

# Konfigurieren von DCPMM in VMware ESXi für den AppDirect-Modus

## Inhalt

[Einführung](#)

[Voraussetzungen](#)

[Anforderungen](#)

[Verwendete Komponenten](#)

[Hintergrundinformationen](#)

[Konfiguration](#)

[Konfigurieren des Serviceprofils](#)

[ESXi überprüfen](#)

[Konfigurieren von NVDIMMs für virtuelle Systeme](#)

[Konfigurieren des Namespaces im virtuellen System](#)

[Fehlerbehebung](#)

[Zugehörige Informationen](#)

## Einführung

Dieses Dokument beschreibt den Prozess zur Konfiguration von ESXi auf Servern der UCS B-Serie mit Intel® Optane™ Persistent Memory (PMEM) im verwalteten Host-Modus.

## Voraussetzungen

### Anforderungen

Cisco empfiehlt, dass Sie über Kenntnisse in folgenden Bereichen verfügen:

- UCS B-Serie
- Konzepte des Intel® Optane™ Data Center Persistent Memory Module (DCPMM)
- VMware ESXi- und vCenter-Serveradministration

Stellen Sie sicher, dass Sie diese Anforderungen erfüllen, bevor Sie versuchen, diese Konfiguration durchzuführen:

- Siehe die PMEM-Richtlinien im B200/B480 M5-[Spezifikationsleitfaden](#).
- Stellen Sie sicher, dass es sich bei der CPU um Intel® Xeon® skalierbare Prozessoren der zweiten Generation handelt.
- Das PMEM/Dynamic Random Access Memory (DRAM)-Verhältnis erfüllt die Anforderungen gemäß [KB 67645](#).
- ESXi ist 6,7 U2 + Express Patch 10 (ESXi670-201906002) oder höher. Frühere 6.7-Versionen werden nicht unterstützt.
- UCS Manager und Server sind ab Version 4.0(4) erhältlich. Die aktuelle empfohlene Version

finden Sie unter [www.software.cisco.com/](http://www.software.cisco.com/).

## Verwendete Komponenten

Die Informationen in diesem Dokument basierend auf folgenden Software- und Hardware-Versionen:

- UCS B480 M5
- UCS Manager 4.1(2b)

Die Informationen in diesem Dokument beziehen sich auf Geräte in einer speziell eingerichteten Testumgebung. Alle Geräte, die in diesem Dokument benutzt wurden, begannen mit einer gelöschten (Nichterfüllungs) Konfiguration. Wenn Ihr Netzwerk in Betrieb ist, stellen Sie sicher, dass Sie die potenziellen Auswirkungen eines Befehls verstehen.

## Hintergrundinformationen

In für den App Direct-Modus konfigurierten UCS-Servern können virtuelle VMware ESXi-Systeme auf optische DCPMM-Module mit permanentem Speicher zugreifen, ohne dass Volatile Dual-In-Line Memory-Module (NVDIMMs) erforderlich sind.

Intel Optane DCPMM kann über das IPMCTL-Verwaltungsprogramm über die Unified Extensible Firmware Interface (UEFI)-Shell oder über die Betriebssystemdienstprogramme konfiguriert werden. Dieses Tool wurde entwickelt, um einige der folgenden Aktionen durchzuführen:

- Erkennung und Verwaltung von Modulen
- Aktualisieren und Konfigurieren der Modulfirmware
- Überwachung des Systemstatus
- Bereitstellen und Konfigurieren von Zielen, Regionen und Namespaces
- Fehlerbehebung und Fehlerbehebung für PMEM

Zur Vereinfachung der Verwendung kann das UCS mithilfe einer persistenten Speicherrichtlinie konfiguriert werden, die mit dem Serviceprofil verknüpft ist.

Das Open-Source Non-Volatile Device Control (NDCTL)-Dienstprogramm wird zur Verwaltung des LIBNVDIMM Linux-Kernel-Subsystems verwendet. Das NDCTL-Dienstprogramm ermöglicht einem System die Bereitstellung und Ausführung von Konfigurationen als Regionen und Namespaces für die Betriebssystemnutzung.

Der einem ESXi-Host hinzugefügte persistente Arbeitsspeicher wird vom Host erkannt, formatiert und als lokaler PMem-Datenspeicher bereitgestellt. Zur Verwendung des PMEM verwendet ESXi das Dateisystemformat Virtual Machine Flying System (VMFS)-L, und pro Host wird nur ein lokaler PMEM-Datenspeicher unterstützt.

Anders als andere Datenspeicher unterstützt der PMEM-Datenspeicher keine Aufgaben als herkömmliche Datenspeicher. Das VM-Stammverzeichnis mit den Dateien vmx und vmware.log kann nicht im PMEM-Datenspeicher gespeichert werden.

PMEM kann einer VM in zwei verschiedenen Modi präsentiert werden: Direktzugriffsmodus und Virtual Disk-Modus.

- Direktzugriffsmodus

Für diesen Modus können virtuelle Systeme konfiguriert werden, indem die PMEM-Region in Form eines NVDIMMs angezeigt wird. VM-Betriebssystem muss PMem-kompatibel sein, um diesen Modus verwenden zu können. Daten, die auf NVDIMM-Modulen gespeichert sind, können über Betriebszyklen hinweg fortbestehen, da das NVDIMM als Byte-adressierbarer Speicher fungiert. NVDIMMs werden beim Formatieren des PMEM automatisch im vom ESXi erstellten PMem-Datenspeicher gespeichert.

- Virtueller Festplattenmodus

Für traditionelle und ältere Betriebssysteme auf VM vorgesehen, um alle Hardwareversionen zu unterstützen. Das VM-Betriebssystem muss nicht PMEM-fähig sein. In diesem Modus kann eine herkömmliche virtuelle Small Computer System Interface (SCSI)-Festplatte erstellt und vom VM-Betriebssystem verwendet werden.

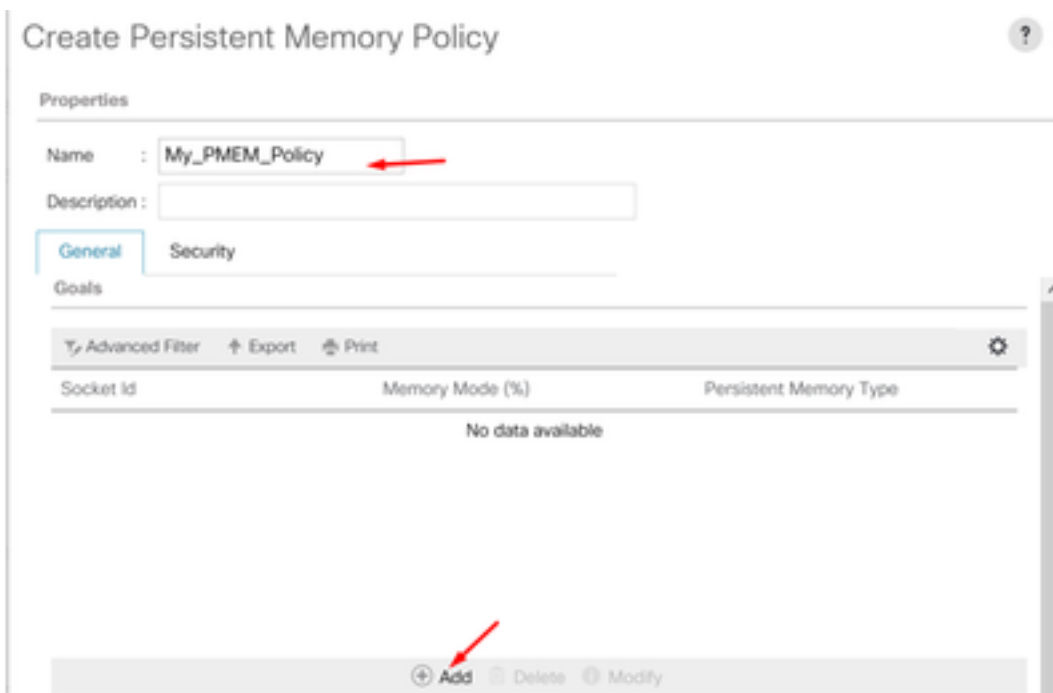
Dieses Dokument beschreibt die Konfiguration zur Verwendung eines virtuellen Systems im Direktzugriffsmodus.

## Konfiguration

Dieses Verfahren beschreibt die Konfiguration von ESXi auf Servern der UCS Blade-Serie mit Intel Optane DCPMM.

### Konfigurieren des Serviceprofils

1. Navigieren Sie in der UCS Manager-GUI zu **Servers > Persistent Memory Policy (Server > Persistente Speicherrichtlinie)**, und klicken Sie auf **Add (Hinzufügen)**, wie im Bild gezeigt.



2. Create **Goal (Ziel erstellen)**, stellen Sie sicher, dass der **Speichermodus** wie im Bild gezeigt 0 % beträgt.

## Create Goal



### Properties

Socket ID :  All Sockets

Memory Mode (%) :

Persistent Memory Type :  App Direct  App Direct Non Interleaved

OK

Cancel

3. Fügen Sie dem gewünschten Serviceprofil die PMEM-Richtlinie hinzu.

Navigieren Sie zu **Serviceprofil > Richtlinien > Persistent Memory Policy** und fügen Sie die erstellte Richtlinie hinzu.

4. Überprüfen Sie die Gesundheit der Region.

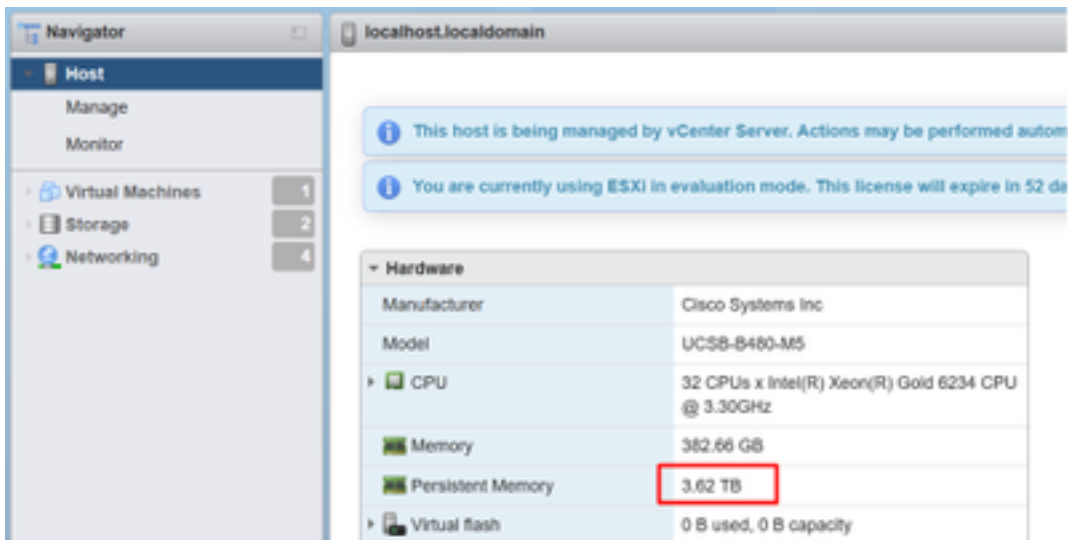
Navigieren Sie zu **Server > Inventory > Persistent Memory > Regions**. Der Typ AppDirect wird angezeigt. Mit dieser Methode wird pro CPU-Socket ein Bereich erstellt.

The screenshot shows the vSphere Web Client interface. The navigation path is: General > Inventory > Persistent Memory > Regions. The table below displays the configuration for four memory regions, all of which are AppDirect type and in a Healthy state.

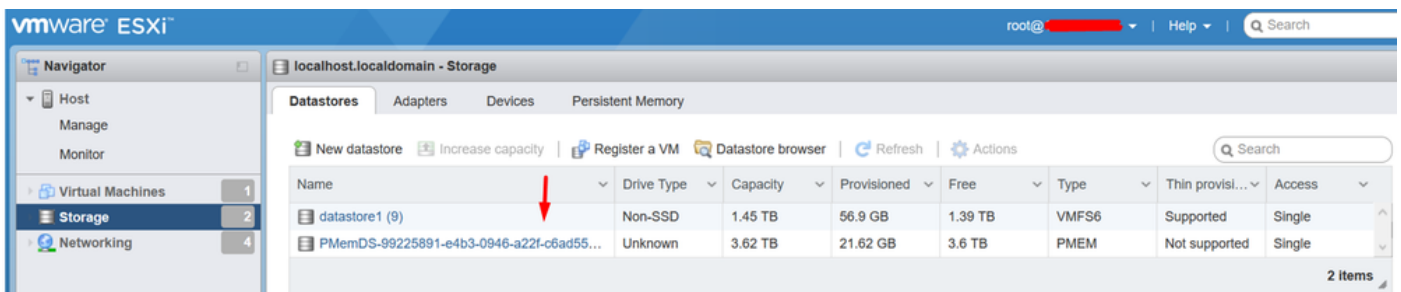
Id	Socket Id	Local DIMM Slot	DIMM Locator Id	Type	Total Capacity (..)	Free Capacity (..)	Health Status
1	Socket 1	Not Applicable	DIMM_A2.DIMM..	AppDirect	928	928	Healthy
2	Socket 2	Not Applicable	DIMM_G2.DIMM..	AppDirect	928	928	Healthy
3	Socket 3	Not Applicable	DIMM_N2.DIMM..	AppDirect	928	928	Healthy
4	Socket 4	Not Applicable	DIMM_U2.DIMM..	AppDirect	928	928	Healthy

## ESXi überprüfen

1. In der Webkonsole zeigt der Host den insgesamt verfügbaren PMEM an.



2. ESXi zeigt einen speziellen Datenspeicher an, der aus der Gesamtzahl von PMEM besteht, wie im Bild gezeigt.



## Konfigurieren von NVDIMMs für virtuelle Systeme

1. In ESXi greifen virtuelle Systeme als NVDIMMs auf Optane DCPMM PMEM zu. Um einem virtuellen System einen NVMDIMM zuzuweisen, greifen Sie über vCenter auf das virtuelle System zu, navigieren Sie zu **Aktionen > Einstellungen bearbeiten**, klicken Sie auf **NEUES GERÄT HINZUFÜGEN**, und wählen Sie **NVDIMM** wie im Bild gezeigt aus.



**Hinweis:** Stellen Sie beim Erstellen eines virtuellen Systems sicher, dass die Kompatibilität des Betriebssystems der Mindestversion entspricht, die den Intel® Optane™ Persistent Memory unterstützt. Andernfalls wird die **NVDIMM**-Option nicht in den auswählbaren Elementen angezeigt.

2. Legen Sie die Größe von NVDIMM wie im Bild gezeigt fest.



## Konfigurieren des Namespaces im virtuellen System

1. Das Dienstprogramm **NDCTL** dient zum Verwalten und Konfigurieren des PMEM oder NVDIMM.

Im Beispiel wird Red Hat 8 für die Konfiguration verwendet. Microsoft bietet PowerShell-cmdlets für das persistente Speicher-namespace-Management.

Laden Sie das **NDCTL**-Dienstprogramm mit dem verfügbaren Tool wie in der Linux Distribution beschrieben herunter.

Beispiele:

```
# yum install ndctl # zypper install ndctl # apt-get install ndctl
```

2. Überprüfen Sie den NVDIMM-Bereich und den standardmäßig von ESXi erstellten Namespace. Wenn das NVDIMM dem virtuellen System zugewiesen ist, überprüfen Sie die Übereinstimmung des Speicherplatzes mit der Konfiguration. Stellen Sie sicher, dass der Modus des Namespaces auf **raw** festgelegt ist, was bedeutet, dass ESXi den Namespace erstellt hat. Verwenden Sie zum Überprüfen den folgenden Befehl:

```
# ndctl list -RuN
```

```
admin@localhost:/etc
File Edit View Search Terminal Help
}
]
[admin@localhost etc]$ ndctl list -RuN
{
  "regions":[
    {
      "dev":"region0",
      "size":"20.00 GiB (21.47 GB)",
      "available_size":0,
      "max_available_extent":0,
      "type":"pmem",
      "persistence_domain":"unknown",
      "namespaces":[
        {
          "dev":"namespace0.0",
          "mode":"raw",
          "size":"20.00 GiB (21.47 GB)",
          "blockdev":"pmem0"
        }
      ]
    }
  ]
}
```

3. (Optional) Wenn der Namespace noch nicht erstellt wurde, kann mit dem folgenden Befehl ein Namespace erstellt werden:

```
# ndctl create-namespace
```

Der Befehl **ndctl create-namespace** erstellt standardmäßig einen neuen Namespace im **fsdax**-Modus und erstellt ein neues **/dev/pmem([x].[y])** Gerät. Wenn bereits ein Namespace erstellt wurde, kann dieser Schritt übersprungen werden.

4. Wählen Sie den PMEM-Zugriffsmodus aus. Zur Konfiguration stehen folgende Modi zur Verfügung:

- Sektormodus:

Stellt Speicher als schnelles Blockgerät dar. Dies ist nützlich für ältere Anwendungen, die noch immer keinen persistenten Speicher verwenden können.

- FSDX-Modus:

Ermöglicht den direkten Zugriff auf das NVDIMM durch die persistenten Speichergeräte. Für den direkten Dateisystemzugriff ist der **fsdax**-Modus erforderlich, um die Verwendung des Programmiermodells für den direkten Zugriff zu ermöglichen. Dieser Modus ermöglicht die Erstellung eines Dateisystems auf dem NVDIMM.

- Devdax-Modus:

Bietet Rohzugriff auf persistenten Speicher mithilfe eines DAX-Zeichengeräts. Dateisysteme können auf Geräten nicht im **devdax**-Modus erstellt werden.

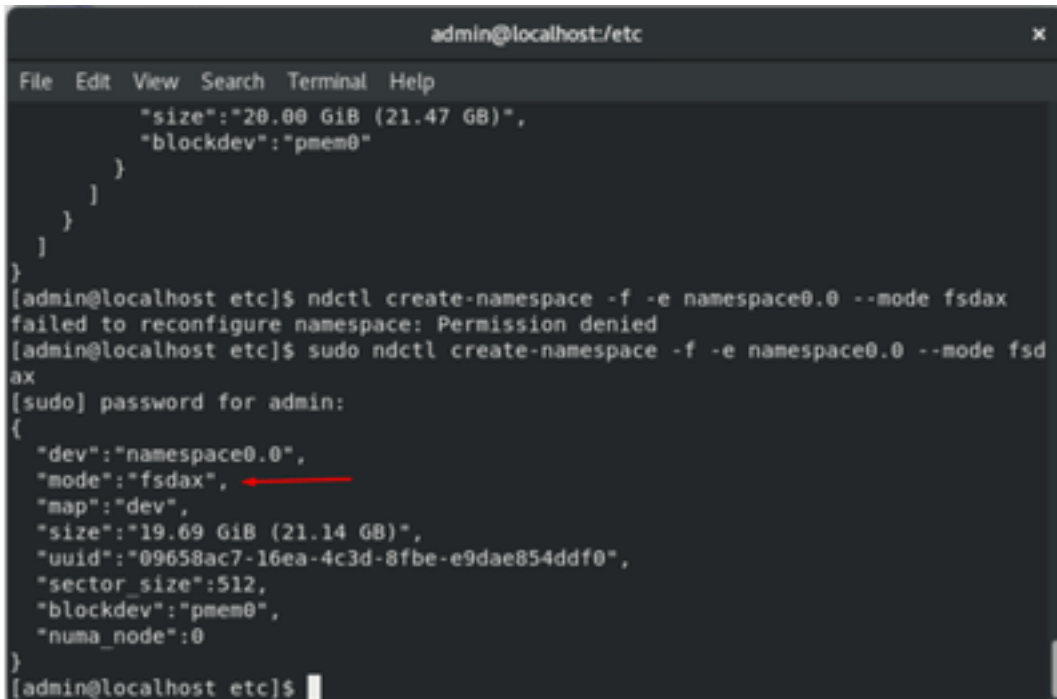
- Raw-Modus:

Dieser Modus hat mehrere Einschränkungen und wird nicht für die Verwendung von persistentem Speicher empfohlen.

Um den Modus in den **fsdax**-Modus zu ändern, verwenden Sie den folgenden Befehl:

```
ndctl create-namespace -f -e
```

Wenn ein **dev** bereits erstellt ist, wird der dev-Namespace zum Formatieren und Ändern des Modus in **fsdax** verwendet.



```
admin@localhost:/etc
File Edit View Search Terminal Help
    "size": "20.00 GiB (21.47 GB)",
    "blockdev": "pmem0"
  }
}
}
}
}
[admin@localhost etc]$ ndctl create-namespace -f -e namespace0.0 --mode fsdax
failed to reconfigure namespace: Permission denied
[admin@localhost etc]$ sudo ndctl create-namespace -f -e namespace0.0 --mode fsdax
[sudo] password for admin:
{
  "dev": "namespace0.0",
  "mode": "fsdax",
  "map": "dev",
  "size": "19.69 GiB (21.14 GB)",
  "uuid": "09658ac7-16ea-4c3d-8fbc-e9dae854ddf0",
  "sector_size": 512,
  "blockdev": "pmem0",
  "numa_node": 0
}
[admin@localhost etc]$
```

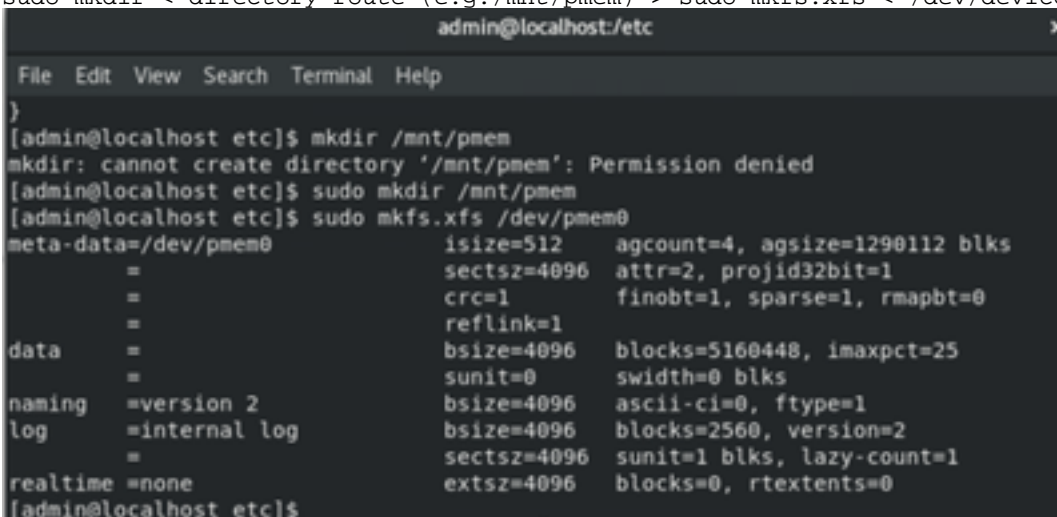
**Hinweis:** Diese Befehle erfordern, dass das Konto über Root-Berechtigungen verfügt, möglicherweise muss der Befehl sudo ausgeführt werden.

### 5. Erstellen Sie ein Verzeichnis und ein Dateisystem.

Direct Access oder DAX ist ein Mechanismus, der Anwendungen den direkten Zugriff auf persistente Medien über die CPU (über Lasten und Speicher) ermöglicht und den traditionellen E/A-Stack umgeht. DAX-fähige persistente Speicherdateisysteme umfassen ext4, XFS und Windows NTFS.

Beispiel eines erstellten und gemounteten XFS-Dateisystems:

```
sudo mkdir < directory route (e.g. /mnt/pmем) > sudo mkfs.xfs < /dev/devicename (e.g. pmем0) >
```



```
admin@localhost:/etc
File Edit View Search Terminal Help
}
[admin@localhost etc]$ mkdir /mnt/pmем
mkdir: cannot create directory '/mnt/pmем': Permission denied
[admin@localhost etc]$ sudo mkdir /mnt/pmем
[admin@localhost etc]$ sudo mkfs.xfs /dev/pmем0
meta-data=/dev/pmем0      isize=512    agcount=4, agsize=1290112 blks
                =          sectsz=4096   attr=2, projid32bit=1
                =          crc=1        finobt=1, sparse=1, rmapbt=0
                =          reflink=1
data         =          bsize=4096  blocks=5160448, imaxpct=25
                =          sunit=0     swidth=0 blks
naming      =version 2   bsize=4096  ascii-ci=0, ftype=1
log         =internal log bsize=4096  blocks=2560, version=2
                =          sectsz=4096  sunit=1 blks, lazy-count=1
realtime    =none       extsz=4096  blocks=0, rtextents=0
[admin@localhost etc]$
```



6. Installieren Sie das Dateisystem, und überprüfen Sie, ob es erfolgreich ist.

sudo mount

```
admin@localhost:/etc
File Edit View Search Terminal Help
[admin@localhost etc]$ sudo mount /dev/pmem0 /mnt/pmem/
[admin@localhost etc]$ // verify the mount was successful
bash: //: Is a directory
[admin@localhost etc]$ df -h /mnt/pmem/
Filesystem      Size  Used Avail Use% Mounted on
/dev/pmem0      20G  173M   20G   1% /mnt/pmem
[admin@localhost etc]$
```

Das virtuelle System ist bereit für die Verwendung von PMEM.

## Fehlerbehebung

Wird generell empfohlen, dieses DAX-fähige Dateisystem mit der Option **-o dax** mount zu mouneten, wenn ein Fehler gefunden wird.

```
[admin@localhost etc]$ sudo mount -o dax /dev/pmem0 /mnt/pmem/
mount: /mnt/pmem: wrong fs type, bad option, bad superblock on /dev/pmem0, missi
ng codepage or helper program, or other error.
```

Die Dateisystemreparatur wird ausgeführt, um Integrität zu gewährleisten.

```
[admin@localhost etc]$ sudo xfs_repair /dev/pmem0
[sudo] password for admin:
Phase 1 - find and verify superblock...
Phase 2 - using internal log
  - zero log...
  - scan filesystem freespace and inode maps...
  - found root inode chunk
Phase 3 - for each AG...
  - scan and clear agi unlinked lists...
  - process known inodes and perform inode discovery...
  - agno = 0
  - agno = 1
  - agno = 2
  - agno = 3
  - process newly discovered inodes...
Phase 4 - check for duplicate blocks...
  - setting up duplicate extent list...
  - check for inodes claiming duplicate blocks...
  - agno = 0
  - agno = 1
  - agno = 2
  - agno = 3
Phase 5 - rebuild AG headers and trees...
  - reset superblock...
Phase 6 - check inode connectivity...
  - resetting contents of realtime bitmap and summary inodes
  - traversing filesystem ...
  - traversal finished ...
  - moving disconnected inodes to lost+found ...
Phase 7 - verify and correct link counts...
done
[admin@localhost etc]$
```

Als Problemumgehung kann das Mount ohne die Option **-o dax** montiert werden.

**Hinweis:** In `xfsprogs` Version 5.1 wird standardmäßig XFS-Dateisysteme mit aktivierter `reflink`-Option erstellt. Zuvor war sie standardmäßig deaktiviert. Die Optionen `reflink` und `dax` schließen sich gegenseitig aus, was dazu führt, dass das Mount fehlschlägt.

"DAX und reflink können nicht zusammen verwendet werden!" Der Fehler wird in `dmesg` angezeigt, wenn der Mount-Befehl fehlschlägt:

```
admin@localhost:/etc
File Edit View Search Terminal Help
log      =internal log          bsize=4096   blocks=2560, version=2
         =                    sectsz=4096  sunit=1 blks, lazy-count=1
realtime =none                extsz=4096   blocks=0, rtextents=0
[admin@localhost etc]$ mount -o dax /dev/pmem0 /mnt/pmem
mount: only root can use "--options" option
[admin@localhost etc]$ sudo mount -o dax /dev/pmem0 /mnt/pmem/
mount: /mnt/pmem: wrong fs type, bad option, bad superblock on /dev/pmem0, missing
codepage or helper program, or other error.
[admin@localhost etc]$ dmesg -T | tail
[mar nov 10 00:12:18 2020] VFS: busy inodes on changed media or resized disk sr0
[mar nov 10 00:12:22 2020] ISO 9660 Extensions: Microsoft Joliet Level 3
[mar nov 10 00:12:22 2020] ISO 9660 Extensions: RRIP_1991A
[mar nov 10 01:47:35 2020] pmem0: detected capacity change from 0 to 21137195008
[mar nov 10 01:51:19 2020] XFS (pmem0): DAX enabled. Warning: EXPERIMENTAL, use
at your own risk
[mar nov 10 01:51:19 2020] XFS (pmem0): DAX and reflink cannot be used together!
[mar nov 10 01:53:06 2020] XFS (pmem0): DAX enabled. Warning: EXPERIMENTAL, use
at your own risk
[mar nov 10 01:53:06 2020] XFS (pmem0): DAX and reflink cannot be used together!
[mar nov 10 01:59:29 2020] XFS (pmem0): DAX enabled. Warning: EXPERIMENTAL, use
at your own risk
[mar nov 10 01:59:29 2020] XFS (pmem0): DAX and reflink cannot be used together!
[admin@localhost etc]$
```

Entfernen Sie als Problemumgehung die Option `-o dax`.

```
admin@localhost:/etc
File Edit View Search Terminal Help
[admin@localhost etc]$ sudo mount /dev/pmem0 /mnt/pmem/
[admin@localhost etc]$ // verify the mount was successful
bash: //: Is a directory
[admin@localhost etc]$ df -h /mnt/pmem/
Filesystem      Size  Used Avail Use% Mounted on
/dev/pmem0      20G  173M   20G   1% /mnt/pmem
[admin@localhost etc]$
```

Montage mit `ext4` FS.

Das `EXT4`-Dateisystem kann als Alternative verwendet werden, da es die `reflink`-Funktion nicht implementiert, aber `DAX` unterstützt.

```
[admin@localhost etc]$ sudo mkfs.ext4 /dev/pmem0
mke2fs 1.44.3 (10-July-2018)
/dev/pmem0 contains a xfs file system
Proceed anyway? (y,N) y
Creating filesystem with 5160448 4k blocks and 1291808 inodes
Filesystem UUID: 164c6d57-0462-45a0-9b94-703719272816
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
    4096000

Allocating group tables: done
Writing inode tables: done
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done

[admin@localhost etc]$ sudo mount /dev/pmem0 /mnt/pmem/
[admin@localhost etc]$ df -h /mnt/pmem/
Filesystem      Size  Used Avail Use% Mounted on
/dev/pmem0      20G  45M   19G   1% /mnt/pmem
[admin@localhost etc]$
```

## Zugehörige Informationen

- [Schnellstartanleitung: Intel® Optane™ Persistent-Memory-Technologie für Rechenzentren](#)
- [Persistente Speicherkonfiguration](#)
- [Management-Utilities ipmctl und ndctl für Intel® Optane™ Persistent Memory](#)
- [Technischer Support und Dokumentation für Cisco Systeme](#)