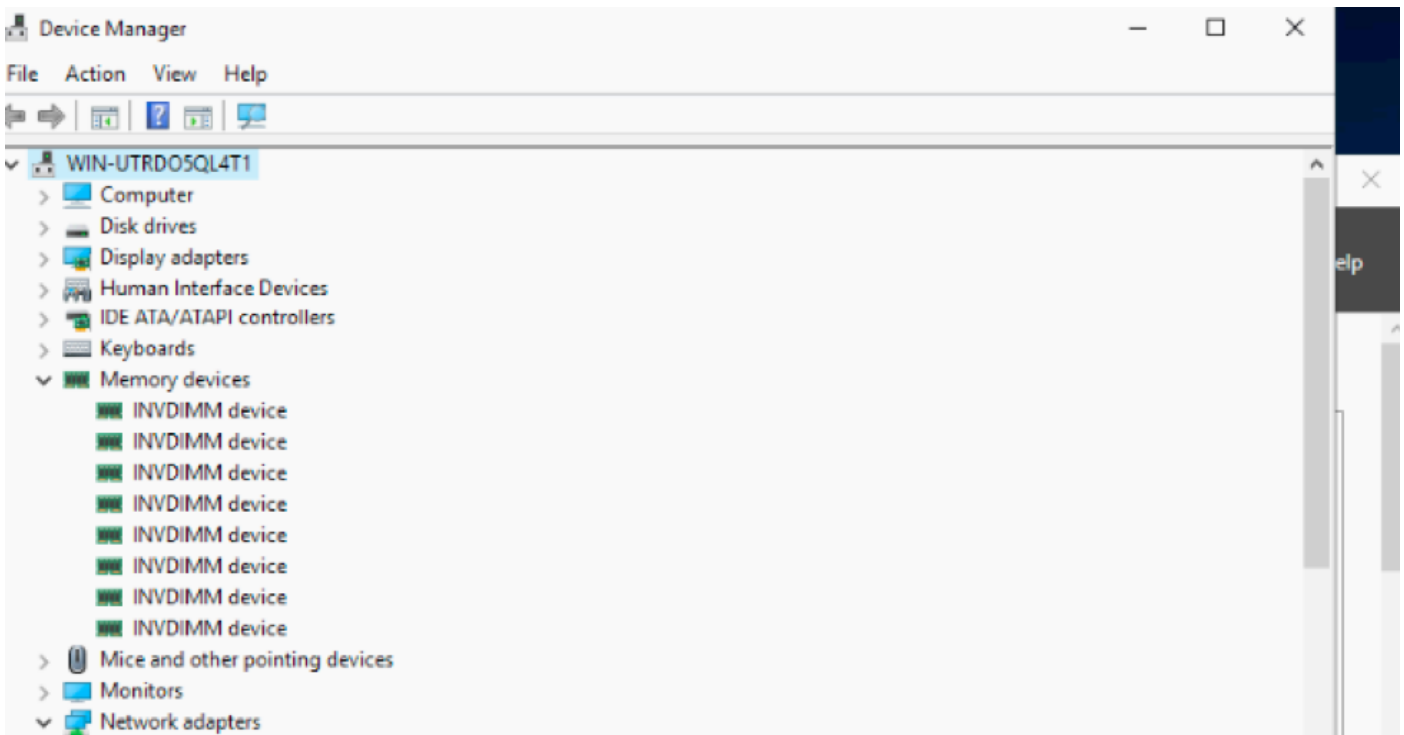
[DIMMS](#)
[Configuration](#)
[Regions](#)
[Namespace](#)

[Advanced Filter](#)
[Export](#)
[Print](#)

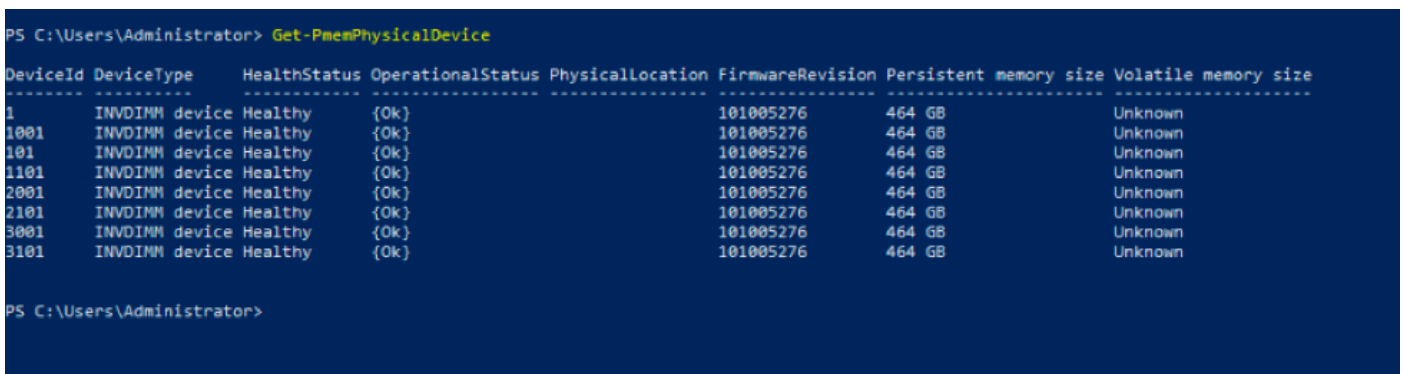
	Socket Id	Local DIMM Slot Id	DIMM Locator Ids	Type	Total Capacity (GiB)	Free Capacity (GiB)	Health Status:
1	Socket 1	Not Applicable	DIMM_A2,DIMM_D2	AppDirect	928	928	Healthy
2	Socket 2	Not Applicable	DIMM_G2,DIMM_K2	AppDirect	928	928	Healthy
3	Socket 3	Not Applicable	DIMM_N2,DIMM_R2	AppDirect	928	928	Healthy
4	Socket 4	Not Applicable	DIMM_U2,DIMM_X2	AppDirect	928	928	Healthy



4. In Windows navigate to **Device Manager > Memory devices** to see the memories.



5. Use PowerShell to verify the memory physical status with the command **Get-PmemPhysicalDevice**.



6. Use the command **Get-PmemUnusedRegion** to return the regions that are available to be assigned to a logical persistent memory device on the system.


```
PS C:\Users\Administrator> Get-PmemUnusedRegion

RegionId TotalSizeInBytes DeviceId
-----
1        996432412672 {1, 101}
3        996432412672 {1001, 1101}
4        996432412672 {2001, 2101}
5        996432412672 {3001, 3101}

PS C:\Users\Administrator> .
```

7. Use the command **New-PmemDisk** to create a namespace on a region in order to enable the capacity.

The namespace is visible to the Windows operating system and can be used by applications.

```
PS C:\Users\Administrator> Get-PmemUnusedRegion | New-PmemDisk
Creating new persistent memory disk. This may take a few moments.
Creating new persistent memory disk. This may take a few moments.
Creating new persistent memory disk. This may take a few moments.
Creating new persistent memory disk. This may take a few moments.
```

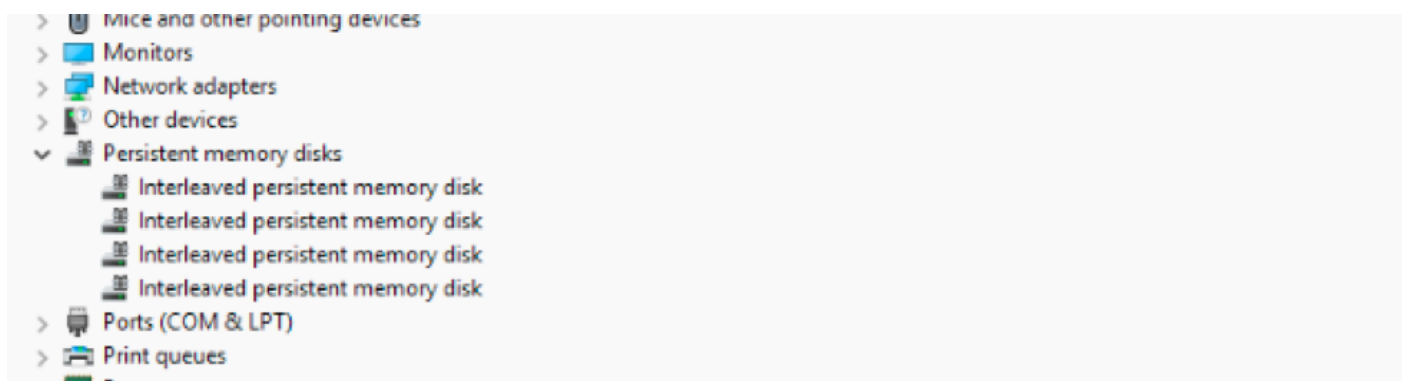
8. Verify with the command **Get-PmemDisk** Persistent Memory Disk (Namespace).

```
PS C:\Users\Administrator> Get-pmemdisk

DiskNumber Size HealthStatus AtomicityType CanBeRemoved PhysicalDeviceIds UnsafeShutdownCount
-----
4          928 GB Healthy None True {1, 101} 0
5          928 GB Healthy None True {1001, 1101} 0
6          928 GB Healthy None True {2001, 2101} 0
7          928 GB Healthy None True {3001, 3101} 0

PS C:\Users\Administrator>
```

9. (Optional) Navigate to the **Device Manager** and check the persistent memory disk under the **Persistent memory disk**.

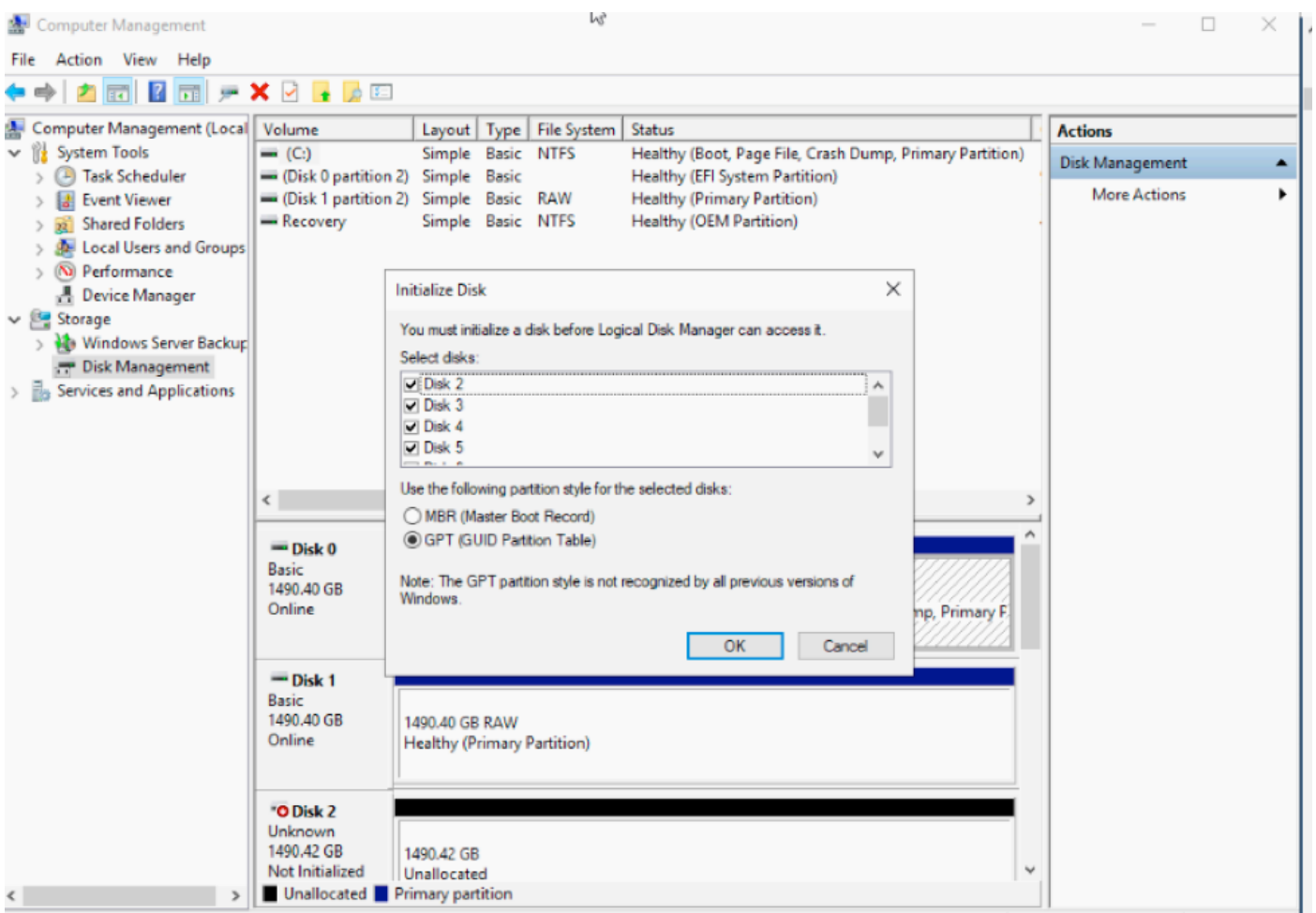


10. In UCS Manager you see the **namespace** created under the regions.

Navigate to **Server > Inventory > Persistent memory > Namespace** you see the Regions with the namespace attached.

General				Inventory	Virtual Machines	Installed Firmware	CIMC Sessions	SEL Logs	VIF Paths	Health	Diagnostics	Faults	Events	FSM			
Motherboard				CIMC	CPUs	GPUs	Memory	Adapters	HBAs	NICs	ISCSI vNICs	Security	Storage	Persistent Memory			
DIMMS				Configuration	Regions	Namespace											
+ - Advanced Filter				Export		Print										⚙️	
Name	Mode	Capacity (GiB)	Health Status:														
▼ Region 1																	
Namespace PmemDisk1	Raw	928	Healthy														
▼ Region 2																	
Namespace PmemDisk1	Raw	928	Healthy														
▼ Region 3																	
Namespace PmemDisk1	Raw	928	Healthy														
▼ Region 4																	
Namespace PmemDisk1	Raw	928	Healthy														

11. In Windows, navigate to the **Disk Management** console to view the new disk. initialize the disk using **MBR** or **GPT** partitioning before the logical disk manager can access it.



Verify

There is currently no verification procedure available for this configuration.

Troubleshoot

1. The command **Remove-PmemDisk** removes a specific persistent memory disk, which can be used if you have to replace a failed module.

```
PS C:\Users\Administrator> Get-pmemdisk 4 | Remove-PmemDisk

This will remove the persistent memory disk(s) from the system and will result in data loss.
Remove the persistent memory disk(s)?
[Y] Yes [A] Yes to All [N] No [L] No to All [S] Suspend [?] Help (default is "Y"): Y
Removing the persistent memory disk. This may take a few moments.
```

Caution: Remove a persistent memory disk cause data loss on that disk.

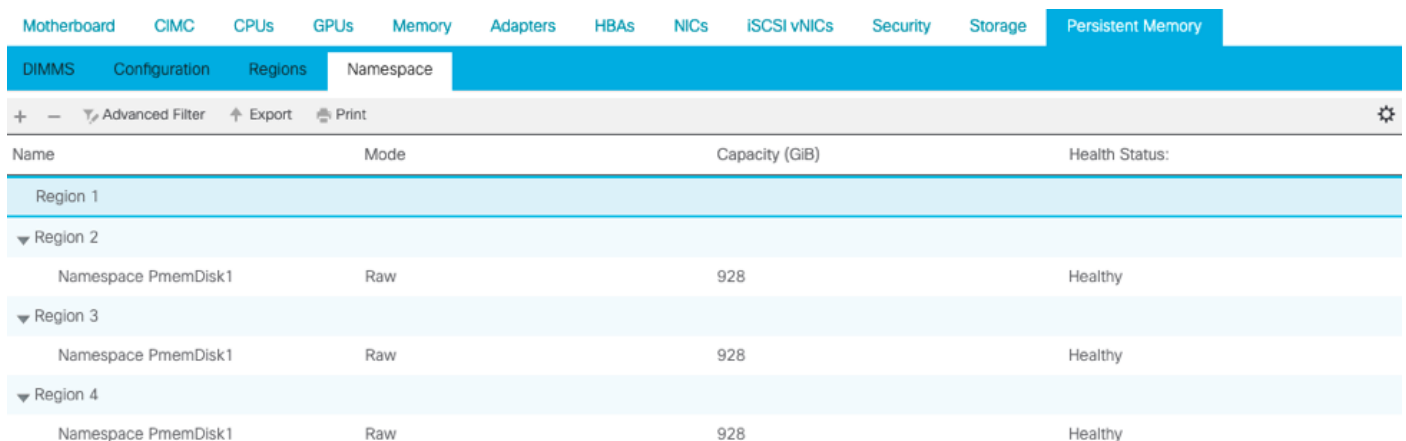
2. Verify with the command **Get-PmemDisk** the remaining available persistent memory disk.

```
PS C:\Users\Administrator> Get-PmemDisk

DiskNumber Size HealthStatus AtomicityType CanBeRemoved PhysicalDeviceIds UnsafeShutdownCount
-----
4 928 GB Healthy None True {1001, 1101} 0
5 928 GB Healthy None True {2001, 2101} 0
6 928 GB Healthy None True {3001, 3101} 0

PS C:\Users\Administrator>
```

3. In UCS Manager under **Persistent Memory** you see the region no longer has the namespace assigned as shown in the image.



Name	Mode	Capacity (GiB)	Health Status:
Region 1			
Region 2			
Namespace PmemDisk1	Raw	928	Healthy
Region 3			
Namespace PmemDisk1	Raw	928	Healthy
Region 4			
Namespace PmemDisk1	Raw	928	Healthy

4. Alternatively, use the **IPMCTL** utility to configure and manage the Intel Optane DC persistent memory modules.

Note: **IPMCTL** can be launched from a Unified Extensible Firmware Interface (UEFI) shell or a terminal window in an operating system.

5. The **ipmctl show -dimm** command displays the persistent memory modules discovered in the system and verifies that software can communicate with them. Among other information, this command outputs each DIMM ID, capacity, health state, and firmware version.

```
Shell> ipmctl show -dimm
DimmID | Capacity | LockState | HealthState | FWVersion
=====
0x0001 | 502.5 GiB | Disabled, Frozen | Healthy | 01.01.00.5276
0x0101 | 502.5 GiB | Disabled, Frozen | Healthy | 01.01.00.5276
0x1001 | 502.5 GiB | Disabled, Frozen | Healthy | 01.01.00.5276
0x1101 | 502.5 GiB | Disabled, Frozen | Healthy | 01.01.00.5276
0x2001 | 502.5 GiB | Disabled, Frozen | Healthy | 01.01.00.5276
0x2101 | 502.5 GiB | Disabled, Frozen | Healthy | 01.01.00.5276
0x3001 | 502.5 GiB | Disabled, Frozen | Healthy | 01.01.00.5276
0x3101 | 502.5 GiB | Disabled, Frozen | Healthy | 01.01.00.5276
Shell> _
```

6. The **ipmctl show -memoryresources** command displays the capacity provisioned.

```
Shell> ipmctl show -memoryresources
Capacity=3.9 TiB
MemoryCapacity=0 B
AppDirectCapacity=3.6 TiB
UnconfiguredCapacity=0 B
InaccessibleCapacity=4.7 GiB
ReservedCapacity=304.0 GiB
Shell> _
```

7. The **ipmctl show -region** command displays available regions, you see the region 1 has **free capacity**.

```
Shell> ipmctl show -region
RegionID | SocketID | PersistentMemoryType | Capacity | FreeCapacity | HealthState
=====
====
0x0001 | 0x0000 | AppDirect | 928.0 GiB | 928.0 GiB | Healthy
0x0002 | 0x0001 | AppDirect | 928.0 GiB | 0 B | Healthy
0x0003 | 0x0002 | AppDirect | 928.0 GiB | 0 B | Healthy
0x0004 | 0x0003 | AppDirect | 928.0 GiB | 0 B | Healthy
Shell>
```

8. The command **ipmctl create -namespace** create a namespace in the available regions.

```

Shell> ipmctl create -namespace -region 1
Current namespace configuration
---NamespaceId=0x0101---
  HealthState=Healthy
  Name=
  Capacity=928.0 GiB
  RegionID=1
  BlockSize=4096 B
  Mode=None
  LabelVersion=1.2
  NamespaceGuid=20428566-F645-43F3-A788-20032C6E9A7C
Shell> _

```

9. Now all the regions are assigned to the namespace, as shown in the image

```

Shell> ipmctl show -region
  RegionID | SocketID | PersistentMemoryType | Capacity | FreeCapacity | HealthSt
ate
=====
====
  0x0001  | 0x0000  | AppDirect            | 928.0 GiB | 0 B          | Healthy
  0x0002  | 0x0001  | AppDirect            | 928.0 GiB | 0 B          | Healthy
  0x0003  | 0x0002  | AppDirect            | 928.0 GiB | 0 B          | Healthy
  0x0004  | 0x0003  | AppDirect            | 928.0 GiB | 0 B          | Healthy
Shell> _

```

10. In UCS manager we can check the namespace created under **Persistent Memory** as shown in the image.

Name	Mode	Capacity (GiB)	Health Status:
▼ Region 1			
Namespace	Raw	928	Healthy
▼ Region 2			
Namespace PmemDisk1	Raw	928	Healthy
▼ Region 3			
Namespace PmemDisk1	Raw	928	Healthy
▼ Region 4			
Namespace PmemDisk1	Raw	928	Healthy

Note: Review all the available commands for IPMCTL : [IPMCTL User Guide](#)

Reference

- [UCSM Configuring and Managing DC Persistent Memory Modules](#)
- [Quick Start Guide: Provision Intel® Optane™ DC Persistent Memory](#)
- [Windows Server: Understand and deploy persistent memory](#)
- [Technical Support & Documentation - Cisco Systems](#)