Enabling High Bandwith Memory in Flat Memory Mode on Linux

Planning / Implementation

The Intel Xeon CPU Max Series integrates Intel Xeon Scalable processors with high bandwidth memory (HBM) and is architected to increase memory bandwidth for applications in data-intensive workloads, such as modeling, artificial intelligence, deep learning, high performance computing (HPC) and data analytics.

It integrates 64 GB of high bandwidth in-package memory (HBM) in the CPU package, as well as other I/O functionalities such as PCI Express 5.0 and CXL 1.1 etc. Xeon Max CPUs provide memory (HBM) capacity for the CPU cores in packages, to fit most common HPC workloads.

More details about the benefits of Intel Xeon CPU Max Series can be found at:
https://www.intel.com/content/www/us/en/products/docs/processors/xeon/xeon-max-series-product-brief.html

Architecture

In this section, we introduce the architecture of the Intel Xeon CPU Max Series and the CPU configuration from the hardware and software perspective.

The processor contains four HBM2e stacks with a total high bandwidth memory (HBM) capacity of 64 GB per processor package, in addition to eight DDR memory channels, as shown in the following figure.

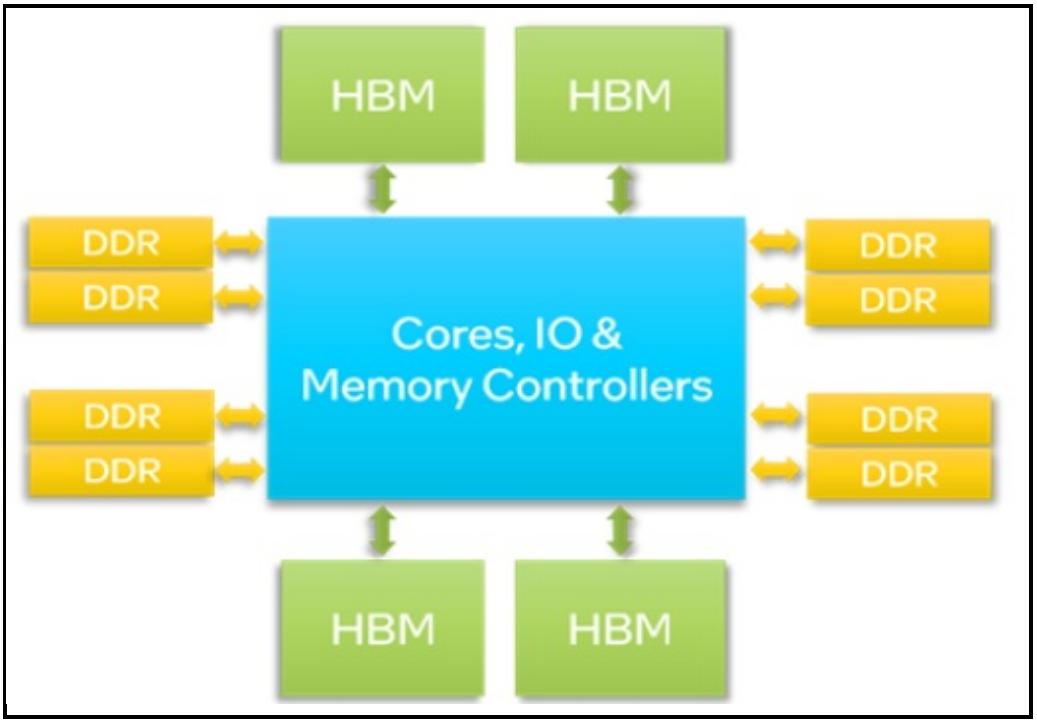

Figure 1. CPU block diagram of the Intel Xeon CPU Max Series

For a two-socket system, the two processors are connected by up to four Intel Ultra Path Interconnect (Intel UPI) links in the system. A two-socket system has a total of 128 GB of HBM capacity.

CPU memory modes configuration: Hardware view

This section describes each configurable CPU memory mode from a hardware point of view.

Depending on the DDR memory DIMMs population, the hardware configuration in the UEFI and software setup in OS, there are three HBM Memory Modes (HBM-only, Flat_1LM and Cache_2LM, see the Software view section for a detailed explanation) that can be configured, as shown in the following figure.

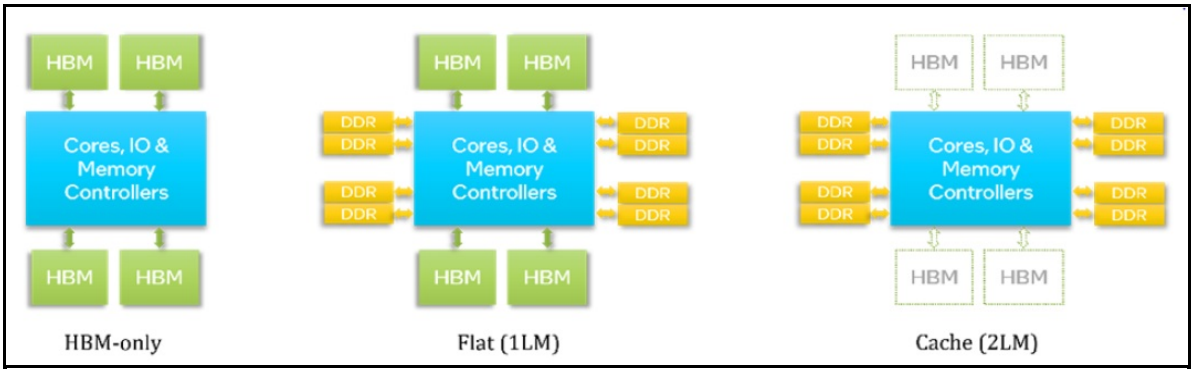

Figure 2. HBM Memory Modes

There is another Xeon Max CPU configuration option SNC (Sub NUMA Cluster) in the UEFI which will change the NUMA node partitions from the OS perspective: when the SNC is disabled, each processor package acts as a single NUMA node (as with any ordinary processor); when SNC is enabled, each processor package in the system is partitioned into 4 sub-NUMA nodes, as shown in the figure below.

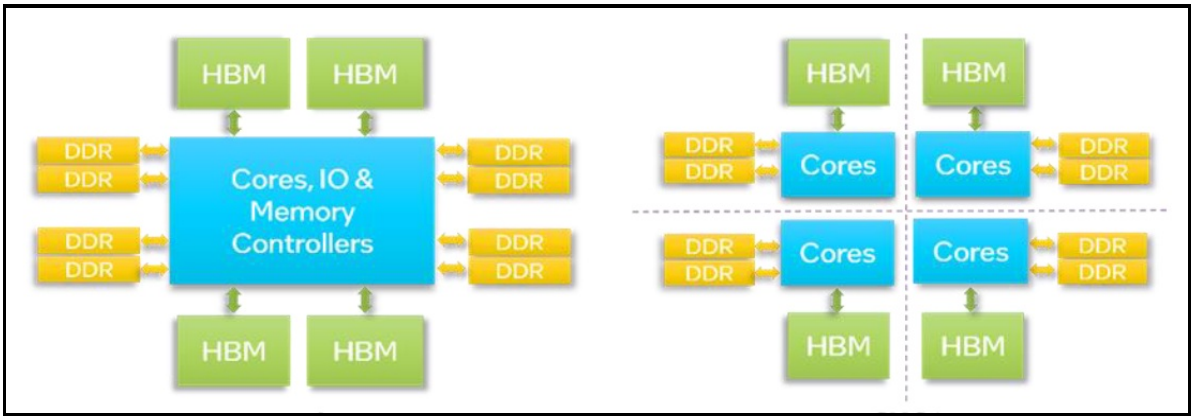

Figure 3. Sub NUMA Cluster modes (left: disabled; right: enabled)

In our test setup, we will disable SNC in the UEFI to simplify the setup, configuration and verification.

CPU memory modes configuration: Software view

From the software perspective, the processor exposes HBM to software (OS and applications) using three memory modes.

- HBM-only mode

  When no DDR is installed, HBM-only mode is automatically selected. The only memory available to the OS and applications in this mode is HBM. The OS may see all the installed HBM in this mode, while applications will see what the OS exposes. Hence the OS and the applications can readily utilize HBM. However, the OS, background services, and applications must share the available HBM capacity (64GB per processor).

- Cache or 2-Level Memory (2LM) mode

  When DDR is installed, it is possible to use HBM as a memory side cache for DDR by setting Memory Hierarchy to Cache mode in the UEFI setup menu before booting to OS. In this mode, only DDR address space is visible to software and HBM functions as a transparent memory-side cache for DDR. Therefore, applications in the OS do not need modifications to use the cache mode. In this mode, the HBM acts a transparent cache and is transparent to the OS.

- Flat or 1-Level Memory (1LM) mode

  When DDR memory is installed, it is possible to expose both HBM and DDR to software by setting Memory Hierarchy to Flat mode in the UEFI setup menu before booting to OS. HBM and DDR are exposed to software as separate address spaces in this mode. DDR is exposed as a separate address space (NUMA node) and HBM as another address space (NUMA node). Users need to use NUMA-aware tools (e.g., numactl) or libraries to utilize HBM in this mode. Additional OS configuration is necessary before HBM can be accessed as part of the regular memory pool.

HBM support on Lenovo ThinkSystem servers

Intel HBM CPUs are supported on the following Lenovo ThinkSystem servers:

- ThinkSystem SD650 V3
- ThinkSystem SD650-I V3 (only supports Cache mode)

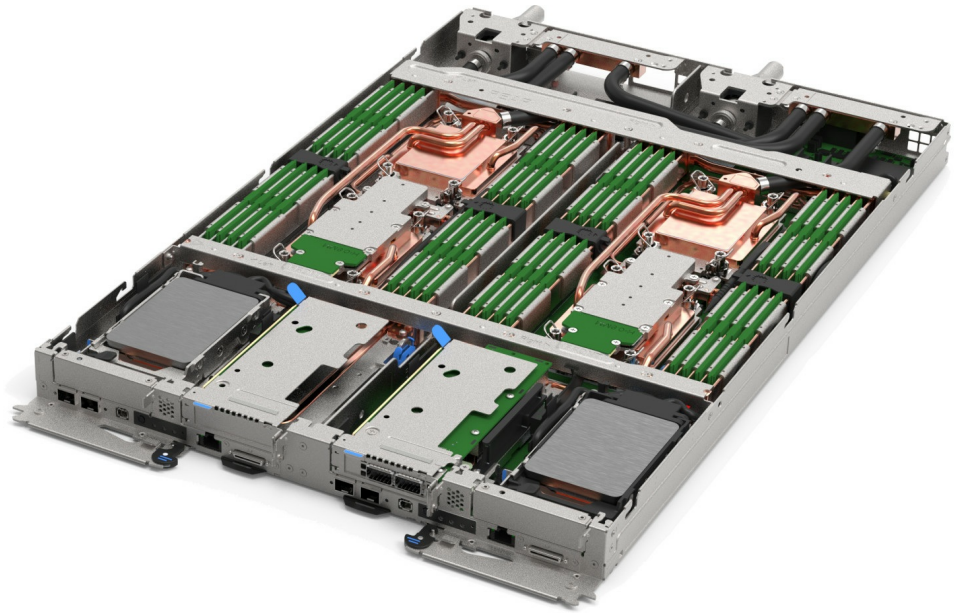

Figure 4. ThinkSystem SD650 V3 server tray with two distinct two-socket nodes

Configuring UEFI to enable HBM in Flat Memory mode

In this section, we will demonstrate how to set up and use HBM Flat memory mode on a Lenovo ThinkSystem SD650 V3 server with Intel Xeon CPU Max series processors.

In this test setup, two Xeon Max CPUs are installed in a two-socket node, providing a total of 128 GB of HBM capacity for the node, and the SD650 V3 server tray hosting the Xeon Max CPUs node is installed in a DW612S enclosure, as shown in the following figure.

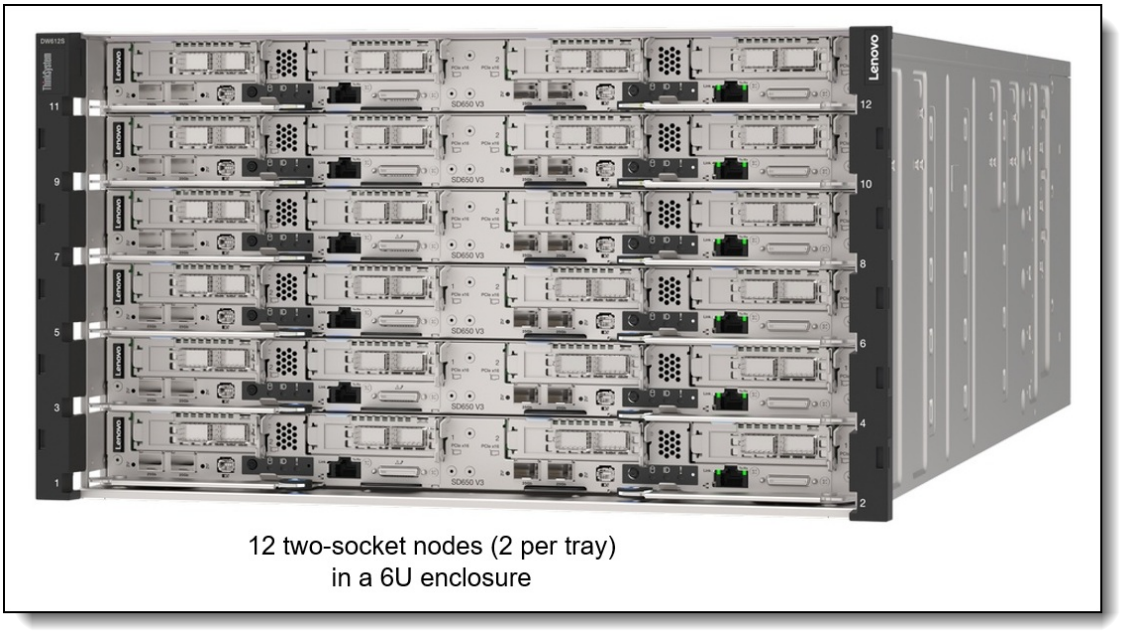

Figure 5. The DW612S enclosure with six trays of SD650 V3 servers (each tray is two nodes)

The following Linux operating systems are the minimum OS versions supported on the ThinkSystem SD650 V3 that also support HBM Flat Memory (1LM) mode:

- Red Hat Enterprise Linux 9.0 or later
- SUSE Linux Enterprise Server 15 SP4 or later
- Ubuntu 22.04 or later

The steps to configure the server UEFI to enable HBM in Flat Memory Mode are as follows.

1. When prompted during server UEFI boot, press F1 to enter System Setup

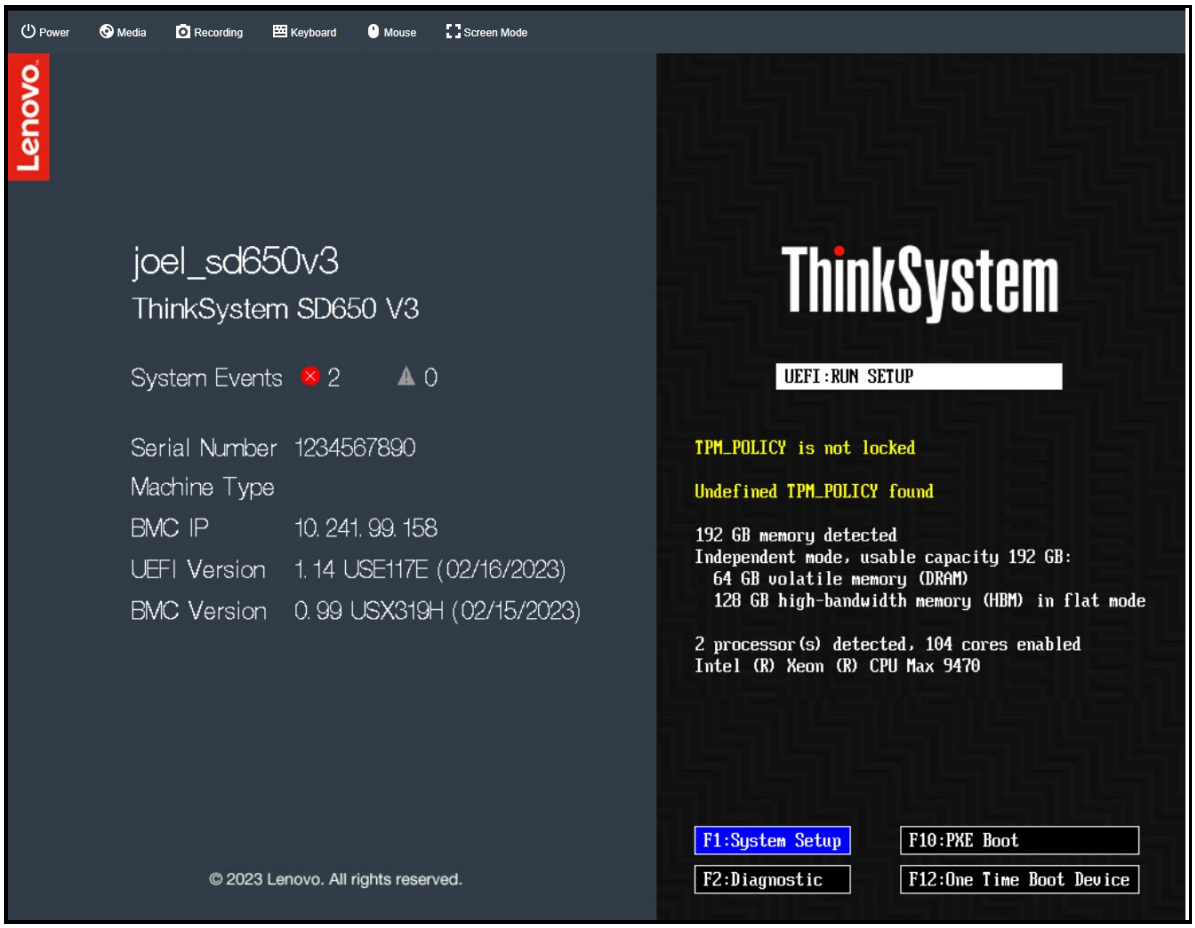

Figure 6. UEFI screen and F1:System Setup

Note that, as you can see from the right side of the UEFI screen, 64 GB DRAM and 128 GB HBM are installed on the system.

2. Select System Settings > Memory > Memory Hierarchy. The following figure is displayed.

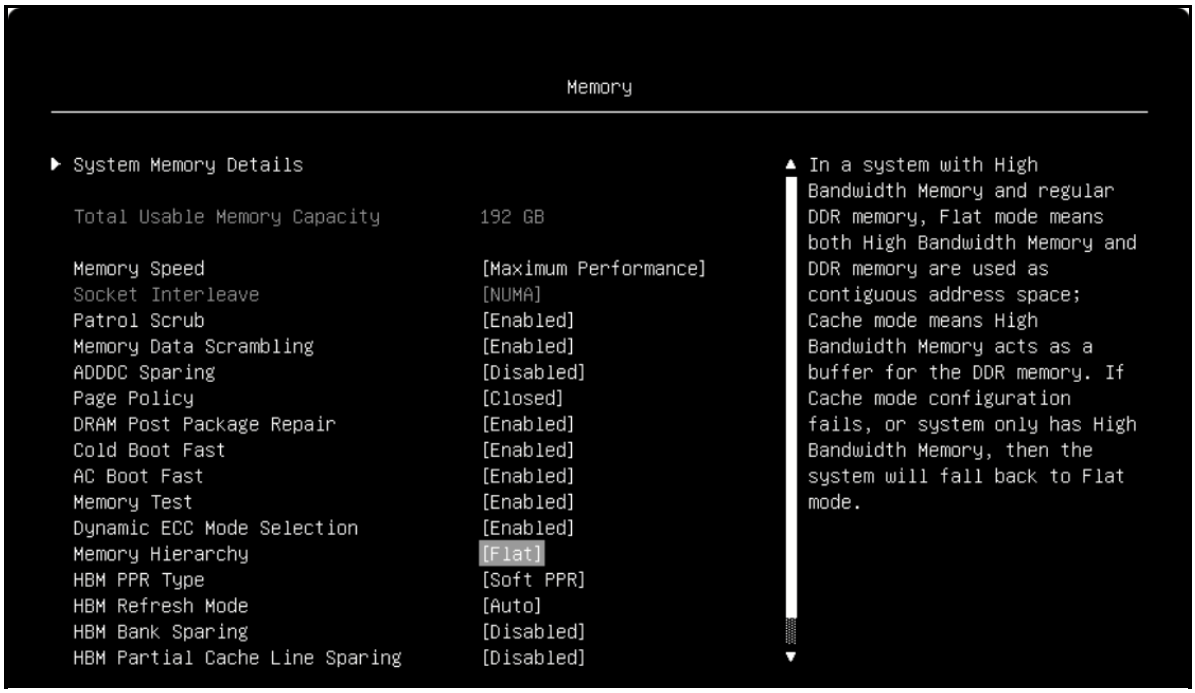

Figure 7. Memory Hierarchy setup

3. Press Enter to select Flat if it isn't already.
4. Use ESC key to go back to the beginning of UEFI setup and Select System Settings > Processors > SNC
5. Press Enter to select Disabled if it isn't already, as shown in the figure below.

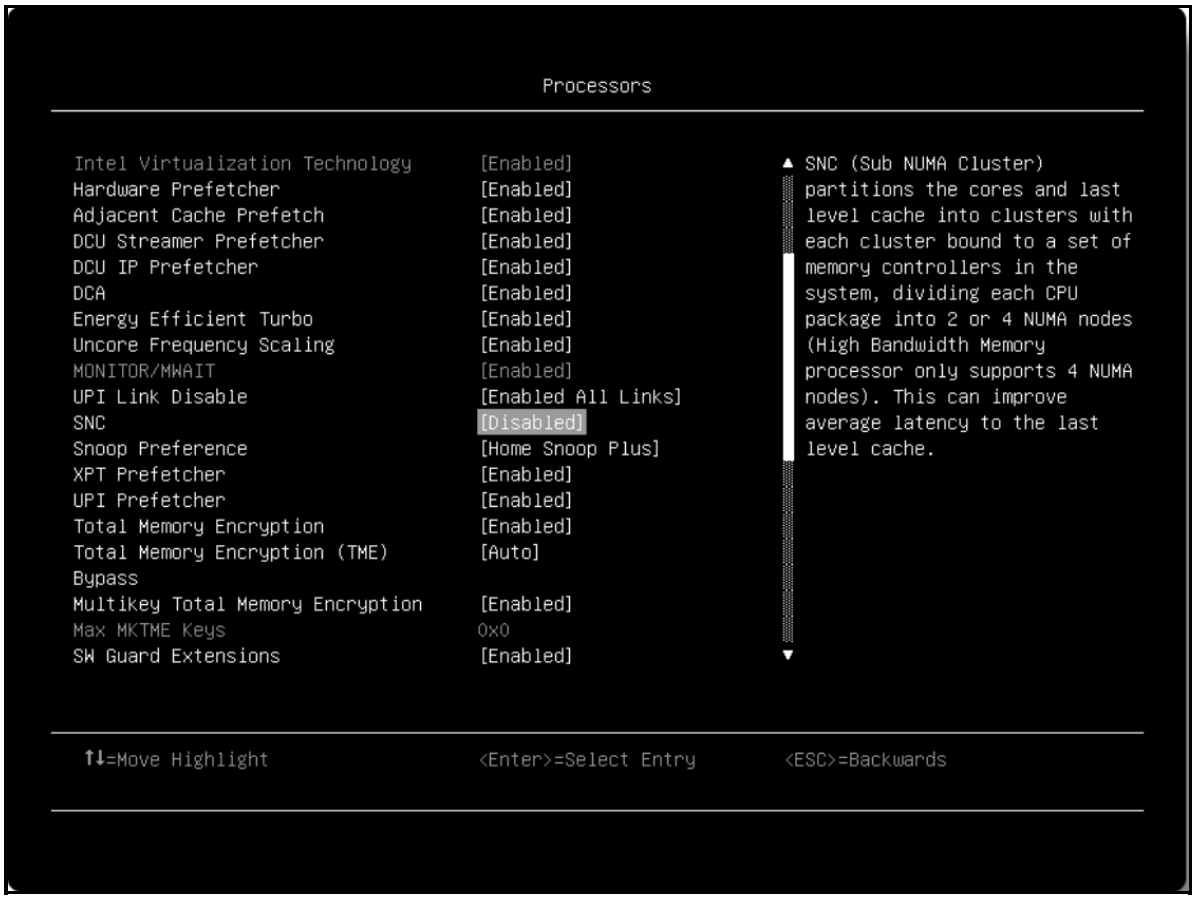

Figure 8. SNC setup

6. Save the settings and reboot the system.

At this point, you can start to install RHEL 9.1. Just follow the standard RHEL 9.x OS installation process to complete the OS installation. For more details, refer to: https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/9/html/performing_a_standard_rhel_9_installation/index

Exposing HBM in Flat Memory Mode in the Linux OS

After the RHEL 9.1 OS installation, boot the OS, and do the following to enable HBM in the OS:

1. Check the total available memory and the NUMA nodes
   Use the following command to check the total memory available to the OS:

```
free
```

   Output is shown below:

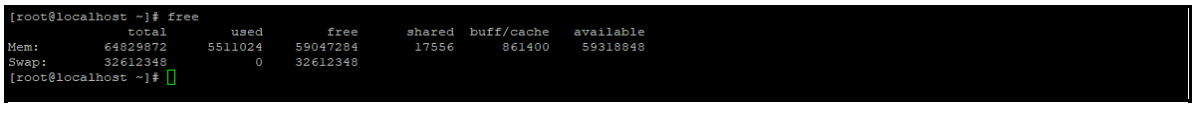

Figure 9. free memory before configuration in OS

As can be seen, the total available memory size under OS is around 64GB, which is the total DDR DRAM memory size, excluding HBM capacity.

Use the following command to check whether the HBM NUMA nodes are visible:

```
numactl -H
```

```
[root@yyy ~]# numactl -H
available: 2 nodes (0-1)
node 0 cpus: 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48
49 50 51 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126 127 128 129 130 131 132 133 134 135 136 137 138 1
39 140 141 142 143 144 145 146 147 148 149 150 151 152 153 154 155
node 0 size: 31351 MB
node 0 free: 27667 MB
node 1 cpus: 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97
 98 99 100 101 102 103 156 157 158 159 160 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180 181 182 183 184 185 186 187
 188 189 190 191 192 193 194 195 196 197 198 199 200 201 202 203 204 205 206 207
node 1 size: 31959 MB
node 1 free: 29678 MB
node distances:
node   0   1
  0:  10  21
  1:  21  10
[root@yyy ~]# ls /dev/kmem
ls: cannot access '/dev/kmem': No such file or directory
[root@yyy ~]#
```

Figure 10. NUMA nodes before configuration in OS

At this moment, the HBM NUMA nodes are not visible either.

Note that, after selecting **Flat** as the Memory Hierarchy in UEFI (Figure 7) and booting to the OS, the system boots up with only DDR exposed to the OS and applications. The HBM is still not visible in the default memory pool since HBM is marked as special-purpose memory. This design choice was made to prevent the OS from allocating and reserving valuable HBM memory during the OS boot process. Since the HBM is "hidden" from the OS during the boot process, the OS cannot allocate or reserve HBM memory.

So, the above outputs are expected results, as we have not yet installed the necessary software packages in the OS, nor have we done the necessary configuration.

2. Install the necessary software packages in the OS with the following commands:

```
dnf install daxctl ndctl
```

Output is shown below.

```
Installed:
  daxctl-71.1-7.el9.x86_64          daxctl-libs-71.1-7.el9.x86_64          ndctl-71.1-7.el9.x86_64          ndctl-libs-71.1-7.el9.x86_64

Complete!
[root@yyy ~]#
```

Figure 11. Software packages installed in OS

3. Execute the configuration commands
The above instructions only describe how to install the necessary software packages in the OS. To enable the "hidden" HBM to appear in the OS, additional configuration is required with the following commands (one command for one HBM NUMA node, thus we need two commands for a two-socket system):

```
daxctl reconfigure-device -m system-ram dax0.0
daxctl reconfigure-device -m system-ram dax1.0
```

The above two commands reconfigured dax devices dax0.0 and dax1.0 to the system RAM mode, which converted dax devices to be regular system memory in hot−pluggable system RAM mode. Each dax device needs to be reconfigured so that OS can see its associated memory.

For more details about the daxctl command utility usage, please refer to the document below:
https://docs.pmem.io/ndctl-user-guide/daxctl-man-pages/daxctl-reconfigure-device

4. If SNC mode was enabled in the Processor page in UEFI (Figure 8), further configuration is required, since each processor package in the system is partitioned into 4 sub-NUMA nodes (as described in the

Architecture section), so total 8 sub-NUMA nodes in a two-socket system need to be reconfigured:

```
daxctl reconfigure-device -m system-ram dax2.0
daxctl reconfigure-device -m system-ram dax3.0
daxctl reconfigure-device -m system-ram dax4.0
daxctl reconfigure-device -m system-ram dax5.0
daxctl reconfigure-device -m system-ram dax6.0
daxctl reconfigure-device -m system-ram dax7.0
```

Verifying the results

Perform the following steps to verify HBM is visible to the OS.

1. Verify the total memory
   Use the **free** command to check the total available memory for OS. Below is the command output.

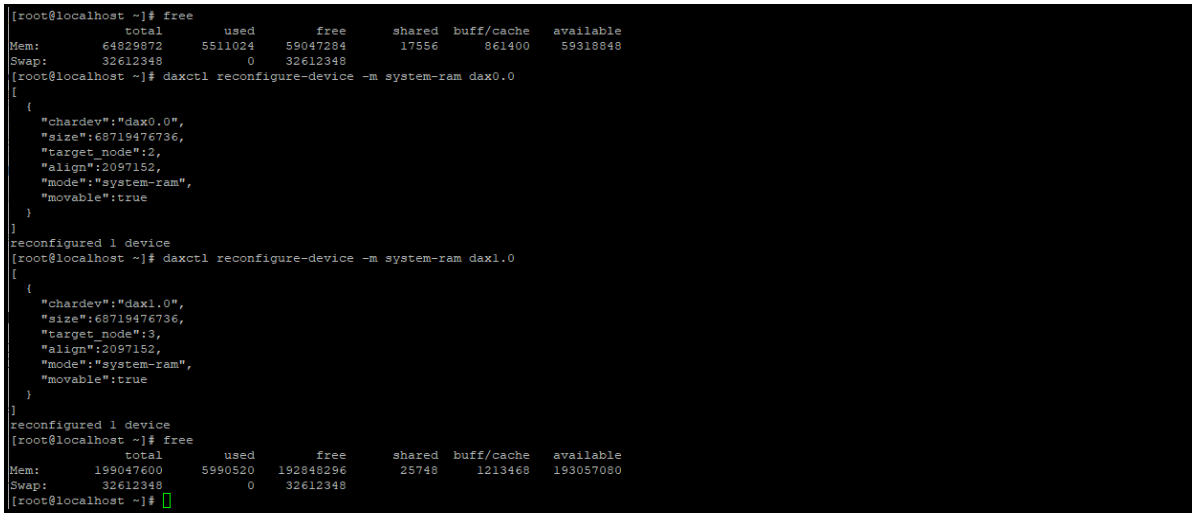

   Figure 12. free memory after configuration in OS

   As can be seen from figure 12, additional 128 GB (from HBM capacity) was added to the original 64 GB DRAM capacity and the total memory added up to around 192 GB.

2. Verify the NUMA nodes
   Use the following command to verify that the HBM nodes:

```
numactl -H
```

   The entire HBM capacity are visible, as below figure shows (for the UEFI SNC disabled case).

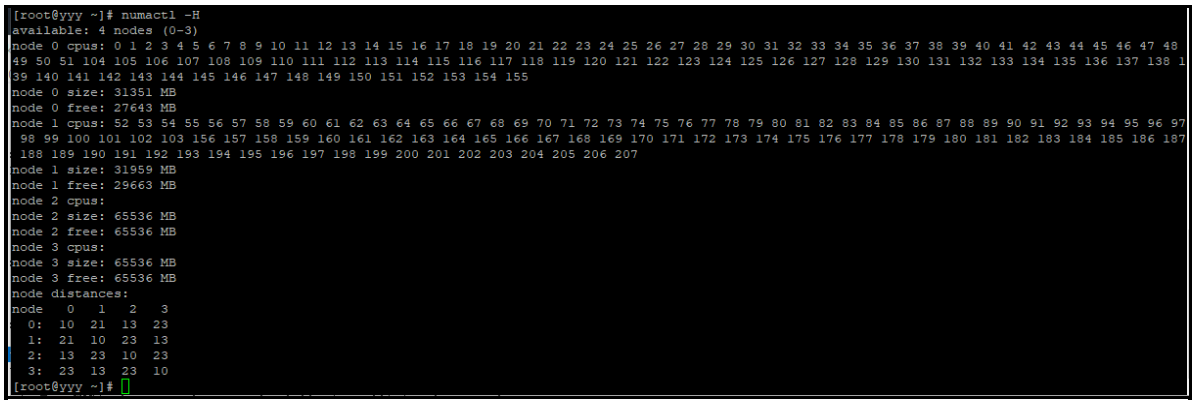

Figure 13. NUMA nodes after configuration in OS

Two more HBM NUMA nodes were just being added. The total number of available NUMA nodes grew from 2 (Figure 10) to 4 (Figure 13) after configuration.

3. The daxctl configuration steps described in step 3 in Exposing HBM in Flat Memory Mode in the Linux OS need to be carried out each time the system boots.

   Therefore, it would be convenient to put the above configuration commands in a script file and enable a systemd service file to execute it each time after the OS boots, as follows:

   a. Create the hbm_startup.sh script file with contents shown below and place it in /usr/local/bin

```
#!/usr/bin/bash

#########################################################
# hbm_startup.sh
# This program should be placed in /usr/local/bin
#########################################################
#
#
# Configure the HBM memory
daxctl reconfigure-device -m system-ram dax0.0
daxctl reconfigure-device -m system-ram dax1.0

# Unmark below commands to make them effective if SNC mode was enable
d in the Processor page in UEFI
# daxctl reconfigure-device -m system-ram dax2.0
# daxctl reconfigure-device -m system-ram dax3.0
# daxctl reconfigure-device -m system-ram dax4.0
# daxctl reconfigure-device -m system-ram dax5.0
# daxctl reconfigure-device -m system-ram dax6.0
# daxctl reconfigure-device -m system-ram dax7.0
```

   b. Create a systemd service file: /etc/systemd/system/hbm_startup.service and add the contents shown below:

```
#########################################################
# hbm_startup.service
# This program should be placed in /etc/systemd/system
#########################################################
#
#

[Unit]
Description=runs /usr/local/bin/hbm_startup.sh

[Service]
ExecStart=/usr/local/bin/hbm_startupp.sh

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

c. Enable the service with below command so that the hbm_startup.sh script will get started to configure HBM each time after the OS boots.

```
systemctl enable hbm_startup.service
```

The above test setup demonstrated the HBM Flat memory mode enabling and using the Intel Xeon CPU Max Series processors and SD650 V3 server.

More information

For more details about the configuration and tuning of the Intel Xeon CPU Max Series processors, refer to:

Intel Xeon CPU Max Series Configuration and Tuning Guide
https://www.intel.com/content/www/us/en/content-details/769060/intel-xeon-cpu-max-series-configuration-and-tuning-guide.html?DocID=769060

For a discussion on the performance of HBM memory in the Intel Xeon Max Series processors compared to DDR5 memory in 4th Gen Intel Xeon Scalable processors, refer to the following document:

Implementing High Bandwidth Memory and Intel Xeon Processors Max Series on Lenovo ThinkSystem Servers
https://lenovopress.lenovo.com/lp1738-implementing-intel-high-bandwidth-memory

For more details about the Lenovo ThinkSystem SD650 V3 Neptune DWC Server, refer to:

Product Guide - Lenovo ThinkSystem SD650 V3 Neptune DWC Server
https://lenovopress.lenovo.com/lp1603-thinksystem-sd650-v3-server

Author

Kelvin Shieh is the OS Development Technical Lead for the Lenovo Infrastructure Solutions Group, based in Taipei, Taiwan.

Related product families

Product families related to this document are the following:

- Processors
- ThinkSystem SD650 V3 server

Notices

Lenovo may not offer the products, services, or features discussed in this document in all countries. Consult your local Lenovo representative for information on the products and services currently available in your area. Any reference to a Lenovo product, program, or service is not intended to state or imply that only that Lenovo product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any Lenovo intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any other product, program, or service. Lenovo may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

Lenovo (United States), Inc.
8001 Development Drive
Morrisville, NC 27560
U.S.A.
Attention: Lenovo Director of Licensing

LENOVO PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. Lenovo may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

The products described in this document are not intended for use in implantation or other life support applications where malfunction may result in injury or death to persons. The information contained in this document does not affect or change Lenovo product specifications or warranties. Nothing in this document shall operate as an express or implied license or indemnity under the intellectual property rights of Lenovo or third parties. All information contained in this document was obtained in specific environments and is presented as an illustration. The result obtained in other operating environments may vary. Lenovo may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Any references in this publication to non-Lenovo 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 Lenovo 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 result 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 measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

This document, LP1810, was created or updated on September 14, 2023.

Send us your comments in one of the following ways:

- Use the online Contact us review form found at:
  https://lenovopress.lenovo.com/LP1810
- Send your comments in an e-mail to:
  comments@lenovopress.com

This document is available online at https://lenovopress.lenovo.com/LP1810.

Trademarks

Lenovo and the Lenovo logo are trademarks or registered trademarks of Lenovo in the United States, other countries, or both. A current list of Lenovo trademarks is available on the Web at https://www.lenovo.com/us/en/legal/copytrade/.

The following terms are trademarks of Lenovo in the United States, other countries, or both:
Lenovo®
ThinkSystem®

The following terms are trademarks of other companies:

Intel® and Xeon® are trademarks of Intel Corporation or its subsidiaries.

Linux® is the trademark of Linus Torvalds in the U.S. and other countries.

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