NVIDIA Grace Software with Red Hat Enterprise Linux 9

Installation Guide

DI-11454-001_04 | March 2024

Document History

DI-11454-001_03

Version	Date	Description of Change
01	July 21, 2023	Initial Release
02	September 18, 2023	Updated Appendix D with the minimum kernel versions that contain the fix for those issues.
03	December 18, 2023	Added Appendix A.1 and RHEL 9.3.
04	March 8, 2024	Added the following content: ● Clarifications to the install steps. ● A boot workaround to Appendix D.1. ● A note to Appendix A.1 about explicitly requiring the open-source GPU driver.

Table of Contents

Introduction

NVIDIA® Grace systems can run Red Hat® Enterprise Linux® (RHEL) and take advantage of the advanced Grace features. This document explains how to install and configure Grace systems with Red Hat Enterprise Linux 9.2 and later.

Attention: Although it might be possible to use other Linux distributions that are related to (or inspired by) RHEL, for example CentOS Stream, only RHEL has been pre-validated with the steps described by this guide for installation on Grace platforms.

Related Documentation

Refer to the Red Hat Enterprise Linux 9 Documentation Portal for more information about RHEL.

Prerequisites

This section lists the required (or recommended) prerequisites:

Red Hat Subscription

A Red Hat subscription is required to install and use RHEL 9 on the Grace platform. A subscription allows you to obtain update packages and additional packages for RHEL. To purchase a subscription or obtain a free evaluation subscription, go to the Red Hat Software & Download Center.

Access to Repositories

The repositories can be accessed from the internet. If you are using a proxy server, follow the instructions in the Red Hat Advanced Installation Guide to ensure that the system can access the necessary URIs.

Red Hat Repositories

To install software for the Grace platform over RHEL 9, you need access to the following repositories:

- Red Hat Enterprise BaseOS Repository: `rhel-9-for-aarch64-baseos-rpms`
- Red Hat Enterprise AppStream Repository: `rhel-9-for-aarch64-appstream-rpms`

Installing Red Hat Enterprise Linux

Red Hat provides several methods to install RHEL (refer to the RHEL Installation Guide for more information). **Before you install**, review the platform-dependent workarounds section in this document to determine whether any modifications are required for your environment.

This section describes how to install RHEL using the Quick Install method and reclaim the disk space used by an existing installation in the process. It describes a minimal installation. If you have a preferred method to install RHEL, you can skip this section but ensure that you reclaim the disk space that is used by an existing OS installation.

This method installs Red Hat Enterprise Linux on the Grace system remotely through facilities hosted on the BMC.

Obtaining Red Hat Enterprise Linux 9

Obtain the RHEL ISO for Arm® image (aarch64) and store it on your local disk. Refer to Downloading Red Hat Enterprise Linux for the instructions.

Remotely Booting the Red Hat Enterprise Linux 9 ISO

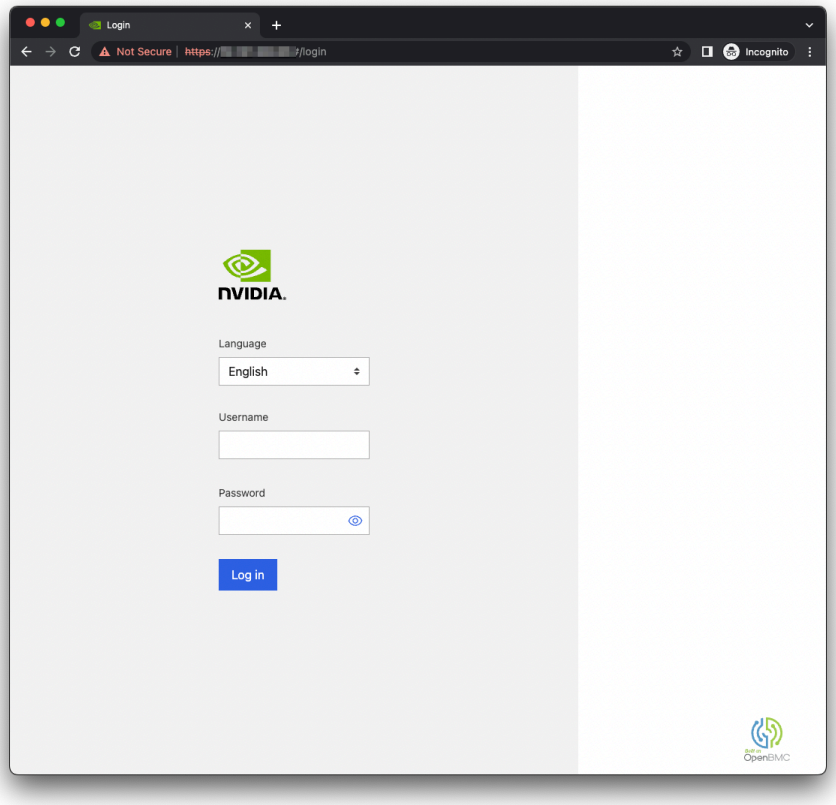
Attention: Here is some important information to know **before** you begin:

- The Grace BMC comes with default login credentials, and NVIDIA recommends that you create a unique user ID and password.
  Contact the system vendor if you have issues logging into the BMC.
- As a performance consideration, for networks that are distributed across a wide geographical area, such as a corporate VPN, we recommend that you use a browser from a host near the target server.

	This can be accomplished by using a remote application software solution, for example VNC, or by exporting the application X session and tunneling over SSH. • This sequence is intended for the NVIDIA reference BMC, and your results might vary when using an IBV BMC.

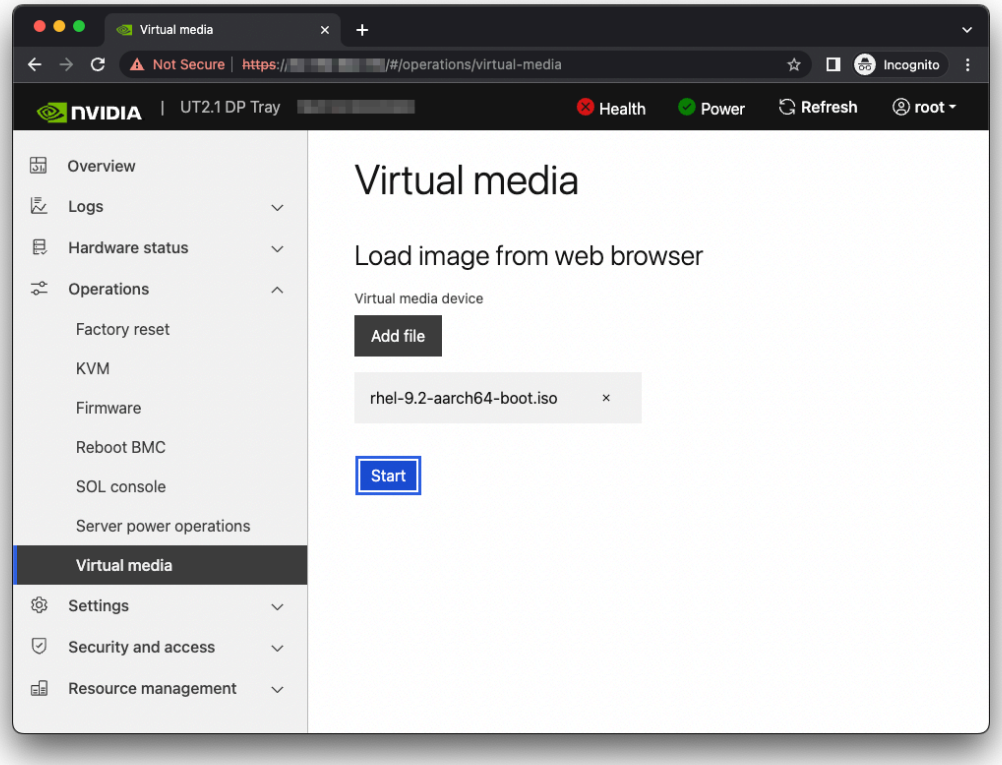
1. Connect to the BMC.
   a. Open a browser in your LAN, navigate to `https://<BMC-IP-address>/`, and log in.

Figure 1. BMC Login



2. Download the ISO image to a location that can be accessed by the browser.
3. Set up the ISO image as virtual media.
   a. From the left hand menu, expand **Operations**.
   b. Select **Virtual media**.
   c. Click **Add file**, navigate to the downloaded ISO image, and select it.
   d. To begin serving the ISO image to the target server, click **Start**.

Figure 2. BMC Virtual Media



4. Boot from the virtual media.
   Typically, the default boot order does not boot the CD-ROM image. You can change this in the BIOS or as a one-time option in the boot menu.
   a. Connect to the console.
      i. From the left hand menu, expand **Operations,** and select the SOL **console**.
      ii. Alternatively, SSH to the BMC from a terminal emulator, log in, and at the prompt, run the `obmc-console-client` command.
   b. To bring up the boot menu, press **Escape** or **F11** at the beginning of the boot process.

Figure 3. Console Splash Screen

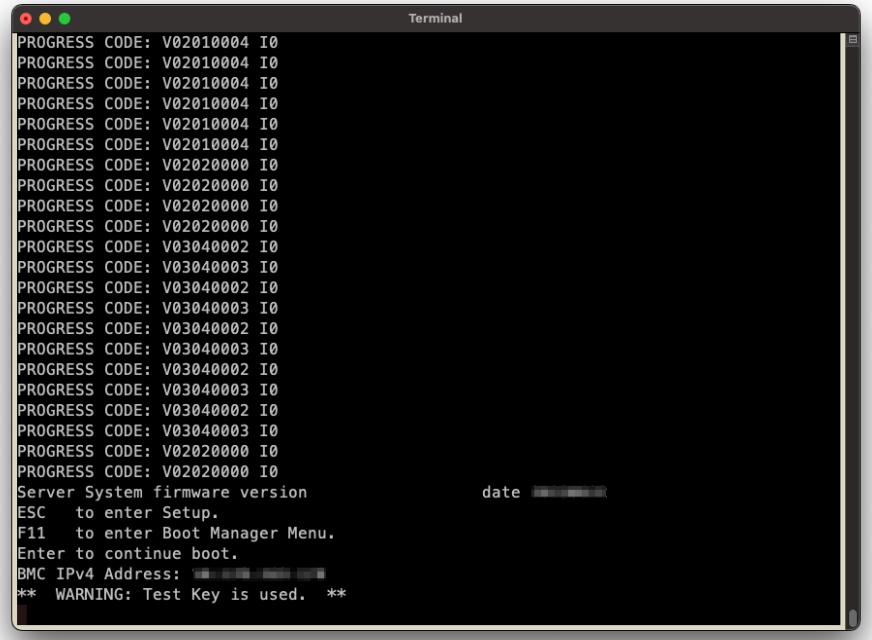
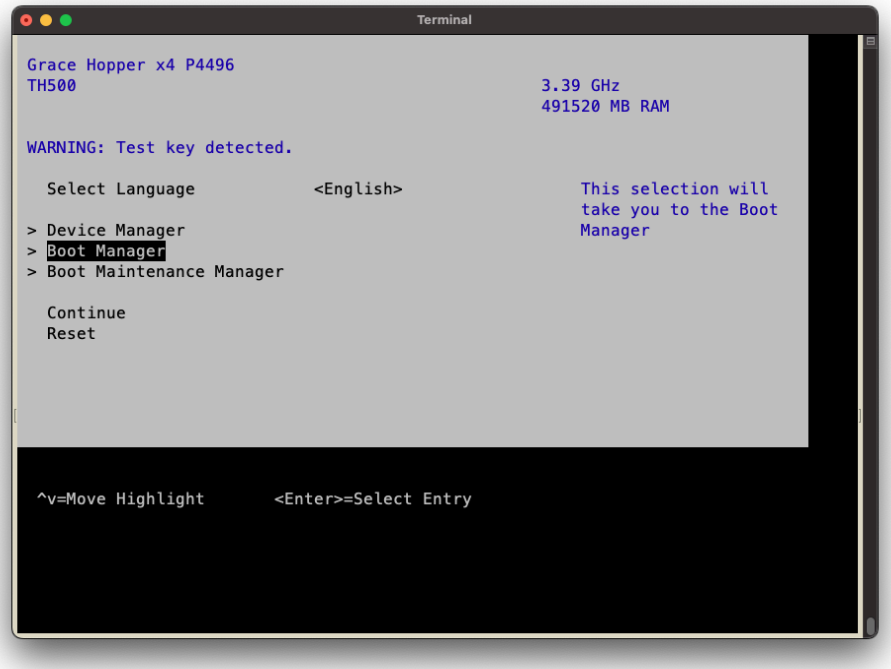
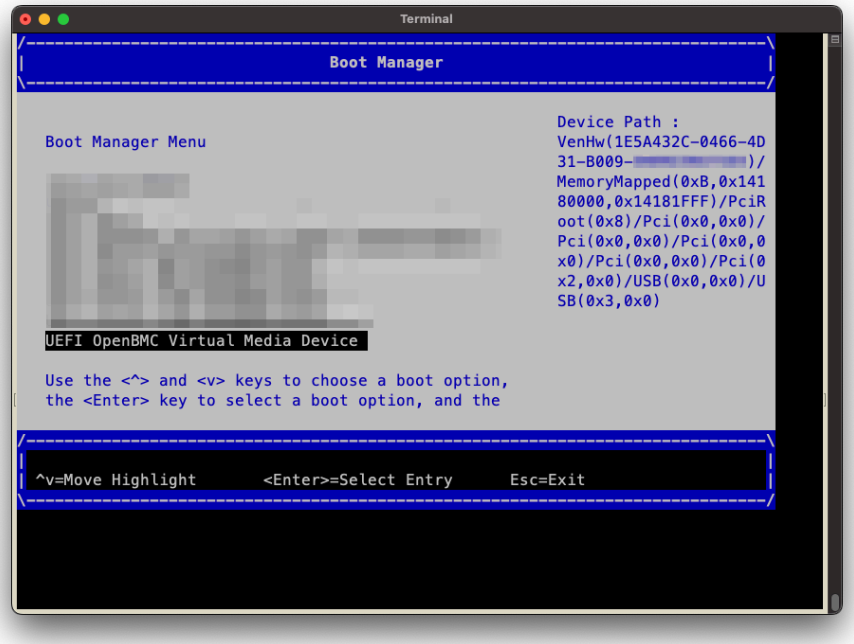


Figure 4. Boot Manager Menu



c. In the boot menu, select **UEFI OpenBMC Virtual Media Device** as the boot device and press **Enter.**
d. Follow the instructions in Installing Red Hat Enterprise Linux.

Figure 5. Boot Manager

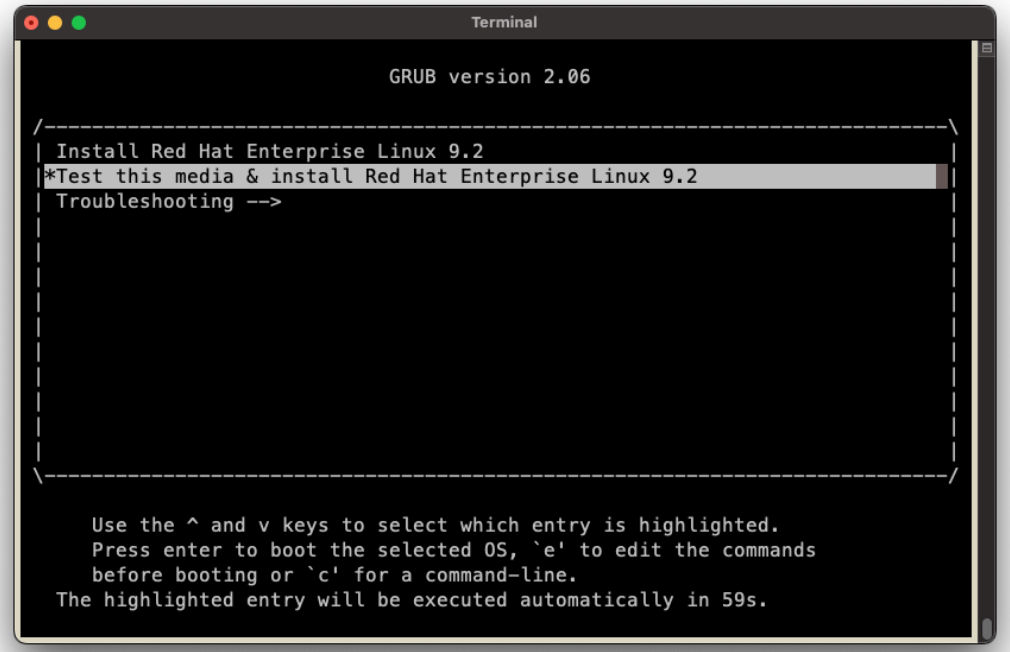


Installing Red Hat Enterprise Linux

Prerequisites: This section assumes you have already booted the RHEL ISO image.

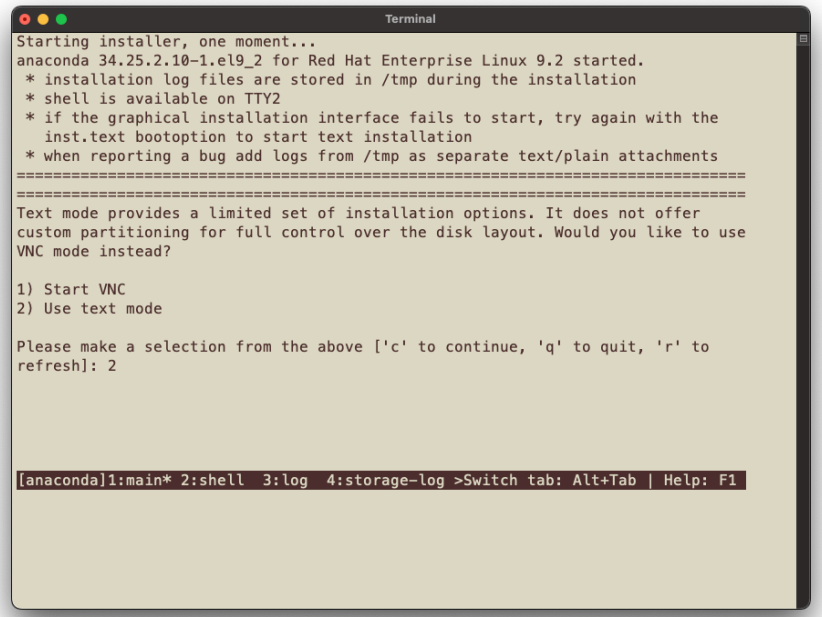
1. After booting the ISO image, the GRUB menu for the installer will appear. The menu will time out after 60 seconds.
2. If no action is taken, it will proceed with booting the **Test this media & install Red Hat Enterprise Linux 9.2** default selection.

Figure 6. Install GRUB Menu



3. To automate the installation with a Kickstart configuration file, refer to Installing with Kickstart.
4. Select one of the **Install** entries at the grub menu or wait for the timeout.
   Refer to the Red Hat Enterprise Linux Quick Installation Guide for guidance on using the installer.
5. Follow the installer prompts to configure the manual installation.
6. Select **Use text mode**.

Figure 7. Installer Menu



7. Select the **Language** and the **Timezone**.

Figure 8. Language Selection

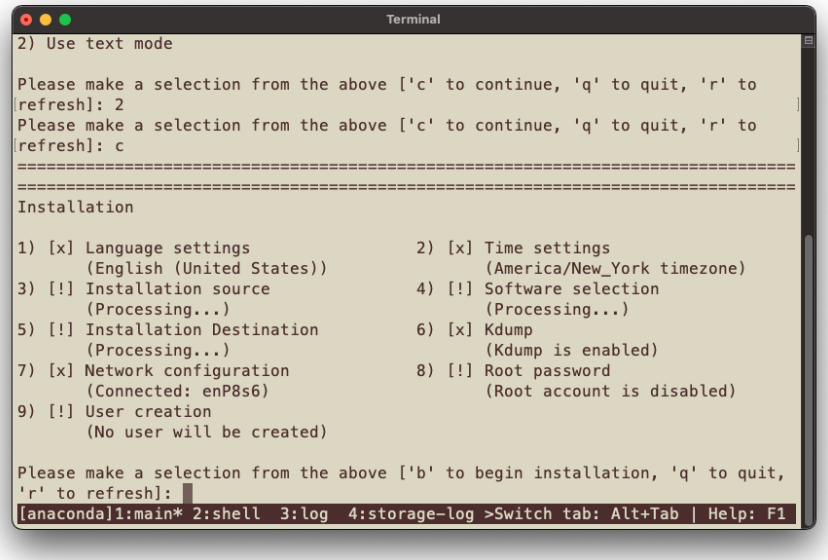
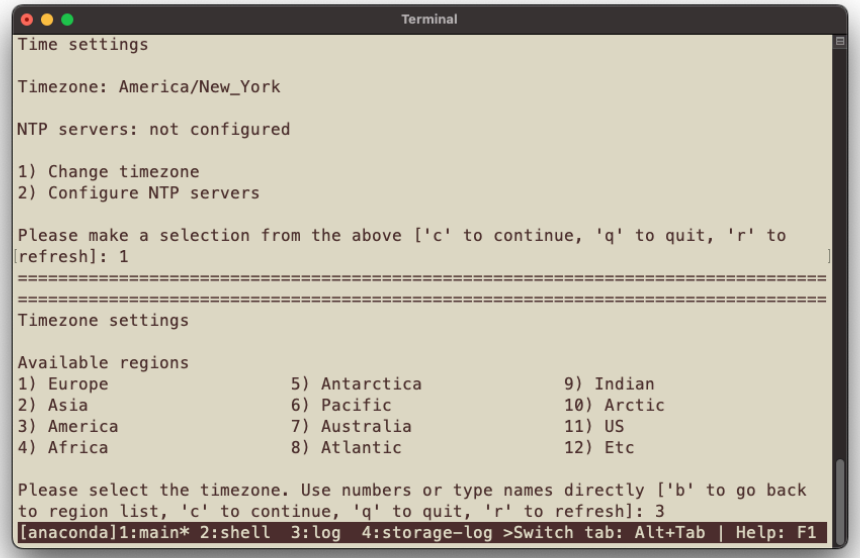


Figure 9. Timezone Selection



8. Select the Base environment and any additional software you want to install.

Figure 10. Base Environment Selection

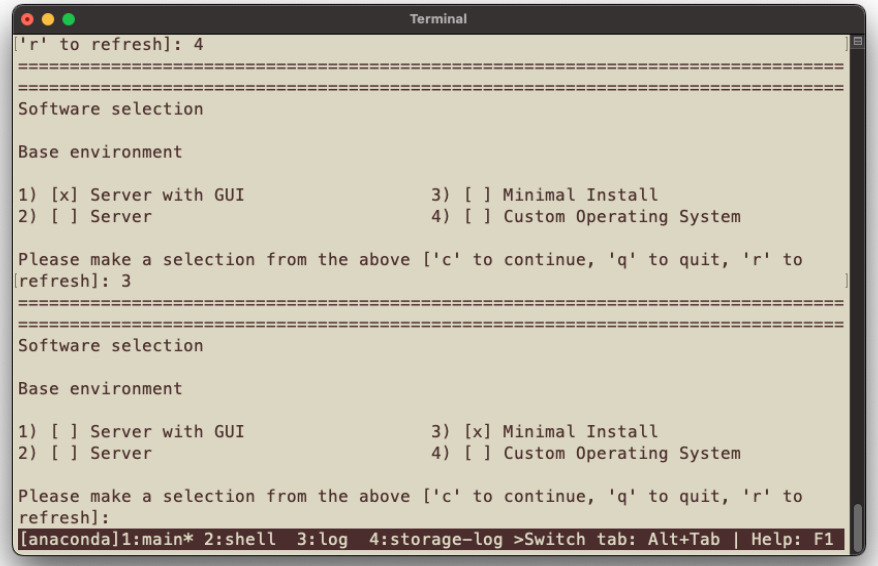
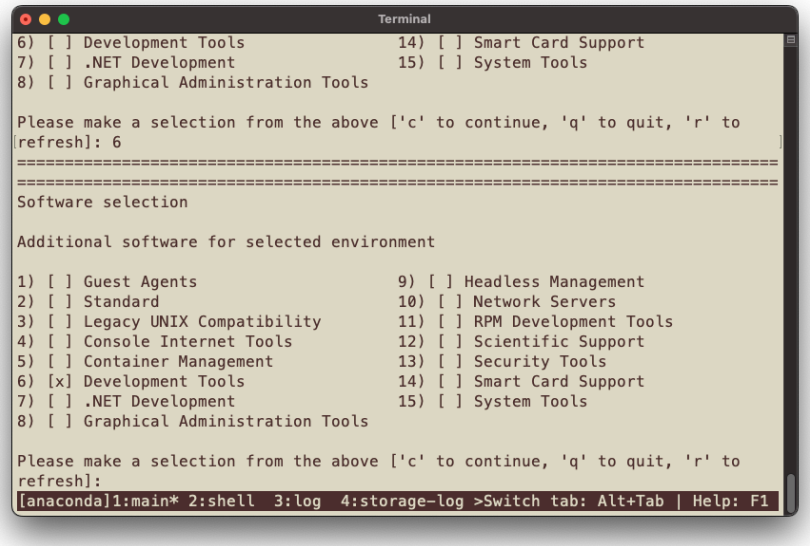
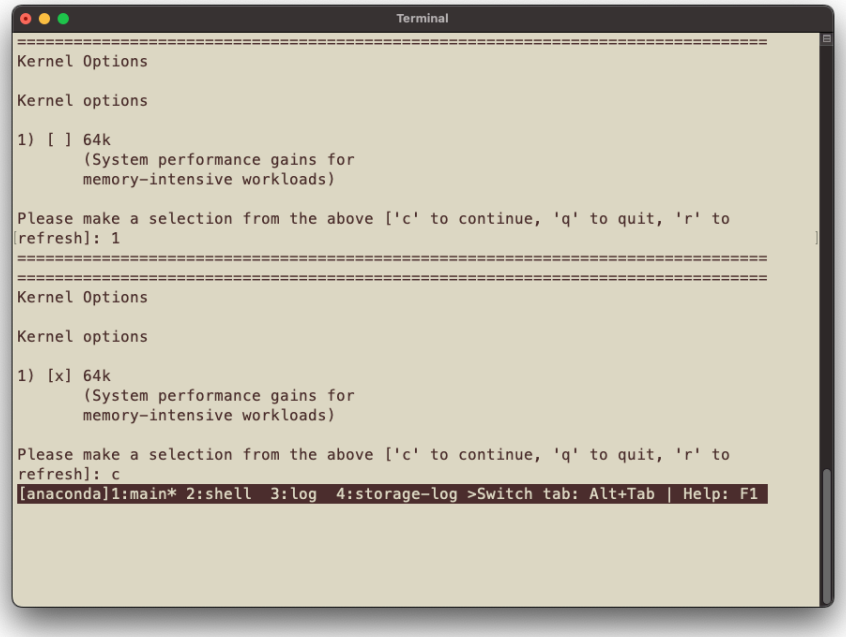


Figure 11. Software Selection



9. On **RHEL 9.3 and later**, select the 64k kernel option.

Figure 12. Kernel Options



10. Select the installation destination.

Figure 13. Destination Selection

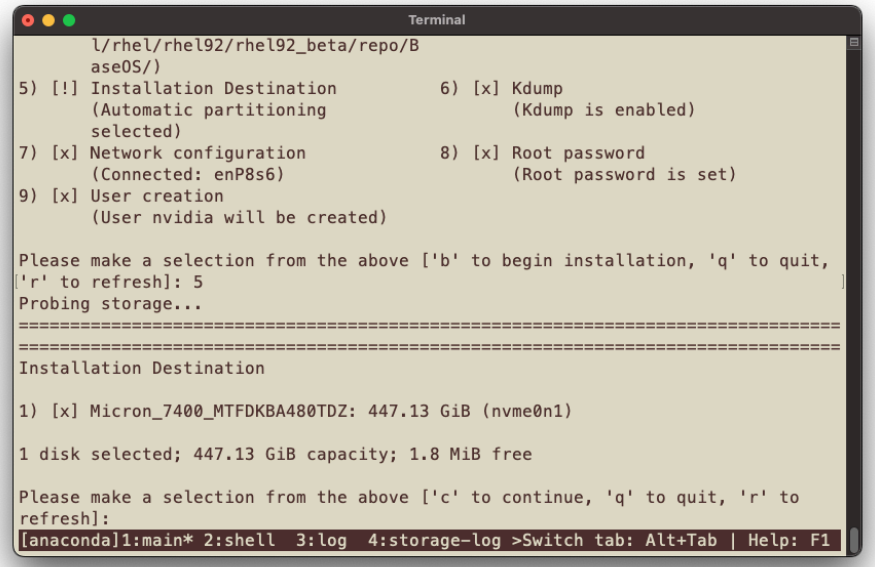


Figure 14. Partition Scheme Selection

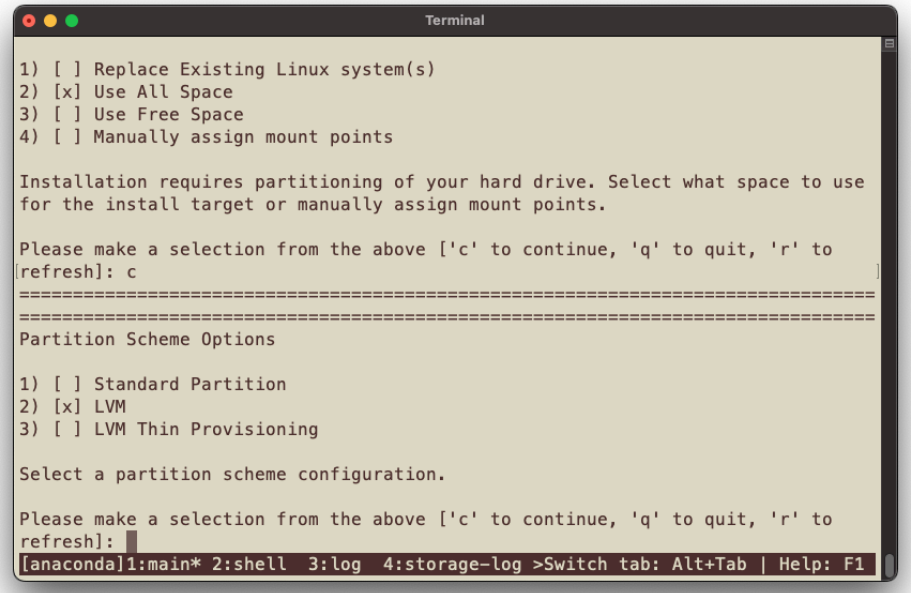
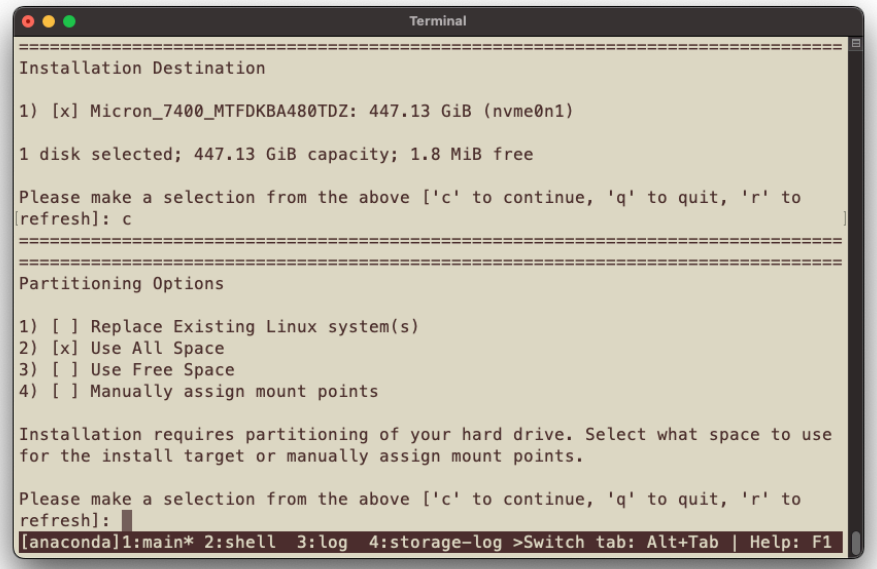
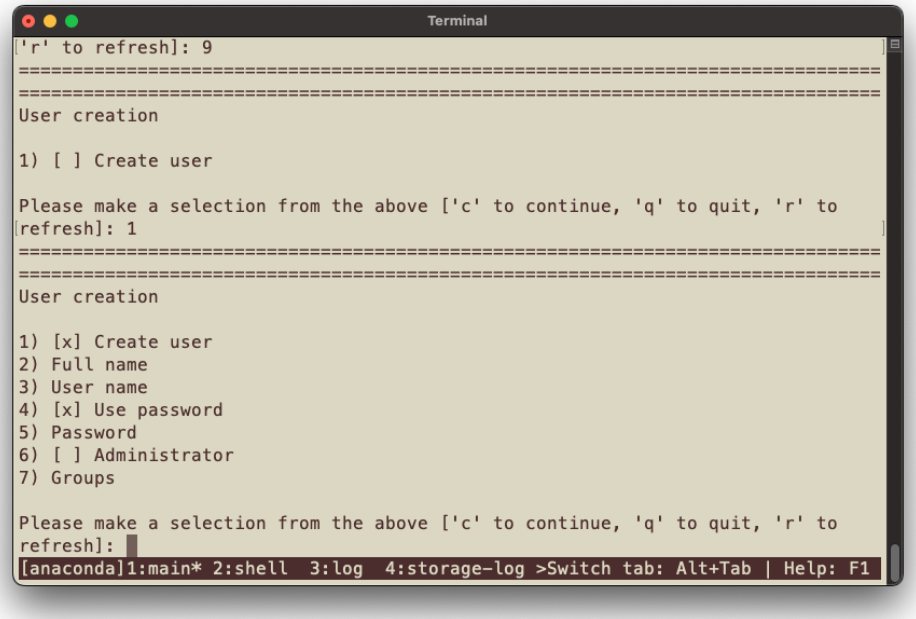


Figure 15. Partition Option Selection



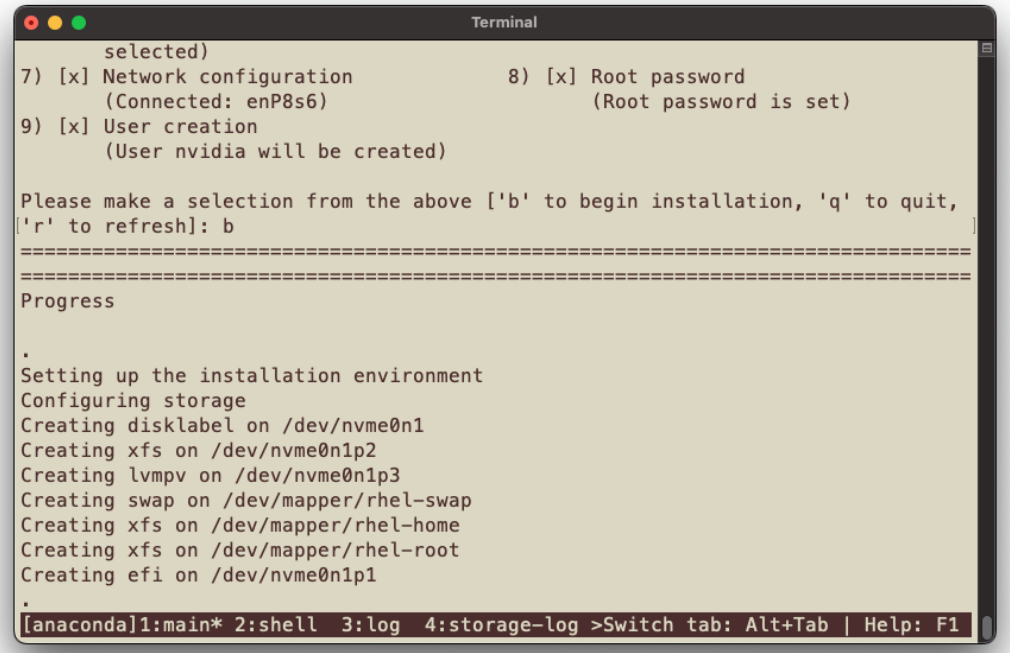
11. Create the users and set the root password.

Figure 16. User and Password Selection



12. To initiate the installation, enter **b**.

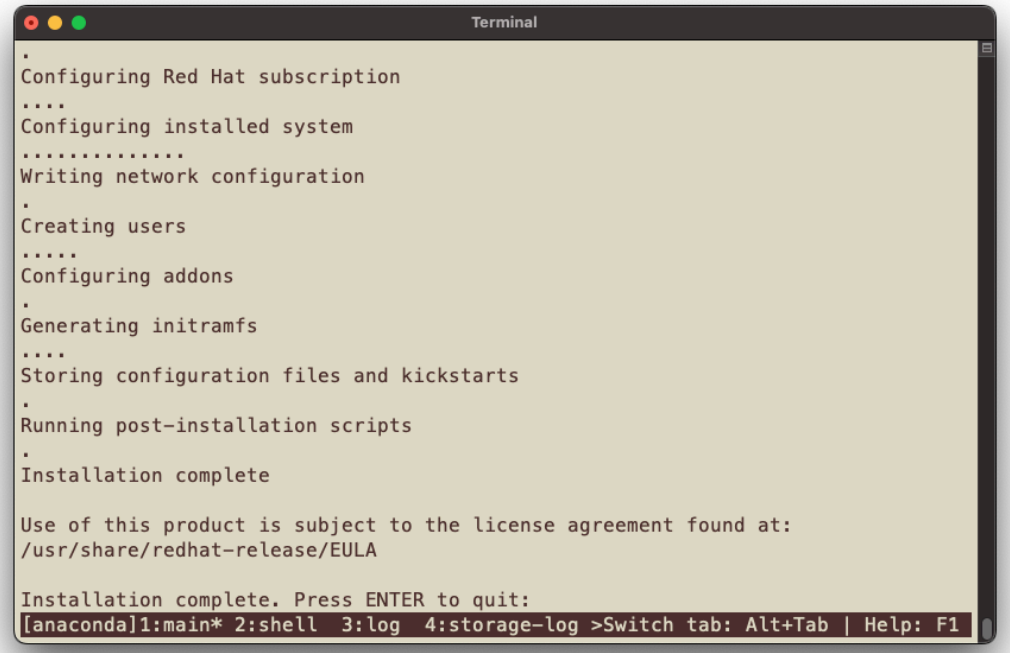
Figure 17. Installation Progress



13. After the installation is complete, the system will display a EULA prompt.

14. Press **ENTER** to quit the installer and reboot the system.

Figure 18. Installation Complete



15. If you are using **RHEL 9.3 and later**, and you selected the 64k kernel in step 8, you can skip to step 19.
16. After the system reboots, log in and install the 64k kernel as the default by running the following commands:

```
sudo dnf install -y kernel-64k
k=$(echo /boot/vmlinuz*64k)
sudo grubby --set-default=$k --update-kernel=$k --args="crashkernel=auto"
grub2-editenv - unset menu_auto_hide
sudo reboot
```

17. After the system reboots, confirm that the system is running the 64k kernel by evaluating the page size.

```
$ getconf PAGESIZE
65536
```

18. With the 64k kernel now booted, the 4k kernel must be removed per Red Hat's recommendation by running the following command.

```
dnf erase kernel
```

19. The installation on Grace is complete.

Appendix A: Installing Software

Red Hat Enterprise Linux uses the `dnf` package manager to install, update, and remove packages. The utility can also be used to manage repositories. For more information about using `dnf`, refer to the Red Hat Managing Software with the DNF tool article.

A.1 NVIDIA GPU Driver and CUDA Toolkit

Refer to the NVIDIA CUDA Installation Guide for Linux for the instructions to install the NVIDIA GPU driver and CUDA support for RHEL. The `R535.129.03` driver is the minimum level required for the Hopper GPU.

The following commands can be used to install the minimum levels required for the Hopper GPU:

```
sudo dnf install kernel-64k-devel-$(uname -r| sed 's/+64k//g')
sudo dnf install kernel-headers-$(uname -r| sed 's/+64k//g')
sudo dnf install
https://dl.fedoraproject.org/pub/epel/epel-release-latest-9.noarch.rpm
sudo dnf config-manager --add-repo
https://developer.download.nvidia.com/compute/cuda/repos/rhel9/sbsa/cuda-rhel9.repo
sudo dnf clean expire-cache
sudo dnf install cuda-toolkit-12-2
sudo dnf module install nvidia-driver:535-open
sudo systemctl status nvidia-persistenced
sudo reboot now
```

Note: The open-source GPU driver is required for Hopper GPUs.

Appendix B: Changing the BMC Login

The NVIDIA Grace servers include a base management controller (BMC) for out-of-band management of the Grace system. NVIDIA recommends that you create a unique username and password as soon as possible.

Appendix C: Automated Installation

This section provides information about automated installations.

C.1 Installing with Kickstart

Kickstart provides a way to automate the installation process by providing a configuration file with the answers to commonly asked installation questions. NVIDIA provides the following Kickstart template for Grace platforms:
https://repo.download.nvidia.com/baseos/el/el-files/9/el9-ks/grace-ks.cfg

For more information about using Kickstart files with Red Hat Enterprise Linux 9, refer to RHEL on ARM platforms with the 64k kernel.

In these files, there are tags that you must replace with site-specific information including the following:

- Language
- Keyboard
- Timezone
- Hostname

Each tag is in the `<CHANGE_YOUR_xxxx> form and` must be replaced with your information.

1. After you have a kickstart file customized for your installation, place that file in a location that can be accessed by NFS, FTP, HTTP, or HTTPS.
2. After booting from the installation medium, when the grub menu appears, press **e** to edit the grub entry and append `inst.ks=<URL>` to the list of kernel boot parameters.

For example:

```
inst.ks=https://192.168.1.2/kickstart/grace-ks.cfg
```

Appendix D: Platform-Dependent Workarounds

Some Grace platforms require temporary (or permanent) alterations to their configurations to work around known issues, such as hardware errata. These workarounds are described in the following sections by the corresponding Grace platform.

Note: Unless mentioned here, later releases of RHEL 9 do not require platform-dependent workarounds.

D.1 All Grace Platforms

- The RHEL 9.2 installation media does **not** carry a patch that is required to resolve an issue with the `ast` driver that is used to interface with the AST2600 BMC.

  The absence of this patch can manifest a variety of issues, including kernel hangs and distorted output from the on-board VGA port. Until the system is installed and running with kernel version 5.14.0-284.30.1.el9_2 or later (available through the RHEL 9.2.z update stream), a temporary workaround is required on **all** Grace platforms to avoid undefined behaviors (refer to RHSA-2023:5069 - Security Advisory for more information). As a side effect of this workaround, because the on-board VGA port is inaccessible, a serial console solution (for example, SOL) must be used for console access to the system.

  - To temporarily deploy this workaround for the duration of the current boot:
    1. During boot, stop at the grub menu, select the desired boot entry, and press the **e** key to edit the entry.
    2. Append `modprobe.blacklist=ast` to the end of the list of kernel boot parameters.
    3. Boot the entry by clicking **Ctrl-X** or pressing **F10**.
  - To permanently deploy this workaround so that it is always active upon boot:
    1. With administrative privileges, run the following command to add a boot parameter to all kernels:

       ```
       grubby --update-kernel=ALL --args="modprobe.blacklist=ast"
       ```

    2. Reboot the system.
  - To permanently remove this workaround so that it is no longer active upon boot:
    1. With administrative privileges, run the following command to remove the boot parameter from all kernels:

       ```
       grubby --update-kernel=ALL --remove-args="modprobe.blacklist=ast"
       ```

    2. Reboot the system.
  - To verify the presence of the workaround, complete one of the following options:
    - Evaluate the kernel boot parameters that were set for the current boot by running the following command:

      ```
      cat /proc/cmdline | grep ast
      ```

      When nothing is returned, the workaround is **not** active.
    - Alternatively, evaluate the loaded modules for the current boot by running the following command:

      ```
      lsmod | grep ast
      ```

      When nothing is returned, the workaround **is** active.

Note: When this workaround is applied using the temporary deployment method from the RHEL Installer, it is automatically included in the installed system.

Refer to Red Hat Modifying Kernel Boot Parameters for additional guidance about modifying kernel boot parameters.

- Due to a firmware bug that can cause an invalid memory access in the kernel, Grace platforms from some vendors might experience a crash during the installation or at boot time.

  This issue can impact any version of RHEL that is supported on the Grace platform. Until this issue is resolved, a workaround is required to allow a successful boot.

  - To temporarily deploy this workaround for the duration of the current boot:
    1. During boot, stop at the grub menu, select the desired boot entry, and press the **e** key to edit the entry.
    2. Append `video=simplefb:off` to the end of the list of kernel boot parameters.
    3. Boot the entry by clicking **Ctrl-X** or pressing **F10**.
  - To permanently deploy this workaround, so that is always active upon boot:
    1. With administrative privileges, run the following command to add a boot parameter to all kernels.

       ```
       grubby --update-kernel=ALL --args="video=simplefb:off"
       ```

    2. Reboot the system.
  - To verify the presence of the workaround, evaluate the kernel boot parameters set for the current boot by running the following command:

    ```
    cat /proc/cmdline | grep simplefb
    ```

    When nothing is returned, the workaround is **not** active.

Note: When this workaround is applied using the temporary deployment method from the RHEL Installer, it is automatically included in the installed system.

Refer to Red Hat Modifying Kernel Boot Parameters for additional guidance about modifying kernel boot parameters.

D.2 Multi-Socket Grace Platforms

The RHEL 9.2 installation media does **not** carry a patch required to work around NVIDIA hardware erratum T241-FABRIC-4. This errata impacts Grace systems with three-and four-socket configurations, for example, Grace Hopper x4. Until the system is installed and running with kernel version 5.14.0-284.30.1.el9_2 or later (available through the RHEL 9.2.z update stream), a temporary workaround is required on impacted Grace platforms to avoid undefined behaviors (refer to RHSA-2023:5069 - Security Advisory for more information).

- The temporary workaround will restrict the system to one socket configuration, which reduces the server to a quarter of its total compute capacity.

Caution: Be careful when profiling a Grace Hopper x4 system with this temporary workaround.

- To temporarily deploy this workaround for the duration of the current boot:
  1. During boot, stop at the grub menu, select the desired boot entry, and press the **e** key to edit the entry.
  2. Append `nr_cpus=72` to the end of the list of kernel boot parameters.
  3. Boot the entry by clicking **Ctrl-X** or pressing **F10**.
- To permanently deploy this workaround so that is always active upon boot:
  1. With administrative privileges, run the following command to add a boot parameter to all kernels:

     ```
     grubby --update-kernel=ALL --args="nr_cpus=72"
     ```

  2. Reboot the system.
- To permanently remove this workaround, so that it is no longer active upon boot:
  1. With administrative privileges, run the following command to remove the boot parameter from all kernels:

     ```
     grubby --update-kernel=ALL --remove-args="nr_cpus=72"
     ```

  2. Reboot the system.
- To verify the presence of the workaround, complete one of the following tasks
  - Evaluate the kernel boot parameters set for the current boot.

    ```
    cat /proc/cmdline | grep nr_cpus
    ```

    When nothing is returned, the workaround is **not** active.
  - Evaluate the CPU configuration for the current boot.

    ```
    lscpu | grep -E "NUMA node[0-9]+"
    ```

    Only one of the nodes should be populated [with CPUs 0-71].

Note: When this workaround is applied using the temporary deployment method from the RHEL Installer, it is automatically included in the installed system.

Refer to Red Hat Modifying Kernel Boot Parameters for additional guidance about modifying kernel boot parameters.

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 Material (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.

NVIDIA products are not designed, authorized, or warranted to be suitable for use in medical, military, aircraft, space, or life support equipment, nor in applications where failure or malfunction of the NVIDIA product can reasonably be expected to result in personal injury, death, or property or environmental damage. NVIDIA accepts no liability for inclusion and/or use of NVIDIA products in such equipment or applications and therefore such inclusion and/or use is at customer's own risk.

NVIDIA makes no representation or warranty that products based on this document will be suitable for any specified use. Testing of all parameters of each product is not necessarily performed by NVIDIA. It is customer's sole responsibility to evaluate and determine the applicability of any information contained in this document, ensure the product is suitable and fit for the application planned by customer, and perform the necessary testing for the application in order to avoid a default of the application or the product. Weaknesses in customer's product designs may affect the quality and reliability of the NVIDIA product and may result in additional or different conditions and/or requirements beyond those contained in this document. NVIDIA accepts no liability related to any default, damage, costs, or problem which may be based on or attributable to: (i) the use of the NVIDIA product in any manner that is contrary to this document or (ii) customer product designs.

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.

NVIDIA Corporation | 2788 San Tomas Expressway, Santa Clara, CA 95051
http://www.nvidia.com

Arm
Arm, AMBA, and ARM Powered are registered trademarks of Arm Limited. Cortex, MPCore, and Mali are trademarks of Arm Limited. All other brands or product names are the property of their respective holders. "Arm" is used to represent ARM Holdings plc; its operating company Arm Limited; and the regional subsidiaries Arm Inc.; Arm KK; Arm Korea Limited.; Arm Taiwan Limited; Arm France SAS; Arm Consulting (Shanghai) Co. Ltd.; Arm Germany GmbH; Arm Embedded Technologies Pvt. Ltd.; Arm Norway, AS, and Arm Sweden AB.

OpenCL
OpenCL is a trademark of Apple Inc. used under license to the Khronos Group Inc.

NVIDIA Corporation | 2788 San Tomas Expressway, Santa Clara, CA 95051
http://www.nvidia.com