Technology Brief

Data Center | Memory Optimization

Implementing High Availability in VMware vSphere 7.0 U2 for SAP HANA with Intel® Optane™ Persistent Memory

When running SAP HANA in a VMware vSphere environment, Intel® Optane™ persistent memory (PMem) can significantly expand system memory resources. And when combined with the power of virtualization, it can help enterprises lower their total cost of ownership. Intel Optane PMem running in App Direct mode can deliver near-DRAM-level performance and enable far larger SAP HANA in-memory database capacity while simultaneously providing data persistence in the event of unplanned failure. This, in turn, can yield faster restart times and higher uptime because data is already in memory and doesn't need to be restored from external storage resources.

VMware enabled high-availability (HA) functionality in vSphere 7.0 U2 for Intel Optane PMem-enabled SAP HANA VMs.[1] However, to ensure complete data transfer, additional steps are needed to prepare Intel Optane PMem for SAP HANA use so that it can automatically reload the data from shared (conventional) storage after the failover.

VMware added the HA functionality for Intel Optane PMem and SAP HANA VMs because of the novel way SAP HANA data and Intel Optane PMem interact. VMware vSphere does not support HA for direct-attached storage, such as NVDIMMs or local SSDs. This makes logical sense. How could one automatically fail over a direct-attached storage resource onto a physically separate system? However, when Intel Optane PMem is used in App Direct mode, it appears to vSphere as a direct-attached memory device; that matters because application data kept in Intel Optane PMem on the failed system needs to be brought up on the new system running SAP HANA.

To illustrate, consider a four-node cluster where Hosts 1, 2, and 4 have Intel Optane PMem and Host 3 does not. If Host 1 running an SAP HANA VM using Intel Optane PMem fails, vSphere HA failover will restart the VM on either Host 2 or 4 (depending on the load if VMware's Dynamic Resource Scheduler is enabled). But because PMem appears as direct-attached storage, the data will not migrate across hosts and thus will not be available on the PMem on Hosts 2 or 4 (shown in Figure 1). vSphere HA will start a new PMem (NVDIMM) device on the restored VM, and SAP HANA will only be able to reload the PMem data if the same device mappings and permissions are also restored.

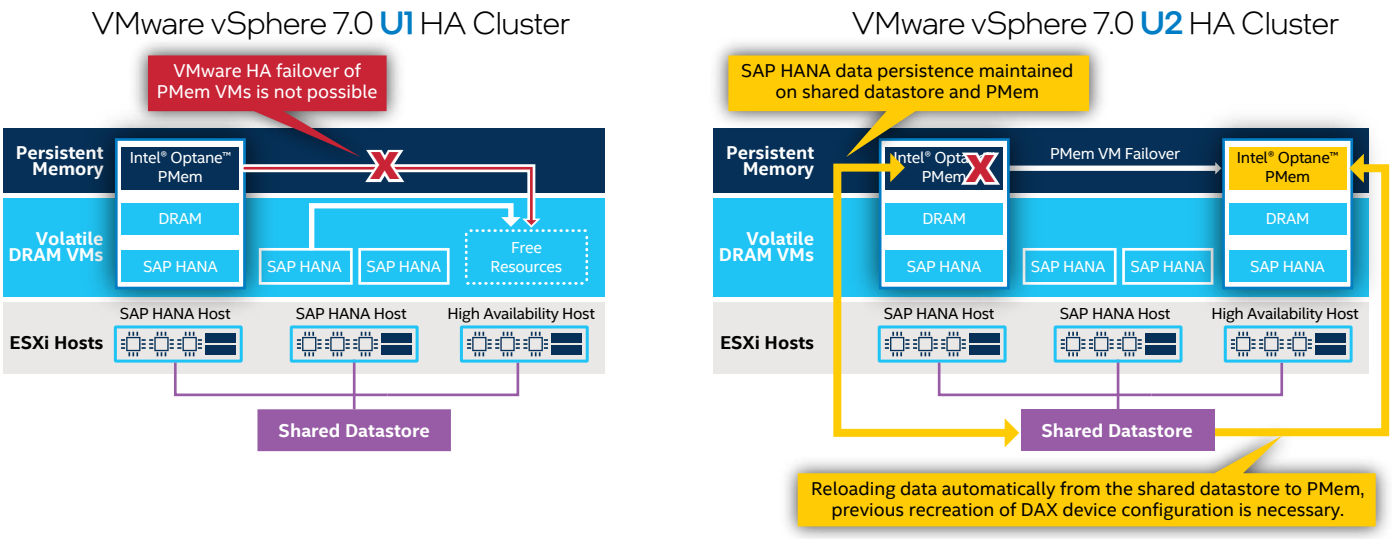

Figure 1. Under vSphere 7.0 U1, there was no way to fail over an SAP HANA VM using Intel® Optane™ PMem into another node, even if that node integrated PMem. With vSphere 7.0 U2 and the execution of functionality-enabling scripts, that data can now be restored from shared storage.

With vSphere 7.0 U2, to allow a virtualized SAP HANA PMem system to start automatically after an HA event, the OS direct-access type (DAX) device configuration required by SAP HANA must be re-created. The next section describes an example script that a customer can use as a basis for their own DAX device preparation script.

Script Elements

There is no single exact way to implement the script to set up the OS to configure SAP HANA with Intel Optane PMem on vSphere. Rather, there are three key elements that need to be implemented:

1. Creating the PMem file system.
2. Defining the file system's mount points.
3. Assigning SAP HANA with the right permissions to the file system.

Appendix A is an example of how to execute the additional OS configuration steps for Intel Optane PMem in a script, which can be automated as needed to execute the entire HA procedure without manual intervention. The intent is to share how users could enable this functionality, but the approach is not meant to be prescriptive or limiting. Customers should evaluate and modify the scripts and procedures to fit their needs.

Step #1: Namespace mode check

This step checks all existing namespaces in the system and ensures that the mode of each one is set to `fsdax` and not `raw`. See Appendix A, lines 3-11:

```
3  do
4    mode=$(ndctl list -n $i | awk '/mode/ {print $0}' | awk
     -F'[:,]' 'gsub(/"/, "") {print $2}')
5    if [ "$mode" == "raw" ]
6    then
7      echo $i
8      ndctl create-namespace -f -e $i --mode=fsdax
9    fi
10 done
11 for i in $(lsblk --output NAME,FSTYPE | awk '/^pmem/ {if
   ($2 !="xfs") {print $1} }')
```

Step #2: PMem filesystem format check

This step checks the filesystem on each PMem device and ensures each one is formatted using `xfs`. See Appendix A, lines 12-15:

```
12 do
13   mkfs.xfs -m reflink=0 -f /dev/$i
14 done
15 for i in $(lsblk | grep ^pmem* | awk '{print $1}')
```

Step #3: Creation and permissions

Finally, the last step involves creation of directories and mounts. Additionally, this step grants the required file access permissions for each PMem device. Appendix A, lines 16-22:

```
16 do
17   mkdir -p /mnt/$i
18   mount -o dax /dev/$i /mnt/$i
19   chown :sapsys /mnt/$i
20   chmod 775 /mnt/$i
21 done
22 exit 0
```

Scripts to control services

The sample script (Appendix B) is executed by service `configureNVDIMM`.

When the VM is deployed from the .ova image, the service must be added to `usr/lib/systemd/system/`, then started and enabled. See Appendix C, lines 7-25 of `install-service.yml`.

```
7  tasks:
8   - name: "Copy service scripts"
9     copy:
10      src: "{{ item.src }}"
11      dest: "{{ item.dest }}"
12      owner: "root"
13      group: "root"
14      mode: "0755"
15    with_items:
16      - {"src":"scripts/configureNVDIMM.service", "dest":"/
          usr/lib/systemd/system/configureNVDIMM.service"}
17      - {"src":"scripts/configureOsForHana.sh", "dest":"/opt/
          configureOsForHana.sh"}
18  - name: "Restart configureNVDIMM service"
19    service:
20      name: "configureNVDIMM"
21      state: "restarted"
22  - name: "Enable configureNVDIMM service"
23    service:
24      name: "configureNVDIMM"
25      enabled: "yes"
```

After every reboot, the service is restarted in the system state, from which all network services are started up and the system will accept logins. See Appendix B, lines 8-9 of `configureNVDIMM.service`.

```
8  [Install]
9  WantedBy=multi-user.target
```

Currently, the service is installed after deploying the SUSE Linux Enterprise Server VM and powering it on. If users want to create a new VM with the configureNVDIMM service, they must either clone the existing VM with this service or install it, then start and enable it manually.

Appendix A: Sample configureOsForHana.sh

```
#!/bin/bash
for i in $(ndctl list | awk '/namespace*/ {print $0}' | awk -F'[:,]' 'gsub(/"/, "") {print $2}')
do
  mode=$(ndctl list -n $i | awk '/mode/ {print $0}' | awk -F'[:,]' 'gsub(/"/, "") {print $2}')
  if [ "$mode" == "raw" ]
  then
    echo $i
    ndctl create-namespace -f -e $i --mode=fsdax
  fi
done
for i in $(lsblk --output NAME,FSTYPE | awk '/^pmem/ {if ($2 !="xfs") {print $1} }')
do
  mkfs.xfs -m reflink=0 -f /dev/$i
done
for i in $(lsblk | grep ^pmem* | awk '{print $1}')
do
  mkdir -p /mnt/$i
  mount -o dax /dev/$i /mnt/$i
  chown :sapsys /mnt/$i
  chmod 775 /mnt/$i
done
exit 0
```

Appendix B: Sample: configureNVDIMM.service

```
[Unit]
 Description=Configure NVDIMMs for hana db
[Service]
 Type=oneshot
 RemainAfterExit=true
 ExecStart=/bin/bash /opt/configureOsForHana.sh
 ExecStop=/bin/true
[Install]
 WantedBy=multi-user.target
```

Appendix C: Sample: install-service.yml

```
---
- hosts: vms
  gather_facts: no
  pre_tasks:
    - name: "Load playbook variables"
      include_vars: "vars.yml"
  tasks:
    - name: "Copy service scripts"
      copy:
        src: "{{ item.src }}"
        dest: "{{ item.dest }}"
        owner: "root"
        group: "root"
        mode: "0755"
      with_items:
        - {"src":"scripts/configureNVDIMM.service", "dest":"/usr/lib/systemd/system/configureNVDIMM.service"}
        - {"src":"scripts/configureOsForHana.sh", "dest":"/opt/configureOsForHana.sh"}
    - name: "Restart configureNVDIMM service"
      service:
        name: "configureNVDIMM"
        state: "restarted"
    - name: "Enable configureNVDIMM service"
      service:
        name: "configureNVDIMM"
        enabled: "yes"
```

[1] For more information go to SAP HANA with Intel Optane Persistent Memory on VMware vSphere - Virtualize Applications.