ISICC | IBM SAP International Competence Center

Power Systems

Planning and Implementation Guide

SAP HANA and PowerVM Virtual Persistent Memory

Jim Nugen
Olaf Rutz

> **Note:** Before using this information and the product it supports, be sure to read the general information under "Copyrights and Trademarks" on page 22 as well as "Disclaimer and Special Notices" on page 22.

Edition Notice: this is version 1.0 of this document.
Focus: IBM PowerVM Virtual Persistent Memory for SAP HANA
Target:

- SAP HANA 2 SPS04 and newer
- POWER9 Systems

Doc Version	Changes
0.1	Draft
1.0	Initial release

Preface

IBM PowerVM Virtual Persistent Memory provides POWER9 servers with the ability to create persistent memory volumes based on conventional DRAM memory modules without investing in additional hardware. This enables applications, such as SAP HANA, to preserve in-memory content across virtual partition reboots and avoid time-consuming reloads.

About This Document

This document is intended for architects and specialists planning to implement IBM PowerVM Virtual Persistent Memory (vPMEM) with SAP HANA®. This guide does not replace existing IBM and SAP HANA documentation, but instead serves as a supplement providing end to end guidance for configuration and implementation of vPMEM with SAP HANA. Guidelines and references are also given for appropriate sizing of vPMEM volumes. This document additionally lists the SAP HANA, IBM Power Systems™, IBM PowerVM®, and Linux Operating System prerequisites for implementation.

The most recent document version can be downloaded from IBM TechDocs:
http://www.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP102502

Feel free to provide feedback and change requests for this document via email to isicc@de.ibm.com.

About the Author(s)

Jim Nugen, jim.nugen@ibm.com
Olaf Rutz, rutz@de.ibm.com

Table of Contents

Introduction

Over the next two to three years, there will be a broad range of new memory technologies that are designed to disrupt the industry, providing persistence characteristics with a range of performance and cost advantages to benefit in-memory workloads such as in SAP HANA. These technologies will come from a multitude of memory partners and IBM has taken a multistep approach to address the various client use cases. IBM and SAP are co-innovating to enable clients running SAP HANA on IBM Power Systems to adopt them and Virtual Persistent Memory (vPMEM) is the first solution in this approach.

Virtual Persistent Memory is an enhancement of IBM's advanced virtualization platform (PowerVM) and introduces the ability to configure persistent volumes using the conventional DRAM memory modules available in every IBM POWER9 system. Consequently, no special or additional hardware components are required. Instead, only a firmware upgrade must be performed to enable vPMEM.

Since vPMEM is built on DRAM technology, it has the same performance characteristics as DRAM, which enables IBM POWER9 users to significantly speed up restarts of SAP HANA during planned maintenance as well as unplanned outages, without compromising performance in production. This innovation substantially improves the serviceability of SAP HANA and Linux software maintenance.

Technology Overview

The following sections provide background into persistent memory and vPMEM and their usage in an SAP HANA environment.

Persistent Memory

Persistent memory is non-volatile, byte-addressable, low latency memory.
- Non-volatile memory refers to the ability to maintain contents after a power shutdown.
- Byte addressable means that the contents can be accessed using CPU load and store instructions.
- Low latency refers to memory speeds similar to that of DRAM.

Some SAP documentation refers to persistent memory as non-volatile memory (NVM), while IBM documentation often uses the term storage class memory (SCM). The term non-volatile DIMM (NVDIMM) persistent memory is also used.

The Storage Networking Industry Association (SNIA) has defined a programming model which describes an architecture of how operating systems can provide persistent memory services and how application software can utilize them. The PowerVM/Linux on Power implementation of this programming model can be seen in Figure 1.
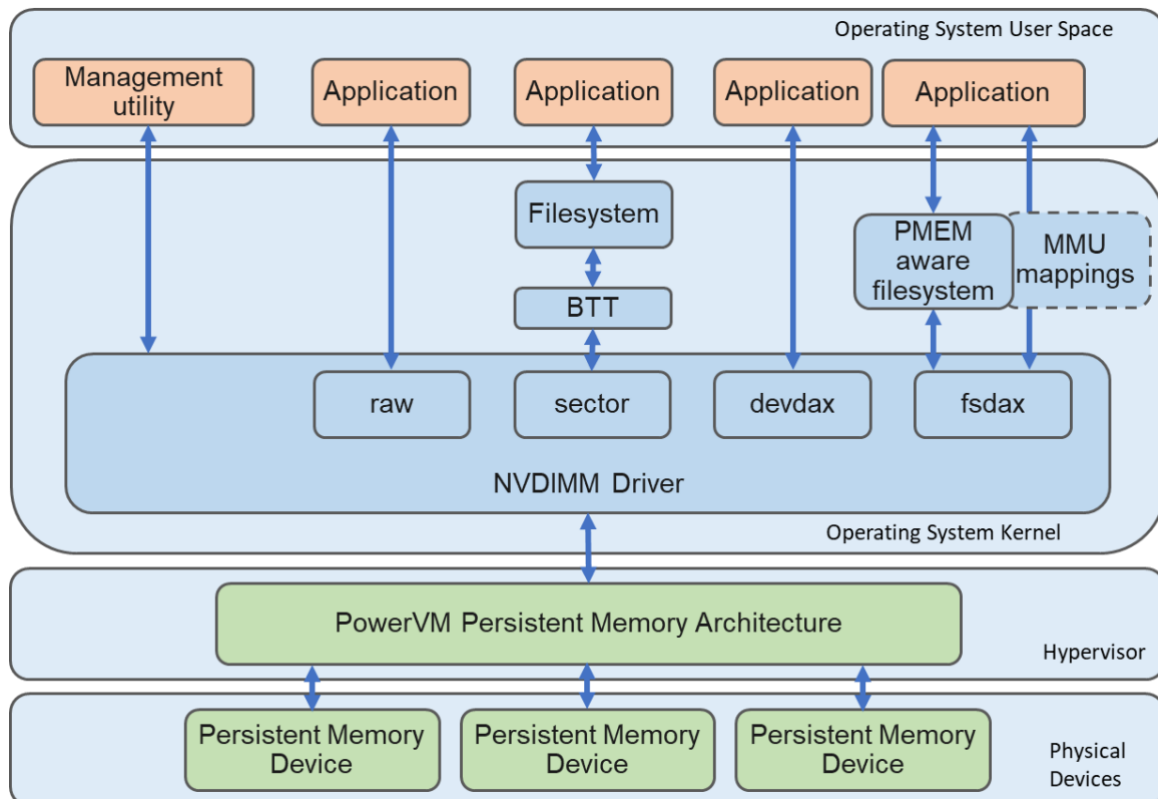
Figure 1: PowerVM / Linux persistent memory architecture

At the bottom of Figure 1, the PowerVM hypervisor presents the persistent memory devices to the operating system in a technology agnostic manner. This is referred to as the PowerVM Persistent Memory Architecture. This abstraction enables the adoption of new persistent memory technologies, attachment technologies, and device form factors with minimal impact on the operating system and virtualization management code.

Depending on the physical device capabilities, the PowerVM hypervisor may also be able to virtualize persistent memory devices and segmenting them into smaller capacity volumes, which can be assigned to different logical partitions (LPARs).

Once persistent memory is assigned to an LPAR, individual devices are presented by the Linux operating system as generic non-volatile DIMM devices, /dev/nmem<#>. The management tool ndtcl is used to interface with the nvdimm driver to configure and provision these "nvdimm" devices into *region*s, *namespaces*, and persistent memory volumes.

A region is a grouping of one or more nvdimm devices. Commonly, a region is formed from devices from the same numa node.

A namespace is a partition of a region either whole or part. Namespaces are associated with a *mode,* which enables different access methods to the persistent memory. Four modes are available:
- **fsdax (filesystem direct access)**
  Persistent memory is presented as a block device (/dev/pmem<#>) and supports XFS and EXT4 filesystems. This mode provides direct access (DAX) support, which

bypasses the Linux page cache and performs reads and writes directly to the device. For direct access through load and store instructions, the device can be mapped into the address space of the application process with mmap(). The default mode of a namespace is fsdax.

- **devdax (device direct access)**
  Persistent memory is presented as a character device (/dev/dax<#>.<#>). This mode also provides DAX support.
- **sector**
  Persistent memory is presented as a block device (/dev/pmem<#>s) and supports any filesystem. This mode is useful for applications which are not persistent memory aware.
- **raw**
  This mode provides a memory disk with no DAX support.

For SAP HANA, only the fsdax mode is employed. Figure 2 depicts an example of the fsdax stack exposing nvdimm devices to applications.
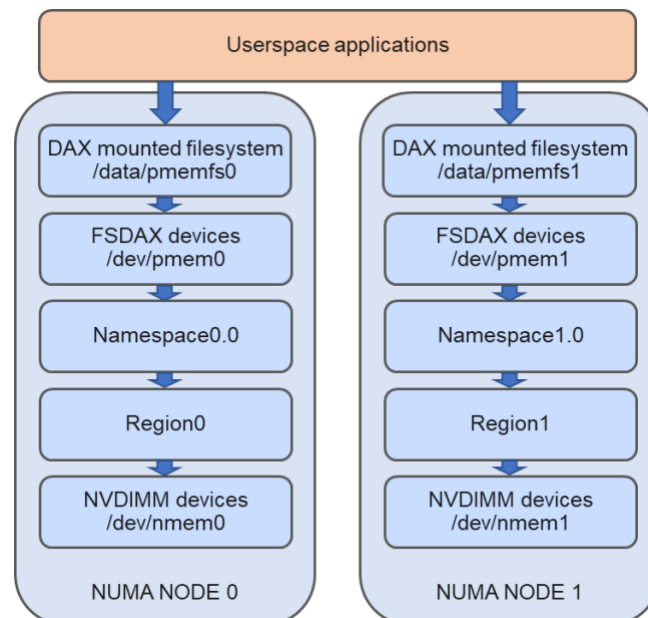

Figure 2: fsdax mode stack

Virtual Persistent Memory

Virtual Persistent Memory (vPMEM) is a PowerVM feature offered on Power9 servers which presents a portion of the installed standard system DRAM DIMMs as nvdimm devices to the operating system. The "virtual" qualifier denotes that this differs from true persistent memory since system DRAM is volatile memory. System DRAM will lose its contents when the physical server is powered off.

Powering down the physical system in a PowerVM virtualized environment is, however, a relatively infrequent event. Maintenance is significantly more often performed at the level of LPAR logical partitions and shutdowns or reboots of the operating system do not involve powering down the physical server. As such, vPMEM nvdimm devices, also referred to as

vPMEM volumes, maintain their contents over these operations. Table 1 summarizes the different levels of data persistence of vPMEM as compared to true persistent memory.

	vPMEM persistence	PMEM persistence
Application restart	✓	✓
LPAR restart	✓	✓
Physical server restart	✗	✓

Table 1: vPMEM data persistence

vPMEM volumes are managed on the system Hardware Management Console (HMC). They are defined per LPAR and are not directly sharable or transferable to other LPARs.

vPMEM volumes are sized on *logical memory block* (LMB) granularity where an LMB is the unit of memory used by the hypervisor to manage DRAM memory. By default, an LMB is 256MB system wide.

On creation, vPMEM volumes are specified to be striped across NUMA nodes or to be NUMA node contained. For NUMA aware applications, like SAP HANA, vPMEM volumes should be provisioned on a NUMA node basis so that their NUMA node associativity is clearly defined. That is, as shown in Figure 3, the PowerVM hypervisor should allocate DRAM exclusively from one NUMA node to serve as single vPMEM volume.
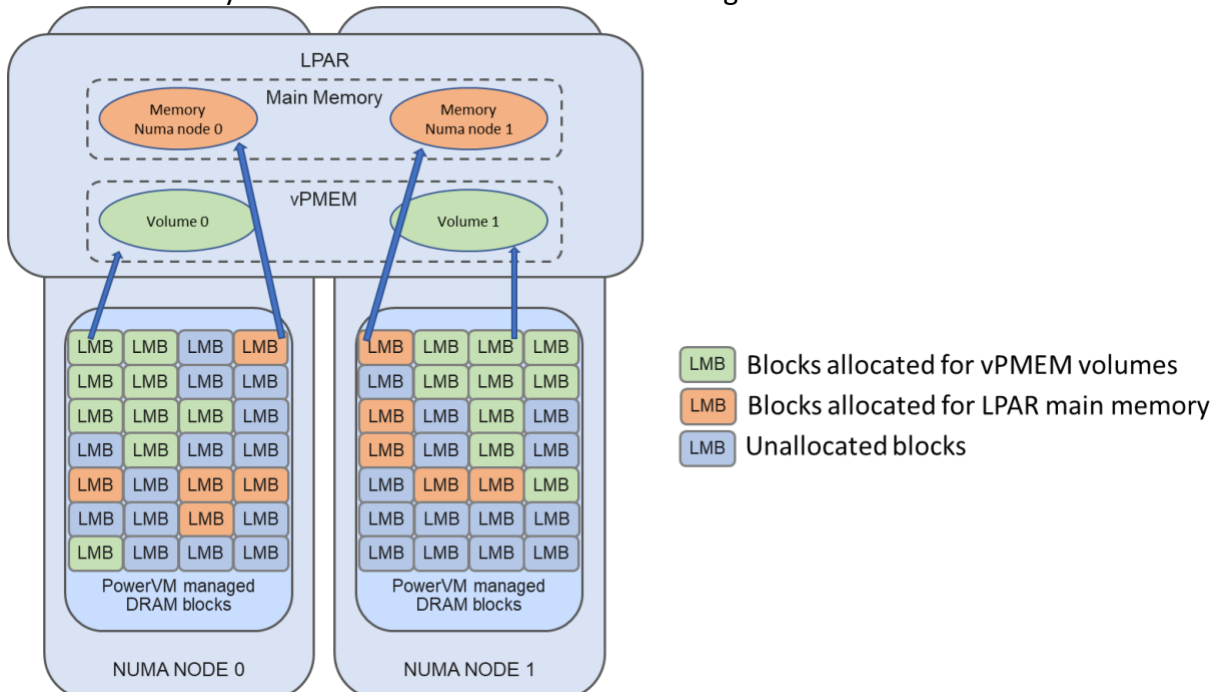

Figure 3: vPMEM placement by NUMA node

vPMEM volumes cannot be resized. Instead, they are deleted and new vPMEM volumes can be created with the desired size.

IBM Systems Solution for SAP HANA on POWER and IBM System Storage
© Copyright IBM Corporation, 2020

Persistent Memory and SAP HANA

SAP HANA leverages persistent memory to reduce operational downtime. By retaining data in persistent memory after a shutdown, SAP HANA can avoid time-consuming data reloads from disk storage on startup. For a large multi-terabyte SAP HANA database, this can reduce startup time from well over an hour to just a few minutes which is significant for systems that have strict SLA requirements.

Specifically, SAP HANA supports placing *column-store main data* structures in persistent memory. The main data structures are highly compressed, read-only (after creation), and represent 95% of database data.

SAP HANA requires persistent memory to be configured in fsdax mode as shown earlier in Figure 2. Additionally, in order to take advantage of SAP HANA NUMA optimizations, it is required that the vPMEM volumes are configured per NUMA node shown in Figure 3. Filesystems are created on the persistent memory fsdax devices and mounted with the DAX option.
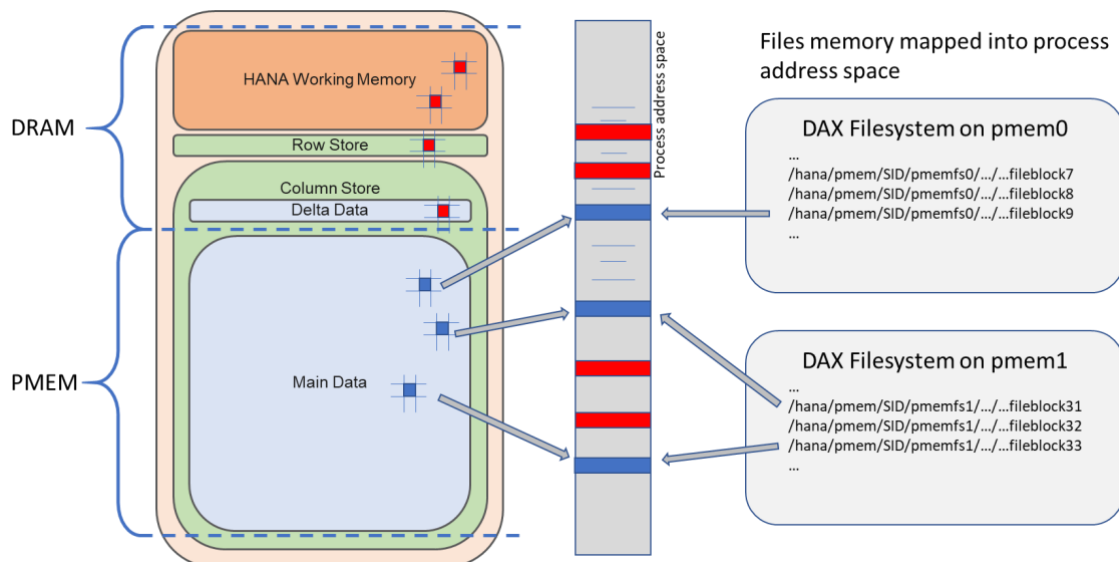

Figure 4 SAP HANA memory components and persistent memory data

SAP HANA main data, which is organized in column-wise data structures, can be written to files in the DAX filesystem. However, instead of using standard file I/O read and write calls, SAP HANA employs memory-mapped file I/O as seen in Figure 4. By mapping the files directly into its address space, the application can use load and store CPU operations to manipulate the data.

Implementing vPMEM with SAP HANA

The following chapter provides instructions on how to enable vPMEM usage for SAP HANA.

Prerequisites

The following list details the minimum hardware and software levels required to configure and implement SAP HANA with IBM PowerVM Virtual Persistent Memory:

IBM Systems Solution for SAP HANA on POWER and IBM System Storage
© Copyright IBM Corporation, 2020

- IBM POWER9 System with Firmware FW940
- IBM HMC V9.1.940
- SAP HANA 2 SPS04 Revision 44
- SLES 15 SP1
    - Kernel version 4.12.14-197.21.1
    - ndctl version 64.1-3.3.1

To run the SAP Hardware and Cloud Measurement Tool (HCMT) with vPMEM, the minimum tool version is SAP HANA 2 SPS04 Revision 46.

Recommended Workflow for Using vPMEM with SAP HANA

The recommended workflow to enable vPMEM usage with SAP HANA is as follows:
1. Determine the amount of persistent memory to configure (see Sizing vPMEM for SAP HANA)
2. Prepare the LPAR profile for persistent memory (see Configuring LPAR profile settings for vPMEM)
3. Create vPMEM volumes (see Managing vPMEM volumes on the Hardware Management Console)
4. Build vPMEM based filesystems (see Automated rebuild and mount of vPMEM based filesystems)
5. Enable SAP HANA to use persistent memory (see Configuring SAP HANA to use vPMEM)

Sizing vPMEM for SAP HANA

Before configuring vPMEM volumes for use with SAP HANA, the appropriate volume sizes must be determined. In general, vPMEM volumes should be as large as the anticipated main data of SAP HANA system's column store plus spare capacity for growth and delta merge operations.

SAP note 2786237 details several tools to assist in the proper sizing of persistent memory volumes:

- SAP HANA Quicksizer for greenfield deployments
- Sizing report for SoH and S/4HANA: SAP Note 1872170
- Sizing report for BWoH and BW/4HANA: SAP Note 2296290
- SQL reports attached to the note 2786237 for an overview of memory usage in a current system

Note that the ratio restrictions between DRAM and PMEM documented in SAP note 2786237 does not apply to the Power platform.

Configuring LPAR profile settings for vPMEM

When preparing an LPAR for vPMEM usage, some considerations must be taken into account in the LPAR definition.

Each LPAR has an associated hardware page table (HPT) which translates the effective memory addresses of the LPAR to real physical addresses in the hardware. The amount of memory that the HPT requires is based on the maximum DRAM that may be used by the partition and the HPT ratio. The HPT ratio is the ratio of the HPT size to the maximum memory value for the logical partition and can affect the performance of the logical partition. A small HPT might lead to increased LPAR CPU consumption as operating system may need to reload the HPT entries more frequently. The default HPT ratio for Linux LPARs is 1/128th of the maximum memory LPAR profile value.

Likewise, a secondary hardware page table, called the physical page table (PPT), is used to track pages that have changed during a *logical partition mobility* (LPM) operation. A larger PPT may improve application performance during an online LPM move. It is also sized to the maximum memory value for the logical partition and is specified as a ratio in the LPAR profile. The default PPT ratio for Linux LPARs is 1/4096th of the maximum memory value.

Currently, with Power Systems firmware FW940, HPT and PPT sizing is automatically calculated based on the LPAR profile value for the maximum amount of DRAM memory. However, although vPMEM resides in DRAM, its memory allocation is not included in the automatic size calculations and any large vPMEM volumes will cause the HPT and PPT to be too small. Therefore, it is recommended to use the hpt_for_vpmem.py script to determine appropriate values for the HPT ratio, the PPT ratio, and the maximum memory size. The options for the script are as follows:

```
hpt_for_vpmem.py [options]*
    -h, --help              Displays the help text.
    -m GB, --memory GB      Desired DRAM memory for the partition, in GB.
    -l [Nx]GB, --lun [Nx]GB VPMEM LUN size in GB, can be specified multiple
                            times.  Can also be specified with a preceding
                            replication factor N, as in '3x1000' for three
                            1000GB LUNs.
    -i, --ibmi              Partition type is IBMi.
    -n, --linux             Partition type is Linux.
    -a, --aix               Partition type is AIX.
```

The following example shows the recommended settings for an LPAR with a maximum memory profile value of 700GB and 4 vPMEM volumes, each of size 1,536GB.

```
$ ./hpt_for_vpmem.py --memory 700 --lun 4x1536 --linux

Inputs:
    desired_memory_size = 700GB
    vpmem_size          = 6144GB
    hpt_ratio           = 1/128 (7)
    ppt_ratio           = 1/4096 (6)

Goals:
    target_hpt_size     = 64GB

Outputs:
    max_memory_size     = 1281GB
```

IBM Systems Solution for SAP HANA on POWER and IBM System Storage
© Copyright IBM Corporation, 2020

```
    hpt_ratio           = 1/32 (5)
    ppt_ratio           = 1/1024 (4)
    actual_hpt_size     = 64GB

ELMM Tree Structure:
    elmm_base_address   = 4TB
    PCI/VAS/XIVE        = 4TB..8TB
    LUN 1536GB          = 8TB..10TB
    LUN 1536GB          = 10TB..12TB
    LUN 1536GB          = 12TB..14TB
    LUN 1536GB          = 14TB..16TB
    elmm_end_address    = 16TB

Recommendations:
    Change the HPT ratio from 1/128 to 1/32.
    Change the PPT ratio from 1/4096 to 1/1024.
    Change the maximum memory size from 700GB to 1281GB.
```

The example's recommendations can be applied to the LPAR profile with the chsyscfg command on the Hardware Management Console:

```
$ chsyscfg -r prof -m <managed-system name> -i "name=<profile
name>,lpar_name=<partition name>,max_mem=1281,hpt_ratio=1:32,ppt_ratio=1:1024"
```

Initially, this script will be made available by email request to the IBM SAP International Competence Center (ISICC) Infoservice (isicc@de.ibm.com).

Firmware FW950 will incorporate the sizes of the vPMEM devices into the calculation of the HPT and PPT sizes. In that case, the above script will no longer be necessary.

### Managing vPMEM volumes with the Hardware Management Console

Configuration and management of virtual persistent memory volumes is performed on a Hardware Management Console (HMC) running HMC V9.1.940, or later. The target POWER9 system needs firmware level FW940, or later.

vPMEM volumes are configured at the level of logical partitions (LPARs). Currently, creation, renaming, and deletion of vPMEM volumes is supported. To perform these operations, the LPAR must be in the state **Not activated**.

IBM Systems Solution for SAP HANA on POWER and IBM System Storage
© Copyright IBM Corporation, 2020

To create vPMEM volumes:

1. In the navigation pane, click the **Resources** icon .
2. Click **All Systems** and select the managed system. The **Partitions** page for the managed system is displayed.
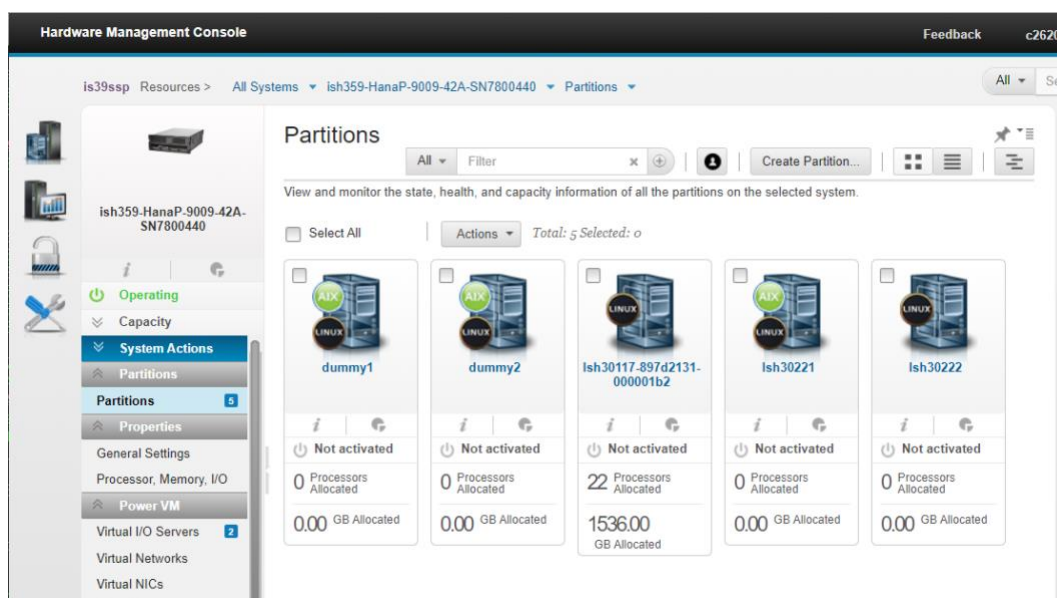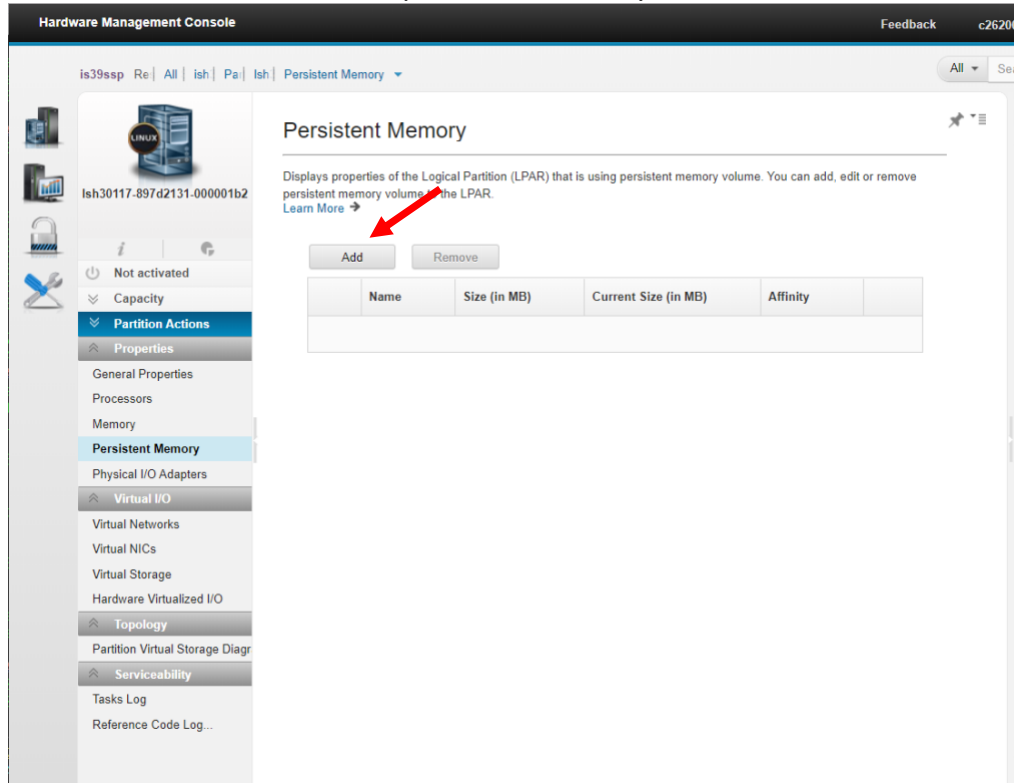

3. In the Partitions pane, select the partition for which you want to view or change the properties, and click **Actions** > **View Partition Properties**. The **Properties** page is displayed. You can view and change the properties that are listed under the **Properties** area.
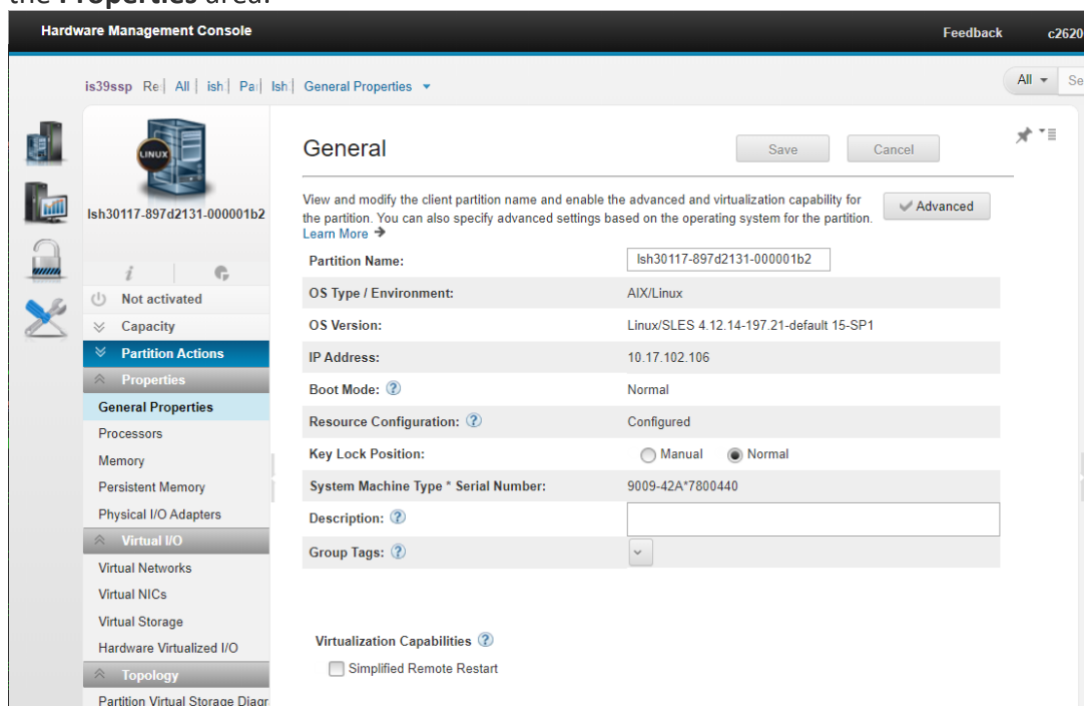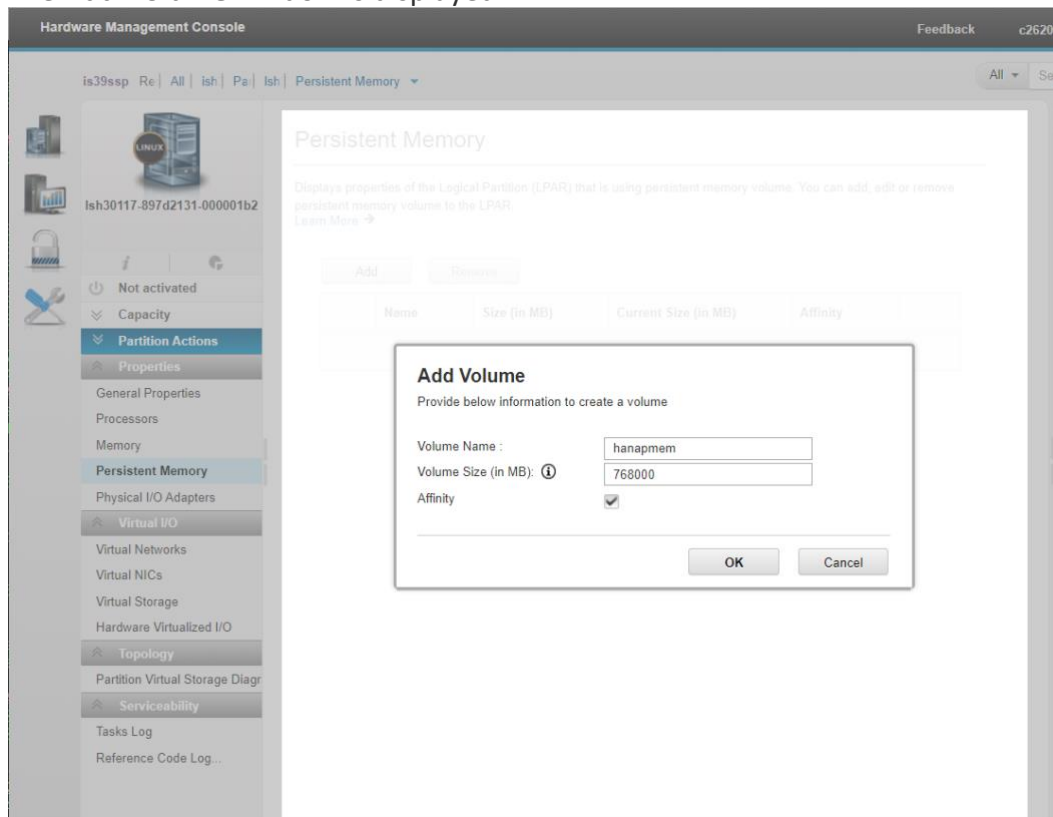

4. In the **Properties** at the left, click **Persistent Memory** to view the properties of the logical partition that should use a virtual persistent memory volume.

IBM Systems Solution for SAP HANA on POWER and IBM System Storage
© Copyright IBM Corporation, 2020

5. Click **Add** to create the virtual persistent memory.

6. The **Add Volume** window is displayed.

   a. Specify a name for the persistent memory volume in the **Volume Name** field.

   b. Specify a value for the size of the persistent memory volume in the **Volume Size** field. This must be a multiple of the system LMB size – commonly 256MB.

   c. Select the **Affinity** check box if the vPMEM volumes should be provisioned by NUMA node. For vPMEM volumes for use with SAP HANA, one should always check affinity.

   d. Click **OK** to create the persistent memory volume. Alternatively, click **Cancel** to reject the changes and to close the window.

Once a vPMEM volume exist one can subsequently rename or delete it on the **Persistent Memory** page.

In addition to the graphical configuration, the HMC command line tools can also be used. For example, to list all vPMEM volumes of all LPARs on a managed system, use the command lshwres:

```
$ lshwres -r pmem -m ish359-HanaP-9009-42A-SN7800440 --level lpar
lpar_name=lsh30221,lpar_id=21,curr_num_volumes=0,curr_num_dram_volume
s=0,max_num_dram_volumes=4
lpar_name=lsh30222,lpar_id=20,curr_num_volumes=0,curr_num_dram_volume
s=0,max_num_dram_volumes=4
lpar_name=dummy2,lpar_id=5,curr_num_volumes=0,curr_num_dram_volumes=0
,max_num_dram_volumes=4
lpar_name=dummy1,lpar_id=4,curr_num_volumes=0,curr_num_dram_volumes=0
,max_num_dram_volumes=4
lpar_name=lsh30117-897d2131-
000001b2,lpar_id=3,curr_num_volumes=1,curr_num_dram_volumes=1,max_num
_dram_volumes=4
lpar_name=ish359v2,lpar_id=2,curr_num_volumes=0,curr_num_dram_volumes
=0,max_num_dram_volumes=4
lpar_name=ish359v1,lpar_id=1,curr_num_volumes=0,curr_num_dram_volumes
=0,max_num_dram_volumes=4
```

Preparing vPMEM volumes for use with SAP HANA

As previously described, HANA places main data into files located in XFS filesystem mounted with the DAX option. The following section shows how to build such filesystems on vPMEM volumes.

After creating vPMEM volumes with the Hardware Management Console and booting the LPAR, the volumes are presented as non-volatile DIMM devices, /dev/nmem<#>, by the operating system.

The ndctl tool can be used to list those DIMM devices:
```
# ndctl list --dimms
[
  {
    "dev":"nmem1"
  },
  {
    "dev":"nmem0"
  }
]
```

IBM Systems Solution for SAP HANA on POWER and IBM System Storage
© Copyright IBM Corporation, 2020

The associated NUMA nodes can also be listed:

```
# ndctl list --bus all
[
  {
    "provider":"ibm,persistent-memory:ibm,pmemory@44108001",
    "dev":"ndbus1"
  },
  {
    "provider":"ibm,persistent-memory:ibm,pmemory@44104001",
    "dev":"ndbus0"
  }
]
```

To create vPMEM-based filesystems for SAP HANA, execute the following steps:

1. Use the ndctl tool to create the necessary pmem regions and namespaces on the devices:

   For example, for 2 volumes, /dev/nmem0 and /dev/nmem1, the following commands can be run in a Bash shell:

   ```
   # for i in 0 1
   do
     ndctl disable-region region$i
     ndctl zero-labels nmem$i
     ndctl init-labels nmem$i
     ndctl enable-region region$i
     ndctl create-namespace -r region$i
   done
   ```

   At this point, persistent memory block devices have been prepared and are presented by the operating system as /dev/pmen<#>. Again, these can be listed with ndctl:

   ```
   # ndctl list --namespaces --type=pmem
   [
     {
       "dev":"namespace1.0",
       "mode":"fsdax",
       "map":"dev",
       "size":402787401728,
       "uuid":"68080d19-6890-4bce-b612-6b4af4e164b1",
       "sector_size":512,
       "align":16777216,
       "blockdev":"pmem1"
     },
     {
       "dev":"namespace0.0",
       "mode":"fsdax",
       "map":"dev",
       "size":401713659904,
       "uuid":"57d3aea8-b8cd-410c-960e-047b8cc03949",
   ```

```
        "sector_size":512,
        "align":16777216,
        "blockdev":"pmem0"
    }
]
```

2. Now, create and mount XFS filesystems with the DAX option for each pmem namespace. Note that the DAX option skips the page cache and uses file system blocks directly as page cache entries. This requires the block size to be same as operating system pagesize, which is 64K on Power Systems.

   For example, for pmem0 and pmem1:
   ```
   # for i in 0 1
   do
     mkdir -p /hana/pmem/pmem$i
     mkfs.xfs /dev/pmem$i -b size=64k -s size=512
     mount -o dax /dev/pmem$i /hana/pmem/pmem$i
   done
   chown -R <sid>adm:sapsys /hana/pmem
   chmod -R 0700 /hana/pmem
   ```

   > **Caution:** Do not create the pmem filesystem mountpoints under filesystem mountpoints other than "/". In such cases, SAP HANA does not determine the DAX attribute properly and will not use those filesystems to store the data.
   >
   > Note that the block device name, for example, pmem0 and pmem1 from above, may change after reboot. For any automated mounting of the associated filesystems, it is recommended to use the filesystem UUID, as with any other filesystem.

To remove the pmem namespaces, execute the following steps:
   1. Unmount the "pmem" filesystem
   2. Destroy the namespace with ndctl

      ```
      # for i in 0 1
      do
        ndctl destroy-namespace namespace${i}.0 -f
      done
      ```

Specifying vPMEM Usage in Existing SAP HANA Installations

SAP HANA configuration files are stored on the server at the following locations according to layer:

- Default:

  `/usr/sap/<SID>/HDB<instance>/exe/config` (read only)

- System:

  `<sapmnt>/<SID>/SYS/global/hdb/custom/config`

- Database:

  `<sapmnt>/<SID>/SYS/global/hdb/custom/config/DB_<dbname>`

- Host:

  `/usr/sap/<SID>/HDB<instance>/<hostname>`

By default, SAP HANA usage of persistent memory volumes is specified at the host level. All HANA services managed by a single SAP HANA Global Allocation Limit (GAL) will share a set of persistent memory volumes.

1. Use the basepath_persistent_memory_volumes parameter to specify the pmem filesystem in the global.ini configuration file of HANA. For example:

```
...
[persistence]
basepath_datavolumes = /hana/shared/data/JE6
basepath_logvolumes = /hana/shared/log/JE6
basepath_persistent_memory_volumes =
/hana/pmem/pmem0/JE6;/hana/pmem/pmem1/JE6
```

2. Activate persistent memory storage for the database in the indexserver.ini configuration file of HANA.

```
...
[persistent_memory]
table_default = ON
```

   Note that this setting may be overridden by the preference settings on the table, partition or column level.

Note that in order to specify different sets of vPMEM volumes for different SAP HANA tenants, use SAP Note 2175606 to first segment tenants to separate GALs. Then define the persistent memory volumes in the above .ini files at the database level.

Specifying vPMEM Usage in New SAP HANA Installations

For new SAP HANA installations, one can use the installation parameters of hdblcm:

```
--use_pmem --pmempath=<path to pmemX>[:<path to pmemY>…]
```

Verifying vPMEM usage

The following query can be used to verify that the vPMEM-based filesystems are utilized by SAP HANA as expected:

```
hdbsql> select * from M_PERSISTENT_MEMORY_VOLUMES where
PORT=3<instance #>03
```

For example:

```
hdbsql> select * from M_PERSISTENT_MEMORY_VOLUMES where PORT=30603
```

```
HOST,PORT,VOLUME_ID,NUMA_NODE_INDEX,PATH,FILESYSTEM_TYPE,IS_DIRECT_AC
CESS_SUPPORTED,TOTAL_SIZE,USED_SIZE
"lsh30117",30603,3,0,"/hana/shared/pmem/pmem0/JE6/mnt00001/hdb00003.0
0003","xfs","true",401517510656,15582494720
"lsh30117",30603,3,1,"/hana/shared/onen/pmem1/JE6/mnt00001/hdb00003.0
0003","xfs","true",402590728192,15930228736
```

The output shows that HANA has found and is using 2 persistent memory-based XFS filesystems. One filesystem is backed by memory on NUMA node 0, while the other is backed by memory on NUMA node 1.

Operational procedures

The following chapter provides guidance on how to operate SAP HANA using vPMEM devices.

Automated rebuild and mount of vPMEM based filesystems

When maintaining a HANA system, activities such as restarting the operating system (e.g., for applying security fixes) or restarting the managed system are occasionally required. When using vPMEM with HANA, some additional steps must be taken before restarting HANA. In case of an OS reboot, the vPMEM based file systems must be remounted. In case of a managed system restart with Power Off the underlying files systems also need to be rebuilt.

To simplify and automate those actions, a convenient startup script is available. The vpmem_hana_startup.sh script assists and automates the process of verifying the vPMEM-based filesystems, recreating the filesystems if necessary, mounting the filesystems, and updating the HANA configuration file.

This script requires a JSON configuration file as input with the following properties:

```
[
  {
    "sid"  : "<HANA instance name>",
    "puuid": "<parent vpmem volume uuid>",
    "mnt"  : "<filesystem path to mount vpmem filesystems under>"
  }
]
```

The parent UUID of all persistent memory block devices can be listed by the script with the -p option:

```
$ ./vpmem_hana_startup.sh -p
/sys/devices/ndbus0/region0/of_node/ibm,unit-parent-guid
          "71043c70-3d8f-42fa-8d7d-2828c04666f5"
/sys/devices/ndbus1/region1/of_node/ibm,unit-parent-guid
          "71043c70-3d8f-42fa-8d7d-2828c04666f5"
```

When executed, the script will perform the following actions:
1. Scan the configuration file to determine the parent UUID of the vPMEM volumes.

IBM Systems Solution for SAP HANA on POWER and IBM System Storage
© Copyright IBM Corporation, 2020

2. Search the device tree to locate the vPMEM devices associated with the UUID.
3. For each child volume, check whether valid filesystems exist.
4. If no valid file systems are found, format the volume with an XFS filesystem.
5. Mount each of the filesystems under a mount point representing their NUMA associativity.
6. Update the HANA configuration file to reflect where the vPMEM devices are mounted for each NUMA domain.

It is recommended to establish a systemd service to execute the script on OS startup.
1. Place this script in /usr/sap/vpmem/
2. Create /usr/sap/vpmem/vpmem_hana.cfg
3. Create /etc/systemd/system/vpmem_hana.service

```
[Unit]
Description=Virtual PMEM SAP HANA Startup Script

[Service]
Type=oneshot
ExecStart=/bin/sh -c "/usr/sap/vpmem/vpmem_hana_startup.sh"

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

4. Start the service now and on reboot

```
systemctl start vpmem_hana.service
systemctl enable vpmem_hana.service
```

It is highly encouraged to deploy this script. Initially, it will be made available by email request to the ISICC infoservice (isicc@de.ibm.com).

Additional or referenced documents

IBM product documentation:

- Managing persistent memory volume
  https://www.ibm.com/support/knowledgecenter/en/9040-MR9/p9efd/p9efd_lpar_pmem_settings.htm

SAP product documentation:

- Persistent Memory - SAP HANA Administration Guide for SAP HANA Platform
  https://help.sap.com/viewer/6b94445c94ae495c83a19646e7c3fd56/2.0.04/en-US/1f61b13e096d4ef98e62c676debf117e.html

SAP notes:

- SAP HANA on IBM Power Systems: Allowed Hardware (SAP Note 2188482)
  https://launchpad.support.sap.com/#/notes/2188482
- Sizing SAP HANA with Persistent Memory (SAP Note 2786237)
  https://launchpad.support.sap.com/#/notes/2786237
- SAP HANA Persistent Memory -FAQ (SAP Note 2700084)
  https://launchpad.support.sap.com/#/notes/2700084
- HANA: How to set allocation limit for tenant databases (SAP Note 2175606)
  https://launchpad.support.sap.com/#/notes/2175606

Copyrights and Trademarks

Disclaimer and Special Notices

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

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems.

Furthermore, some measurement may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products.

All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs

ANY INFORMATION HEREIN IS PROVIDED "AS IS" WITHOUT WARRANTY OR INDEMNIFICATION OF ANY KIND BY IBM AND DO NOT ANY EXPRESS OR IMPLIED, WARRANTIES OR CONDITIONS OF MERCHANTABILITY, FITNESS OR USAGE FOR PARTICULAR PURPOSE AND ANY WARRANTY OR CONDITION OF NON-INFRINGEMENT.