



# NVIDIA DGX SuperPOD

## Deployment Guide

Featuring NVIDIA DGX A100 and DGX H100 Systems

DG-11251-001 V10  
2024-01-04  
BCM 10.23.12

# Contents

- 1. Introduction..... 1
- 2. Initial Cluster Setup..... 2
- 3. Head Node Configuration .....22
- 4. High Availability.....25
- 5. Slurm Setup.....40

---

# 1. Introduction

This document details deploying NVIDIA Base Command™ Manager (BCM) on NVIDIA DGX SuperPOD™ configurations.

Physical installation and network switch configuration must be completed before deploying BCM. In addition, information about the intended deployment should be recorded in a site survey.

The DGX SuperPOD Deployment Guides contain essential information and should be consulted before deploying the configuration. The DGX H100 version of the deployment guide is available here: <https://docs.nvidia.com/dgx-superpod/index.html>

## 2. Initial Cluster Setup

The deployment stage of a DGX SuperPOD consists of using BCM to provision and manage the Slurm cluster.

1. Configure the NFS server.

User home directories (`home/`) and shared data (`cm_shared/`) directories must be shared between head nodes (such as the DGX OS image) and must be stored on an NFS filesystem for HA availability. Because DGX SuperPOD does not mandate the nature of the NFS storage, the configuration is outside the scope of this document. This DGX SuperPOD deployment uses the NFS V3 export path provided in the [site survey](#): `/var/nfs/general`.

The following parameters are recommended for the NFS server export file `/etc/exports`.

```
/var/nfs/general *(rw, sync, no_root_squash, no_subtree_check)
```

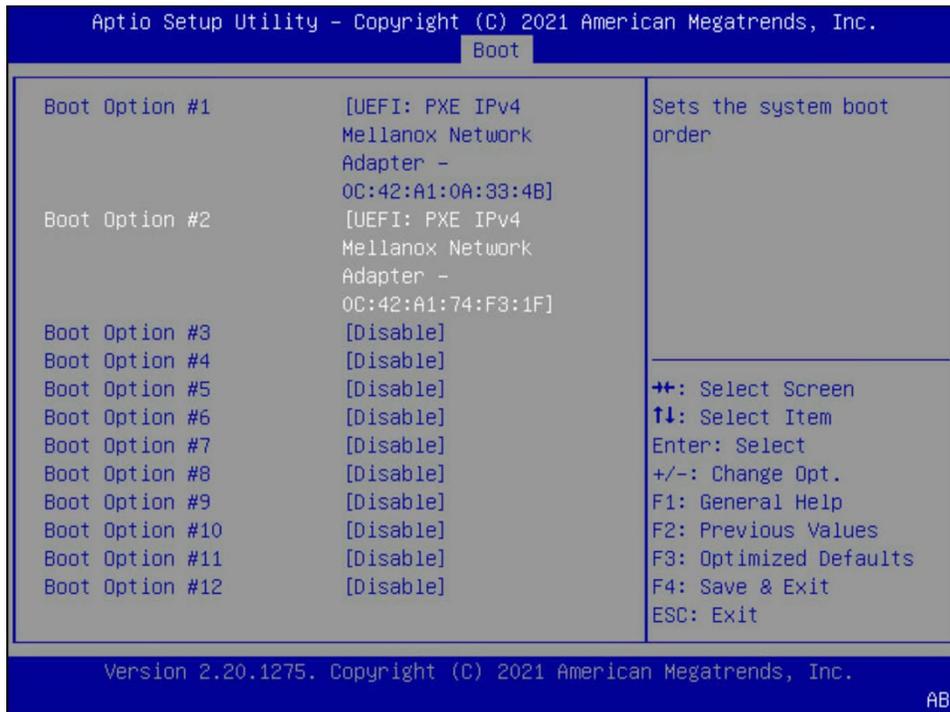
2. Configure the DGX systems to PXE boot by default.

- a. Using either KVM or a crash cart, connect to the DGX system, enter the BIOS menu, and configure Boot Option #1 to be `[NETWORK]`.

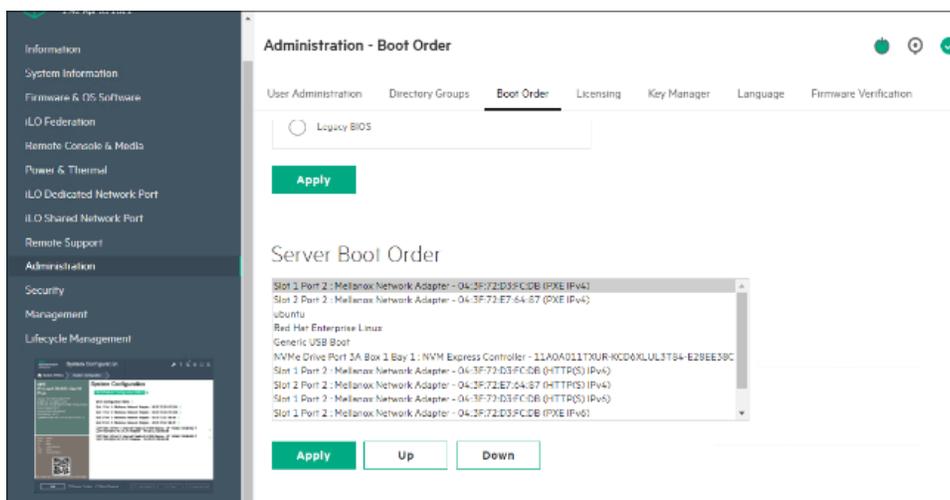


- b. Ensure that other Boot Options are `[Disabled]` and go to the next screen.

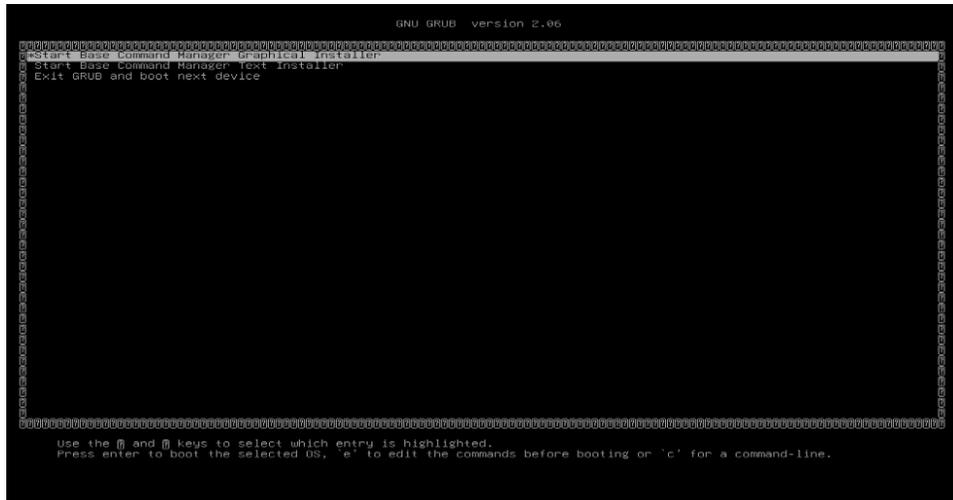
- c. Set Boot Option #1 and Boot Option #2 to use IPv4 for Storage 4-2 and Storage 5-2.



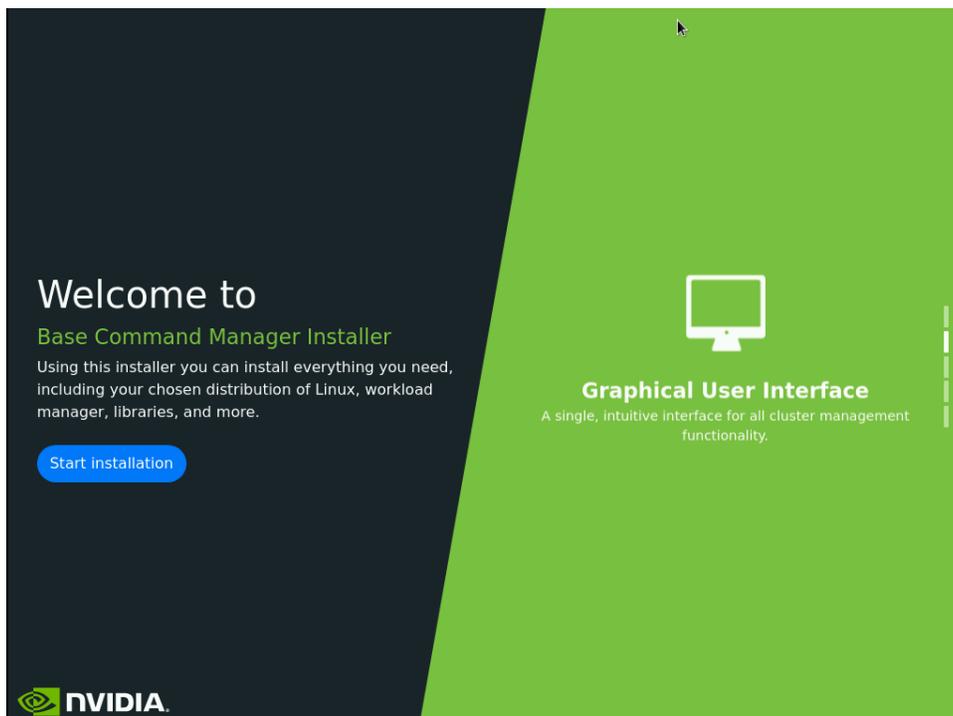
- d. Ensure that other Boot Options are [Disabled].
  - e. Select Save & Exit.
3. On the failover head node and the CPU nodes, ensure that Network boot is configured as the primary option. Ensure that the Mellanox ports connected to the network on the head and CPU nodes are also set to Ethernet mode. This is an example of a system that will boot from the network with Slot 1 Port 2 and Slot 2 Port 2.



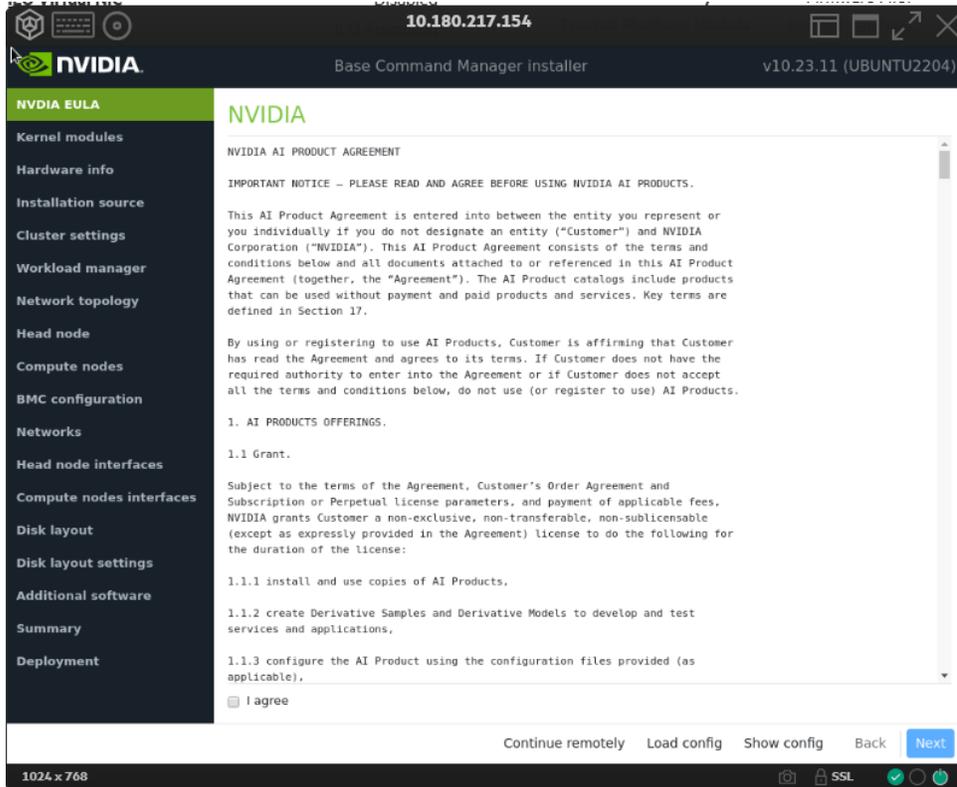
4. Download the BCM installer ISO.
5. Burn the ISO to a DVD or to a bootable USB device.  
It can also be mounted as virtual media and installed using the BMC. The specific mechanism for the latter will vary by vendor.
6. Ensure that the BIOS of the target head node is configured in UEFI mode and that its boot order is configured to boot the media containing the BCM installer image.
7. Boot the installation media.



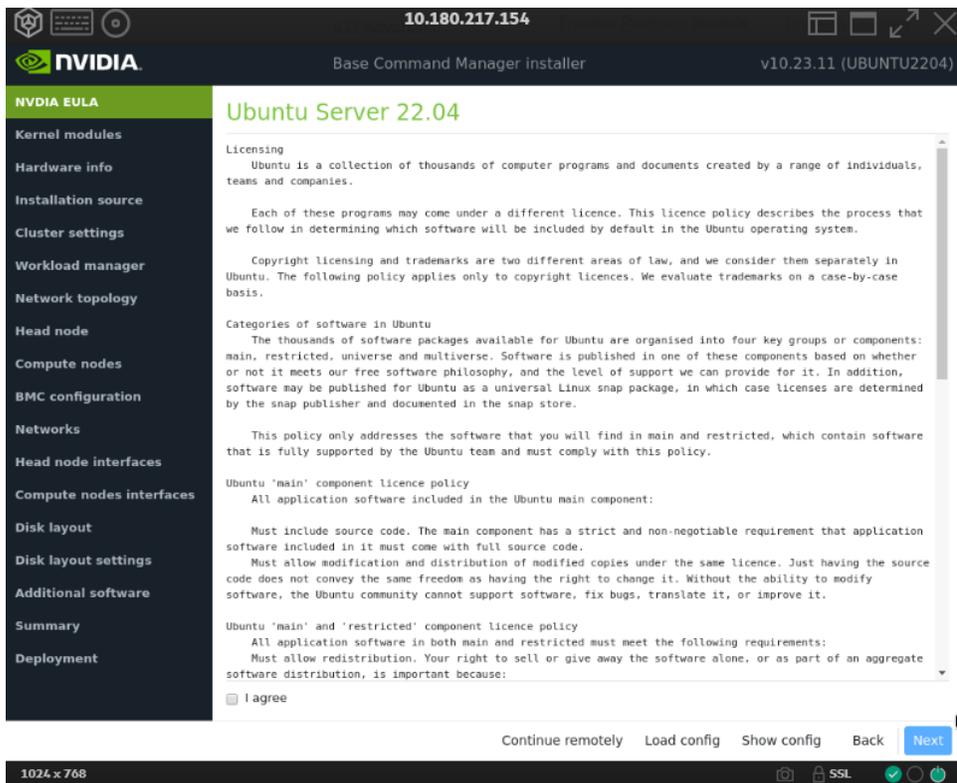
8. At the grub menu, choose Start Base Command Manager Graphical Installer.
9. Select start installation on the splash screen.



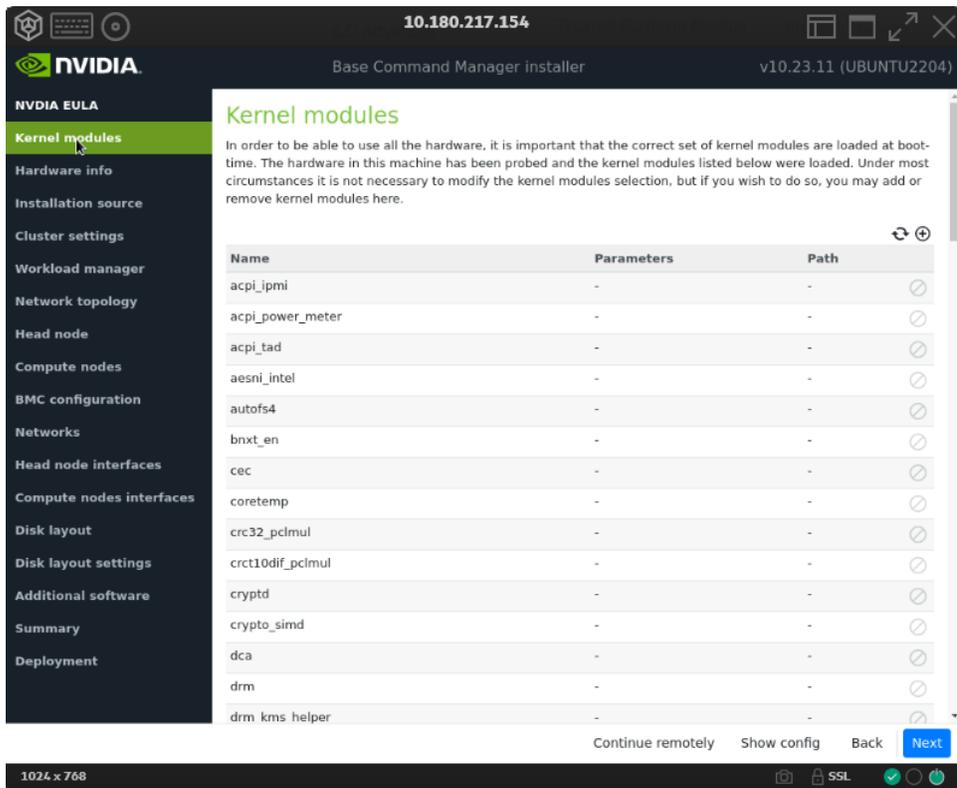
10. Accept the terms of the NVIDIA EULA by checking  I agree and then select Next.



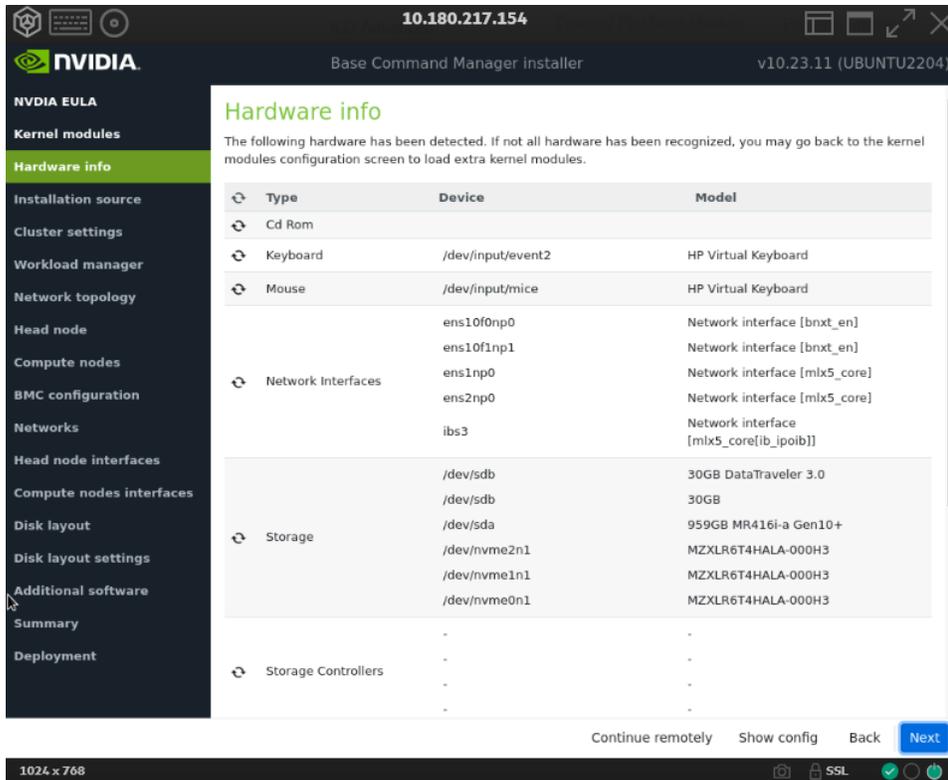
11. Accept the terms of the Ubuntu Server UELA by checking  I agree and then select Next.



12. Unless instructed otherwise, select **Next** without modifying the kernel modules to be loaded at boot time.

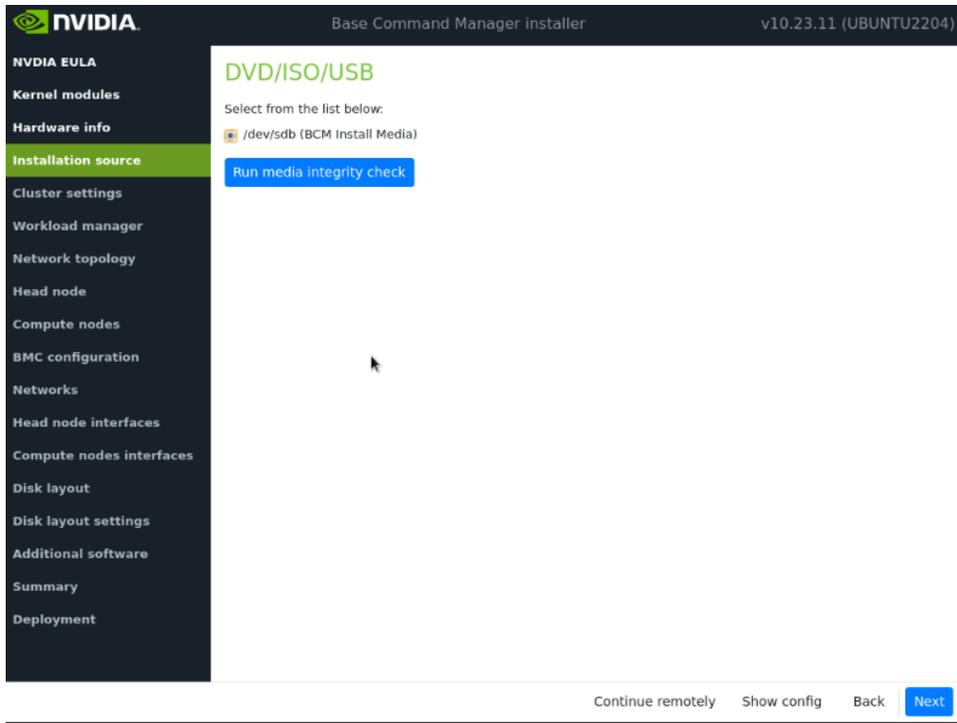


13. Verify that the **Hardware info** is correct and then select **Next**.

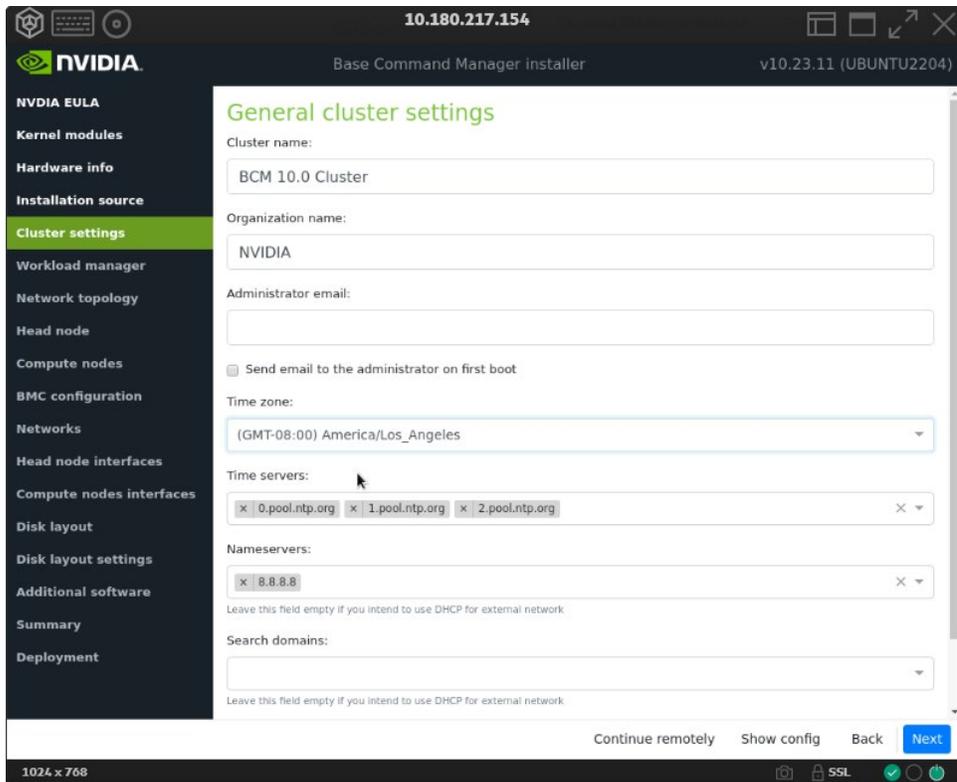


For example, that the target storage device and the cabled host network interfaces are present (in this case three NVMe drives are the target storage device, and ens1np0 and ens2np01 are the cabled host network interfaces).

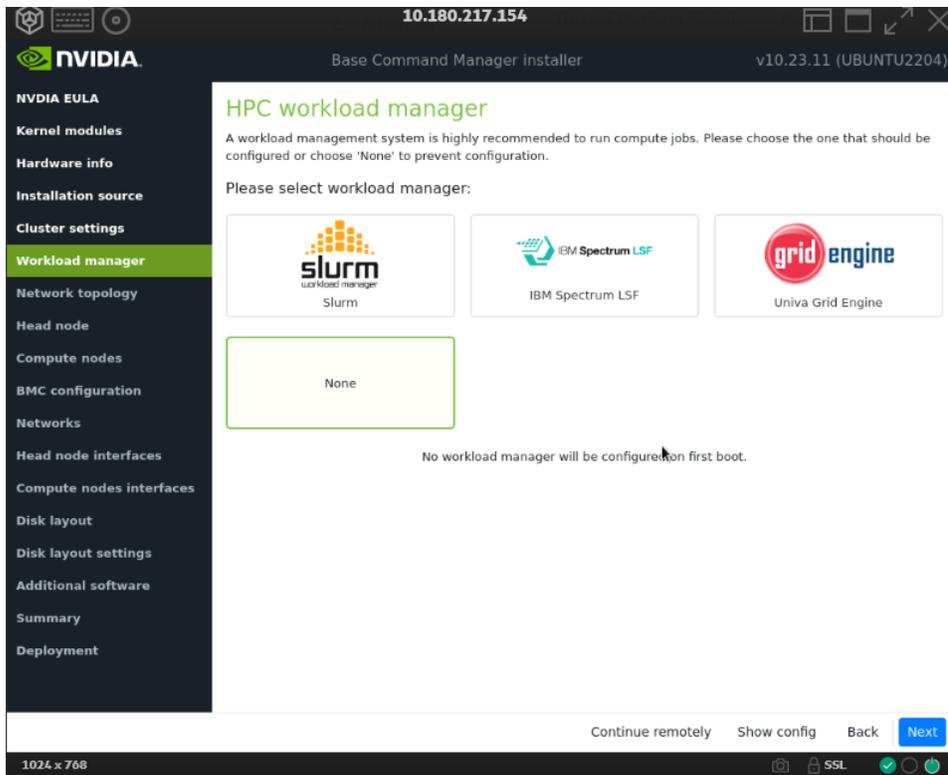
- On the Installation source screen, choose the appropriate source and then select Next. Running a media integrity check is optional.



15. On the Cluster settings screen, enter the required information and then select Next.



16. On the Workload manager screen, choose None and then select Next.



17. On the Network topology screen, choose the network type for the data center environment and then select Next.



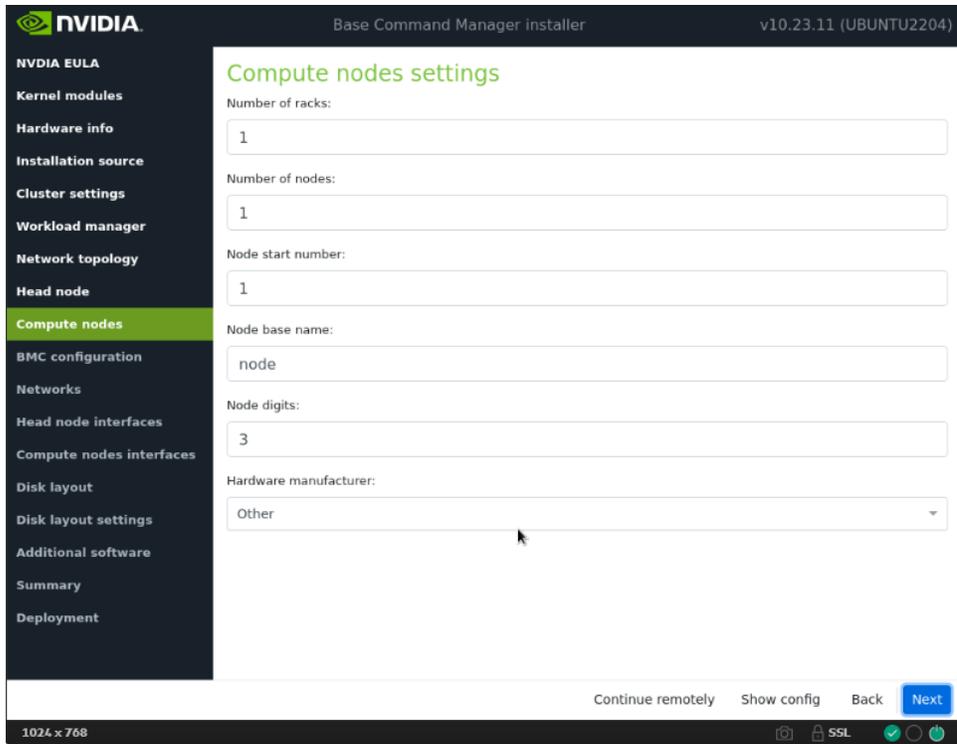
18. On the Head node screen, enter the Hostname, Administrator password, choose Other for Hardware manufacturer, and then select Next

The screenshot displays the 'Head node settings' screen of the NVIDIA Base Command Manager Installer. The window title is '10.180.217.154' and the version is 'v10.23.11 (UBUNTU2204)'. The left sidebar contains a navigation menu with the following items: NVIDIA EULA, Kernel modules, Hardware info, Installation source, Cluster settings, Workload manager, Network topology, **Head node** (highlighted), Compute nodes, BMC configuration, Networks, Head node interfaces, Compute nodes interfaces, Disk layout, Disk layout settings, Additional software, Summary, and Deployment. The main content area is titled 'Head node settings' and contains the following fields:

- Hostname:
- Administrator password:
- Confirm administrator password:
- Hardware manufacturer:

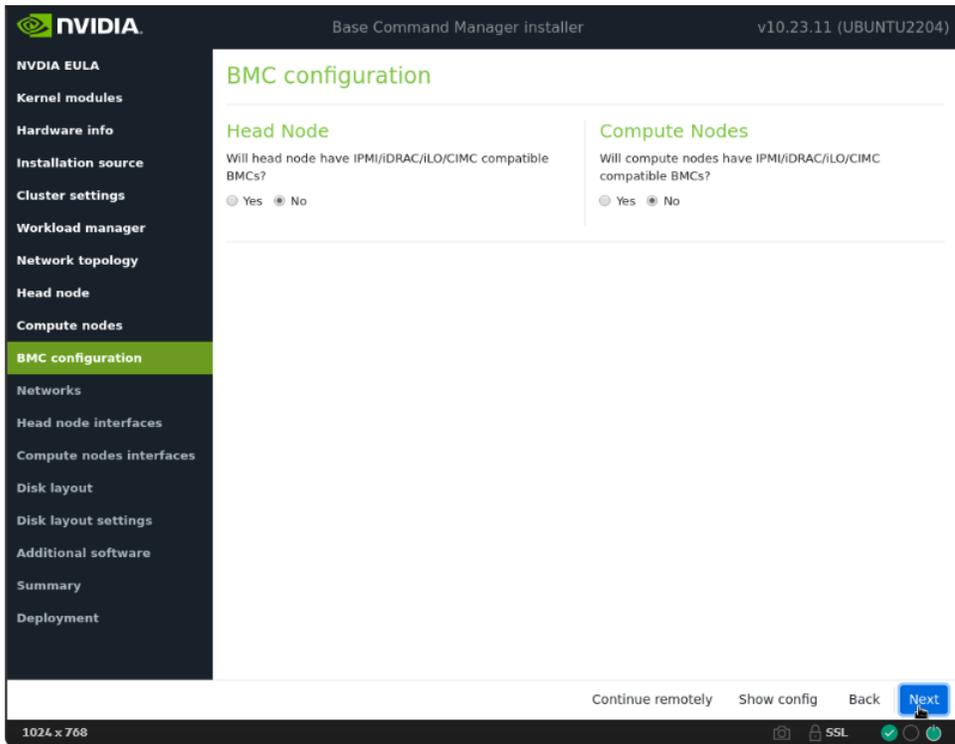
At the bottom right, there are four buttons: 'Continue remotely', 'Show config', 'Back', and 'Next'. The 'Next' button is highlighted in blue. The bottom status bar shows '1024 x 768' resolution and 'SSL' status.

19. Accept defaults in the Compute nodes and then select Next.



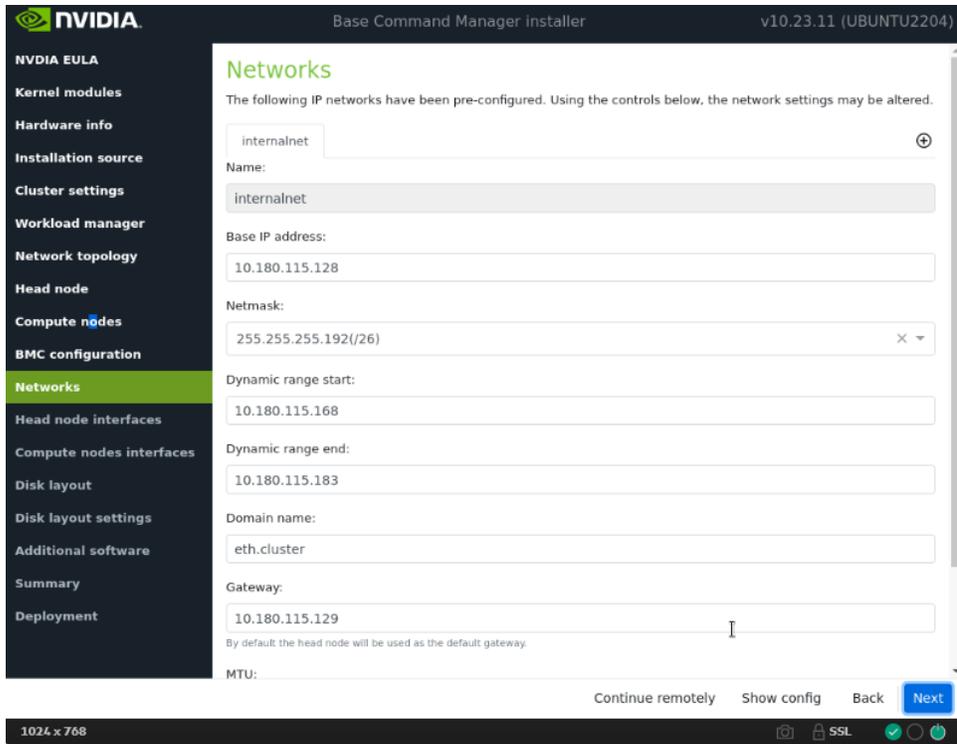
Ensure that the Node base name is node. Other values will be updated later in the installation.

20. On the BMC Configuration screen, choose No for both Head Node and Compute Nodes, and then select Next.



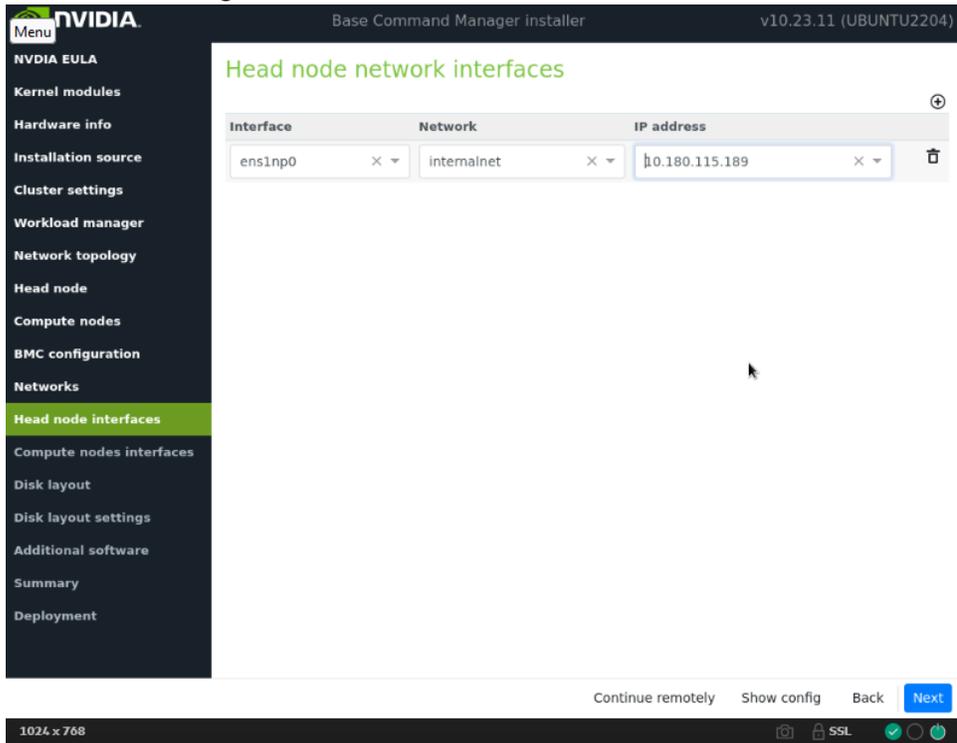
These will be updated later in the post install stages.

21. On the Networks screen, enter the required information for internalnet, and then select Next.



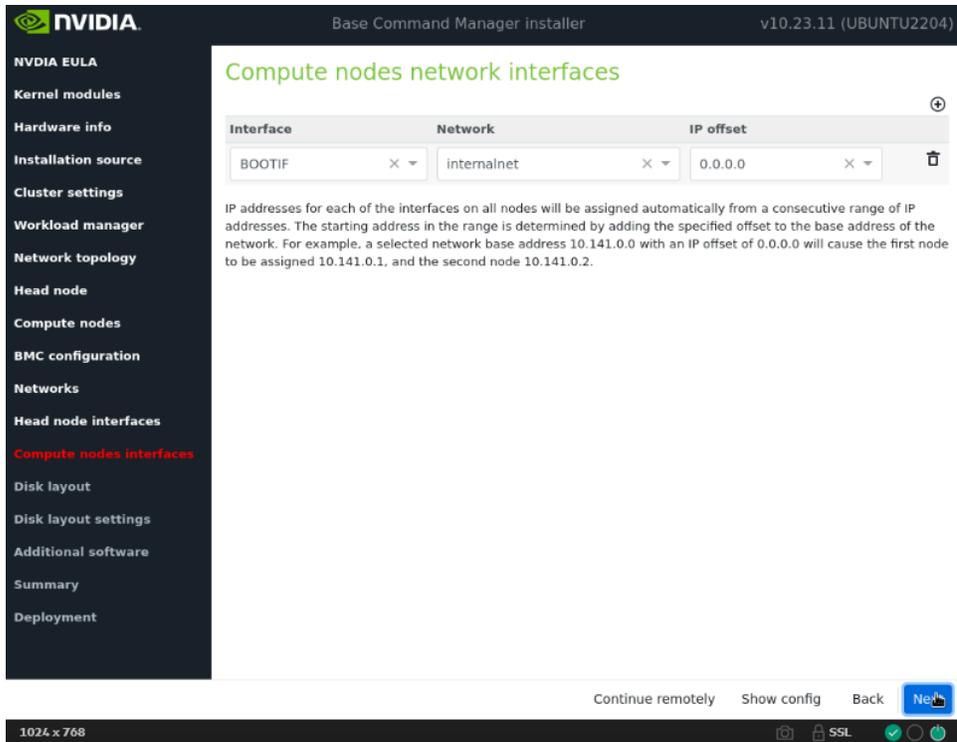
Since a Type 2 network was specified, there are no other network tabs (for example, externalnet or ipminet).

22. On the Head node interfaces screen, ensure that one interface is configured with the head node's target `internalnet` IP, and then select Next.



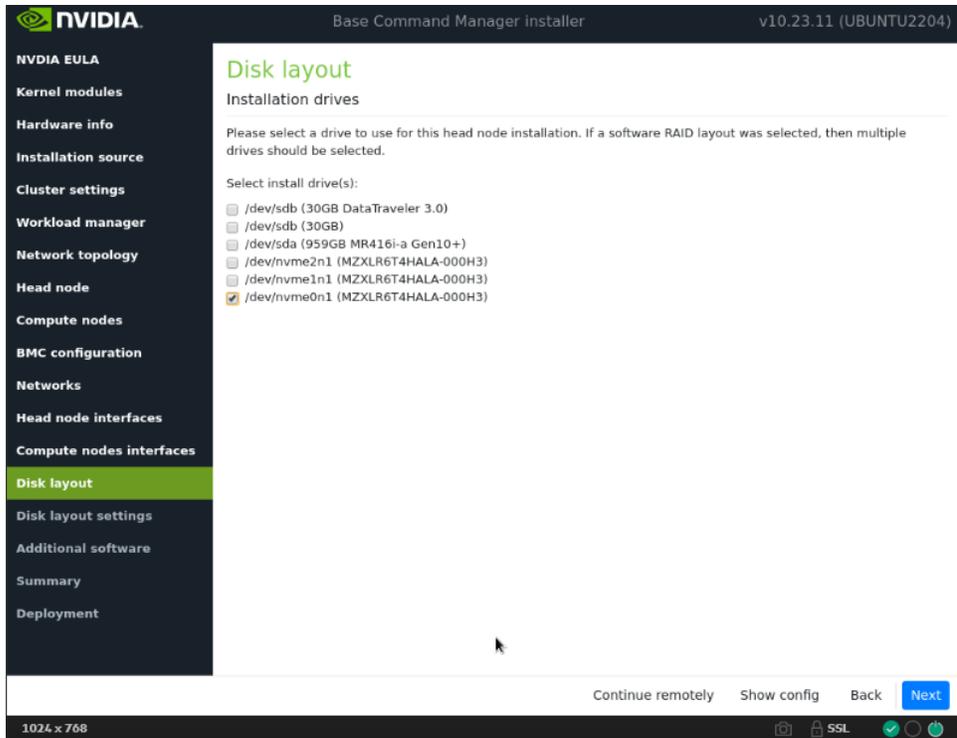
Other interfaces will be configured by the post install script.

23. On the Compute node interfaces screen, leave the default entries, and then select Next.

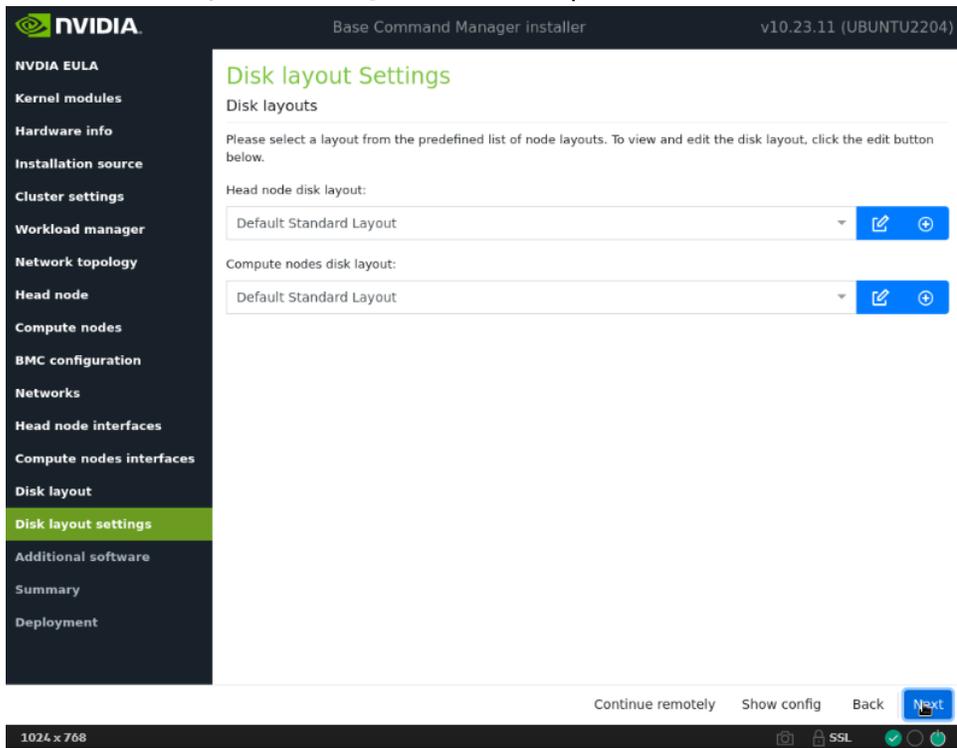


These will be updated post install.

24. On the Disk layout screen, select the target install location (in this case `nvme0n1`) and then select Next.

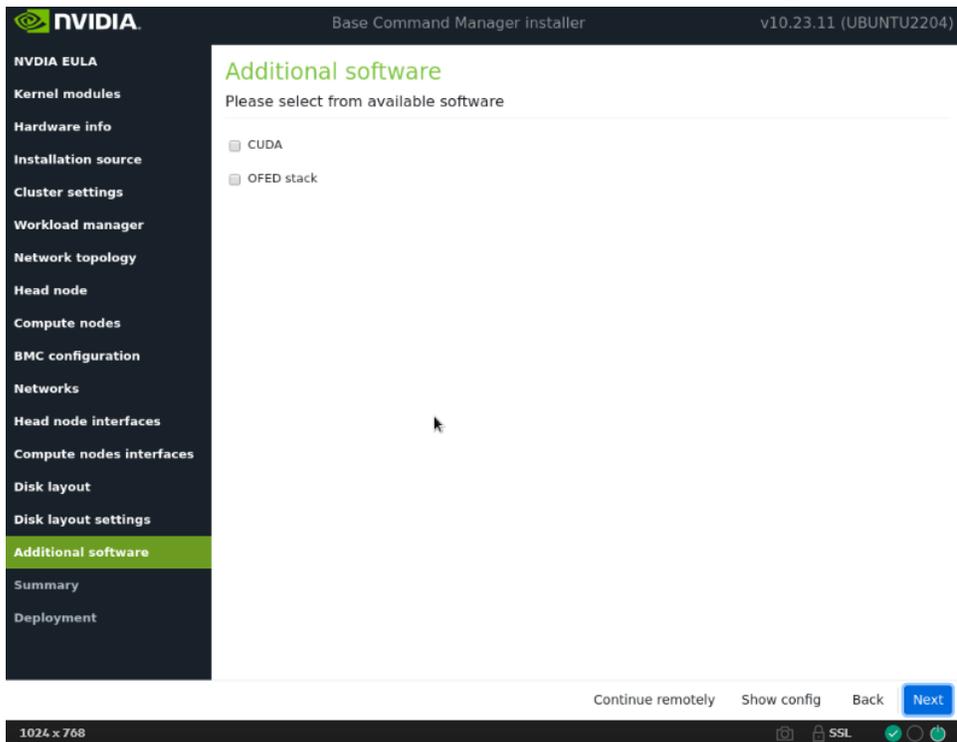


25. On the Disk layout settings screen, accept defaults and then select Next.

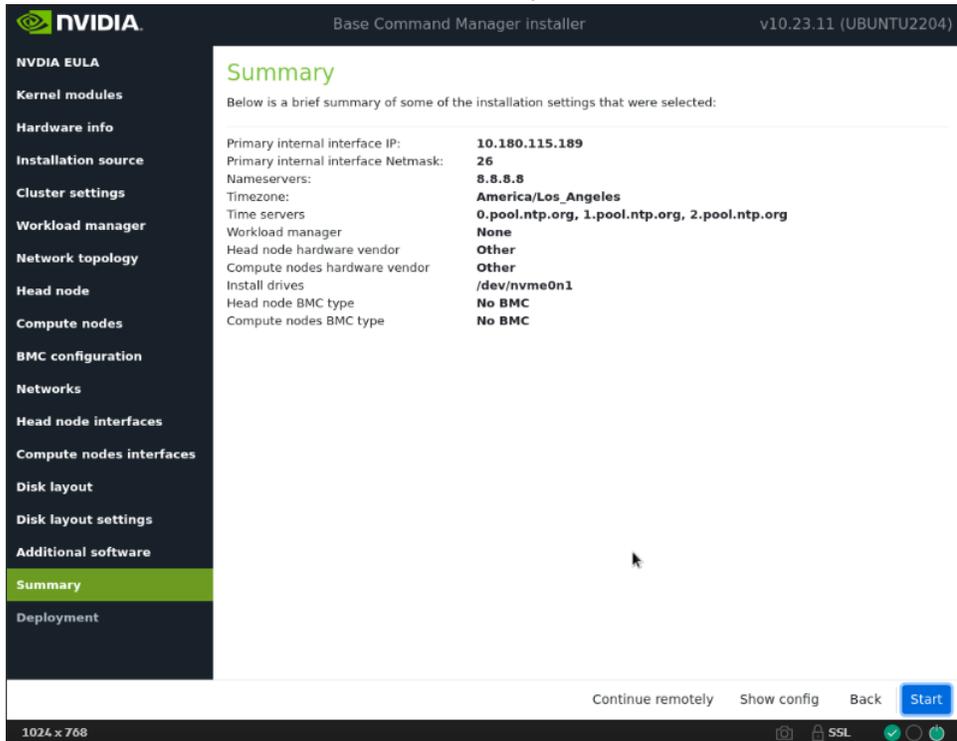


These settings will be updated later in the post installation steps.

26. In the Additional software screen, do not choose anything and then select Next.



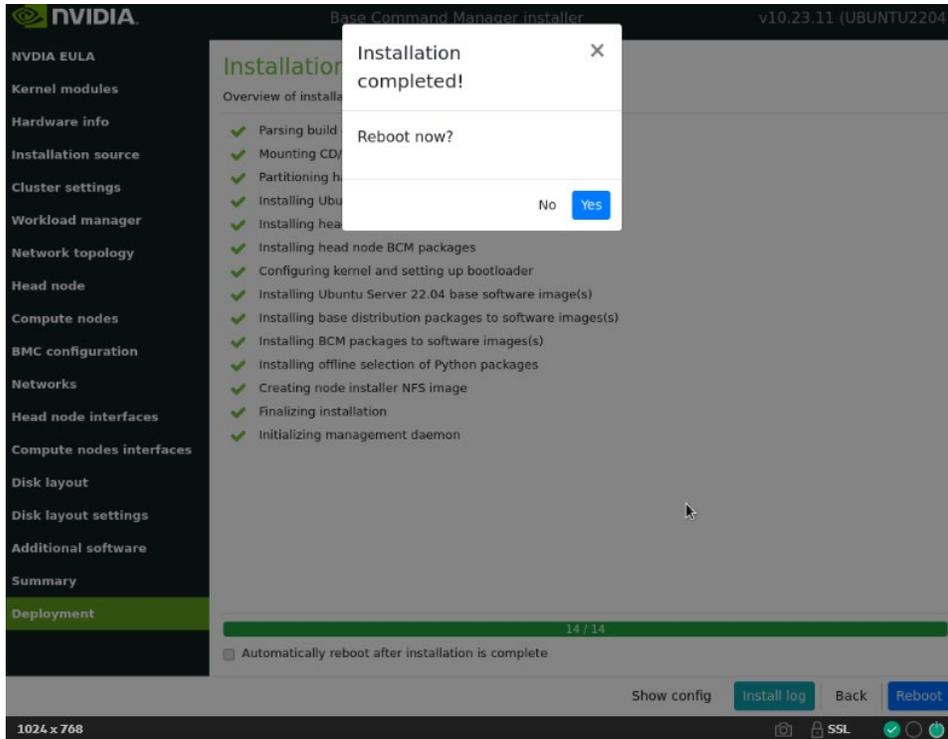
27. Confirm the information on the Summary screen and then select Next.



The Summary screen provides an opportunity to confirm the Head node and basic cluster configuration before deployment begins. This configuration will be updated/modified for DGX SuperPOD after deployment is complete. If any values do

not match expectations, use the Back button to navigate to the appropriate screen to correct any mistakes.

28. Once the deployment is complete, select Reboot.



29. License the cluster by running the request-license and providing the product key.

```
sudo -i request-license  
Product Key (XXXXXX-XXXXXX-XXXXXX-XXXXXX-XXXXXX):
```

30. Load the bcm-superpod-network module.

```
module load bcm-superpod-network
```

31. Run the bcm-netautogen script.

```
bcm-netautogen
```

```

SuperPOD Network Configuration Generator
== What would you like to do? == Generate new network configuration
++ Which type of DGX servers are in the cluster? ++ H100 (Supports 1 - 6 SU, 32 nodes per SU)
++ Enter the number of SU in the cluster (maximum 6): 1
++ How many CPU nodes are in the cluster (this is in addition to the 2 required head nodes)? 3 ++ How
++ Enter the assigned network segment for the cluster, including the CIDR prefix: 10.180.112.0/21
++ Select the storage type for the cluster ++ DDN (DataDirectNetwork), Ethernet MGMT only
++ Enter the number of storage nodes (minimum 2, maximum 30): 2
++ Enter the number of NSDS devices (if any): 0
++ Enter the number of DDN Insight devices (if any): 0
++ Enter the number of managed IB switches: 0
++ Will the customer provide NFS shared storage for config backup? No
++ Enter the number of managed smart PDUs: 0
++ What is the Point-to-Point Connection type for the cluster? ++ BGP
++ Allow sending aggregate route summary prefix to Customer Edge? No
++ What is the uplink type to the customer edge? ++ Numbered - requires IPv4 addresses
++ Will there be more than 2 uplink connections? No
++ Have the edge uplink connection subnets been provided? No
Calculating switch counts
++ Enter the number of additional IPMI switches in the cluster, if any: 0
++ Enter the number of additional TOR switches in the cluster, if any: 0
++ Keep the current internalnet subnet as configured on the head node? Yes
++ Which IPMI switch is the primary? IPMI-01
++ MAC address for IPMI-01 (OK to leave blank if unknown):
++ MAC address for IPMI-02 (OK to leave blank if unknown):
++ MAC address for TOR-01 (OK to leave blank if unknown):
++ MAC address for TOR-02 (OK to leave blank if unknown):

Site network configuration saved to /cm/local/apps/bcm-superpod-network/config/network-configuration.yml
Site network allocations saved to /cm/local/apps/bcm-superpod-network/config/network-allocations.yml
Switch connection mapping saved to /cm/local/apps/bcm-superpod-network/config/switch-connections.yml

Network config updated, switch configs have been cleaned up.

++ Would you like to save the switch configuration files? Yes

Switch configuration files saved in /cm/local/apps/bcm-superpod-network/config

```

32. Download and move cumulus-linux-5.6.0-mlx-amd64.bin to the following directory on the head node. Contact your TAM for access to the correct file and move the file to the following directory on the head node

```
mv cumulus-linux-5.6.0-mlx-amd64.bin /cm/local/apps/cmd/etc/htdocs/switch/image/
```

33. Load the bcm-post-install module.

```
module load bcm-post-install/
```

34. Run the bcm-pod-setup script.

These parameters were used:

- -c sets the base address of the computenet network.
- -s sets the base address of the storagenet network.
- -I sets the installation source.

```
bcm-pod-setup -C 100.126.0.0/16 -S 100.127.0.0/16 -I /dev/sdb
```

35. Check the nodes and their categories.

Extra options are used for device list to make the format more readable.

```
%cmlsh
%[bcm-head-01]%device list -f hostname:20,category:10
hostname(key)      category
bcm-cpu-01         default
bcm-dgx-a100-01    dgx-a100
bcm-dgx-h100-01    dgx-h100
```

36. Confirm the config is correct for bcm-dgx-h100-01 / bcm-dgx-a100-01.

```
[bcm-head-01->device[bcm-dgx-h100-01]]% interfaces
[bcm-head-01->device[bcm-dgx-h100-01]->interfaces]% list
Type      Network device name      IP      Network      Start if
-----
-----
```

bmc	ipmi0		10.0.92.50	ipminet	
always					
bond	bond0 [prov]		10.0.93.12	dgxnet	
always					
physical	enp170s0f1np1 (bond0)	0.0.0.0			always
physical	enp41s0f1np1 (bond0)	0.0.0.0			always
physical	ibp154s0		100.126.5.14	ibnetcompute	always
physical	ibp170s0f0		100.127.2.2	ibnetstorage	always
physical	ibp192s0		100.126.6.14	ibnetcompute	always
physical	ibp206s0		100.126.7.14	ibnetcompute	always
physical	ibp220s0		100.126.8.14	ibnetcompute	always
physical	ibp24s0		100.126.1.14	ibnetcompute	always
physical	ibp41s0f0		100.127.1.2	ibnetstorage	always
physical	ibp64s0		100.126.2.14	ibnetcompute	always
physical	ibp79s0		100.126.3.14	ibnetcompute	always
physical	ibp94s0		100.126.4.14	ibnetcompute	always

37. Check the Ethernet Switch are in the devices.

*cmsh >> device >> list*

```
[bcm-head-01->device]% list
```

Type	Hostname (key)	MAC	Category	Ip	Network
HeadNode	bcm-head-01	B8:CE:F6:FF:98:A0		10.0.105.189	internalnet
PhysicalNode	slogin-01	00:00:00:00:00:00	default	10.0.105.140	internalnet
PhysicalNode	slogin-02	00:00:00:00:00:00	default	10.0.105.141	internalnet
PhysicalNode	storage-cpu-01	00:00:00:00:00:00	default	10.0.105.139	internalnet
PhysicalNode	storage-dgx-01	00:00:00:00:00:00	dgx-a100	10.0.0.11	dgxnet1
PhysicalNode	storage-dgx-02	00:00:00:00:00:00	dgx-a100	10.0.0.12	dgxnet1
Switch	IPMI-01	00:00:00:00:00:00		10.0.255.194	loopback
Switch	SJC4-C25-TOR-04	B8:59:9F:49:CC:9C		10.0.105.3	ipminet2
Switch	TOR-01	00:00:00:00:00:00		10.0.128.61	ipminet
Switch	TOR-02	00:00:00:00:00:00		10.0.128.60	ipminet
Switch	ib01	E8:EB:D3:31:FE:D2		10.0.105.41	ipminet2

```
[bcm-head-01->device]%
```

Validate under the “**Type**” Switches are added after executing *bcm-pod-setup*

3. Add Switch *credential*, under each IPMI, TOR and SPINE switches,

**Example:**

```
[bcm-head-01->device]% use IPMI-01
[bcm-head-01->device[IPMI-01]]% accesssettings
[bcm-head-01->device*[IPMI-01*]->accesssettings*]% set username cumulus
[bcm-head-01->device*[IPMI-01*]->accesssettings*]% set password <pass>
```

*commit*

*quit*

4. Add Infiniband Switch and PDU manually.

**Example:**

```
add switch ib01
set mac <macaddress>
set ip <ip>
set network <ipminet2>
set disablesmp yes
commit
```

Run the *bcm-netautogen* script again

```
module load bcm-superpod-network  
bcm-netautogen
```

Select "Write the switch configs"

```
root@bcm-head-01:~# bcm-netautogen  
  
SuperPOD Network Configuration Generator  
  
== What would you like to do? ==  
Generate new network configuration  
Update network configuration  
Write the network configuration  
> Write the switch configs  
Quit
```

---

# Head Node Configuration

This section addresses configuration steps to be performed on BCM head nodes.

Use the root (not `cmsh`) shell.

38. In `/cm/local/apps/cmd/etc/cmd.conf`, uncomment the `AdvancedConfig` parameter.

```
AdvancedConfig = { "DeviceResolveAnyMAC=1" } # modified value
```

Restart the `CMDaemon` to enable dependable PXE booting from bonded interfaces.

```
# systemctl restart cmd
```

The `cmsh` session will be disconnected because of restarting the `CMDaemon`. Type `connect` to reconnect after the `CMDaemon` has restarted. Or enter `exit` and then restart `cmsh`.

The steps that follow are performed on the head node and should be run for all DGX systems.

On the head node, set the MAC addresses on the physical interfaces.

 **Note:** Double check the MAC address for each interface and the IP number for the `bond0` interface. Mistakes here will be difficult to diagnose.

For DGX A100 systems, the commands should be like this code block.

```
# cmsh
% device
% use bcm-dgx-a100-01
% interfaces
% use enp225s0f1np1
% set mac B8:CE:F6:2F:08:69
% use enp97s0f1np1
% set mac B8:CE:F6:2D:0E:A7
% ..
% commit
```

For DGX H100 systems, the commands should be like this code block.

```
# cmsh
% device
% use bcm-dgx-h100-01
% interfaces
% use enp170s0f1np1
% set mac B8:CE:F6:2F:08:69
% use enp41s0f1np1
% set mac B8:CE:F6:2D:0E:A7
% ..
% commit
```

Verify the configuration.

This example is for a DGX A100 system. The output for a DGX H100 system is similar.

```
[bcm10-headnode->device[dgx-01]]% get provisioninginterface
bond0
% interfaces
% list
Type          Network device name      IP          Network      Start if
-----
bmc           ipmi0                    10.130.111.68 ipminet
always
bond          bond0 [prov]             10.130.122.5  internalnet  always
physical     enp225s0f1np1 (bond0)   0.0.0.0
physical     enp97s0f1np1 (bond0)   0.0.0.0
always
```

Identify the nodes by setting the MAC address for the provisioning interface for each node to the MAC address listed in the site survey.

```
% device
% use bcm-dgx-h100-01
% set mac b8:ce:f6:2f:08:69
% use bcm-dgx-h100-02
% set mac 0c:42:a1:54:32:a7
% use bcm-dgx-h100-03
% set mac 0c:42:a1:0a:7a:51
% use bcm-dgx-h100-04
% set mac 1c:34:da:29:17:6e
% foreach -c dgx-h100 (get mac)
B8:CE:F6:2F:08:69
0C:42:A1:54:32:A7
0C:42:A1:0A:7A:51
1C:34:DA:29:17:6E
```

If all the MAC addresses are set properly, commit the changes.

```
% device commit
% quit
```

Set the MAC addresses for the Ethernet interfaces.

For control nodes connected to DGX A100 systems, use the following commands.

```
% device
% use bcm-cpu-01
% interfaces
% use ens2f0np0
% set mac 88:e9:a4:92:26:ba
% use ens2f1np1
% set mac 88:e9:a4:92:26:bb
% commit
```

For control nodes connected to DGX H100 systems, use the following commands.

```
% device
% use bcm-cpu-01
% interfaces
% use enp37s0np0
% set mac 88:e9:a4:92:26:ba
% use enp65s0np0
% set mac 88:e9:a4:92:26:bb
% commit
```

Set the IP address for the bond0 interface.

```
% device
% use bcm-cpu-01
% interfaces
% use bond0
% set ip 10.127.3.15
% commit
```

Power on and provision the cluster nodes.

For initial provisioning, the cluster nodes must be powered on either directly or by using a KVM. It will take several minutes for the nodes to go through their BIOS. After that, node status progress will be displayed as the nodes are being provisioned. Monitor the `/var/log/messages` and `/var/log/node-installer` log files to verify that everything is proceeding smoothly.

# High Availability

39. Verify that the head node has power control over the cluster nodes.

```
% device
% power -c dgx-h100 status
[-head1->device]% power -c dgx-h100 status
ipmi0 ..... [ ON ] bcm-dgx-h100-01
ipmi0 ..... [ ON ] bcm-dgx-h100-02
ipmi0 ..... [ ON ] bcm-dgx-h100-03
ipmi0 ..... [ ON ] bcm-dgx-h100-04
[bcm-head-01->device]%
```

Power off the cluster nodes.

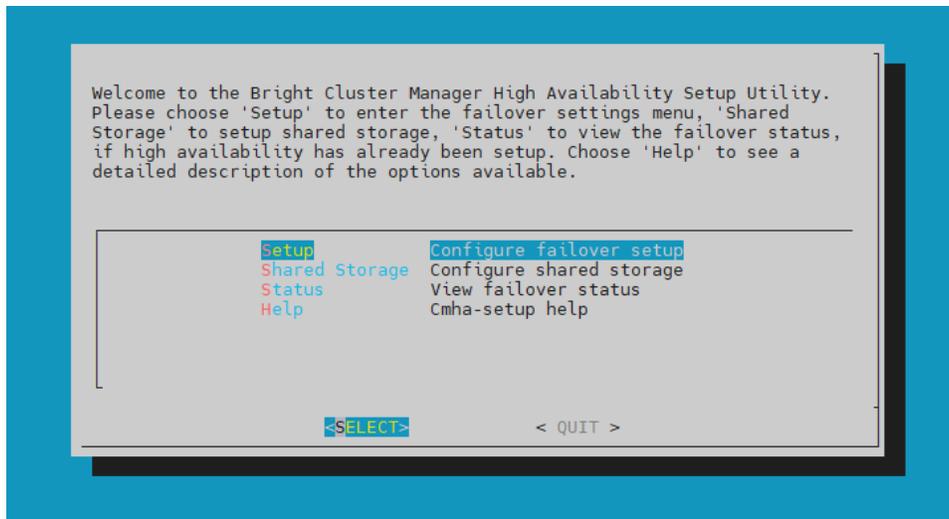
The cluster nodes must be powered off before configuring HA.

```
% power -c dgx-h100 off
ipmi0 ..... [ OFF ] bcm-dgx-h100-01
ipmi0 ..... [ OFF ] bcm-dgx-h100-02
ipmi0 ..... [ OFF ] bcm-dgx-h100-03
ipmi0 ..... [ OFF ] bcm-dgx-h100-04
```

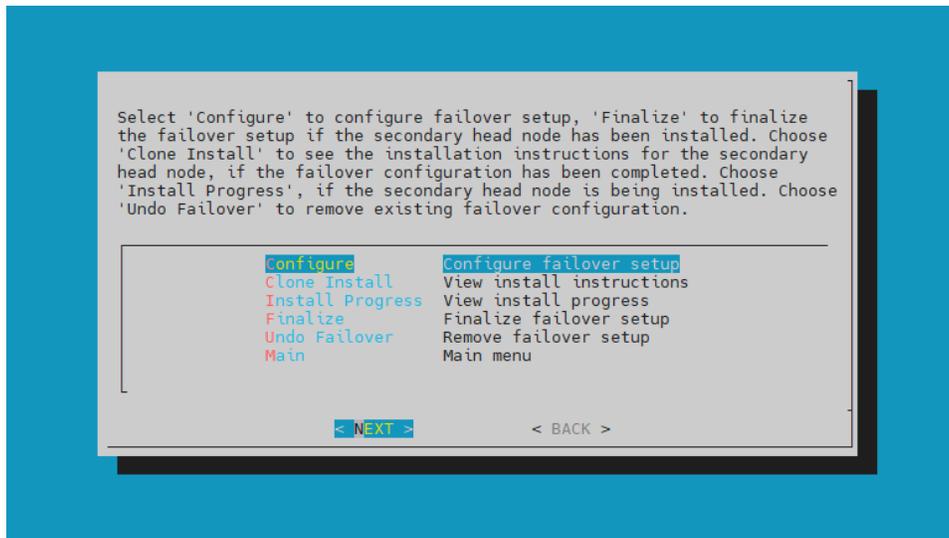
Start the cmha-setup CLI wizard as the root user on the primary head node.

```
# cmha-setup
```

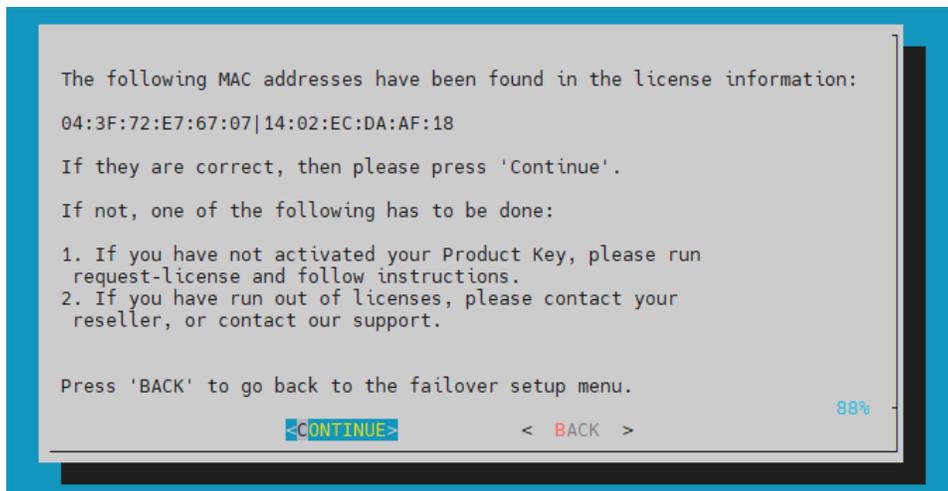
Choose Setup and then select SELECT.



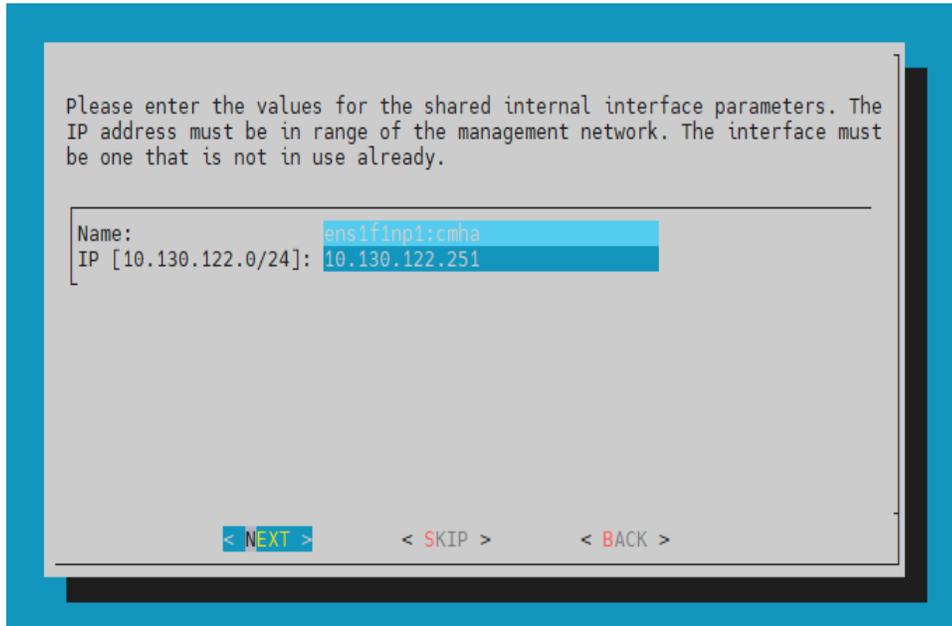
Choose configure and then select NEXT.



Verify that the cluster license information found cmha-setup is correct and then select CONTINUE.



Provide an internal Virtual IP address that is to be used by the active head node in the HA configuration.

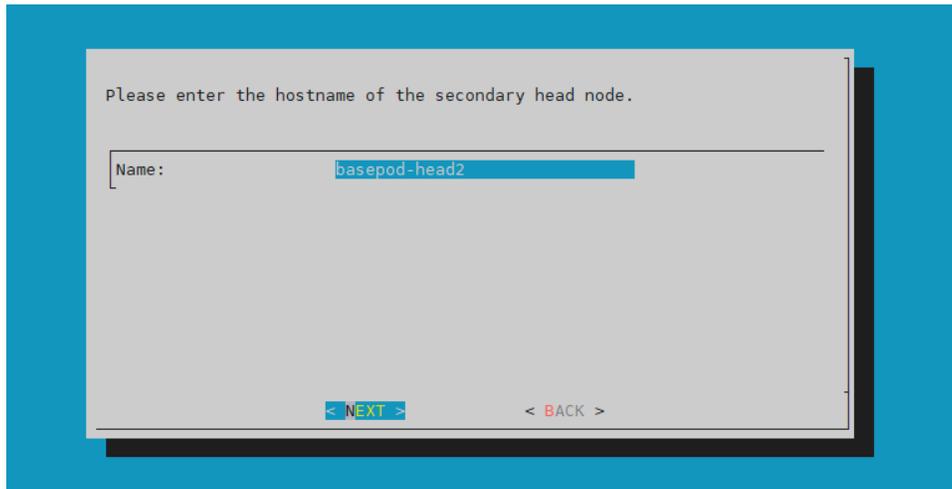


Please enter the values for the shared internal interface parameters. The IP address must be in range of the management network. The interface must be one that is not in use already.

Name:

IP [10.130.122.0/24]:

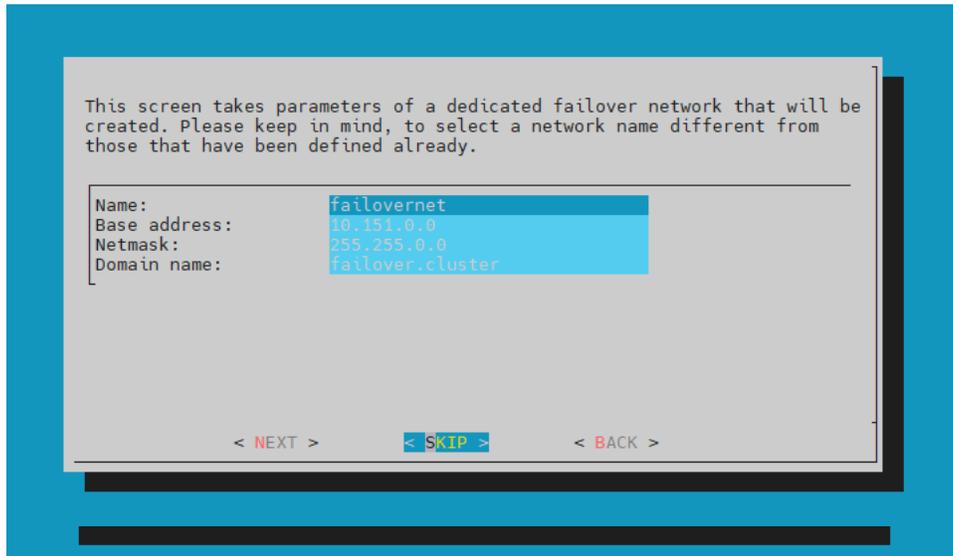
Provide the name of the secondary head node and then select NEXT.



Please enter the hostname of the secondary head node.

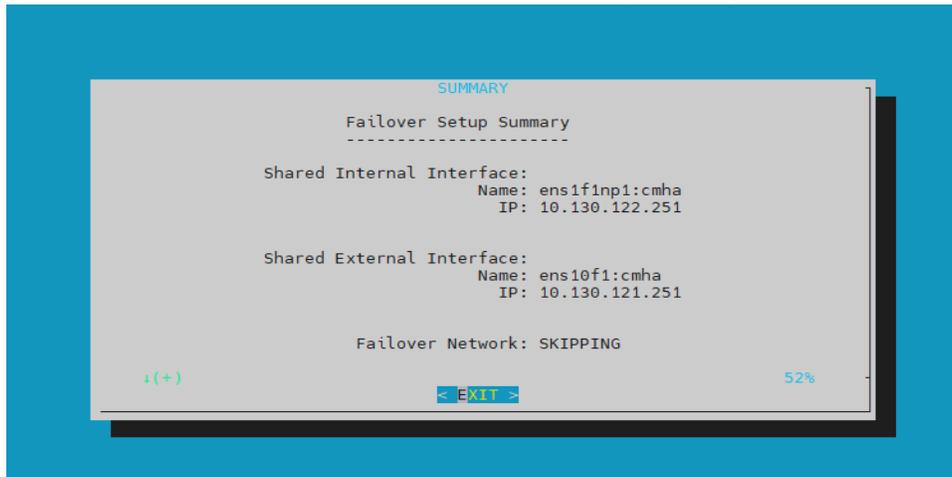
Name:

Because DGX SuperPOD uses the internal network as the failover network, select SKIP.



Configure the IP addresses for the secondary head node that the wizard is about to create and then select NEXT.

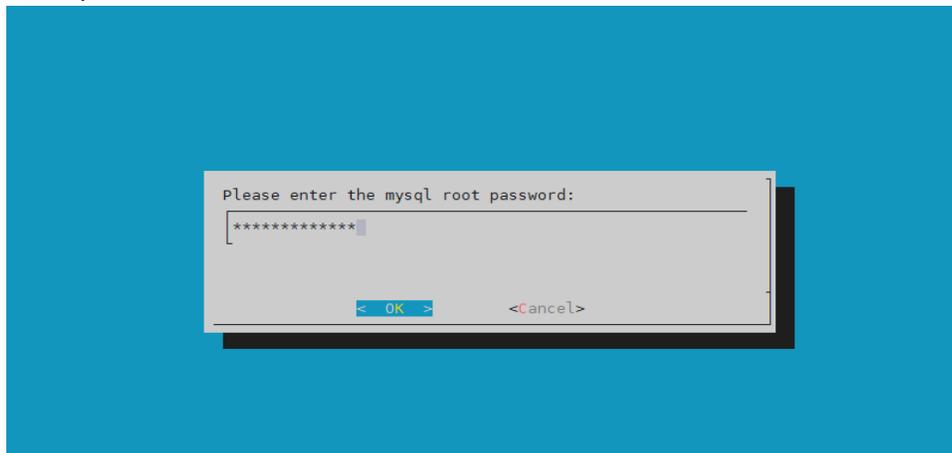
The wizard shows a summary of the information that it has collected. The VIP that will be assigned to the internal and external interfaces, respectively.



Select **Yes** to proceed with the failover configuration.



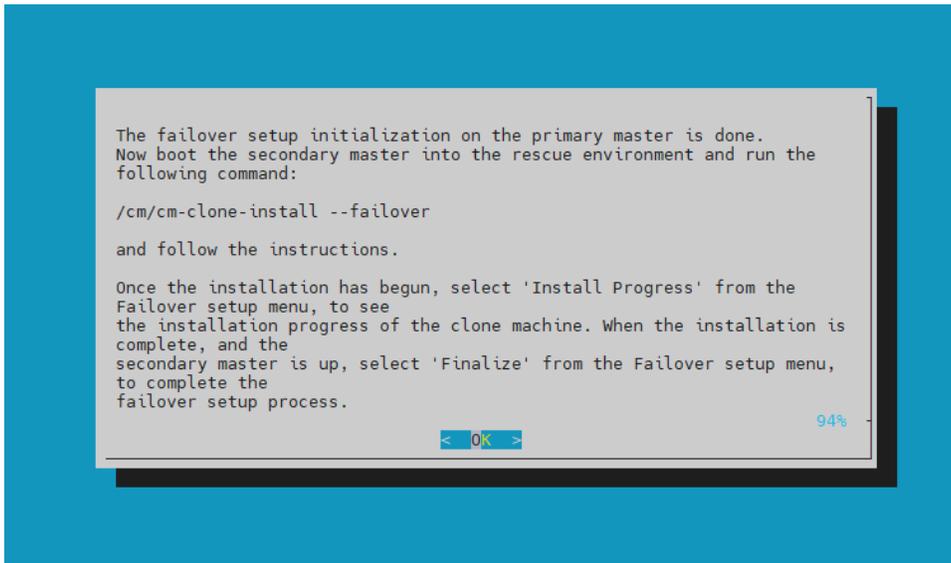
Enter the root password and then select **OK**.



The wizard implements the first steps in the HA configuration. If all the steps show OK, press ENTER to continue. The progress is shown here.

```
Initializing failover setup on master..... [ OK ]
Updating shared internal interface..... [ OK ]
Updating shared external interface..... [ OK ]
Updating extra shared internal interfaces..... [ OK ]
Cloning head node..... [ OK ]
Updating secondary master interfaces..... [ OK ]
Updating Failover Object..... [ OK ]
Restarting cmdaemon..... [ OK ]
Press any key to continue
```

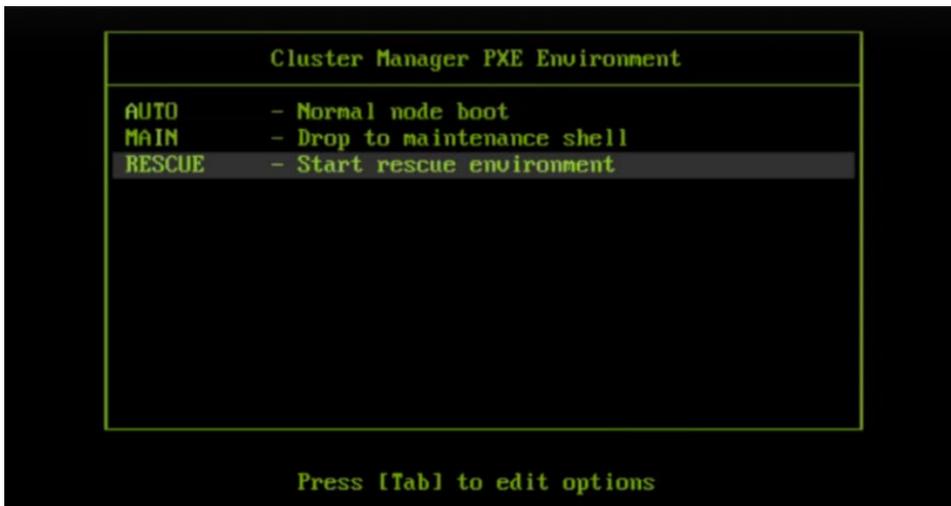
When the failover setup installation on the primary master is complete, select OK to exit the wizard.



PXE boot the secondary head node and then select RESCUE from the grub menu.

Because this is the initial boot of this node, it must be done outside of BCM (BMC or physical power button).

Select RESCUE from the grub menu.



After the secondary head node has booted into the rescue environment, run the `/cm/cm--clone--install --failover` command, then enter yes when prompted.

The secondary head node will be cloned from the primary.

```
|
| *Welcome to the Cluster Manager rescue environment*
|-----|
| Creating failover/clone nodes:
| * Install the secondary head node
|
|   $ /cm/cm-clone-install --failover
|
| * Create a clone of the primary head node
|
|   $ /cm/cm-clone-install --clone --hostname=new-hostname
|
| * Install the secondary (failover) head node and reboot automatically
|
|   $ /cm/cm-clone-install --failover --reboot
|
| * Help
|
|   $ /cm/cm-clone-install --help
|-----|
ClusterManager login: root (Automatic login)

Linux ClusterManager 5.13.0-39-generic #44720.04.1-Ubuntu SMP Thu Mar 24 16:43:35 UTC 2022 x86_64
root@ClusterManager:~# /cm/cm-clone-install --failover
Network interface to use [default: enp0]: ens1f1np1
Please wait while bringing up network...
Please wait while authentication is being set up...
Enter the password of the headnode node to continue.
root@master's password:
Please wait while installation begins...
Verifying license ..... [ OK ]
Getting disk layout ..... [ OK ]
The head node disk layout is saved in /cm/_headnodedisksetup.xml
(u - view, e - edit, c - continue ): c
info: Detecting device '/dev/nvme0n1': found
info: Valid device nvme0n1. All checks have succeeded.
The contents of the following disks will be erased.
/dev/nvme0n1
Do you want to continue [yes/no]? yes_
```

When cloning is completed, enter `y` to reboot the secondary head node.

The secondary must be set to boot from its hard drive. PXE boot should not be enabled.

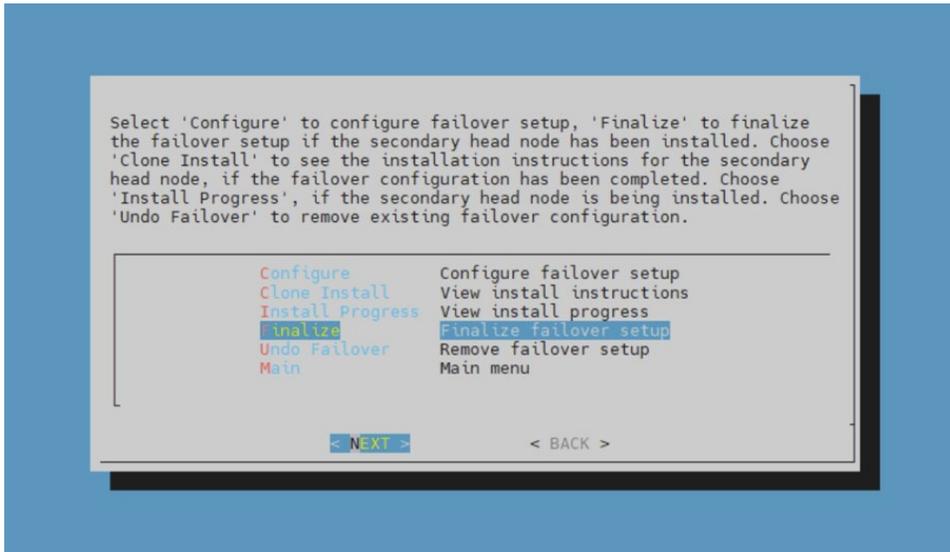
Wait for the secondary head node to reboot and then continue the HA setup procedure on the primary head node.

```
|
| *Welcome to the Cluster Manager rescue environment*
|-----|
| Creating failover/clone nodes:
| * Install the secondary head node
|
|   $ /cm/cm-clone-install --failover
|
| * Create a clone of the primary head node
|
|   $ /cm/cm-clone-install --clone --hostname=new-hostname
|
| * Install the secondary (failover) head node and reboot automatically
|
|   $ /cm/cm-clone-install --failover --reboot
|
| * Help
|
|   $ /cm/cm-clone-install --help
|-----|
ClusterManager login: root (Automatic login)

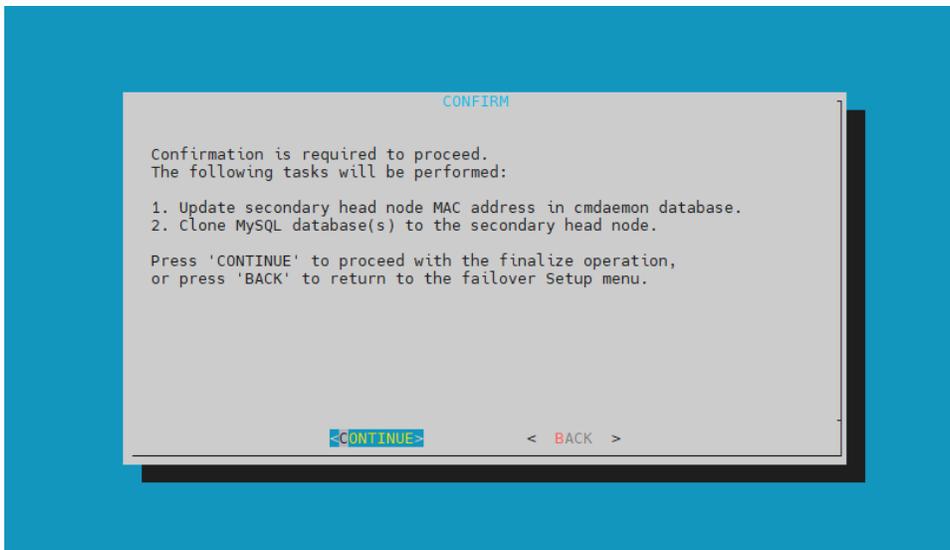
Linux ClusterManager 5.13.0-39-generic #44720.04.1-Ubuntu SMP Thu Mar 24 16:43:35 UTC 2022 x86_64
root@ClusterManager:~# /cm/cm-clone-install --failover
Network interface to use [default: enp0]: ens1f1np1
Please wait while bringing up network...
Please wait while authentication is being set up...
Enter the password of the headnode node to continue.
root@master's password:
Please wait while installation begins...
Verifying license ..... [ OK ]
Getting disk layout ..... [ OK ]
The head node disk layout is saved in /cm/_headnodedisksetup.xml
(u - view, e - edit, c - continue ): c
info: Detecting device '/dev/nvme0n1': found
info: Valid device nvme0n1. All checks have succeeded.
The contents of the following disks will be erased.
/dev/nvme0n1
Do you want to continue [yes/no]? yes_
Getting mount points ..... [ OK ]
Partitioning hard drive ..... [ OK ]
Mounting partitions ..... [ OK ]
Syncing hard drive ..... [ OK ]
Finalizing installation ..... [ OK ]
Do you want to reboot[y/n]:y_
```

Select finalize from the cmha-setup menu and then select NEXT.

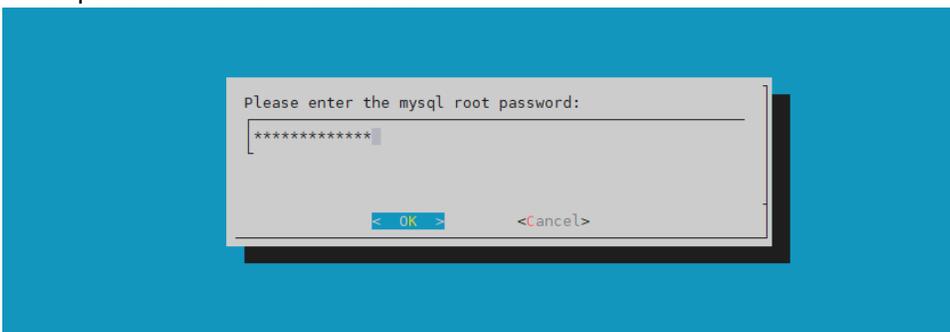
This will clone the MySQL database from the primary to the secondary head node.



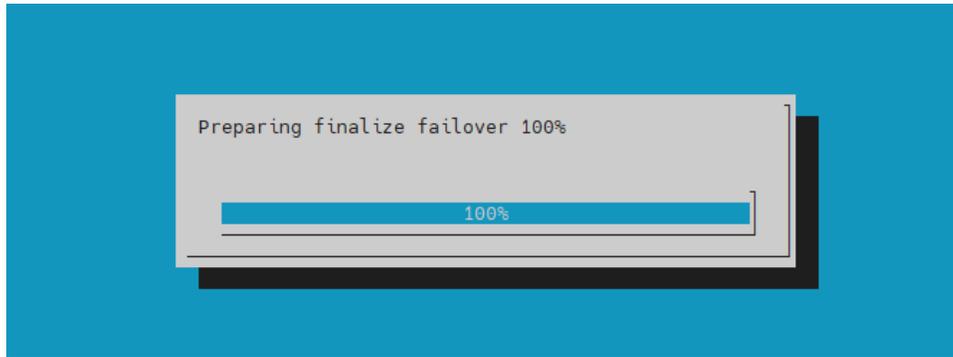
Select CONTINUE on the confirmation screen.



Enter the root password and then select OK.



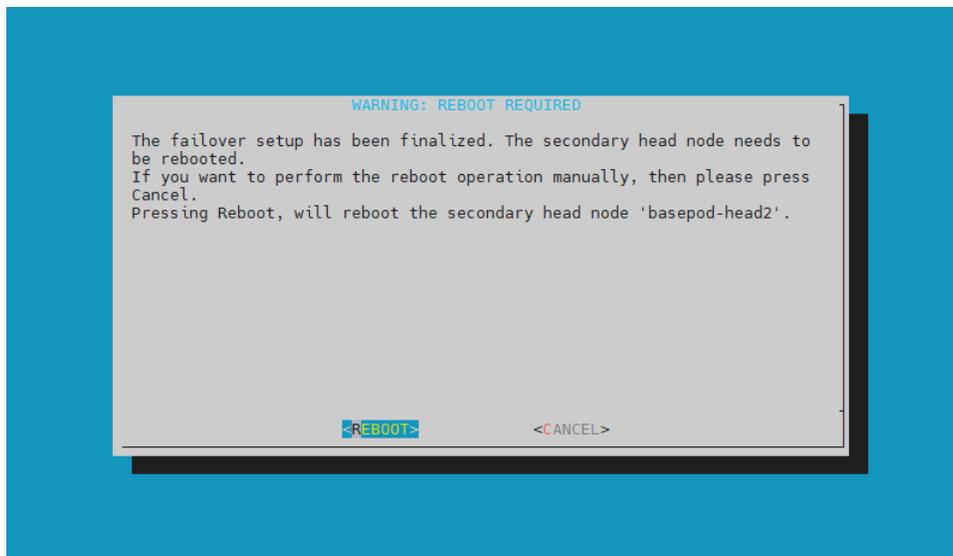
The cmha-setup wizard continues. Press ENTER to continue when prompted.



The progress is shown here:

```
Updating secondary master mac address..... [ OK ]
Initializing failover setup on bcm-head-02..... [ OK ]
Stopping cmdaemon..... [ OK ]
Cloning cmdaemon database..... [ OK ]
Checking database consistency..... [ OK ]
Starting cmdaemon, chkconfig services..... [ OK ]
Cloning workload manager databases..... [ OK ]
Cloning additional databases..... [ OK ]
Update DB permissions..... [ OK ]
Checking for dedicated failover network..... [ OK ]
Press any key to continue
```

The Finalize step is now completed. Select REBOOT and wait for the secondary head node to reboot.

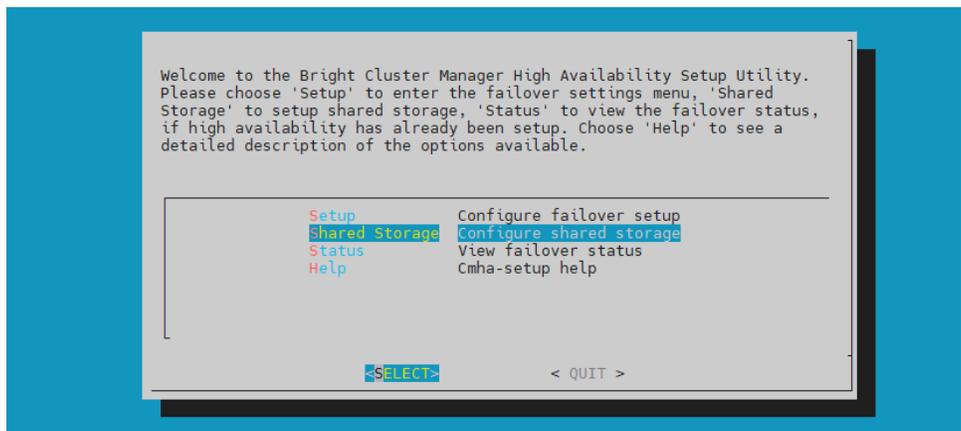


The secondary head node is now UP.

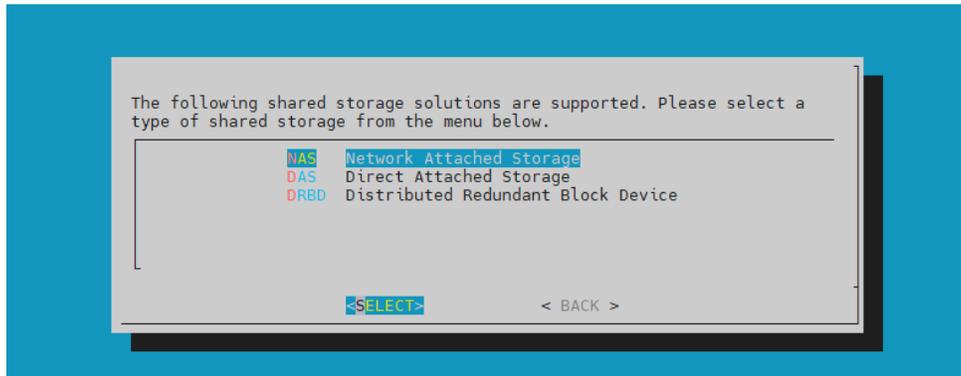
```
% device list -f hostname:20,category:12,ip:20,status:15
hostname (key)      category      ip            status
-----
bcm-head-01        10.130.122.254 [  UP  ]
bcm-head-02        10.130.122.253 [  UP  ]
bcm-dgx-h100-01    dgx-h100     10.130.122.5  [ DOWN ]
bcm-dgx-h100-02    dgx-h100     10.130.122.6  [ DOWN ]
bcm-dgx-h100-03    dgx-h100     10.130.122.7  [ DOWN ]
bcm-dgx-h100-04    dgx-h100     10.130.122.8  [ DOWN ]
```

Select Shared Storage from the cmha-setup menu and then select SELECT.

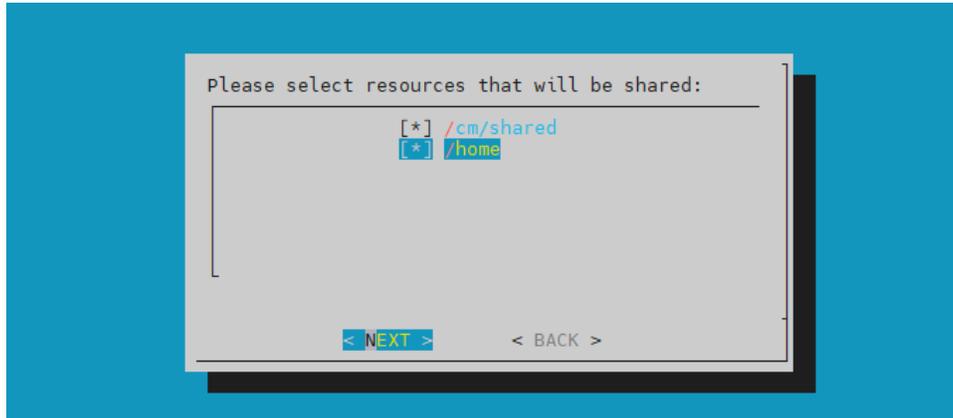
In this final HA configuration step, cmha-setup will copy the /cm/shared and /home directories to the shared storage and configure both head nodes and all cluster nodes to mount it.



Choose NAS and then select SELECT.

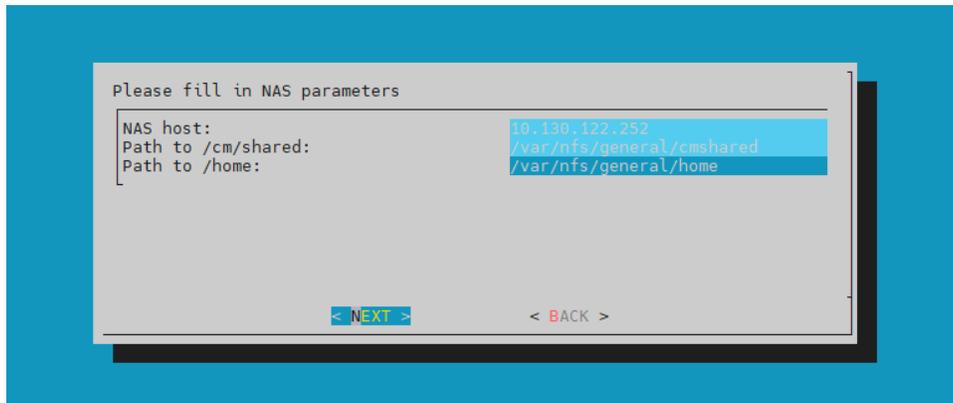


Choose both `/cm/shared` and `/home` and then select NEXT.

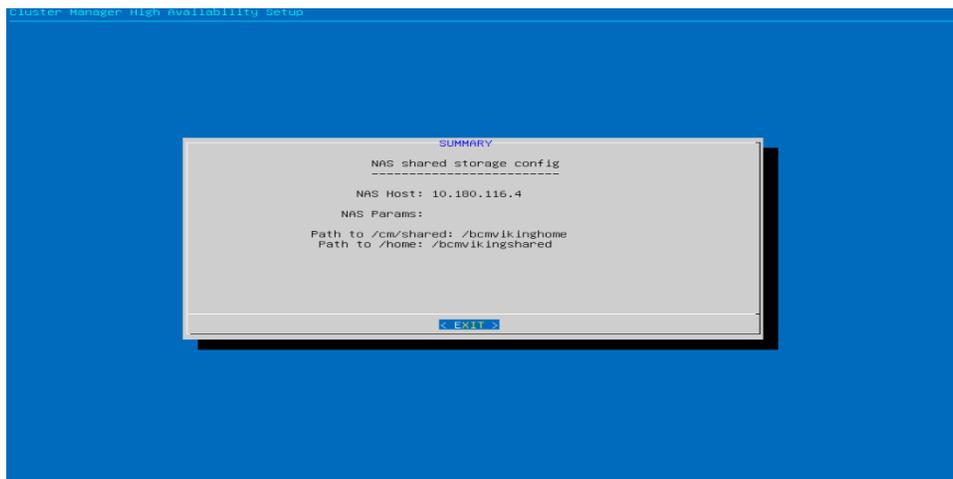


Provide the IP address of the NAS host, the paths that the `/cm/shared` and `/home` directories should be copied to on the shared storage, and then select NEXT.

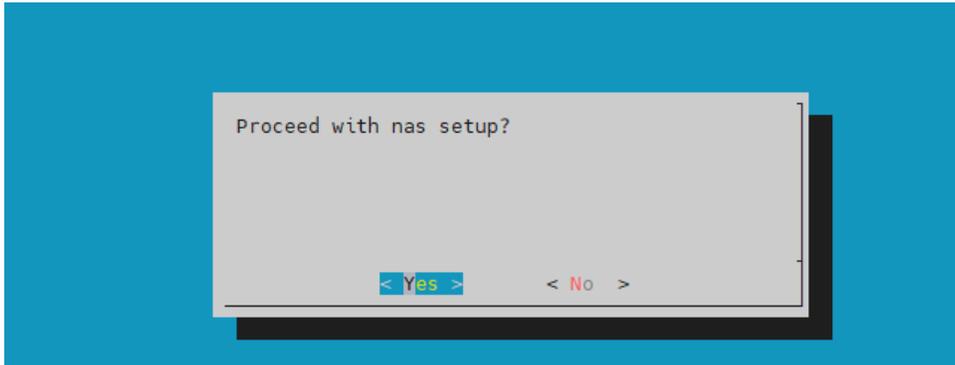
In this case, `/var/nfs/general` is exported, so the `/cm/shared` directory will be copied to `10.130.122.252:/var/nfs/general/cmshared`, and it will be mounted over `/cm/shared` on the cluster nodes.



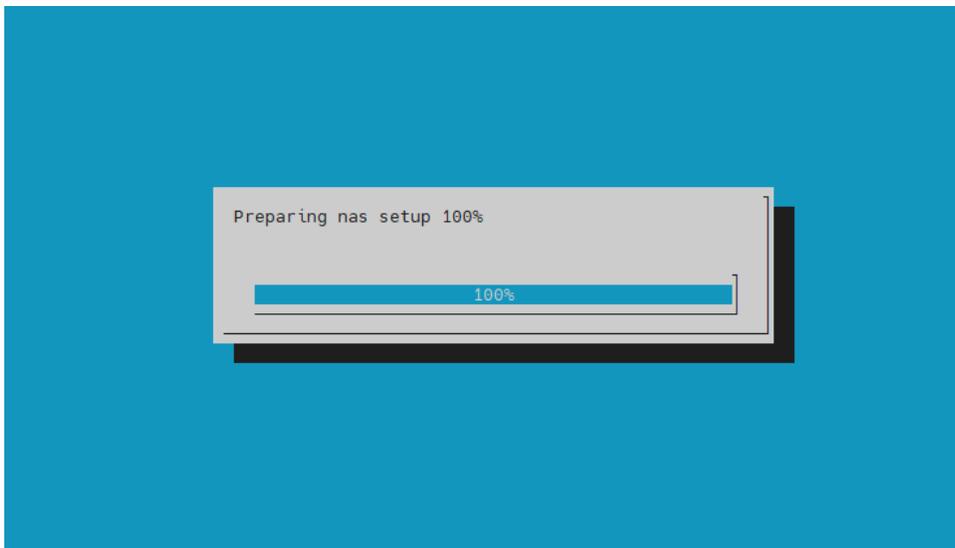
The wizard shows a summary of the information that it has collected. Select EXIT to continue.



When asked to proceed with the NAS setup, select `Yes` to continue.  
This will initiate a copy and update to `fsexports`.



The `cmha-setup` wizard proceeds with its work.

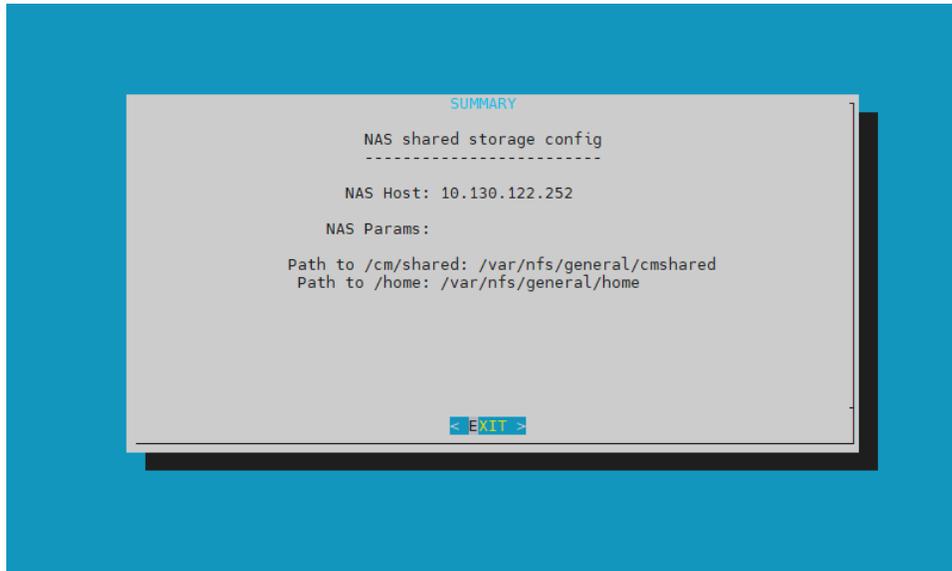


When setup completes, press any key to finish HA setup.

The progress is shown here:

```
Copying NAS data..... [ OK ]
Mount NAS storage..... [ OK ]
Remove old fsmounts..... [ OK ]
Add new fsmounts..... [ OK ]
Remove old fsexports..... [ OK ]
Write NAS mount/unmount scripts..... [ OK ]
Copy mount/unmount scripts..... [ OK ]
Press any key to continue
```

cmha-setup is now complete. EXIT the wizard to return to the shell prompt.



Run the `cmha status` command to verify that the failover configuration is correct and working as expected.

The command tests the configuration from both directions: from the primary head node to the secondary, and from the secondary to the primary. The active head node is indicated by an asterisk.

```
# cmha status
Node Status: running in active mode

bcm-head-01* -> bcm-head-02
failoverping [ OK ]
mysql        [ OK ]
ping         [ OK ]
status       [ OK ]

bcm-head-02 -> bcm-head-01*
failoverping [ OK ]
mysql        [ OK ]
ping         [ OK ]
status       [ OK ]
```

Verify that the /cm/shared and /home directories are mounted from the NAS server.

```
# mount
. . . some output omitted . . .
10.130.122.252:/var/nfs/general/cmshared on /cm/shared type nfs4
(rw,relatime,vers=4.2,rsiz=32768,wsiz=32768,namlen=255,hard,proto=tcp,timeo=600,retrans=2
,sec=sys,clientaddr=10.130.122.253,local_lock=none,addr=10.130.122.252)
10.130.122.252:/var/nfs/general/home on /home type nfs4
(rw,relatime,vers=4.2,rsiz=32768,wsiz=32768,namlen=255,hard,proto=tcp,timeo=600,retrans=2
,sec=sys,clientaddr=10.130.122.253,local_lock=none,addr=10.130.122.252)
```

Login to the head node to be made active and run cmha makeactive.

```
# ssh bcm-head-02
# cmha makeactive
=====
This is the passive head node. Please confirm that this node should become
the active head node. After this operation is complete, the HA status of
the head nodes will be as follows:

bcm-head-02 will become active head node (current state: passive)
bcm-head-01 will become passive head node (current state: active)
=====

Continue(c)/Exit(e)? c

Initiating failover..... [ OK ]

bcm-head-02 is now active head node, makeactive successful
```

Run the cmsh status command again to verify that the secondary head node has become the active head node.

```
# cmsh status
Node Status: running in active mode

bcm-head-02* -> bcm-head-01
failoverping [ OK ]
mysql [ OK ]
ping [ OK ]
status [ OK ]

bcm-head-01 -> bcm-head-02*
failoverping [ OK ]
mysql [ OK ]
ping [ OK ]
status [ OK ]
```

Manually failover back to the primary head node by running `cmha makeactive`.

```
# ssh bcm-head-01
# cmha makeactive

=====
This is the passive head node. Please confirm that this node should become
the active head node. After this operation is complete, the HA status of
the head nodes will be as follows:

bcm-head-01 will become active head node (current state: passive)
bcm-head-02 will become passive head node (current state: active)
=====

Continue(c)/Exit(e)? c

Initiating failover..... [ OK ]

bcm-head-01 is now active head node, makeactive successful
```

Run the `cmsh status` command again to verify that the primary head node has become the active head node.

```
# cmsh status
Node Status: running in active mode

bcm-head-01* -> bcm-head-02
failoverping [ OK ]
mysql       [ OK ]
ping        [ OK ]
status      [ OK ]

bcm-head-02 -> bcm-head-01*
failoverping [ OK ]
mysql       [ OK ]
ping        [ OK ]
status      [ OK ]
```

Power on the cluster nodes.

```
# cmsh -c "device ; power -c dgx-h100 on"
ipmi0 ..... [ ON ] bcm-dgx-h100-01
ipmi0 ..... [ ON ] bcm-dgx-h100-02
ipmi0 ..... [ ON ] bcm-dgx-h100-03
ipmi0 ..... [ ON ] bcm-dgx-h100-04
```

This concludes the setup and verification of HA.

---

# Slurm Setup

40. As a root user, load the `bcm-post-install` module and run `bcm-post-install`.

```
module load bcm-post-install
bcm-install-slurm
```

This script will create the `slogin` nodes.

Update the interface names on the `slogin` nodes.

```
% device use slogin-01
```

If the `slogin-01` does not have the expected interface names, update the interface names.

```
% use networkdevicename
% set networkdevicename new-name
```

Assign the MAC addresses to the `slogin` nodes.

```
device use slogin-01
set mac <MAC address>
```

Power on and install the `slogin` nodes.

Rerun the `bcm-install-slurm` script.

Use the following parameters:

- installation source for the `--bcm-media` parameter. It can be either a USB or a path to a `*.iso` file.
- Use the `-A` parameter to run the script in air-gapped mode.
- If CMHA is set up but has failover ping errors, append `--ignore-ha-errors`.
- If there is only one `slogin` node, append `--ignore-missing-login-node`.

```
bcm-install-slurm -A --bcm-media <path to installer image or usb device to mount>
```

Confirm that the `slurmd` file is present in DGX image before provisioning DGX nodes, if not create it.

The same file is needed for both DGX A100 and DGX H100 systems. This example is for DGX H100 systems. It is observed that NCCL tests with PMIX need this file.

```
vi /cm/images/dgx-os-6.1-h100-image/etc/sysconfig/slurmd
PMIX_MCA_ptl=^usock
PMIX_MCA_psec=none
PMIX_SYSTEM_TMPDIR=/var/empty
PMIX_MCA_gds=hash
```

Reboot the `slogin` and compute nodes.

```
cmsh
device
reboot -c slogin
reboot -c dgx-h100
```

To simplify the configuration, modify the `slurmclient-gpu` role to remove the `slurm-client` role and convert `slurm-client-gpu` to instead use that name.

```
cmsh
configurationoverlay
```

```
remove slurm-client
commit
use slurm-client-gpu
set name slurm-client
commit
```

For DGX A100 systems, clear the type value and set the correct core association with each GPU entry for maximum performance.

```
cmsh
configurationoverlay
use slurm-client
roles
use slurmclient
genericresources
use gpu0
clear type
set cores 48-63,176-191
use gpu1
clear type
set cores 48-63,176-191
use gpu2
clear type
set cores 16-31,144-159
use gpu3
clear type
set cores 16-31,144-159
use gpu4
clear type
set cores 112-127,240-255
use gpu5
clear type
set cores 112-127,240-255
use gpu6
clear type
set cores 80-95,210-223
use gpu7
clear type
set cores 80-95,210-223
commit
```

For DGX H100 systems, generic resources are set to autodetect.

Use this script.

```
cmsh
wlm
set gpuautodetect nvm1
commit
configurationoverlay
use slurm-client
roles
use slurmclient
set gpuautodetect nvm1
commit
genericresources
foreach * (remove)
commit
add autodetected-gpus
set name gpu
set count 8
set addtogresconfig yes
commit
```



**Note:** addtogresconfig is set by default to “yes” and does not need to be set explicitly.

Which should yield output like this.

```
[vikingbcmhead-01->configurationoverlay*[slurm-client*]->roles*[slurmclient*]-
>genericresources*[autodetected-gpus]]% ls
Alias (key)          Name      Type      Count   File
-----
autodetected-gpus  gpu       H100      8
```

The `gres.conf` file will be updated automatically by BCM—these settings align with the expectations of various scripts and tools in the NVIDIA ecosystem and will then maximize the compatibility of this environment with those scripts and tools.

If the `/home` directory is not mounted on the nodes, increase the number of retries.

Due to a race condition between the `bond0` interface being up and `/home` being mounted, sometimes `/home` will not be mounted. Increasing the number of retries should fix the issue.

```
cmsh
category
use dgx-h100
fsmounts
use /home
set mountoptions "x-systemd.mount-timeout=150,defaults,_netdev,retry=5,vers=3"
```

## Notice

This document is provided for information purposes only and shall not be regarded as a warranty of a certain functionality, condition, or quality of a product. NVIDIA Corporation ("NVIDIA") makes no representations or warranties, expressed or implied, as to the accuracy or completeness of the information contained in this document and assumes no responsibility for any errors contained herein. NVIDIA shall have no liability for the consequences or use of such information or for any infringement of patents or other rights of third parties that may result from its use. This document is not a commitment to develop, release, or deliver any Materials (defined below), code, or functionality.

NVIDIA reserves the right to make corrections, modifications, enhancements, improvements, and any other changes to this document, at any time without notice.

Customer should obtain the latest relevant information before placing orders and should verify that such information is current and complete.

NVIDIA products are sold subject to the NVIDIA standard terms and conditions of sale supplied at the time of order acknowledgement, unless otherwise agreed in an individual sales agreement signed by authorized representatives of NVIDIA and customer ("Terms of Sale"). NVIDIA hereby expressly objects to applying any customer general terms and conditions with regards to the purchase of the NVIDIA product referenced in this document. No contractual obligations are formed either directly or indirectly by this document.

No license, either expressed or implied, is granted under any NVIDIA patent right, copyright, or other NVIDIA intellectual property right under this document. Information published by NVIDIA regarding third-party products or services does not constitute a license from NVIDIA to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property rights of the third party, or a license from NVIDIA under the patents or other intellectual property rights of NVIDIA.

Reproduction of information in this document is permissible only if approved in advance by NVIDIA in writing, reproduced without alteration and in full compliance with all applicable export laws and regulations, and accompanied by all associated conditions, limitations, and notices.

THIS DOCUMENT AND ALL NVIDIA DESIGN SPECIFICATIONS, REFERENCE BOARDS, FILES, DRAWINGS, DIAGNOSTICS, LISTS, AND OTHER DOCUMENTS (TOGETHER AND SEPARATELY, "MATERIALS") ARE BEING PROVIDED "AS IS." NVIDIA MAKES NO WARRANTIES, EXPRESSED, IMPLIED, STATUTORY, OR OTHERWISE WITH RESPECT TO THE MATERIALS, AND EXPRESSLY DISCLAIMS ALL IMPLIED WARRANTIES OF NONINFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE. TO THE EXTENT NOT PROHIBITED BY LAW, IN NO EVENT WILL NVIDIA BE LIABLE FOR ANY DAMAGES, INCLUDING WITHOUT LIMITATION ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL, PUNITIVE, OR CONSEQUENTIAL DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF ANY USE OF THIS DOCUMENT, EVEN IF NVIDIA HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Notwithstanding any damages that customer might incur for any reason whatsoever, NVIDIA's aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms of Sale for the product.

## Trademarks

NVIDIA, the NVIDIA logo, DGX, DGX NVIDIA SuperPOD, and NVIDIA Base Command Manager are trademarks and/or registered trademarks of NVIDIA Corporation in the U.S. and other countries. Other company and product names may be trademarks of the respective companies with which they are associated.

## Authors

AY, DS, JB, HS, LH, MK, ML, RA, RS, and VS.

## Copyright

© 2023 NVIDIA Corporation. All rights reserved.