NVIDIA AI Enterprise

Quick Start Guide

QSG-10396-001 _v1.3 | October 2024

Table of Contents

About this Guide v

Chapter 1. Getting NVIDIA AI Enterprise 1

1.1. Before You Begin 1

1.2. Your Order Confirmation Message 1

1.3. NVIDIA Enterprise Account Requirements 3

1.4. Creating your NVIDIA Enterprise Account 4

1.5. Linking an Evaluation Account to an NVIDIA Enterprise Account for Purchased Licenses 6

1.6. Downloading NVIDIA AI Enterprise 7

Chapter 2. Accessing the Enterprise Catalog and the NGC Private Registry 11

2.1. The Enterprise Catalog 11

2.1.1. Setting Up Your Access to the Enterprise Catalog 11

2.1.2. Downloading Software from the Enterprise Catalog 17

2.1.2.1. Accessing the NVIDIA AI Enterprise Collection 17

2.1.2.2. Container Images 19

2.1.2.3. Helm Charts 19

2.1.2.4. Resources 19

2.1.3. Adding Additional Users from Your Organization to the Enterprise Catalog (Admins Only) 20

2.2. The NGC Private Registry 25

2.2.1. Accessing Your NGC Private Registry 25

2.2.2. Managing Teams and Users 27

2.2.2.1. Creating Teams 27

2.2.2.2. Creating Users 27

Chapter 3. Installing Your NVIDIA AI Enterprise License Server and License Files 28

3.1. Introduction to NVIDIA Software Licensing 28

3.2. Creating a License Server on the NVIDIA Licensing Portal 29

3.3. Creating a CLS Instance on the NVIDIA Licensing Portal 32

3.4. Binding a License Server to a Service Instance 34

3.5. Installing a License Server on a CLS Instance 34

3.6. Generating a Client Configuration Token for a CLS Instance 35

Chapter 4. Installing and Configuring NVIDIA vGPU Manager and the Guest Driver 39

4.1. Switching the Mode of a GPU that Supports Multiple Display Modes 39

4.2. Installing the NVIDIA Virtual GPU Manager on VMware vSphere 40

4.3. Disabling and Enabling ECC Memory 41

4.3.1. Disabling ECC Memory..... 42
4.3.2. Enabling ECC Memory..... 43
4.4. Changing the Default Graphics Type in VMware vSphere 6.5 and Later..... 44
4.5. Configuring a vSphere VM with NVIDIA vGPU..... 51
Chapter 5. Installing and Licensing NVIDIA AI Enterprise Components Required in a Guest VM..... 56
5.1. Installing the NVIDIA AI Enterprise Graphics Driver on Linux from a Debian Package.... 56
5.2. Prerequisites for Configuring a Licensed Client of NVIDIA License System with a Networked License..... 57
5.2.1. Configuring a Licensed Client with a Networked License on Linux with Default Settings..... 57
5.2.2. Verifying the NVIDIA AI Enterprise License Status of a Licensed Client..... 59
5.3. Installing NVIDIA Container Toolkit..... 60
5.4. Verifying the Installation of NVIDIA Container Toolkit..... 61
5.5. Installing Software Distributed as Container Images..... 61
5.6. Running ResNet-50 with TensorRT..... 62
5.7. Running ResNet-50 with TensorFlow..... 63
Chapter 6. Additional Information..... 65

List of Figures

Figure 1. Shared default graphics type 46
Figure 2. Host graphics settings for vGPU 48
Figure 3. Shared graphics type 49
Figure 4. Graphics device settings for a physical GPU 50
Figure 5. Shared direct graphics type 51
Figure 6. VM settings for vGPU 53

About this Guide

NVIDIA AI Enterprise Quick Start Guide provides minimal instructions for installing and configuring NVIDIA® virtual GPU software on the Citrix Hypervisor or VMware vSphere hypervisor and for installing and configuring a Cloud License Service (CLS) instance or a standalone Delegated License Service (DLS) instance. The instructions for configuring a DLS instance assume that the VM that hosts the DLS instance has been assigned an IP address automatically. If you need complete instructions, are using other platforms, are hosting a DLS instance on a VM that has not been assigned an IP address automatically, or require high availability for a DLS instance, refer to *NVIDIA AI Enterprise User Guide* and *NVIDIA License System User Guide*. If you want to use the legacy NVIDIA AI Enterprise license server, refer to *Virtual GPU License Server Release Notes* and *Virtual GPU License Server User Guide*.

NVIDIA AI Enterprise Quick Start Guide provides minimal instructions for installing and configuring NVIDIA AI Enterprise on a single node and for configuring a Cloud License Service (CLS) instance. If you need complete instructions, are using multiple nodes, or are using Delegated License Service (DLS) instances to serve licenses, refer to *NVIDIA AI Enterprise User Guide* and *NVIDIA License System User Guide*.

Chapter 1. Getting NVIDIA AI Enterprise

After your order for NVIDIA AI Enterprise is processed, you will receive an order confirmation message from NVIDIA. This message contains information that you need for getting NVIDIA AI Enterprise from the NVIDIA Licensing Portal. To log in to the NVIDIA Licensing Portal, you must have an NVIDIA Enterprise Account.

1.1. Before You Begin

Before following the procedures in this guide, ensure that the following prerequisites are met:

- You have a server platform that is capable of hosting your chosen hypervisor and NVIDIA GPUs that support NVIDIA AI Enterprise. For a list of validated server platforms, refer to NVIDIA GRID Certified Servers.
- One or more NVIDIA GPUs that support NVIDIA AI Enterprise is installed in your server platform.
- A supported virtualization software stack is installed according to the instructions in the software vendor's documentation.
- A virtual machine (VM) running a supported Windows guest operating system (OS) is configured in your chosen hypervisor.

For information about supported hardware and software, and any known issues for this release of NVIDIA AI Enterprise, refer to the *Release Notes* for your chosen hypervisor:

- *NVIDIA AI Enterprise Release Notes*
- *NVIDIA AI Enterprise Release Notes*

For information about supported hardware and software, and any known issues for this release of NVIDIA AI Enterprise, refer to *NVIDIA AI Enterprise Release Notes*.

1.2. Your Order Confirmation Message

After your order for NVIDIA AI Enterprise is processed, you will receive an order confirmation message to which your NVIDIA Entitlement Certificate is attached.

Thank you for your software and/or services order!

Please find enclosed your Entitlement Certificate for the Software and/or Services products you ordered.

Please refer to the attached Entitlement Certificate to register for your software and services.

The following is your order information:

PO Number	NVIDIA Sales Order	NVIDIA Delivery Number

Questions?
NVIDIA Enterprise Support contact information can be found here https://www.NVIDIA.com/en-us/support/enterprise/

Your NVIDIA Entitlement Certificate contains your product activation keys.

NVIDIA Corporation
2788 San Tomas Expressway
SANTA CLARA CA 95051
USA

NVIDIA® Entitlement Certificate
This certificate serves as evidence that NVIDIA has entitled you for the following product(s).

End Customer ( )

NVIDIA Delivery	
Entitlement Date	16 AUG 2021
PO Number	
NVIDIA Sales Order	

No	Entitlement Description	Quantity	Sales Type	Term	Start Date	End Date
1	NVIDIA AI Enterprise Subscription License and Support per CPU Socket PAK ID	2 EA	Initial	3 Years	16 AUG 2021	15 AUG 2024

Please follow the instructions provided in the following section to register your entitlements.

Thank you for your order!

Your NVIDIA Entitlement Certificate also provides instructions for using the certificate.

NOTICE

HOW TO USE THIS CERTIFICATE

Registration Instructions

Please refer to your NVIDIA AI Enterprise Quick Start Guide for information on how to get started, including additional instructions on how to register for your entitlement.

Sales Type: Initial

Already have NVIDIA AI Enterprise entitlements? Please Login.

New to NVIDIA AI Enterproise entitlements? Please register and follow instructions on the registration page.

You will get an email to set up your password for the NVIDIA Application Hub.

After you have successfully registered, please wait for up to 2 business days for a second email to be sent to you to set up your profile and log into the NVIDIA GPU Cloud (NGC) to access your NVIDIA AI Enterprise software in the NGC Enterprise Catalog.

You can also click here if you wish to contact NVIDIA Enterprise Support or access the NVIDIA Support Portal or the NVIDIA Licensing Portal to view your NVIDIA AI Enterprise entitlements.

Questions?
NVIDIA Enterprise Support contact information can be found here.

Rights and restrictions on the use, transfer and copying of the Software are set forth in corresponding product's NVIDIA End User License Agreement. Rights and restrictions on the use of Services are set forth in NVIDIA's corresponding service program's End User Terms and Conditions.

1.3. NVIDIA Enterprise Account Requirements

To get NVIDIA AI Enterprise, you must have a suitable NVIDIA Enterprise Account for accessing your licenses.

Note: For a Support, Upgrade, and Maintenance Subscription (SUMS) renewal, you should already have a suitable NVIDIA Enterprise Account and this requirement should already be met. However, if you have an account that was created for an evaluation license and you want to access licenses that you purchased, you must repeat the registration process.

- If you do not have an account, follow the **Register** link in the instructions for using the certificate to create your account. For details, refer to Creating your NVIDIA Enterprise Account.
- If you have an account that was created for an evaluation license and you want to access licenses that you purchased, follow the **Register** link in the instructions for using the certificate to create an account for your **purchased** licenses. You can choose to create

a separate account for your purchased licenses or link your existing account for an evaluation license to the account for your purchased licenses.

- To create a separate account for your purchased licenses, follow the instructions in Creating your NVIDIA Enterprise Account, specifying a different e-mail address than the address with which you created your existing account.
- To link your existing account for an evaluation license to the account for your purchased licenses, follow the instructions in Linking an Evaluation Account to an NVIDIA Enterprise Account for Purchased Licenses, specifying the e-mail address with which you created your existing account.

- If you already have a suitable NVIDIA Enterprise Account for accessing your licenses, follow the **Login** link in the instructions for using the certificate to log in to the NVIDIA Enterprise Application Hub, go to the NVIDIA Licensing Portal, and download your NVIDIA AI Enterprise. For details, refer to Downloading NVIDIA AI Enterprise.

1.4. Creating your NVIDIA Enterprise Account

If you do not have an NVIDIA Enterprise Account, you must create an account to be able to log in to the NVIDIA Licensing Portal.

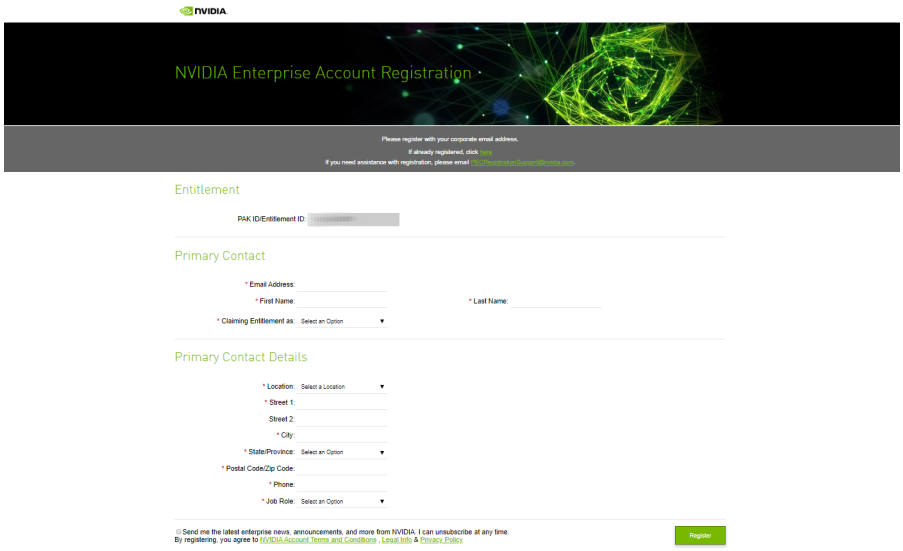
If you already have an account, skip this task and go to Downloading NVIDIA AI Enterprise.

However, if you have an account that was created for an evaluation license and you want to access licenses that you purchased, you must repeat the registration process when you receive your purchased licenses. You can choose to create a separate account for your purchased licenses or link your existing account for an evaluation license to the account for your purchased licenses.

- To create a separate account for your purchased licenses, perform this task, specifying a different e-mail address than the address with which you created your existing account.
- To link your existing account for an evaluation license to the account for your purchased licenses, follow the instructions in Linking an Evaluation Account to an NVIDIA Enterprise Account for Purchased Licenses, specifying the e-mail address with which you created your existing account.

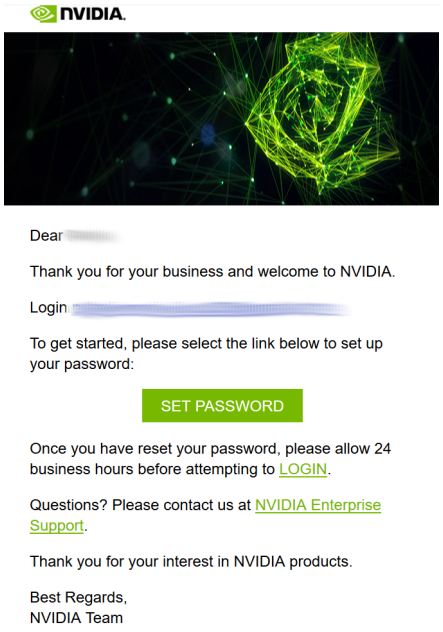
Before you begin, ensure that you have your order confirmation message.

1. In the instructions for using your NVIDIA Entitlement Certificate, follow the **Register** link.
2. Fill out the form on the **NVIDIA Enterprise Account Registration** page and click **Register**.



A message confirming that an account has been created appears, and an e-mail instructing you to set your NVIDIA password is sent to the e-mail address you provided.

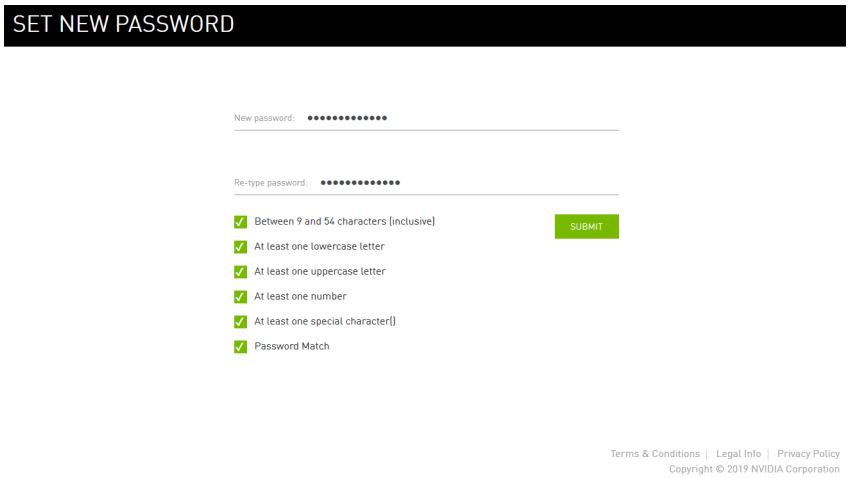
3. Open the e-mail instructing you to set your password and click **SET PASSWORD**.



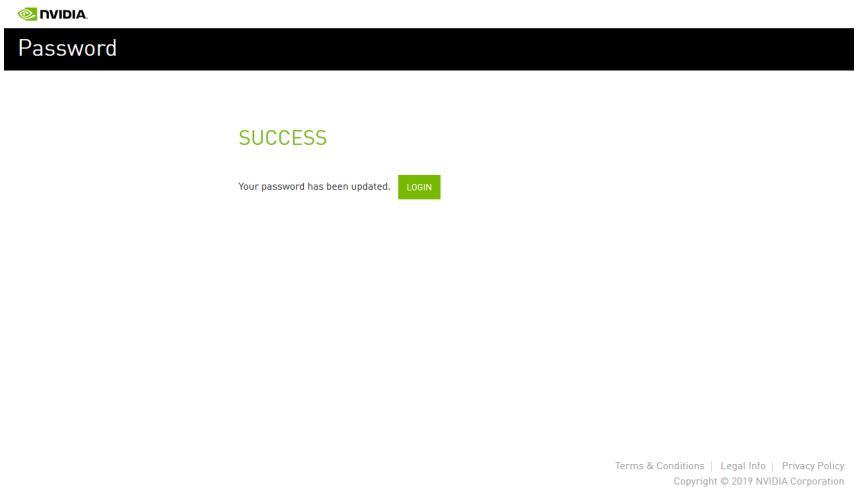
> **Note:** After you have set your password during the initial registration process, you will be able to log in to your account within 15 minutes. However, it may take up to 24 business hours for your entitlement to appear in your account.

For your account security, the **SET PASSWORD** link in this e-mail is set to expire in 24 hours.

4. Enter and re-enter your new password, and click **SUBMIT**.



A message confirming that your password has been set successfully appears.



You are then automatically directed to log in to the NVIDIA Licensing Portal with your new password.

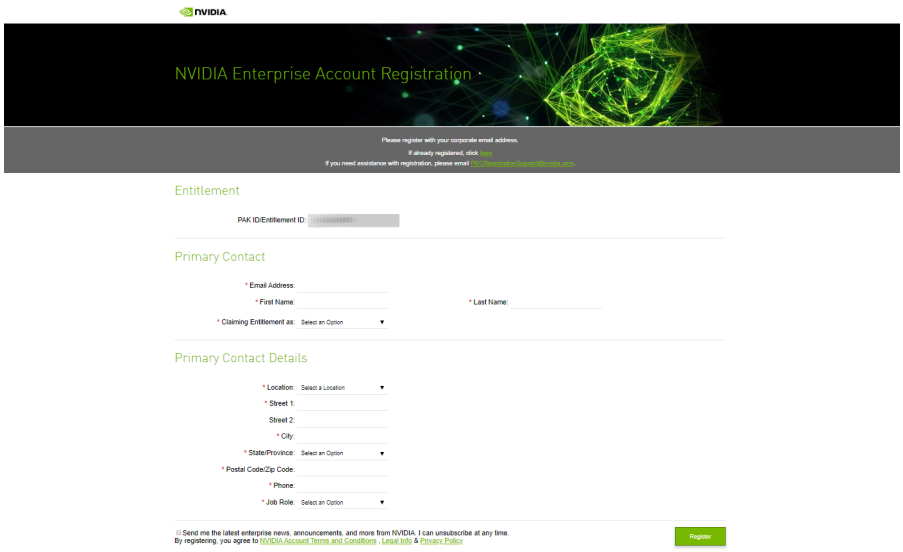
1.5. Linking an Evaluation Account to an NVIDIA Enterprise Account for Purchased Licenses

If you have an account that was created for an evaluation license, you must repeat the registration process when you receive your purchased licenses. To link your existing account

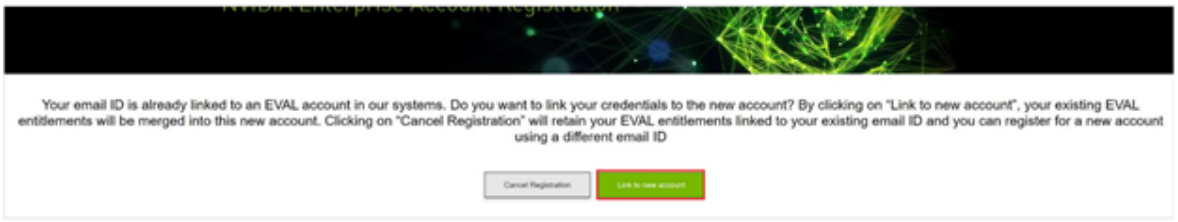
for an evaluation license to the account for your purchased licenses, register for an NVIDIA Enterprise Account with the e-mail address with which you created your existing account.

If you want to create a separate account for your purchased licenses, follow the instructions in Creating your NVIDIA Enterprise Account, specifying a different e-mail address than the address with which you created your existing account.

1. In the instructions for using the NVIDIA Entitlement Certificate **for your purchased licenses**, follow the **Register** link.
2. Fill out the form on the **NVIDIA Enterprise Account Registration** page, specifying the e-mail address with which you created your existing account, and click **Register**.



3. When a message stating that your e-mail address is already linked to an evaluation account is displayed, click **LINK TO NEW ACCOUNT**.

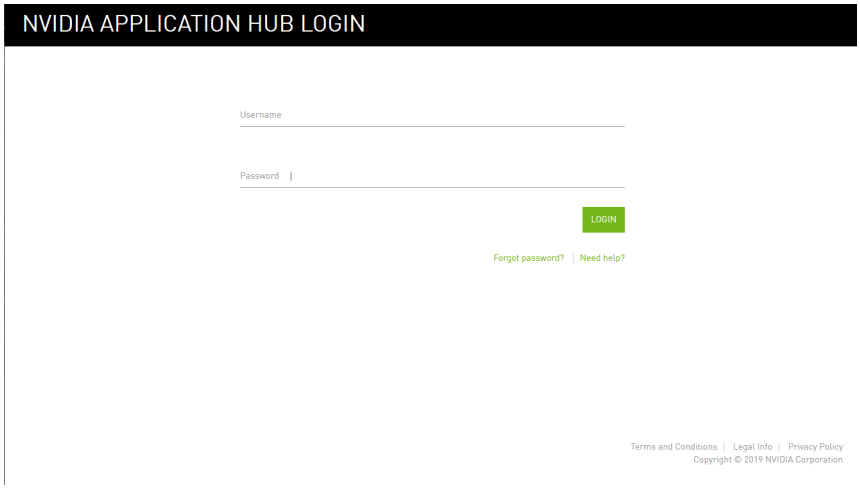


Log in to the NVIDIA Licensing Portal with the credentials for your existing account.

1.6. Downloading NVIDIA AI Enterprise

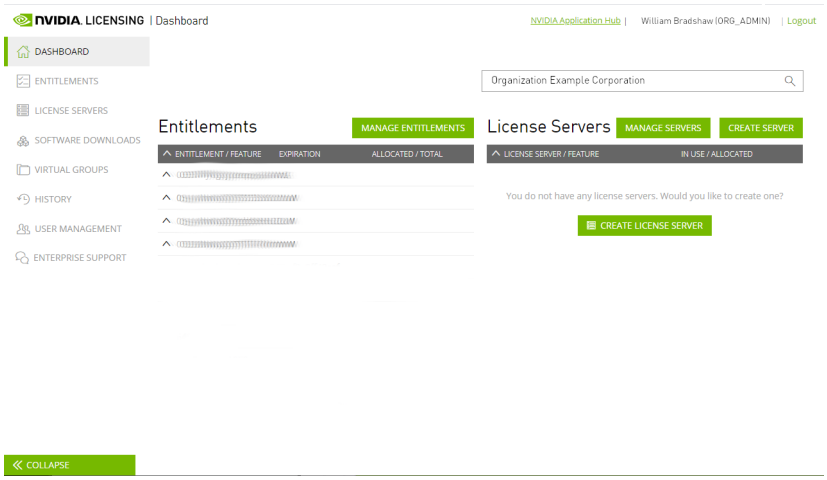
Before you begin, ensure that you have your order confirmation message and have created an NVIDIA Enterprise Account.

1. Visit the NVIDIA Enterprise Application Hub by following the **Login** link in the instructions for using your NVIDIA Entitlement Certificate or when prompted after setting the password for your NVIDIA Enterprise Account.
2. When prompted, provide your e-mail address and password, and click **LOGIN**.



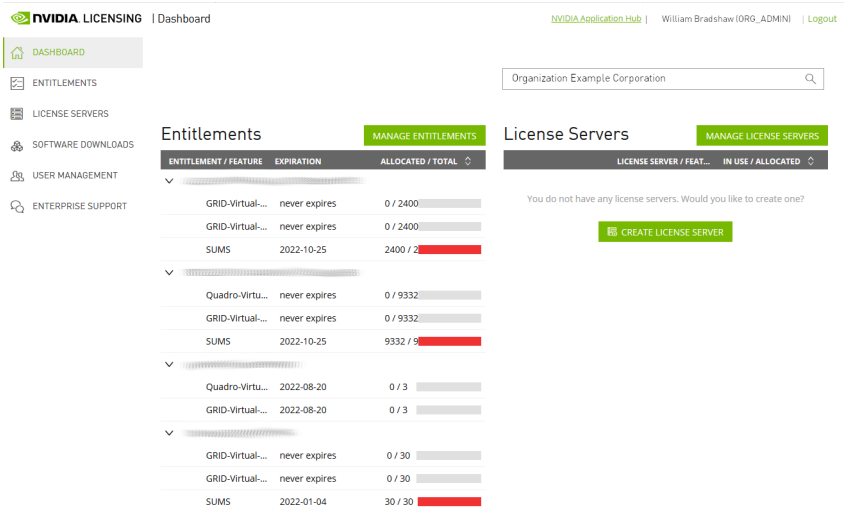
3. On the **NVIDIA APPLICATION HUB** page that opens, click **NVIDIA LICENSING PORTAL**.

   The NVIDIA Licensing Portal dashboard page opens.



> **Note:** Your entitlement might not appear on the NVIDIA Licensing Portal dashboard page until 24 business hours after you set your password during the initial registration process.

4. In the NVIDIA Licensing Portal dashboard page opens, click the down arrow next to each entitlement listed to view details of the NVIDIA AI Enterprise that you purchased.



5. In the left navigation pane of the NVIDIA Licensing Portal dashboard, click **SOFTWARE DOWNLOADS**.
6. On the **Product Download** page that opens, set the **Product Family** option to **vGPU** and follow the **Download** link for the brand and version of your chosen hypervisor for the release of NVIDIA AI Enterprise that you are using, for example, NVIDIA vGPU for vSphere 6.7 for NVIDIA AI Enterprise release 13.4.

> **Note:** To be able to download any additional software that you need for your NVIDIA AI Enterprise deployment, for example, the license server software, you **must** set the **Product Family** option to **vGPU**. Otherwise, the **ADDITIONAL SOFTWARE** button does not appear on the **Product Download** page and the pop-up window for downloading additional software is not opened.

   If the brand and version of your chosen hypervisor for the release of NVIDIA AI Enterprise that you are using aren't displayed, click **ALL AVAILABLE** to display a list of all NVIDIA AI Enterprise available for download. Use the drop-down lists or the search box to filter the software listed.
7. On the **Product Download** page that opens, set the **Product Family** option to **NVAIE** and follow the **Download** link for NVIDIA AI Enterprise.
8. When prompted to accept the license for the software that you are downloading, click **AGREE & DOWNLOAD**.
9. When the browser asks what it should do with the file, select the option to save the file.

   After the download starts, a pop-up window opens for you to download any additional software that you might need for your NVIDIA AI Enterprise deployment.

10. In the pop-up window, follow the links to download any additional software that you need for your NVIDIA AI Enterprise deployment.
    a). If you are using Delegated License Service (DLS) instances to serve licenses, follow the link to DLS 1.0 for your chosen hypervisor, for example, **DLS 1.0 for VMware vSphere**.

    For information about installing and configuring DLS instances, refer to *NVIDIA License System User Guide*.
    b). If you are using NVIDIA GPU Operator, follow the **GPU Operator vGPU Driver Catalogs** link.
    c). Follow the link to the NVIDIA AI Enterprise license server software for your license server host machine's operating system, for example, **License Manager for Windows**.
    d). If you are using an NVIDIA Tesla™ M60 or M6 GPU and think you might need to change its mode, follow the **Mode Change Utility** link.

    For details about when you need to change the mode, see #unique_10.

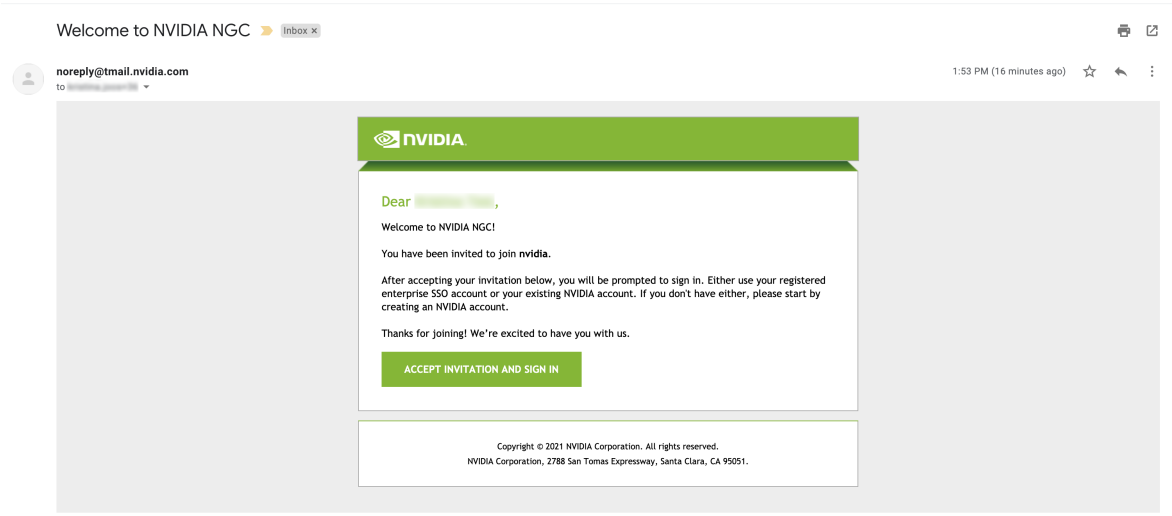
Chapter 2. Accessing the Enterprise Catalog and the NGC Private Registry

2.1. The Enterprise Catalog

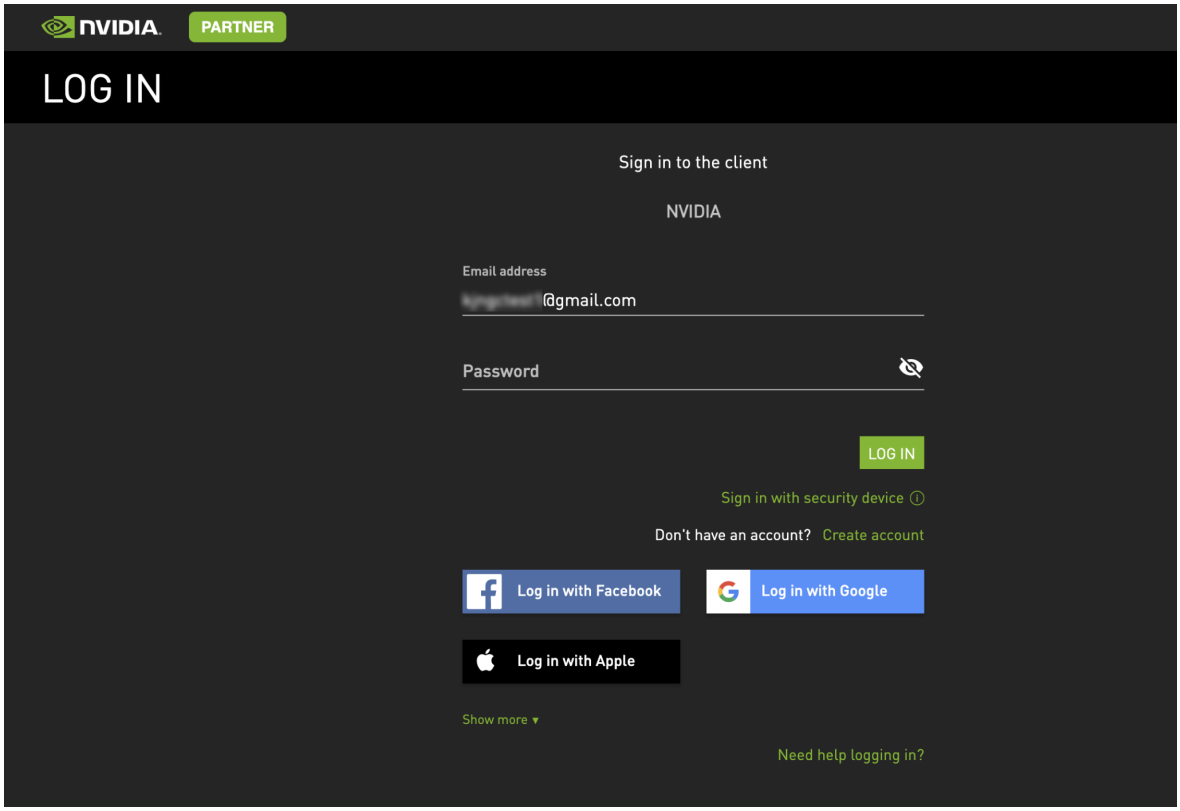
The NVIDIA AI Enterprise Software Suite is distributed through the Enterprise Catalog. After you access the Enterprise Catalog, you will see the NVIDIA AI Enterprise Software Suite collection. Detailed documentation makes it easy to utilize the software, and if additional support is required, users can submit the ticket directly from the portal.

2.1.1. Setting Up Your Access to the Enterprise Catalog

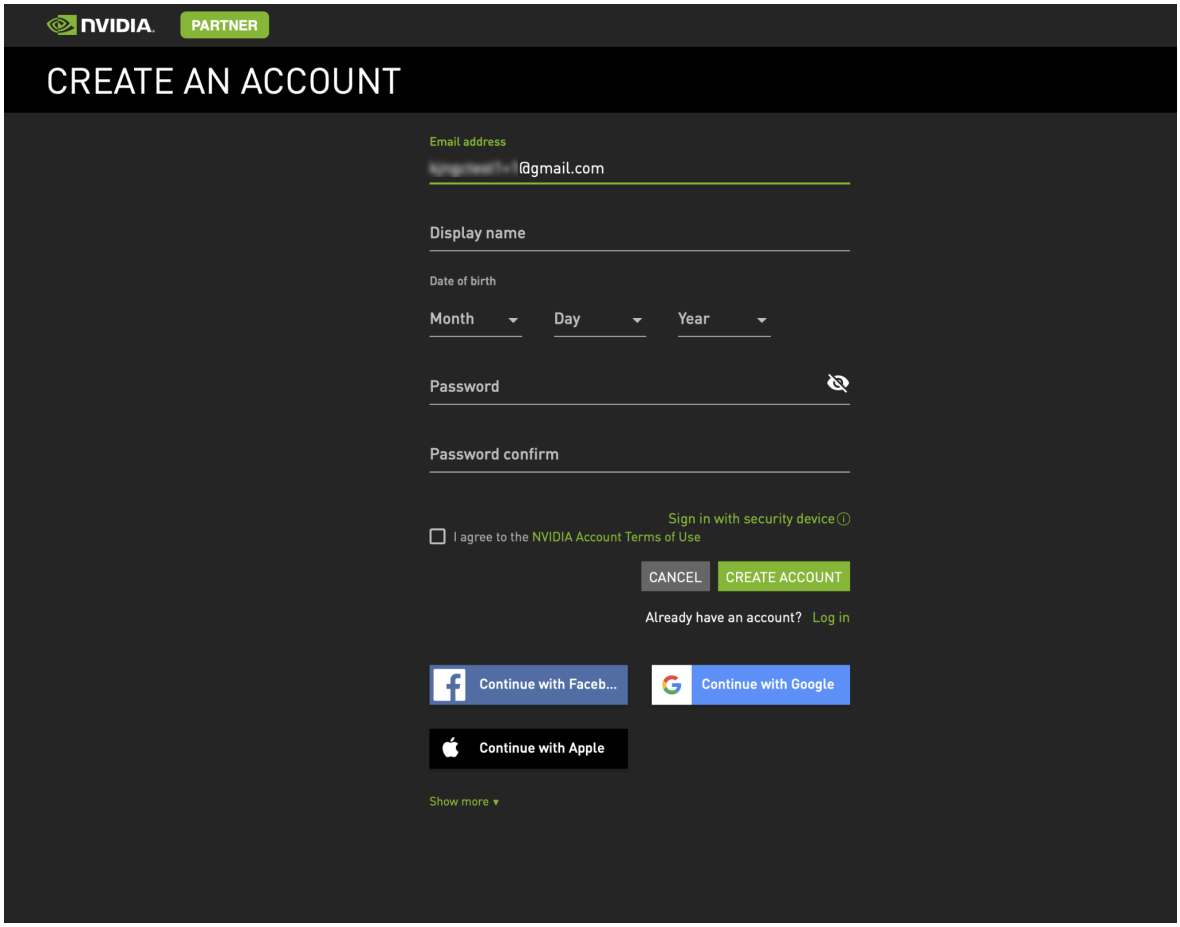
1. After your access was set up, you will receive a welcome email that invites you to continue the login process. Click on **Activate Account**.



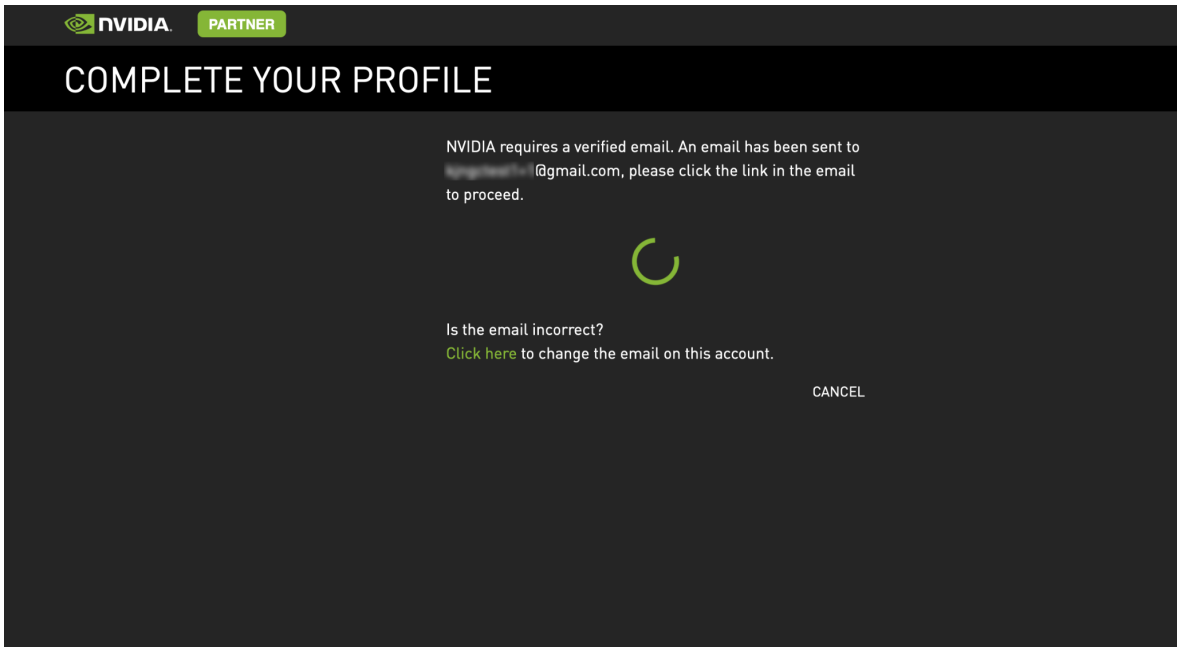
2. Click on **Create Account** to create a new NVIDIA account. *If you already have an existing NVIDIA account linked to this email address, login here.*



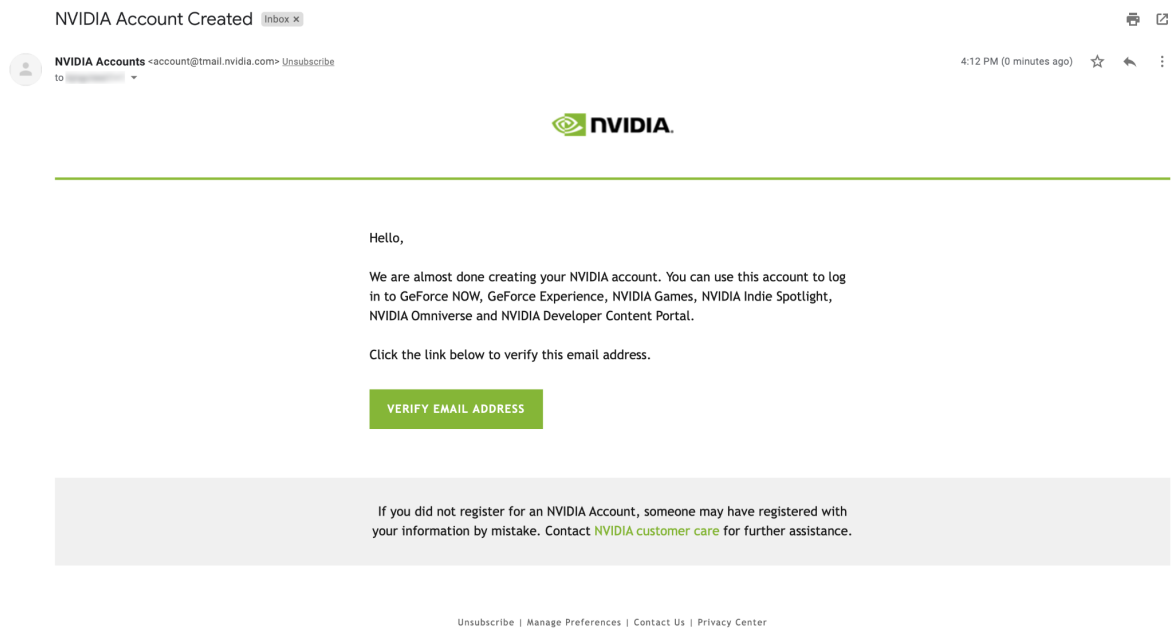
3. Provide account details and accept the NVIDIA Account Terms of Use. Click on **Create Account**.



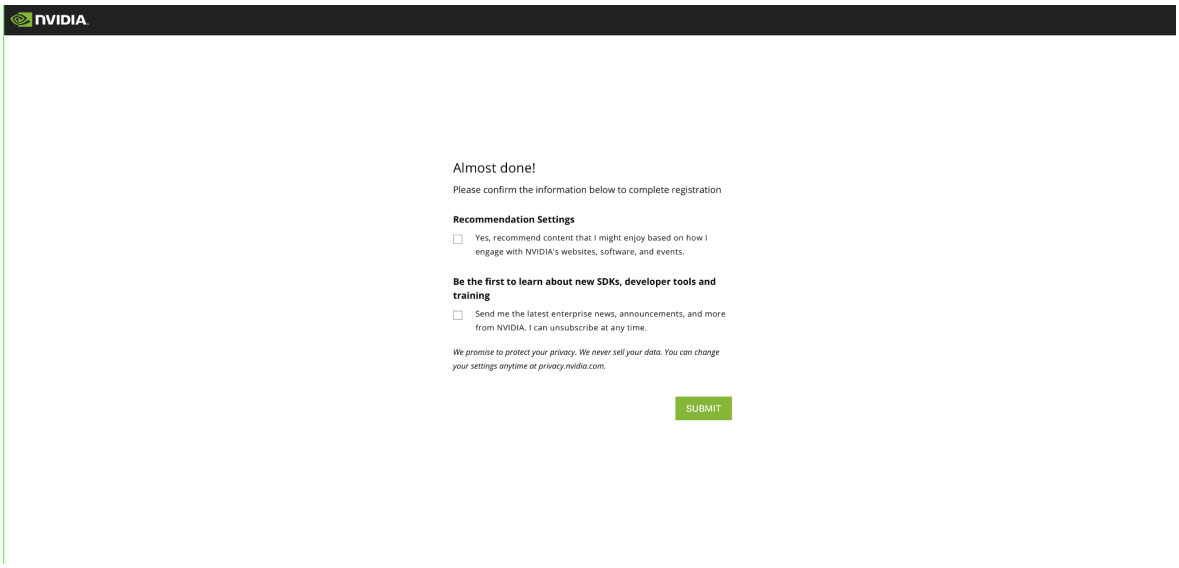
4. To complete your profile, you are asked to verify your account.



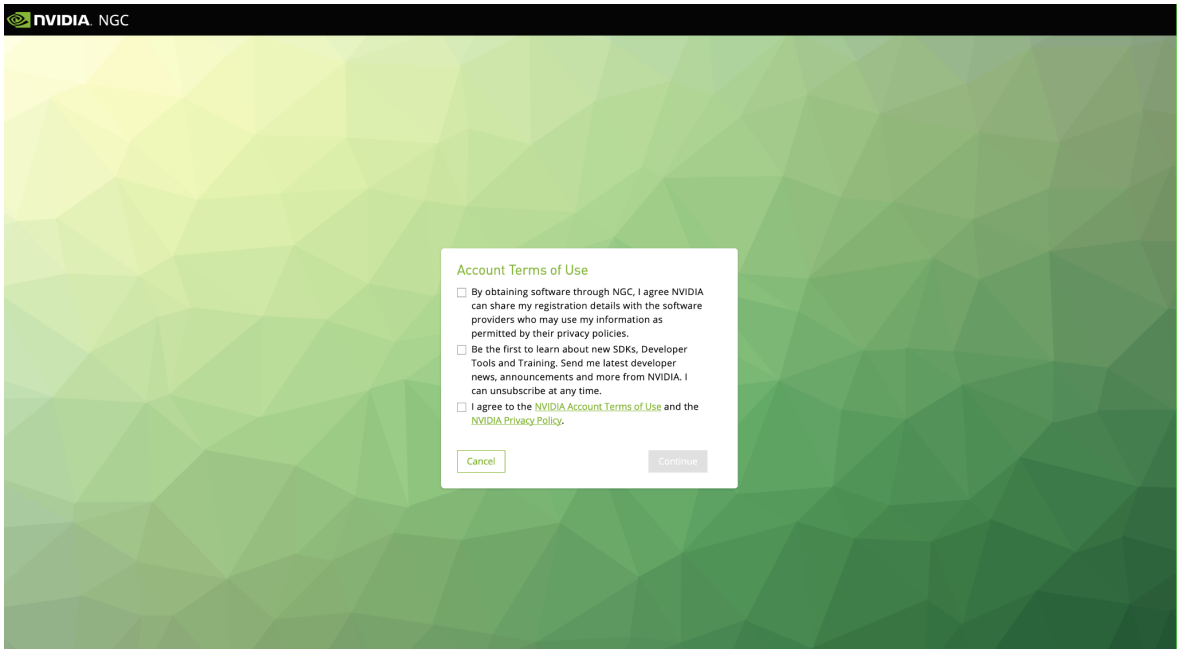
5. Go to your email inbox, open the "NVIDIA Account Created" email, and click on **Verify Email Address**.



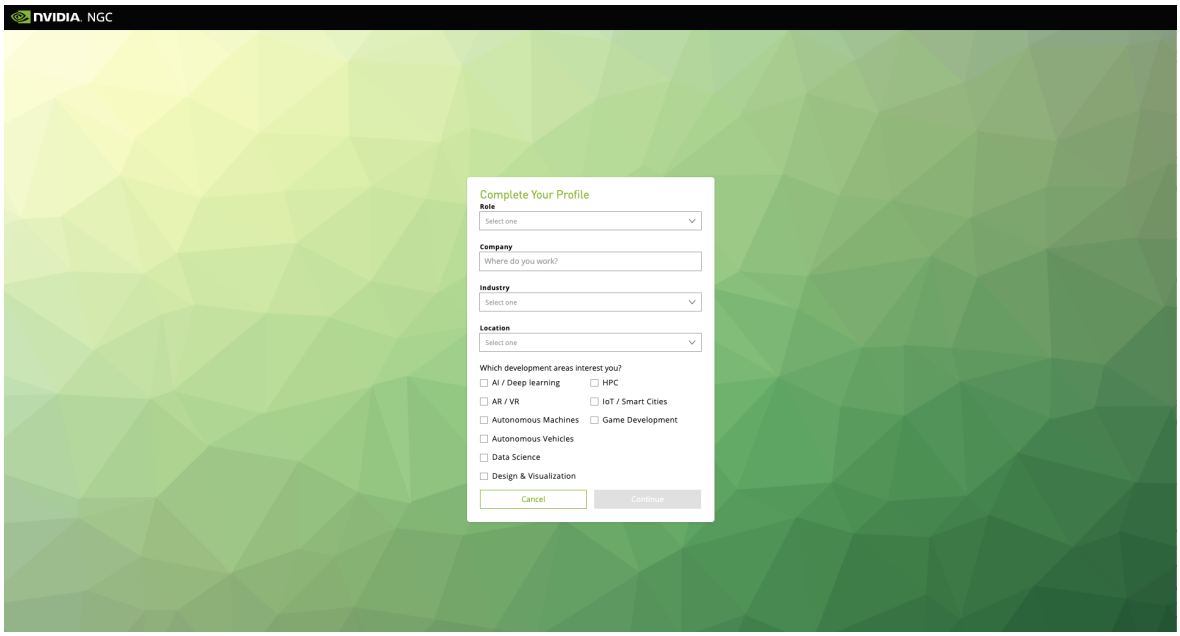
6. You are redirected to the following screen. Set your recommendation settings. Click **Submit**.



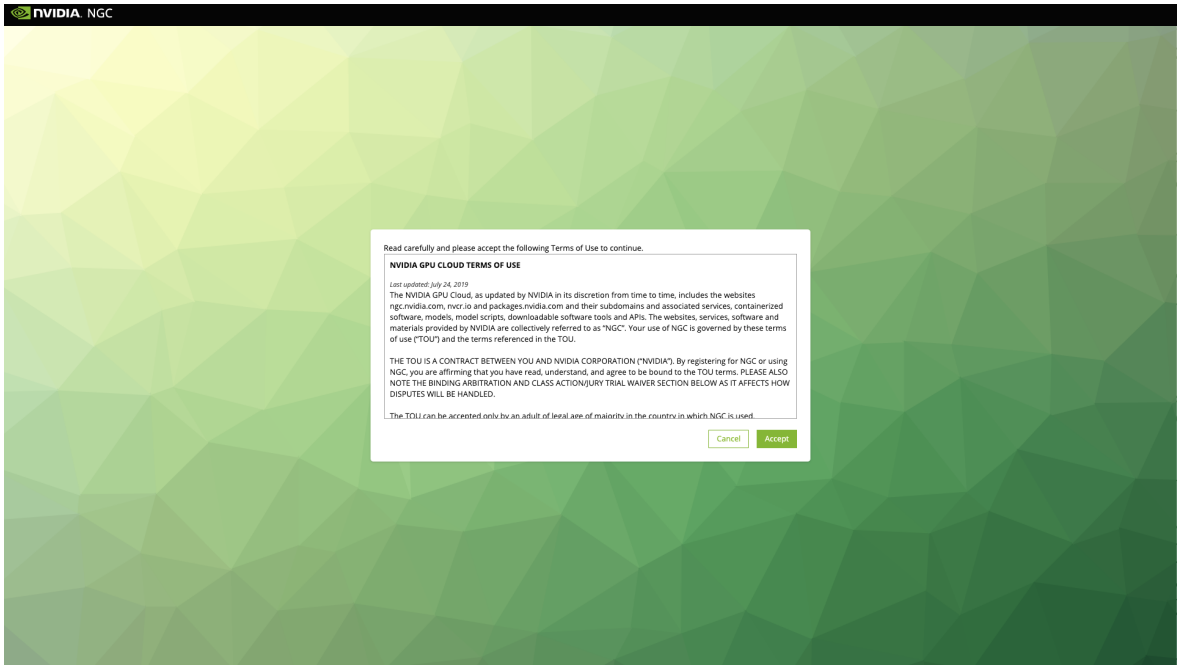
7. Review and accept the **NVIDIA Account Terms of Use** and the **NVIDIA Privacy Policy**.



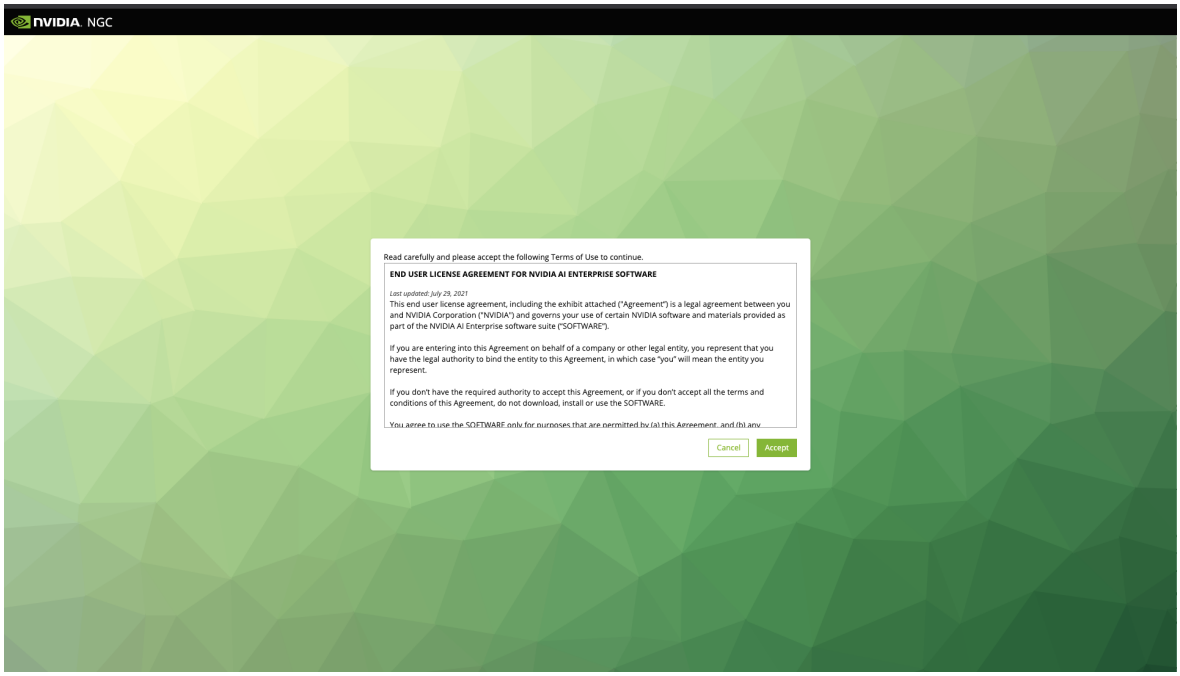
8. Complete your profile by providing the information below. Click **Continue**.



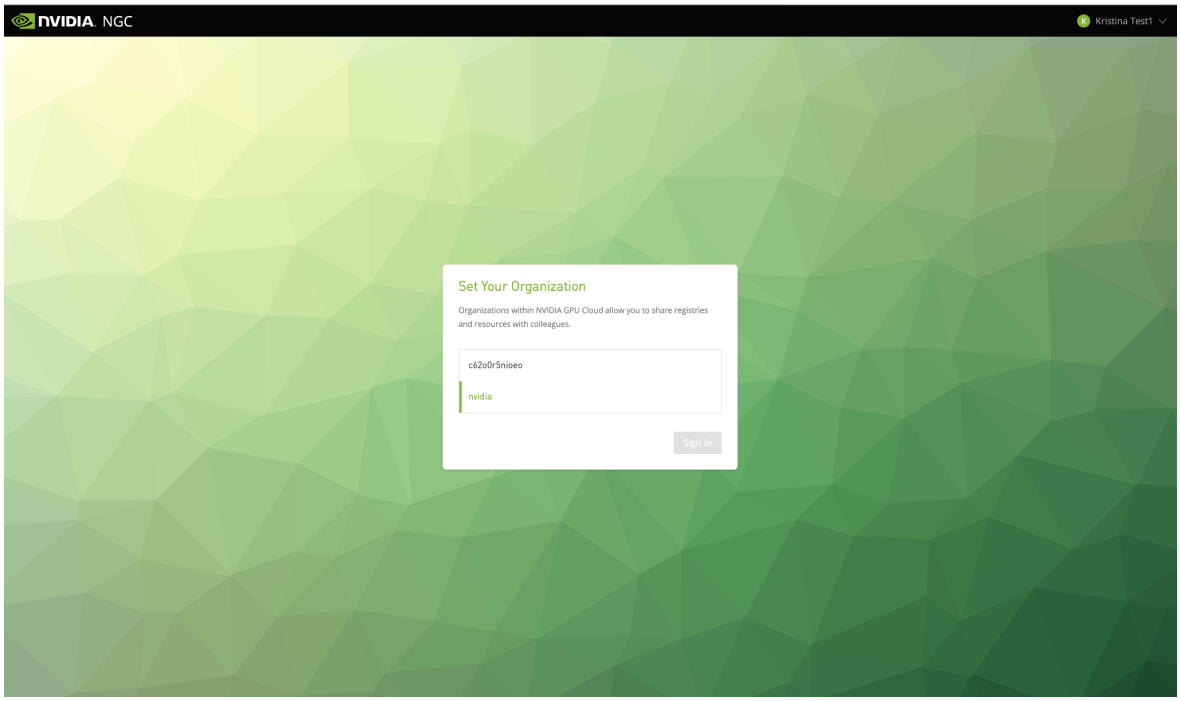
9. Review and **Accept** the NVIDIA GPU Cloud Terms of Use and Consent.



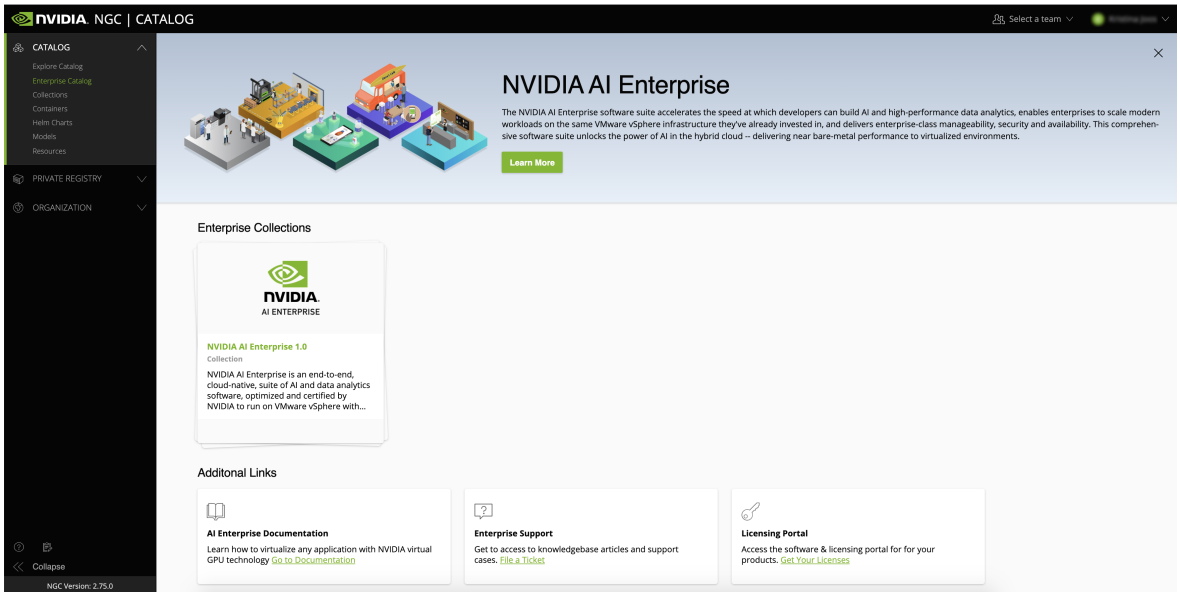
10. Review and **Accept** the NVIDIA AI Enterprise Terms of Use.



11. If asked, set your organization. The name of your organization was defined while setting up your Private Registry. Click **Sign In**.



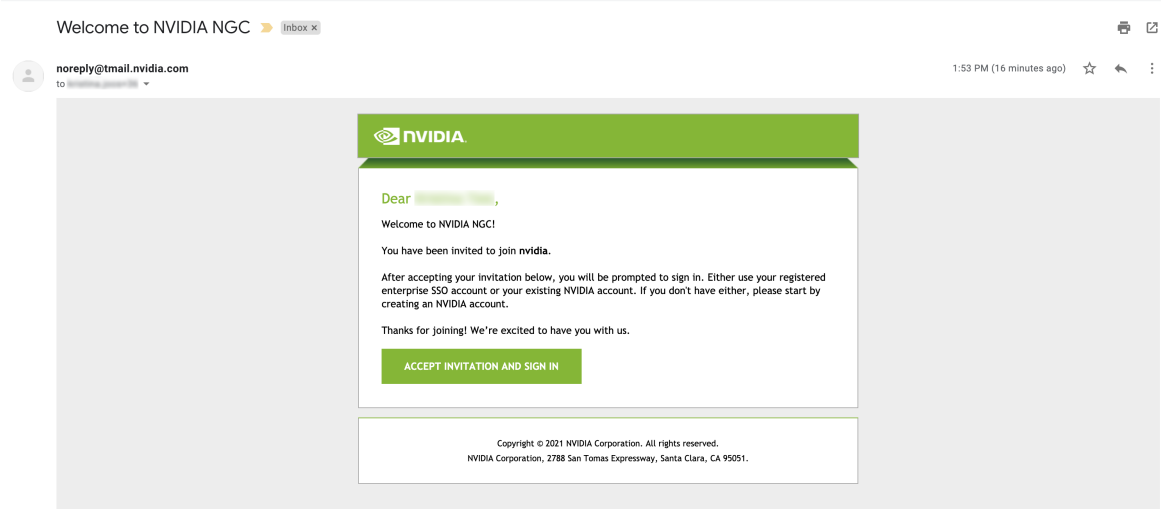
12. Welcome to the Enterprise Catalog.



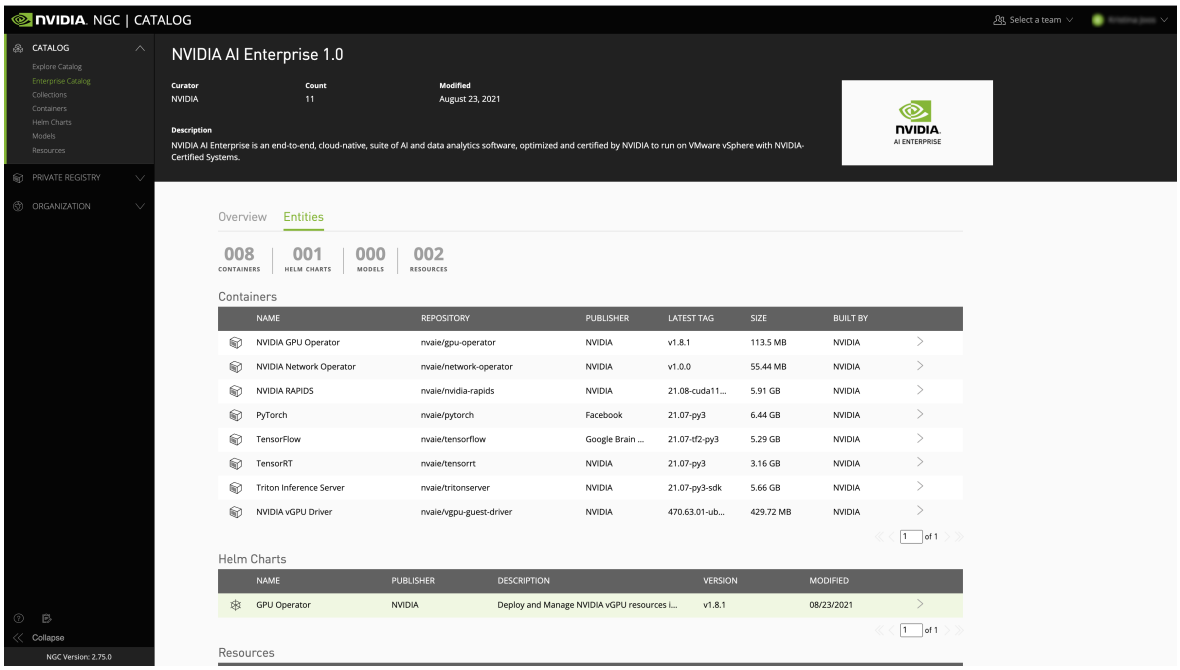
2.1.2. Downloading Software from the Enterprise Catalog

2.1.2.1. Accessing the NVIDIA AI Enterprise Collection

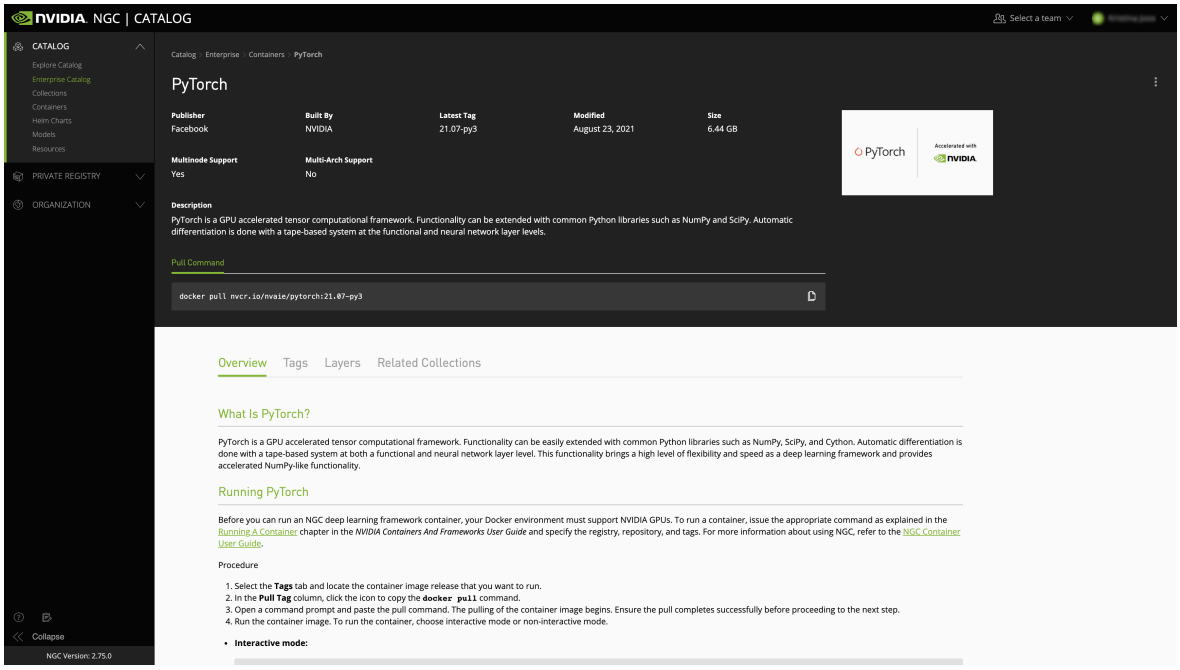
1. Go to https://ngc.nvidia.com/catalog/enterprise and, if prompted, log in. Click on the **NVIDIA AI Enterprise Collection**.



2. Click on the **Entities** tab to review all the software assets part of the NVIDIA AI Enterprise stack.



3. Click on the software asset you are interested in to learn more or download the software in the entities view.



2.1.2.2. Container Images

To pull AI and data science containers using Docker, follow these steps within the VM:

1. Generate your API key.
2. Access the Enterprise Catalog Container Registry.
   a). Log in to the NGC container registry.
   ```
   sudo docker login nvcr.io
   ```
   b). When prompted for your username, enter the text `$oauthtoken`.
   ```
   Username: $oauthtoken
   ```
   c). When prompted for your password, enter your NGC API key.
   ```
   Password: my-api-key
   ```
3. For each AI or data science application that you are interested in, load the container.
   ```
   sudo docker pull nvcr.io/nvaie/tensorflow:21.02-tf2-py3
   ```

2.1.2.3. Helm Charts

1. Go to the Enterprise Catalog.
2. Click on the NVIDIA AI Enterprise Collection.
3. Go to the Entities tab and select the Helm chart you are interested in.
4. Here is how you download a Helm chart from the Enterprise Catalog.

2.1.2.4. Resources

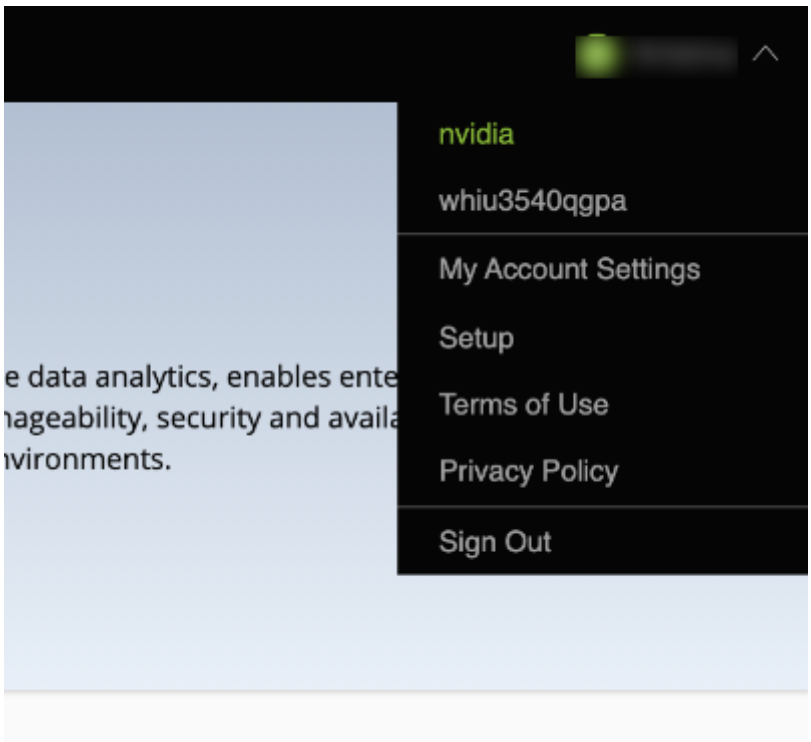
1. Go to the Enterprise Catalog.
2. Click on the NVIDIA AI Enterprise Collection.

3. Go to the Entities tab and select the Resource you are interested in. You can either download the Resource directly from the UI or use the displayed `wget` or CLI commands.

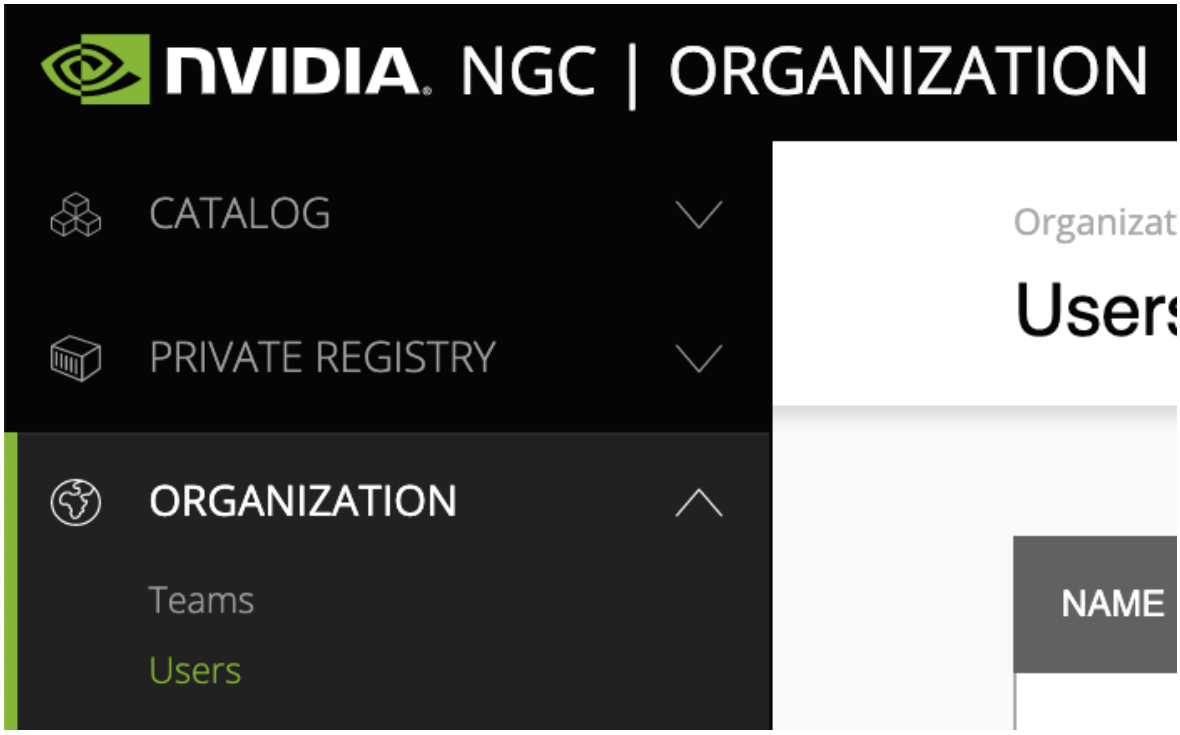
2.1.3. Adding Additional Users from Your Organization to the Enterprise Catalog (Admins Only)

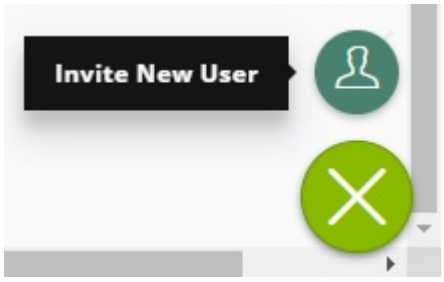
As an admin, you are responsible for giving members of your organization access to the Enterprise Catalog.

1. Make sure you are signed in.
2. Make sure to select your company's organization from the user menu on the top right.

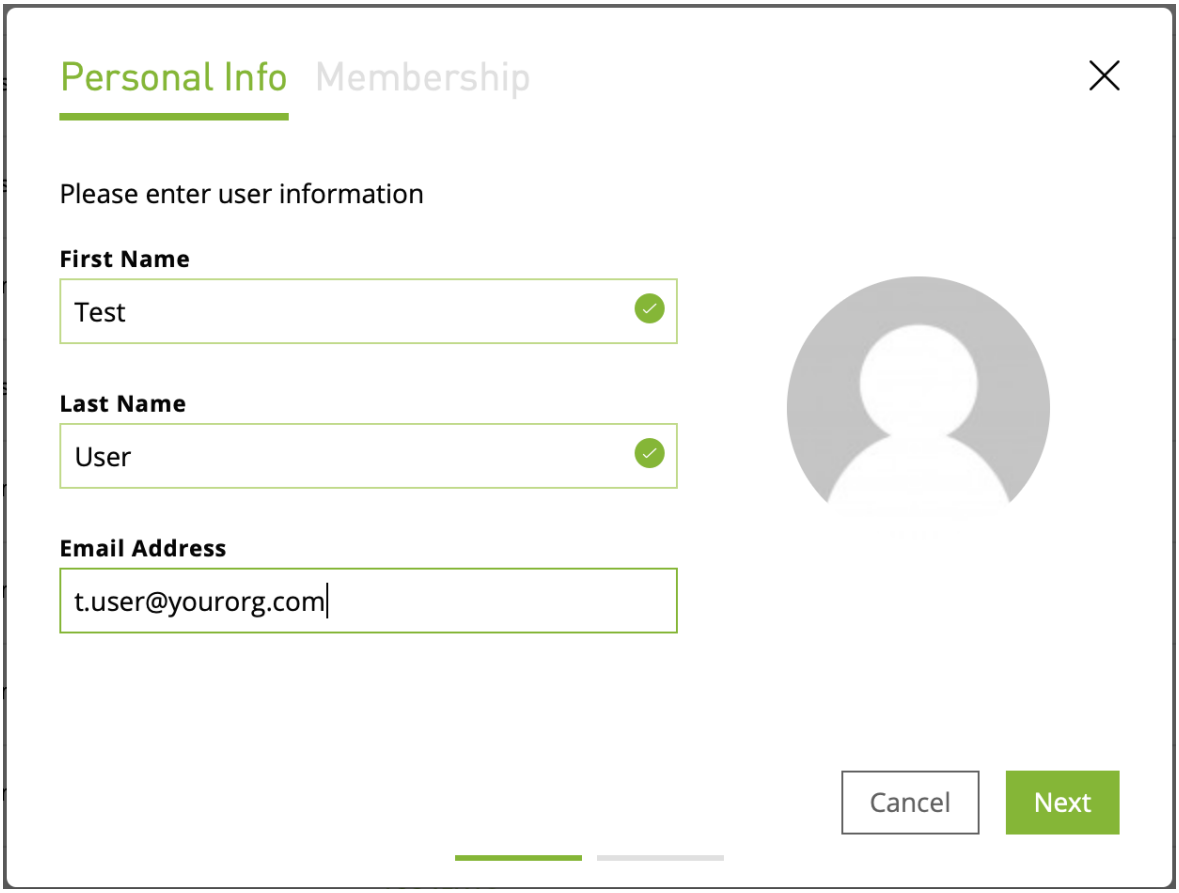


3. On the left side menu, select **Organization** and click on **Users**, then click the + icon at the bottom of the screen and then click the **Invite New User** icon.

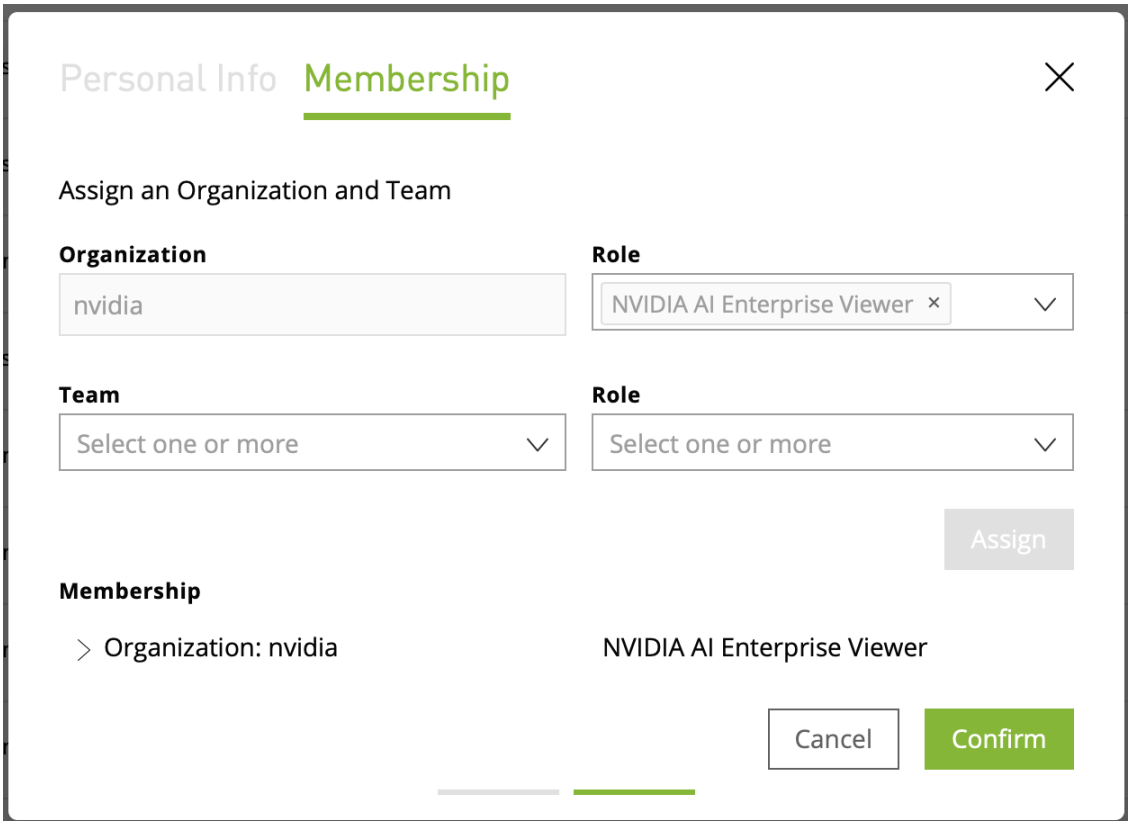




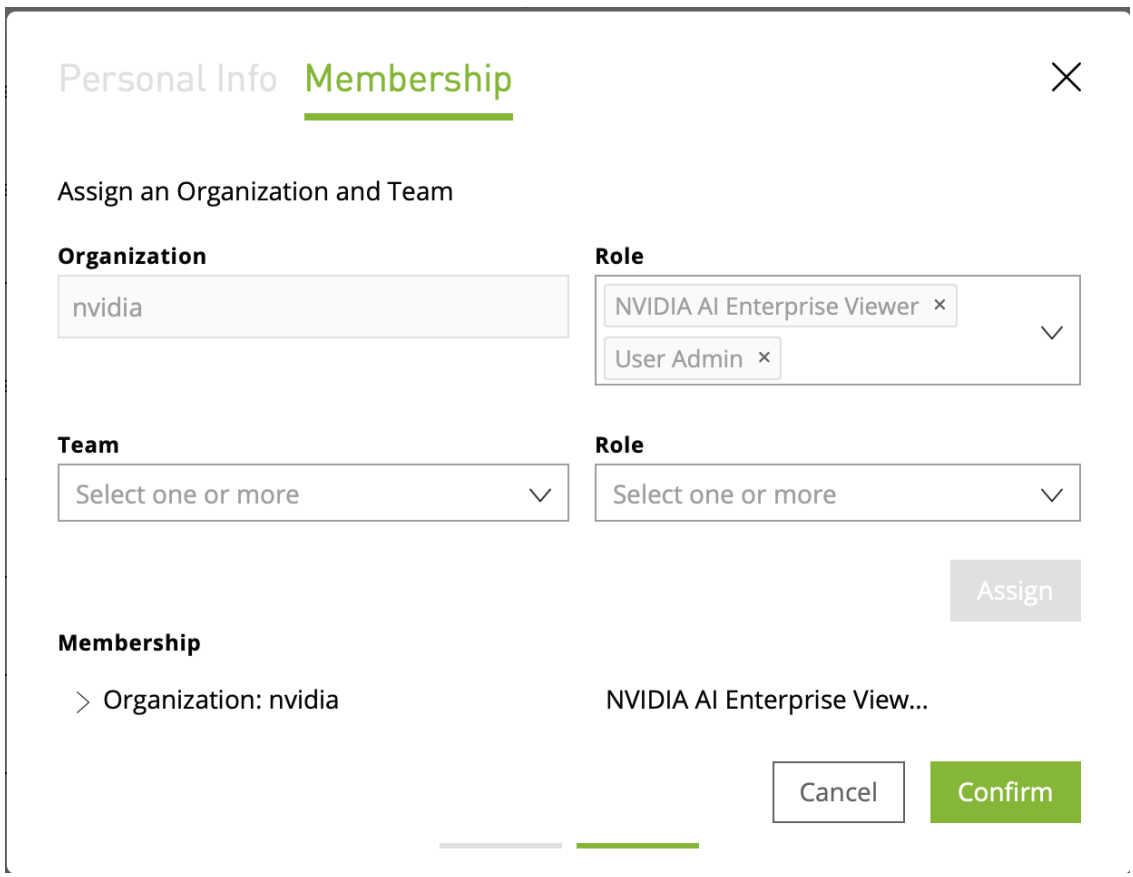
4. Provide the name and email address of the user you would like to add.



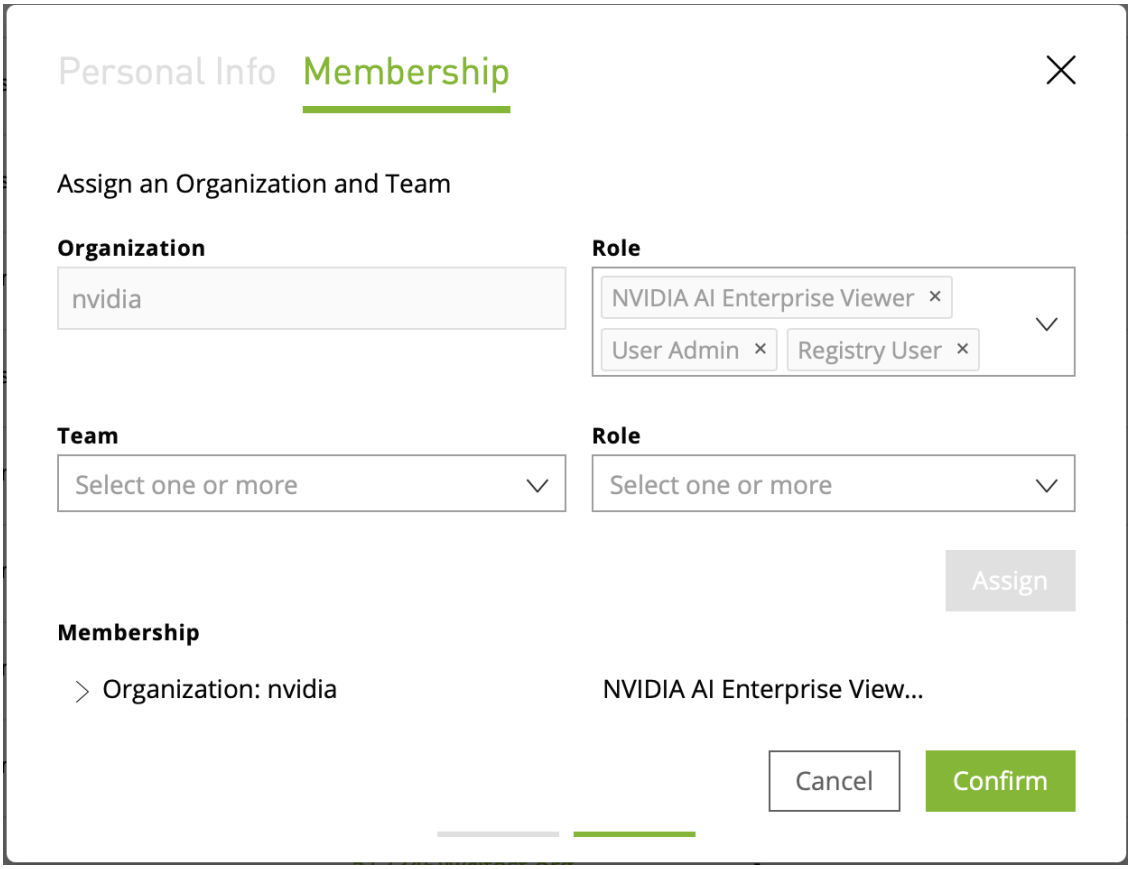
5. Provision user roles for the new user:
   a). To give the new user access to the entities in the Enterprise Catalog, provide them with the user role **NVIDIA AI Enterprise Viewer**.



b). To make them an admin that can add additional users to the Enterprise Catalog, provision the user roles: **NVIDIA AI Enterprise Viewer** and **User Admin**.



c). To give the user access to your organization's Private Registry, see Accessing Your NGC Private Registry. Provisioning access to the Enterprise Catalog and your organization's Private Registry can be done in one or two steps.



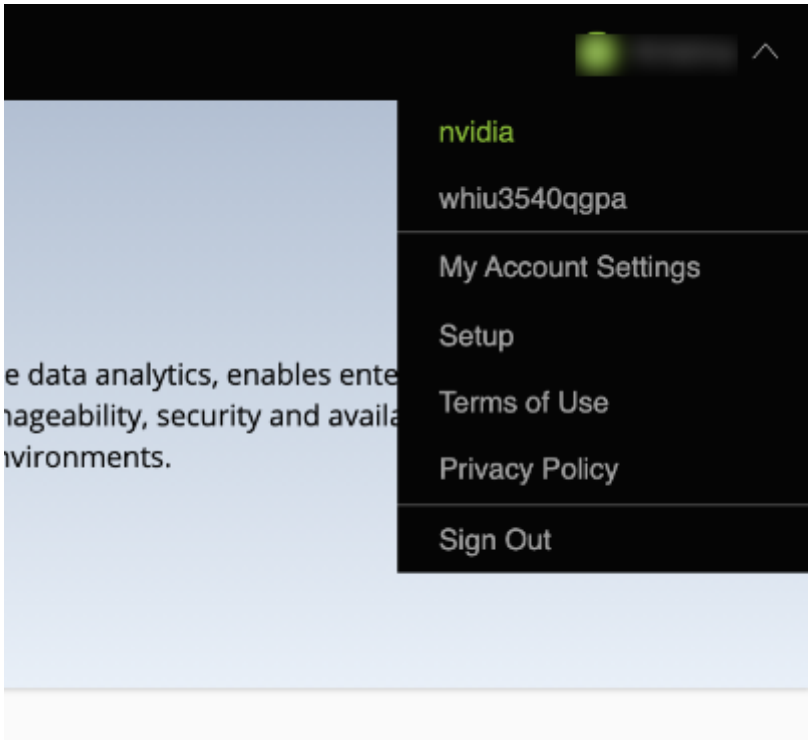
2.2. The NGC Private Registry

As an NVIDIA AI Enterprise user, you have exclusive access to your organization's own NGC Private Registry, which gives authorized users within your organization privileges to store your company's proprietary software and tools, including custom models, frameworks, and helm charts, in one location.

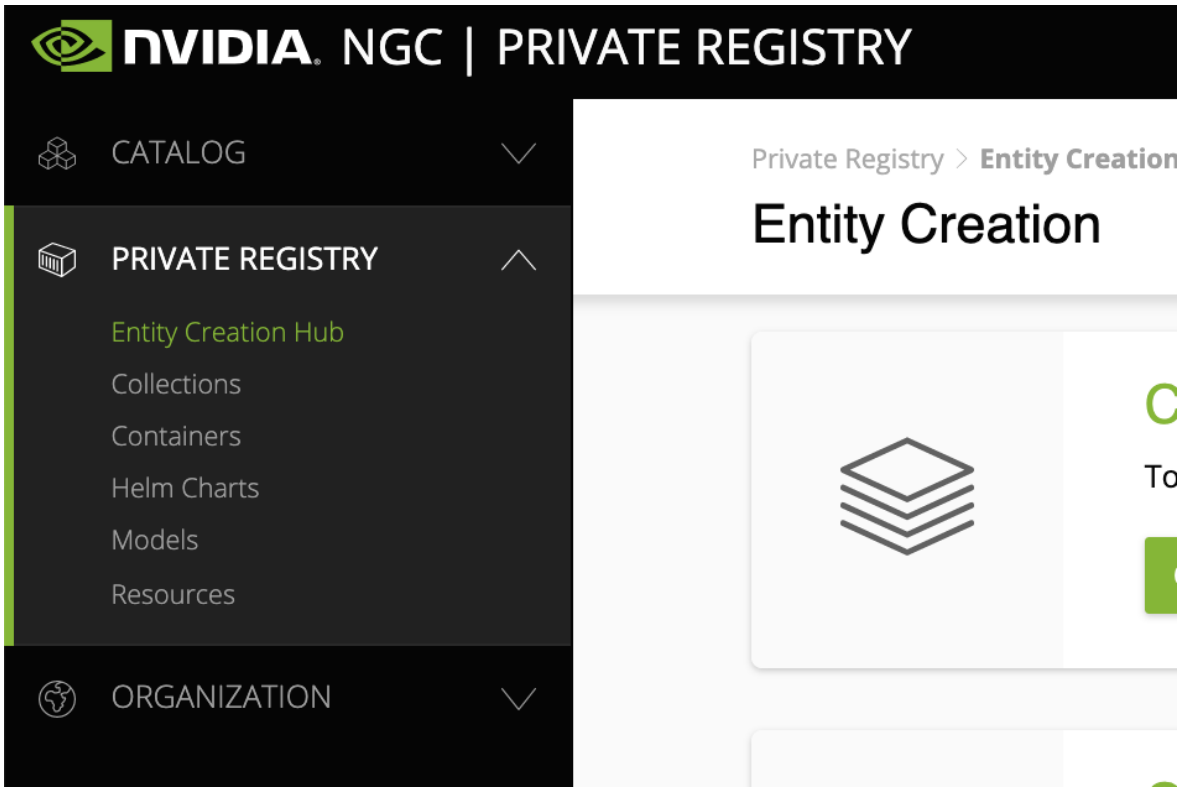
The complete NGC Private Registry user guide can be found here.

2.2.1. Accessing Your NGC Private Registry

1. To access your NGC Private Registry, sign in with your NGC Account.
2. In the top right corner, click your user account icon and select the **orgname**.



3. To view artifacts in your NGC Private Registry, select **Private Registry** in the left-hand menu.



4. You can access the content of the NGC Private Registry by selecting one of the entity types (Collections, Containers, Helm Charts, Models, Resources).
5. To upload entities to your NGC Private Registry, click on **Entity Creation Hub**.

2.2.2. Managing Teams and Users

As an admin, you can add users to your organization's NGC Private Registry and create teams within the NGC Private Registry.

Before adding users and teams, familiarize yourself with the following definitions of each role here.

2.2.2.1. Creating Teams

Creating teams allows users to share images within a team while keeping them invisible to other teams in the same organization. Only organization administrators can create teams.

Here is how you create a team.

2.2.2.2. Creating Users

As the organization administrator, you must create user accounts to allow others to use the NGC container registry within the organization.

Here is how you create a new user.

Chapter 3. Installing Your NVIDIA AI Enterprise License Server and License Files

The NVIDIA License System is used to serve a pool of floating licenses to licensed NVIDIA software products. The NVIDIA License System is configured with licenses obtained from the NVIDIA Licensing Portal.

Note: These instructions cover only the configuration of a Cloud License Service (CLS) instance or a standalone Delegated License Service (DLS) instance. The instructions for configuring a DLS instance assume that the VM that hosts the DLS instance has been assigned an IP address automatically. If you need complete instructions, are hosting a DLS instance on a VM that has not been assigned an IP address automatically, or require high availability for a DLS instance, refer to *NVIDIA License System User Guide*.

Note: These instructions cover only the configuration of a Cloud License Service (CLS) instance. If you need complete instructions or are using Delegated License Service (DLS) instances to serve licenses, refer to *NVIDIA License System User Guide*.

3.1. Introduction to NVIDIA Software Licensing

To activate licensed functionalities, a licensed client must obtain a software license when it is booted.

A client with a network connection obtains a license by leasing it from a NVIDIA License System service instance. The service instance serves the license to the client over the network from a pool of floating licenses obtained from the NVIDIA Licensing Portal. The license is returned to the service instance when the licensed client no longer requires the license.

NVIDIA License System supports the types of licensing for licensed clients:

- **Networked-licensing:** A client with a network connection obtains a license by leasing it from a NVIDIA License System service instance. The service instance serves the license to the client over the network from a pool of floating licenses obtained from the NVIDIA

Licensing Portal. The license is returned to the service instance when the licensed client no longer requires the license.

- **Node locked-licensing:** A client system without a network connection or on an air-gapped network can obtain a node-locked NVIDIA AI Enterprise license from a file installed locally on the client system.

> **Note:** Support for node-locked licensing was introduced in 15.0. It is **not** supported in earlier releases.

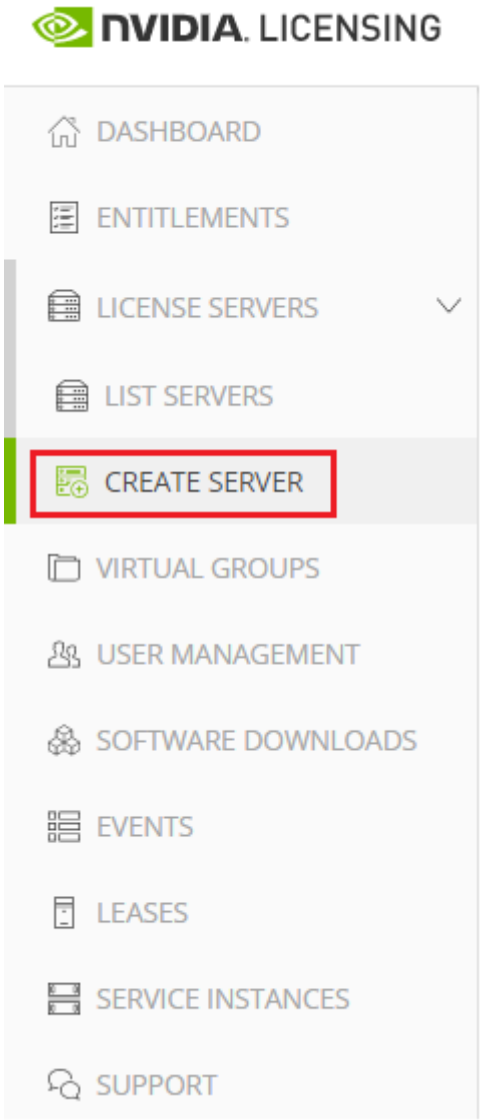
3.2. Creating a License Server on the NVIDIA Licensing Portal

To be able to allot licenses to an NVIDIA License System instance, you must create at least one license server on the NVIDIA Licensing Portal. Creating a license server defines the set of licenses to be allotted.

You can also create multiple servers on the NVIDIA Licensing Portal and distribute your licenses across them as necessary, for example to group licenses functionally or geographically.

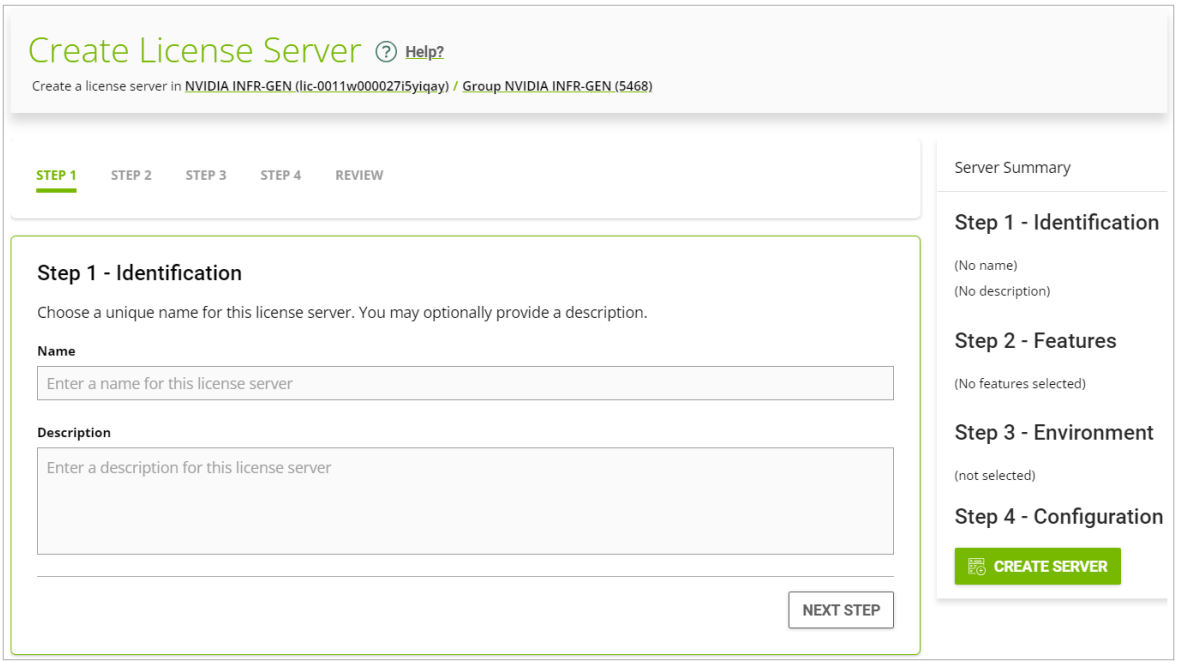
1. In the NVIDIA Licensing Portal, navigate to the organization or virtual group for which you want to create the license server.
   a). If you are not already logged in, log in to the NVIDIA Enterprise Application Hub and click **NVIDIA LICENSING PORTAL** to go to the NVIDIA Licensing Portal.
   b). **Optional:** If your assigned roles give you access to multiple virtual groups, click **View settings** at the top right of the page and in the **My Info** window that opens, select the virtual group from the **Virtual Group** drop-down list, and close the **My Info** window.

   If no license servers have been created for your organization or virtual group, the NVIDIA Licensing Portal dashboard displays a message asking if you want to create a license server.
2. In the left navigation pane of the NVIDIA Licensing Portal dashboard, expand **LICENSE SERVER** and click **CREATE SERVER**. The Create License Server wizard is started.



If you are adding a license server to an organization or virtual group for which a license server has already been created, click **CREATE SERVER**.

The **Create License Server** wizard opens.



3. On the Create License Server page of the wizard, step through the configuration requirements to provide the details of your license server.
   a). **Step 1 – Identification**: In the **Name** field, enter your choice of name for the license server and in the **Description** field, enter a text description of the license server.

   The description is required and will be displayed on the details page for the license server that you are creating.

   b). **Step 2 – Features**: Select one or more available features from your entitlements to allot to this license server.

   c). **Step 3 - Environment**: Select **Cloud (CLS)** or **On-Premises (DLS)** to install this license server.

   To make the selection after the license server has been created, select the **Deferred** option.

   d). **Step 4 – Configuration**: From the **Leasing mode** drop-down list, select one of the following leasing modes:

   Standard Networked Licensing
   Select this mode to simplify the management of licenses on a license server that supports networked licensing. In this mode, no additional configuration of the licenses on the server is required.

   Advanced Networked Licensing
   Select this mode if you require control over the management of licenses on a license server that supports networked licensing. This mode requires additional configuration to create license pools and fulfillment conditions on the server. For more information, refer to #unique_28 and #unique_29.

   Node-Locked Licensing
   Select this mode **only** if the license server will serve clients that cannot obtain a license from a remote license server over a network connection. In this mode,

the clients obtain a node-locked license from a file installed locally on the client system. For more details, refer to #unique_30.

> **CAUTION:** This mode requires additional work to create the license file to be installed locally and to return licenses when the client is shut down. If this mode is set, the mode of the license server **cannot** be changed.

e). Click **REVIEW SUMMARY** to review the configuration summary before creating the license server.

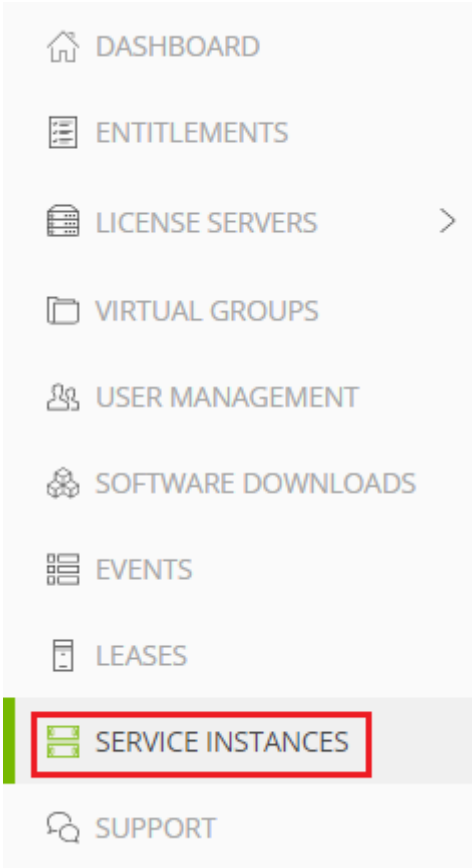
4. On the Create License Server page, from the **Step 4 – Configuration** menu, click the **CREATE SERVER** option to create this license server.

   Alternatively, you can click **CREATE SERVER** on the Server Summary page.

3.3. Creating a CLS Instance on the NVIDIA Licensing Portal

When you create a CLS instance, the instance is automatically registered with the NVIDIA Licensing Portal. This task is only necessary if you are not using the default CLS instance. Service instances belong to an organization. Therefore, this task requires the #unique_32 role.

1. If you are not already logged in, log in to the NVIDIA Enterprise Application Hub and click **NVIDIA LICENSING PORTAL** to go to the NVIDIA Licensing Portal.
2. In the left navigation pane of the NVIDIA Licensing Portal dashboard, click **SERVICE INSTANCES**.



3. On the Service Instances page, from the **Actions** menu, choose **Create cloud (CLS) instance**.

   The **Create cloud (CLS) instance** pop-up window opens.
4. Provide the details of your cloud service instance.
   a). In the **Name** field, enter your choice of name for the service instance.
   b). In the **Description** field, enter a text description of the service instance.

   This description is required and will be displayed on the **Service Instances** page when the entry for service instance that you are creating is expanding.
5. Click **CREATE CLS INSTANCE**.

After creating a CLS instance on the NVIDIA Licensing Portal, follow the instructions in Binding a License Server to a Service Instance.

3.4. Binding a License Server to a Service Instance

Binding a license server to a service instance ensures that licenses on the server are available only from that service instance. As a result, the licenses are available only to the licensed clients that are served by the service instance to which the license server is bound.

You can bind multiple license servers to the same CLS instance but only one license server to the same DLS instance. If you want to use a different license server than the license server that was originally bound to a DLS instance, free the license sever as explained in #unique_34.

This task is necessary only if you are not using the default CLS instance.

1. In the NVIDIA Licensing Portal, navigate to the organization or virtual group to which the **license server** belongs.
   a). If you are not already logged in, log in to the NVIDIA Enterprise Application Hub and click **NVIDIA LICENSING PORTAL** to go to the NVIDIA Licensing Portal.
   b). **Optional:** If your assigned roles give you access to multiple virtual groups, click **View settings** at the top right of the page and in the **My Info** window that opens, select the virtual group from the **Virtual Group** drop-down list, and close the **My Info** window.
2. In the left navigation pane of the NVIDIA Licensing Portal dashboard, expand **LICENSE SERVERS** and click **LIST SERVERS**.
3. In the list of license servers on the **License Servers** page that opens, from the **Actions** menu for the license server, choose **Bind**.
4. In the **Bind Service Instance** pop-up window that opens, select the service instance to which you want to bind the license server and click **BIND**.
   The **Bind Service Instance** pop-up window confirms that the license server has been bound to the service instance.

After a license server has been bound to a service instance, the license server is freed from the service instance when the service instance is deleted. You can also free a license sever as explained in #unique_34.

3.5. Installing a License Server on a CLS Instance

This task is necessary only if you are not using the default CLS instance.

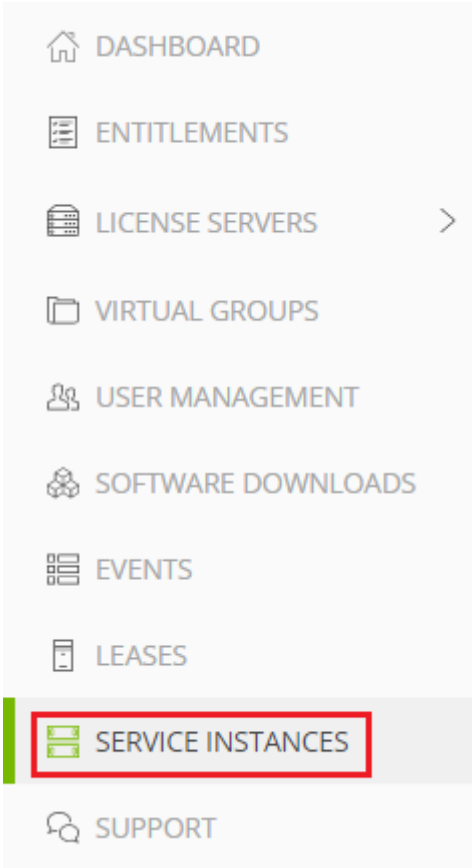
1. In the NVIDIA Licensing Portal, navigate to the organization or virtual group for which you want to install the license server.
   a). If you are not already logged in, log in to the NVIDIA Enterprise Application Hub and click **NVIDIA LICENSING PORTAL** to go to the NVIDIA Licensing Portal.

b). **Optional:** If your assigned roles give you access to multiple virtual groups, click **View settings** at the top right of the page and in the My Info window that opens, select the virtual group from the **Virtual Group** drop-down list, and close the **My Info** window.

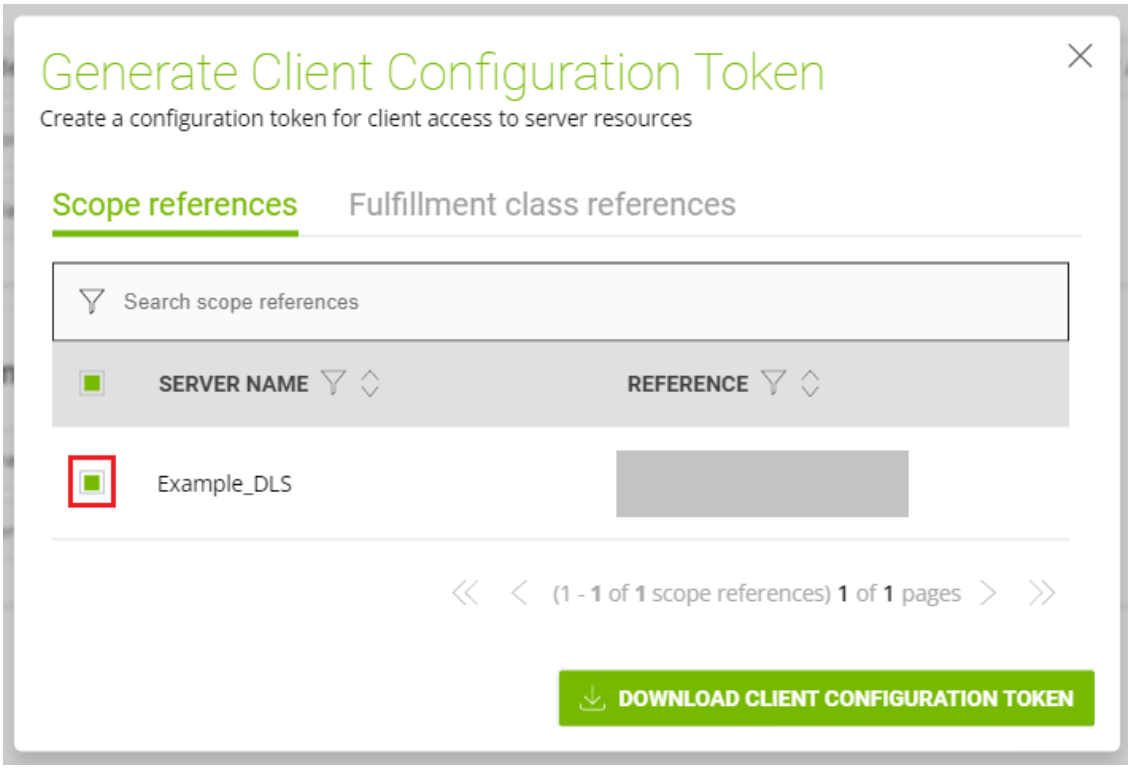
2. In the left navigation pane of the NVIDIA Licensing Portal dashboard, expand **LICENSE SERVER** and click **LIST SERVERS**.
3. In the list of license servers on the **License Servers** page that opens, click the name of the license server that you want to install.
4. In the **License Server Details** page that opens, from the **Actions** menu, choose **Install**.
5. In the **Install License Server** pop-up window that opens, click **INSTALL SERVER**.

3.6. Generating a Client Configuration Token for a CLS Instance

1. Log in to the NVIDIA Enterprise Application Hub and click **NVIDIA LICENSING PORTAL** to go to the NVIDIA Licensing Portal.
2. If your assigned roles give you access to multiple virtual groups, select the virtual group for which you are managing licenses from the list of virtual groups at the top right of the NVIDIA Licensing Portal dashboard.
3. In the left navigation pane, click **SERVICE INSTANCES**.



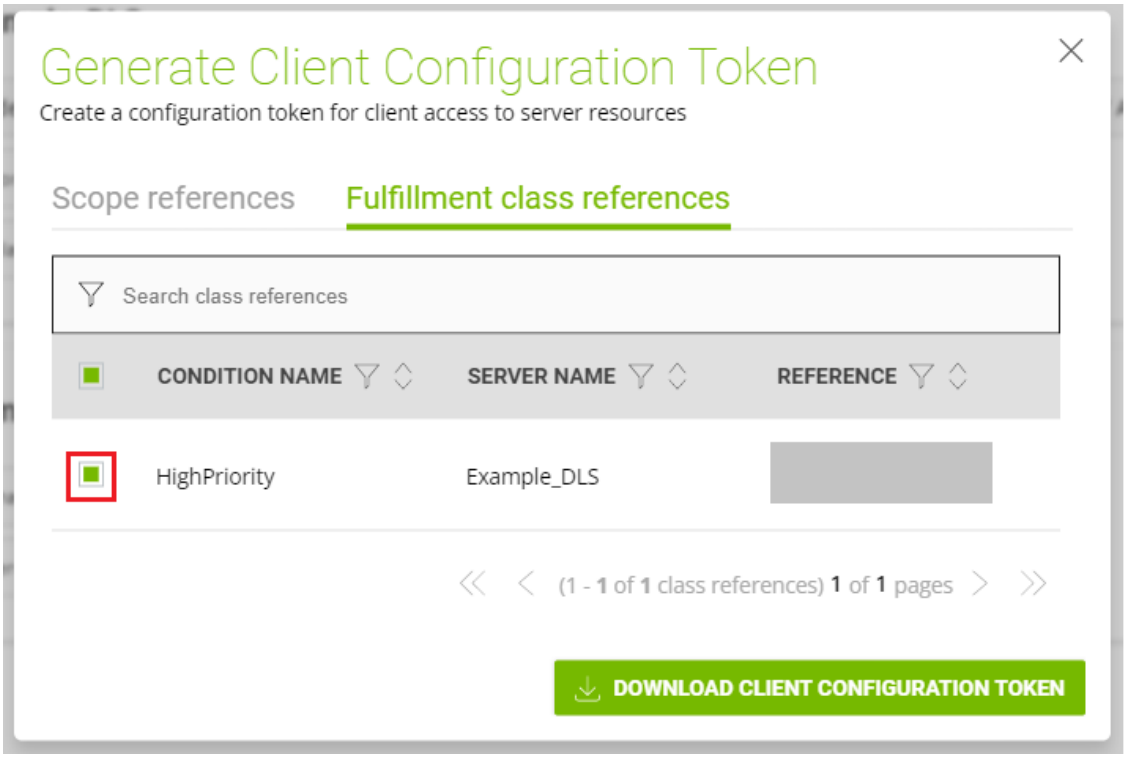
4. On the Service Instances page that opens, from the **Actions** menu for the CLS instance for which you want to generate a client configuration token, choose **Generate client configuration token**.
5. In the **Generate Client Configuration Token** pop-up window that opens, select the references that you want to include in the client configuration token.
   a). From the list of scope references, select the scope references that you want to include.



You must select **at least one** scope reference.

Each scope reference specifies the license server that will fulfil a license request.

b). **Optional:** Click the **Fulfillment class references** tab, and from the list of fulfillment class references, select the fulfillment class references that you want to include.



Including fulfillment class references is optional.

c). **Optional:** In the **Expiration** section, select an expiration date for the client configuration token. If you do not select a date, the default token expiration time is 12 years.

d). Click **DOWNLOAD CLIENT CONFIGURATION TOKEN**.

A file named `client_configuration_token_`*`mm-dd-yyyy-hh-mm-ss`*`.tok` is saved to your default downloads folder.

After creating a client configuration token from a service instance, copy the client configuration token to each licensed client that you want to use the combination of license servers and fulfillment conditions specified in the token. For more information, see Prerequisites for Configuring a Licensed Client of NVIDIA License System with a Networked License.

Chapter 4. Installing and Configuring NVIDIA vGPU Manager and the Guest Driver

Before installing and configuring NVIDIA vGPU Manager and the guest driver, ensure that a VM running a supported Windows guest OS is configured in your chosen hypervisor.

The factory settings of some supported GPU boards are incompatible with NVIDIA AI Enterprise. Before configuring NVIDIA AI Enterprise on these GPU boards, you must configure the boards to change these settings.

4.1. Switching the Mode of a GPU that Supports Multiple Display Modes

Some GPUs support displayless and display-enabled modes but must be used in NVIDIA AI Enterprise deployments in displayless mode.

The GPUs listed in the following table support multiple display modes. As shown in the table, some GPUs are supplied from the factory in displayless mode, but other GPUs are supplied in a display-enabled mode.

GPU	Mode as Supplied from the Factory
NVIDIA A40	Displayless
NVIDIA RTX A5000	Display enabled
NVIDIA RTX A6000	Display enabled

A GPU that is supplied from the factory in displayless mode, such as the NVIDIA A40 GPU, might be in a display-enabled mode if its mode has previously been changed.

To change the mode of a GPU that supports multiple display modes, use the `displaymodeselector` tool, which you can request from the NVIDIA Display Mode Selector Tool page on the NVIDIA Developer website.

Note:

Only the following GPUs support the `displaymodeselector` tool:

- NVIDIA A40
- NVIDIA RTX A5000
- NVIDIA RTX A6000

Other GPUs that support NVIDIA AI Enterprise do not support the `displaymodeselector` tool and, unless otherwise stated, do not require display mode switching.

4.2. Installing the NVIDIA Virtual GPU Manager on VMware vSphere

For all supported VMware vSphere releases, the NVIDIA Virtual GPU Manager package is distributed as a software component in a ZIP archive. For supported releases **before** VMware vSphere 7.0, the NVIDIA Virtual GPU Manager package is also distributed as a vSphere Installation Bundle (VIB) file.
Before you begin, ensure that the following prerequisites are met:

- The ZIP archive that contains NVIDIA AI Enterprise has been downloaded from the NVIDIA Licensing Portal.
- The NVIDIA Virtual GPU Manager package has been extracted from the downloaded ZIP archive.

1. Copy the NVIDIA Virtual GPU Manager package file to the ESXi host.
2. Put the ESXi host into maintenance mode.

   ```
   $ esxcli system maintenanceMode set --enable true
   ```
3. Run the `esxcli` command to install the NVIDIA Virtual GPU Manager from the package file.

   ```
   $ esxcli software vib install -d /vmfs/volumes/datastore/software-component.zip
   ```

   datastore
   The name of the VMFS datastore to which you copied the software component.

   software-component
   The name of the file that contains the NVIDIA Virtual GPU Manager package in the form of a software component. Ensure that you specify the file that was extracted from the downloaded ZIP archive. For example, for VMware vSphere 7.0.2, *software-component* is **`NVD.NVIDIA_bootbank_NVIDIA-VMware_470.141.05-1OEM.702.0.0.8169922-offline_bundle-build-number`**.

   - For a software component, run the following command:

     ```
     $ esxcli software vib install -d /vmfs/volumes/datastore/software-component.zip
     ```

     datastore
     The name of the VMFS datastore to which you copied the software component.

     software-component
     The name of the file that contains the NVIDIA Virtual GPU Manager package in the form of a software component. Ensure that you specify the file that was extracted from the downloaded ZIP archive. For example, for VMware vSphere 7.0.2, *software-component* is **`NVD.NVIDIA_bootbank_NVIDIA-VMware_470.141.05-1OEM.702.0.0.8169922-offline_bundle-build-number`**.

- For a VIB file, run the following command:

```
$ esxcli software vib install -v directory/NVIDIA**.vib
```

directory

The absolute path to the directory to which you copied the VIB file. You must specify the absolute path even if the VIB file is in the current working directory.

4. Exit maintenance mode.

```
$ esxcli system maintenanceMode set --enable false
```

5. Reboot the ESXi host.

```
$ reboot
```

6. Verify that the NVIDIA kernel driver can successfully communicate with the physical GPUs in your system by running the `nvidia-smi` command without any options.

```
$ nvidia-smi
```

If successful, the `nvidia-smi` command lists all the GPUs in your system.

4.3. Disabling and Enabling ECC Memory

Some GPUs that support NVIDIA AI Enterprise support error correcting code (ECC) memory with NVIDIA vGPU. ECC memory improves data integrity by detecting and handling double-bit errors. However, not all GPUs, vGPU types, and hypervisor software versions support ECC memory with NVIDIA vGPU.

On GPUs that support ECC memory with NVIDIA vGPU, ECC memory is supported with C-series and Q-series vGPUs, but not with A-series and B-series vGPUs. Although A-series and B-series vGPUs start on physical GPUs on which ECC memory is enabled, enabling ECC with vGPUs that do not support it might incur some costs.

On physical GPUs that do not have HBM2 memory, the amount of frame buffer that is usable by vGPUs is reduced. All types of vGPU are affected, not just vGPUs that support ECC memory.

The effects of enabling ECC memory on a physical GPU are as follows:

- ECC memory is exposed as a feature on all supported vGPUs on the physical GPU.
- In VMs that support ECC memory, ECC memory is enabled, with the option to disable ECC in the VM.
- ECC memory can be enabled or disabled for individual VMs. Enabling or disabling ECC memory in a VM does not affect the amount of frame buffer that is usable by vGPUs.

GPUs based on the Pascal GPU architecture and later GPU architectures support ECC memory with NVIDIA vGPU. To determine whether ECC memory is enabled for a GPU, run **`nvidia-smi -q`** for the GPU.

Tesla M60 and M6 GPUs support ECC memory when used without GPU virtualization, but NVIDIA vGPU does not support ECC memory with these GPUs. In graphics mode, these GPUs are supplied with ECC memory disabled by default.

Some hypervisor software versions do not support ECC memory with NVIDIA vGPU.

If you are using a hypervisor software version or GPU that does not support ECC memory with NVIDIA vGPU and ECC memory is enabled, NVIDIA vGPU fails to start. In this situation, you must ensure that ECC memory is disabled on all GPUs if you are using NVIDIA vGPU.

4.3.1. Disabling ECC Memory

If ECC memory is unsuitable for your workloads but is enabled on your GPUs, disable it. You must also ensure that ECC memory is disabled on all GPUs if you are using NVIDIA vGPU with a hypervisor software version or a GPU that does not support ECC memory with NVIDIA vGPU. If your hypervisor software version or GPU does not support ECC memory and ECC memory is enabled, NVIDIA vGPU fails to start.

Where to perform this task depends on whether you are changing ECC memory settings for a physical GPU or a vGPU.

- For a physical GPU, perform this task from the hypervisor host.
- For a vGPU, perform this task from the VM to which the vGPU is assigned.

Note: ECC memory must be enabled on the physical GPU on which the vGPUs reside.

Before you begin, ensure that NVIDIA Virtual GPU Manager is installed on your hypervisor. If you are changing ECC memory settings for a vGPU, also ensure that the NVIDIA AI Enterprise graphics driver is installed in the VM to which the vGPU is assigned.

1. Use `nvidia-smi` to list the status of all physical GPUs or vGPUs, and check for ECC noted as enabled.

```
# nvidia-smi -q

==============NVSMI LOG==============

Timestamp                           : Mon Oct 17 18:36:45 2022
Driver Version                      : 470.141.05

Attached GPUs                       : 1
GPU 0000:02:00.0

[...]

    Ecc Mode
        Current                     : Enabled
        Pending                     : Enabled

[...]
```

2. Change the ECC status to off for each GPU for which ECC is enabled.
   - If you want to change the ECC status to off for all GPUs on your host machine or vGPUs assigned to the VM, run this command:

     ```
     # nvidia-smi -e 0
     ```
   - If you want to change the ECC status to off for a specific GPU or vGPU, run this command:

     ```
     # nvidia-smi -i id -e 0
     ```

     id is the index of the GPU or vGPU as reported by `nvidia-smi`.

     This example disables ECC for the GPU with index `0000:02:00.0`.

     ```
     # nvidia-smi -i 0000:02:00.0 -e 0
     ```
3. Reboot the host or restart the VM.

4. Confirm that ECC is now disabled for the GPU or vGPU.

```
# nvidia–smi –q

==============NVSMI LOG==============

Timestamp                           : Mon Oct 17 18:37:53 2022
Driver Version                      : 470.141.05

Attached GPUs                       : 1
GPU 0000:02:00.0
[...]

    Ecc Mode
        Current                     : Disabled
        Pending                     : Disabled

[...]
```

If you later need to enable ECC on your GPUs or vGPUs, follow the instructions in Enabling ECC Memory.

4.3.2. Enabling ECC Memory

If ECC memory is suitable for your workloads and is supported by your hypervisor software and GPUs, but is disabled on your GPUs or vGPUs, enable it.

Where to perform this task depends on whether you are changing ECC memory settings for a physical GPU or a vGPU.

- For a physical GPU, perform this task from the hypervisor host.
- For a vGPU, perform this task from the VM to which the vGPU is assigned.

Note: ECC memory must be enabled on the physical GPU on which the vGPUs reside.

Before you begin, ensure that NVIDIA Virtual GPU Manager is installed on your hypervisor. If you are changing ECC memory settings for a vGPU, also ensure that the NVIDIA AI Enterprise graphics driver is installed in the VM to which the vGPU is assigned.

1. Use `nvidia-smi` to list the status of all physical GPUs or vGPUs, and check for ECC noted as disabled.

```
# nvidia-smi -q

==============NVSMI LOG==============

Timestamp                           : Mon Oct 17 18:36:45 2022
Driver Version                      : 470.141.05

Attached GPUs                       : 1
GPU 0000:02:00.0

[...]

    Ecc Mode
        Current                     : Disabled
        Pending                     : Disabled

[...]
```

2. Change the ECC status to on for each GPU or vGPU for which ECC is enabled.
   - If you want to change the ECC status to on for all GPUs on your host machine or vGPUs assigned to the VM, run this command:
     ```
     # nvidia-smi -e 1
     ```
   - If you want to change the ECC status to on for a specific GPU or vGPU, run this command:
     ```
     # nvidia-smi -i id -e 1
     ```
     id is the index of the GPU or vGPU as reported by `nvidia-smi`.

     This example enables ECC for the GPU with index `0000:02:00.0`.
     ```
     # nvidia-smi -i 0000:02:00.0 -e 1
     ```
3. Reboot the host or restart the VM.
4. Confirm that ECC is now enabled for the GPU or vGPU.
   ```
   # nvidia–smi –q

   ==============NVSMI LOG==============

   Timestamp                           : Mon Oct 17 18:37:53 2022
   Driver Version                      : 470.141.05

   Attached GPUs                       : 1
   GPU 0000:02:00.0
   [...]

       Ecc Mode
           Current                     : Enabled
           Pending                     : Enabled

   [...]
   ```

If you later need to disable ECC on your GPUs or vGPUs, follow the instructions in Disabling ECC Memory.

4.4. Changing the Default Graphics Type in VMware vSphere 6.5 and Later

The vGPU Manager VIBs for VMware vSphere 6.5 and later provide vSGA and vGPU functionality in a single VIB. After this VIB is installed, the default graphics type is Shared, which provides vSGA functionality. To enable vGPU support for VMs in VMware vSphere 6.5, you must change the default graphics type to Shared Direct. If you do not change the default graphics type, VMs to which a vGPU is assigned fail to start and the following error message is displayed:

```
The amount of graphics resource available in the parent resource pool is
 insufficient for the operation.
```

Note:

If you are using a supported version of VMware vSphere earlier than 6.5, or are configuring a VM to use vSGA, omit this task.

Change the default graphics type **before** configuring vGPU. Output from the VM console in the VMware vSphere Web Client is not available for VMs that are running vGPU.

Before changing the default graphics type, ensure that the ESXi host is running and that all VMs on the host are powered off.

1. Log in to vCenter Server by using the vSphere Web Client.
2. In the navigation tree, select your ESXi host and click the **Configure** tab.
3. From the menu, choose **Graphics** and then click the **Host Graphics** tab.
4. On the **Host Graphics** tab, click **Edit**.

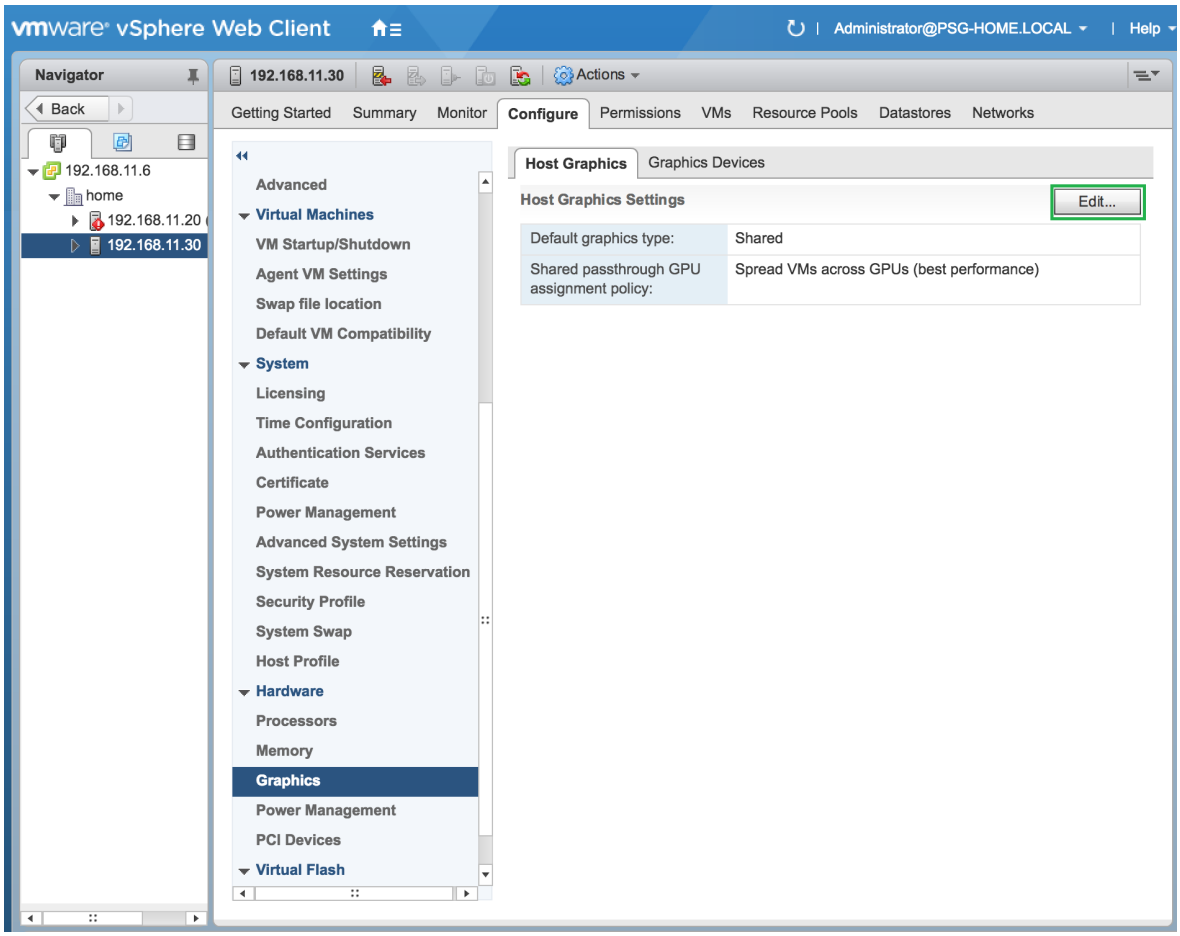
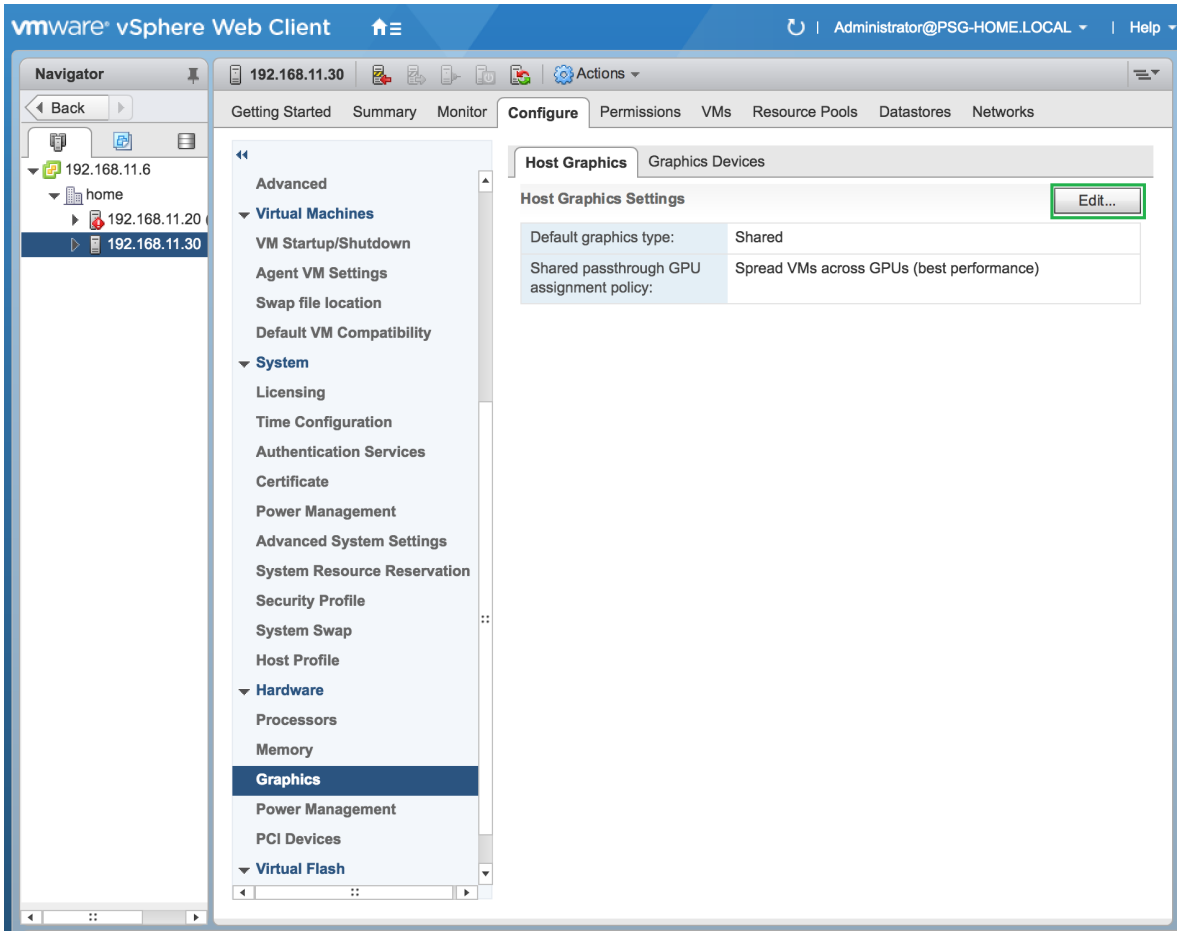


Figure 1. Shared default graphics type



5. In the **Edit Host Graphics Settings** dialog box that opens, select **Shared Direct** and click **OK**.

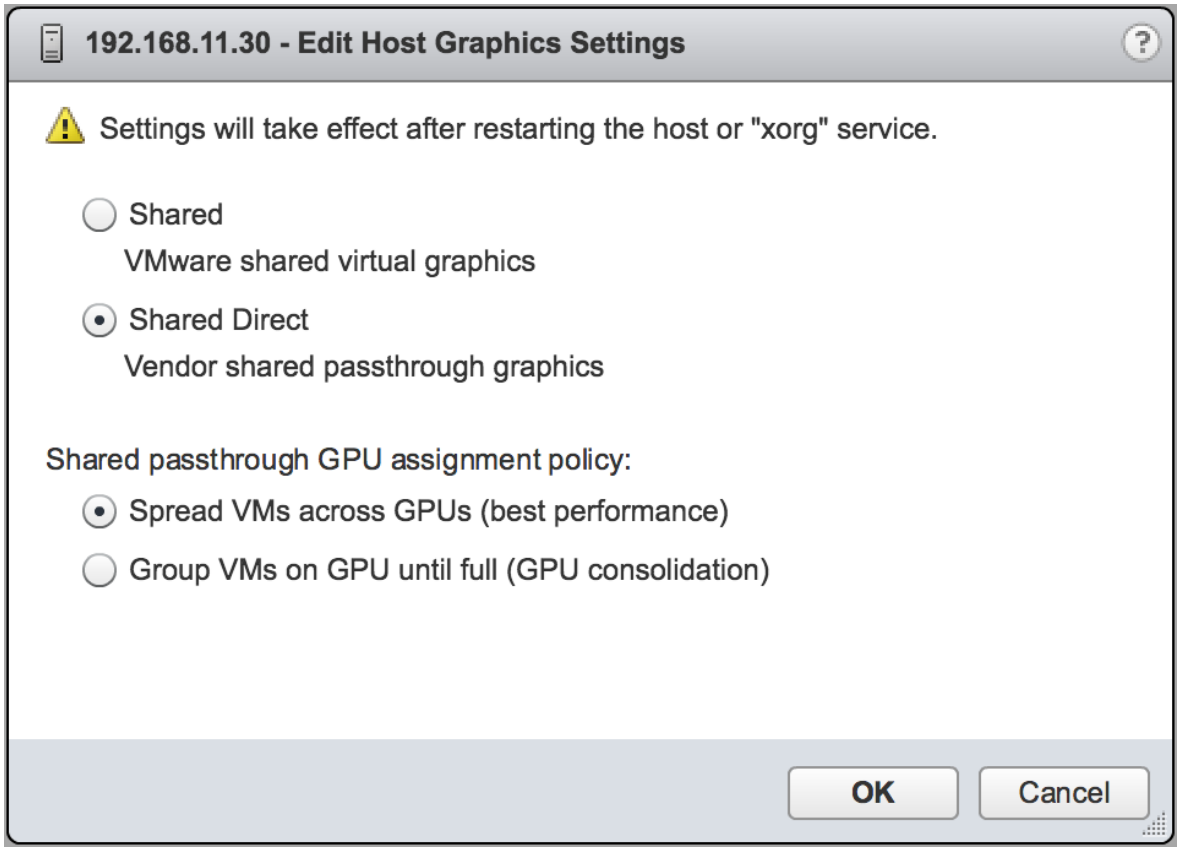
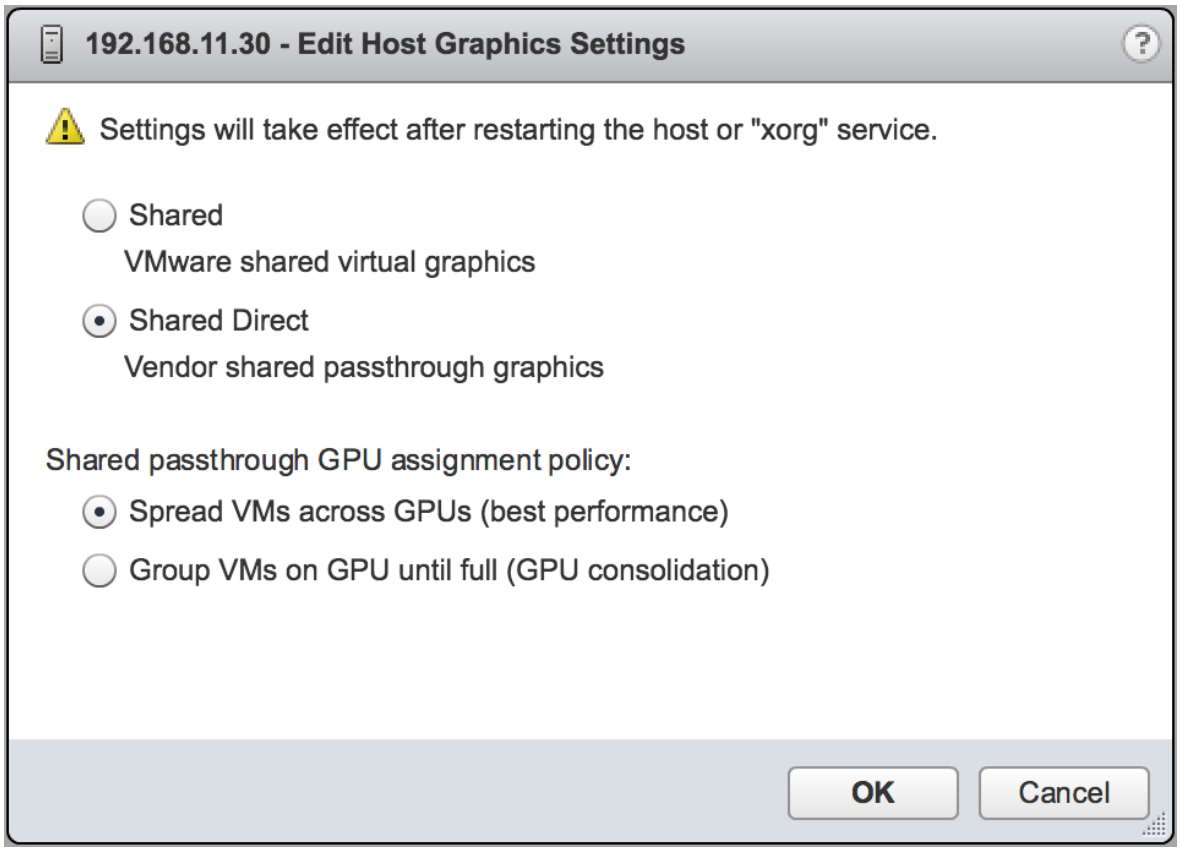

192.168.11.30 - Edit Host Graphics Settings
Settings will take effect after restarting the host or "xorg" service.
Shared
VMware shared virtual graphics
Shared Direct
Vendor shared passthrough graphics
Shared passthrough GPU assignment policy:
Spread VMs across GPUs (best performance)
Group VMs on GPU until full (GPU consolidation)
OK
Cancel

Figure 2. Host graphics settings for vGPU



Note: In this dialog box, you can also change the allocation scheme for vGPU-enabled VMs. For more information, see #unique_45.

After you click OK, the default graphics type changes to Shared Direct.

6. Click the **Graphics Devices** tab to verify the configured type of each physical GPU on which you want to configure vGPU.

   The configured type of each physical GPU must be Shared Direct. For any physical GPU for which the configured type is Shared, change the configured type as follows:

   a). On the **Graphics Devices** tab, select the physical GPU and click the **Edit icon**.

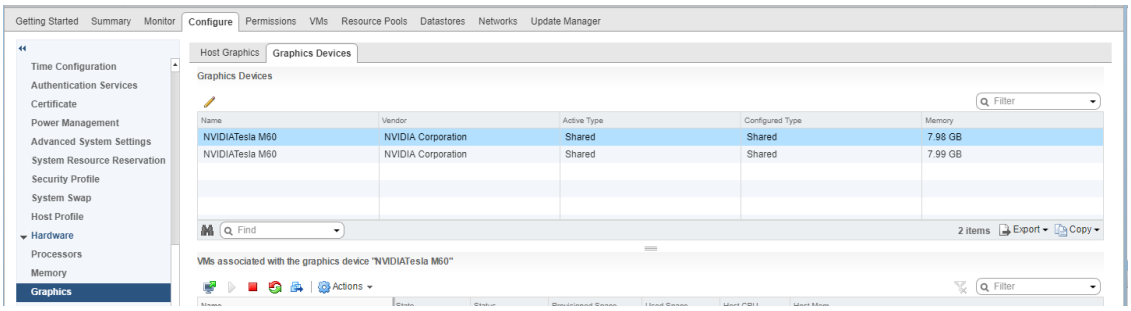
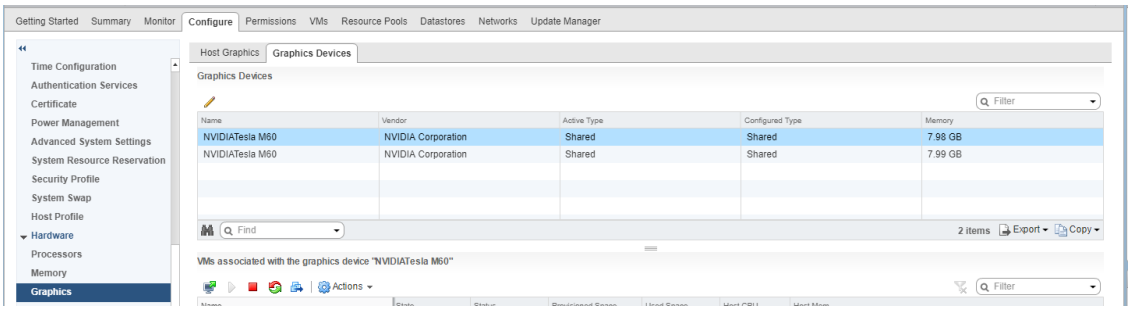


Figure 3. Shared graphics type



b). In the **Edit Graphics Device Settings** dialog box that opens, select **Shared Direct** and click **OK**.

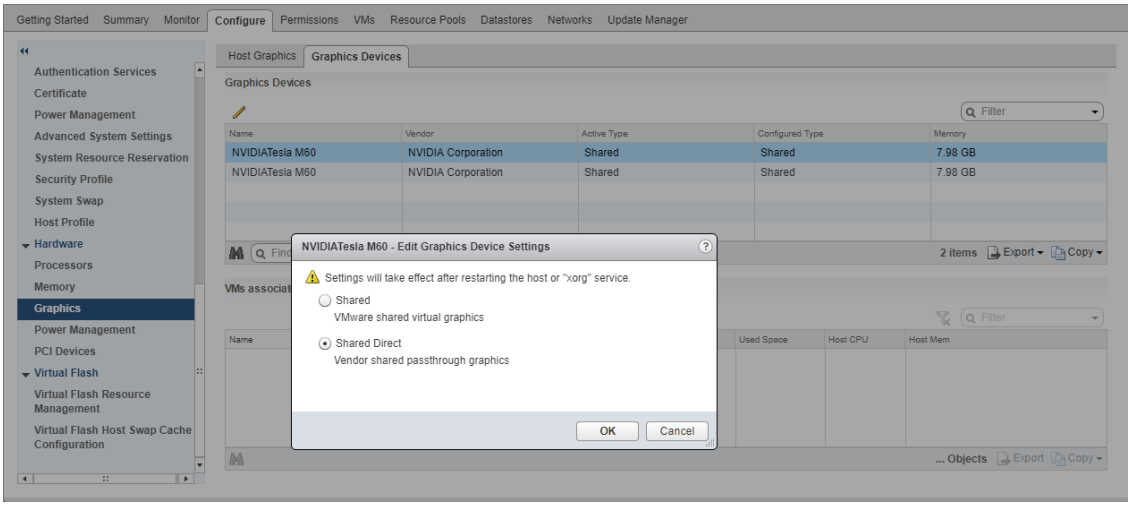
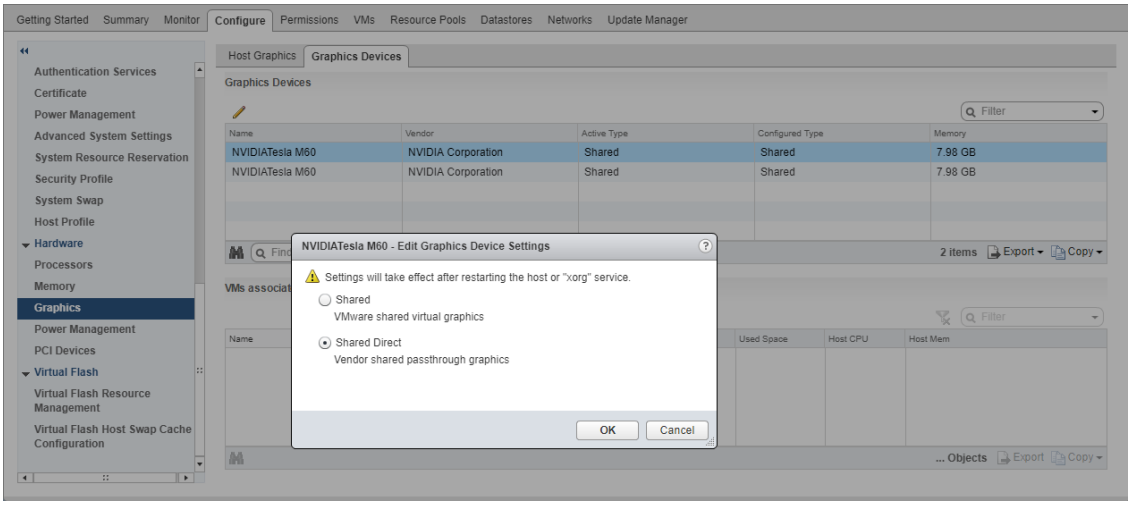


Figure 4. Graphics device settings for a physical GPU



7. Restart the ESXi host **or** stop and restart the Xorg service if necessary and `nv-hostengine` on the ESXi host.

   To stop and restart the Xorg service and `nv-hostengine`, perform these steps:

   a). **VMware vSphere releases before 7.0 Update 1 only:** Stop the Xorg service.

   As of VMware vSphere 7.0 Update 1, the Xorg service is no longer required for graphics devices in NVIDIA vGPU mode.

   b). Stop `nv-hostengine`.

   ```
   [root@esxi:~] nv-hostengine -t
   ```

   c). Wait for 1 second to allow `nv-hostengine` to stop.

   d). Start `nv-hostengine`.

   ```
   [root@esxi:~] nv-hostengine -d
   ```

   e). **VMware vSphere releases before 7.0 Update 1 only:** Start the Xorg service.

   As of VMware vSphere 7.0 Update 1, the Xorg service is no longer required for graphics devices in NVIDIA vGPU mode.

   ```
   [root@esxi:~] /etc/init.d/xorg start
   ```

8. In the **Graphics Devices** tab of the VMware vCenter Web UI, confirm that the active type and the configured type of each physical GPU are Shared Direct.

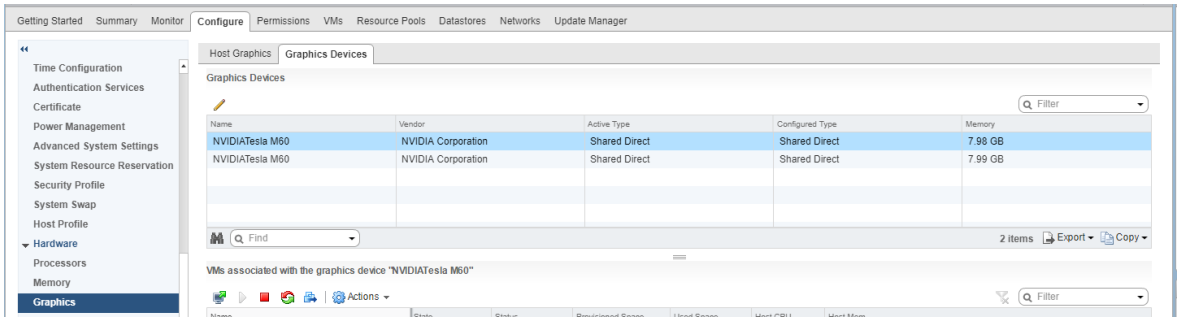
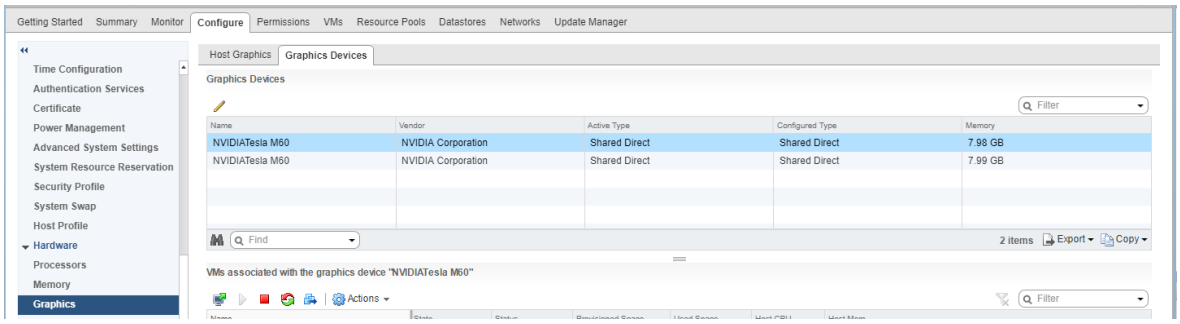


Figure 5. Shared direct graphics type



After changing the default graphics type, configure vGPU as explained in Configuring a vSphere VM with NVIDIA vGPU.

See also the following topics in the VMware vSphere documentation:

- Log in to vCenter Server by Using the vSphere Web Client
- Configuring Host Graphics

4.5. Configuring a vSphere VM with NVIDIA vGPU

To support applications and workloads that are compute or graphics intensive, you can add multiple vGPUs to a single VM.

For details about which VMware vSphere versions and NVIDIA vGPUs support the assignment of multiple vGPUs to a VM, see *NVIDIA AI Enterprise Release Notes*.

If you upgraded to VMware vSphere 6.7 Update 3 from an earlier version and are using VMs that were created with that version, change the VM compatibility to **vSphere 6.7 Update 2 and later**. For details, see Virtual Machine Compatibility in the VMware documentation.

If you are adding multiple vGPUs to a single VM, perform this task for each vGPU that you want to add to the VM.

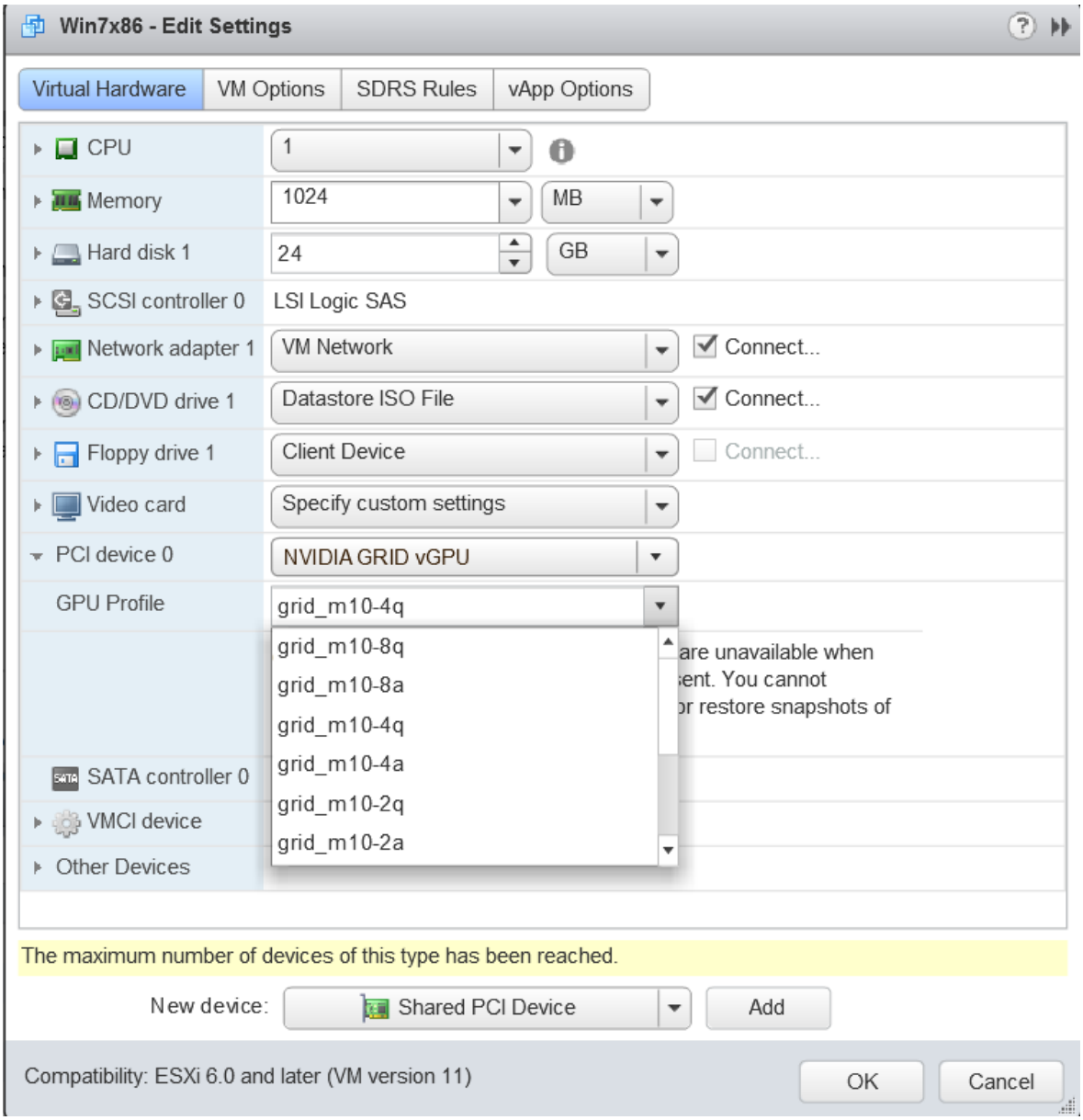
CAUTION: Output from the VM console in the VMware vSphere Web Client is not available for VMs that are running vGPU. Make sure that you have installed an alternate means of accessing the VM (such as VMware Horizon or a VNC server) before you configure vGPU.

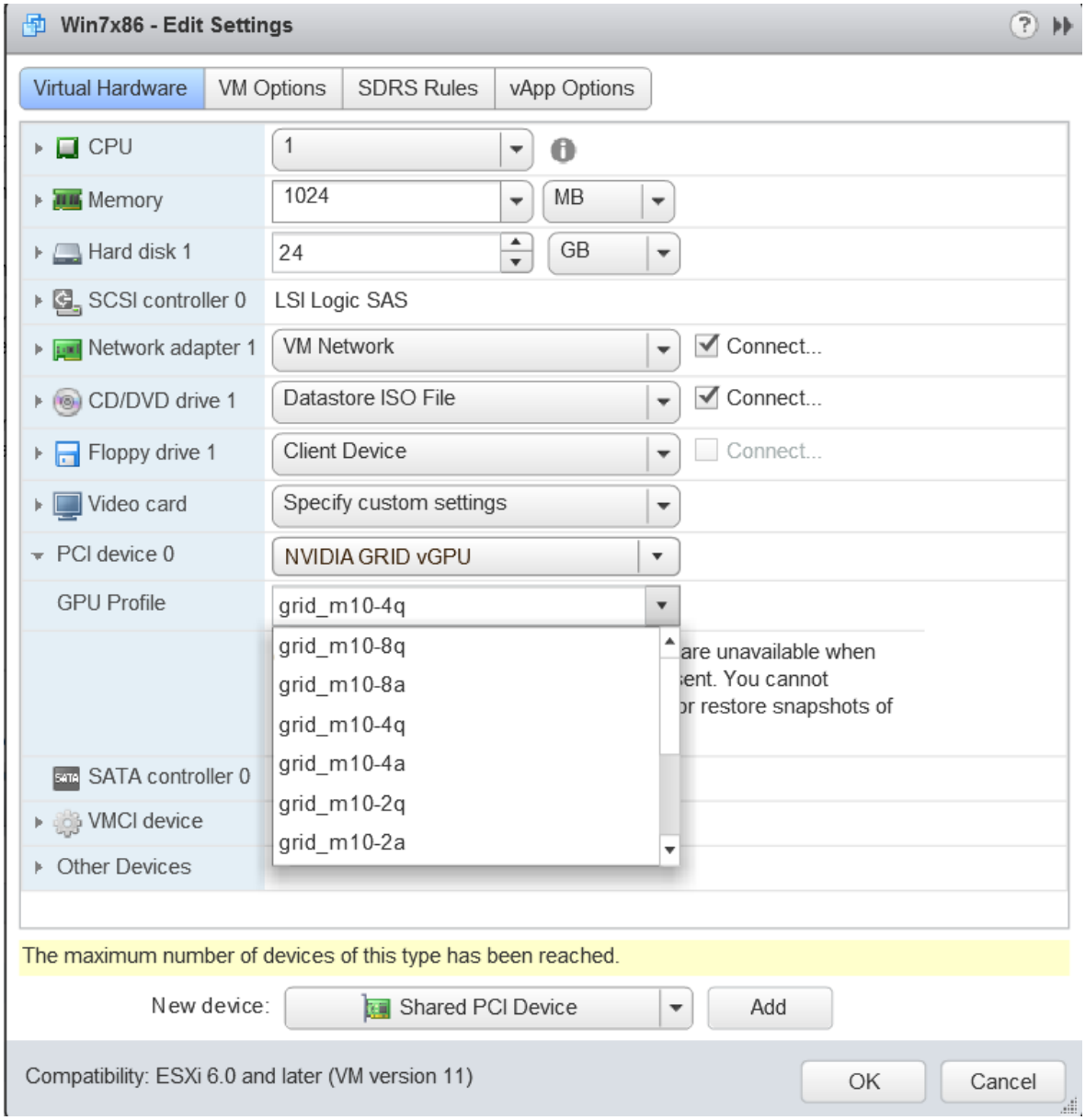
VM console in vSphere Web Client will become active again once the vGPU parameters are removed from the VM's configuration.

Note: If you are configuring a VM to use VMware vSGA, omit this task.

1. Open the vCenter Web UI.
2. In the vCenter Web UI, right-click the VM and choose **Edit Settings**.
3. Click the **Virtual Hardware** tab.
4. In the **New device** list, select **Shared PCI Device** and click **Add**.
   The **PCI device** field should be auto-populated with **`NVIDIA GRID vGPU`**.

Figure 6. VM settings for vGPU





5. From the **GPU Profile** drop-down menu, choose the type of vGPU you want to configure and click **OK**.

> **Note:** VMware vSphere does **not** support vCS. Therefore, C-series vGPU types are not available for selection from the **GPU Profile** drop-down menu.

6. Ensure that VMs running vGPU have all their memory reserved:
   a). Select **Edit virtual machine settings** from the vCenter Web UI.
   b). Expand the **Memory** section and click **Reserve all guest memory (All locked)**.

After you have configured a vSphere VM with a vGPU, start the VM. VM console in vSphere Web Client is not supported in this vGPU release. Therefore, use VMware Horizon or VNC to access the VM's desktop.

After the VM has booted, install the NVIDIA AI Enterprise graphics driver as explained in #unique_47Installing and Licensing NVIDIA AI Enterprise Components Required in a Guest VM.

Chapter 5. Installing and Licensing NVIDIA AI Enterprise Components Required in a Guest VM

5.1. Installing the NVIDIA AI Enterprise Graphics Driver on Linux from a Debian Package

The NVIDIA AI Enterprise graphics driver for Linux is distributed a Debian package file. This task requires `sudo` privileges.

1. Copy the NVIDIA AI Enterprise Linux driver package, for example `nvidia-linux-grid-470_470.63.01_amd64.deb`, to the guest VM where you are installing the driver.
2. Log in to the guest VM as a user with `sudo` privileges.
3. Open a command shell and change to the directory that contains the NVIDIA AI Enterprise Linux driver package.
4. From the command shell, run the command to install the package.

   ```
   $ sudo apt-get install ./nvidia-linux-grid-470_470.63.01_amd64.deb
   ```
5. Verify that the NVIDIA driver is operational.
   a). Reboot the system and log in.
   b). After the system has rebooted, confirm that you can see your NVIDIA vGPU device in the output from the `nvidia-smi` command.

   ```
   $ nvidia-smi
   ```

5.2. Prerequisites for Configuring a Licensed Client of NVIDIA License System with a Networked License

A client with a network connection obtains a license by leasing it from a NVIDIA License System service instance. The service instance serves the license to the client over the network from a pool of floating licenses obtained from the NVIDIA Licensing Portal. The license is returned to the service instance when the licensed client no longer requires the license.

Note: NVIDIA AI Enterprise releases earlier than 13.0 do **not** support NVIDIA License System. For full details of NVIDIA AI Enterprise releases that support NVIDIA License System, refer to .

Before configuring a licensed client, ensure that the following prerequisites are met:

- The NVIDIA AI Enterprise graphics driver is installed on the client.
- The client configuration token that you want to deploy on the client has been created from the NVIDIA Licensing Portal or the DLS as explained in *NVIDIA License System User Guide*.
- Ports 443 and 80 in your firewall or proxy must be open to allow HTTPS traffic between a service instance and its the licensed clients. These ports must be open for both CLS instances and DLS instances.

  Note: For DLS releases **before** DLS 1.1, ports 8081 and 8082 were also required to be open to allow HTTPS traffic between a DLS instance and its licensed clients. Although these ports are no longer required, they remain supported for backward compatibility.

The graphics driver creates a default location in which to store the client configuration token on the client. If you want to use this location for the client configuration token and, on Windows, are configuring the client with NVIDIA vGPU, you can configure the client with default settings. Otherwise, you must configure the client with custom settings as explained in #unique_50.

The process for configuring a licensed client is the same for CLS and DLS instances but depends on the OS that is running on the client.

5.2.1. Configuring a Licensed Client with a Networked License on Linux with Default Settings

Perform this task from the client.

1. As root, open the file `/etc/nvidia/gridd.conf` in a plain-text editor, such as `vi`.

```
$ sudo vi /etc/nvidia/gridd.conf
```

Note: You can create the `/etc/nvidia/gridd.conf` file by copying the supplied template file `/etc/nvidia/gridd.conf.template`.

2. Add the `FeatureType` configuration parameter to the file `/etc/nvidia/gridd.conf` on a new line as `FeatureType="value"`.

   value depends on the type of the GPU assigned to the licensed client that you are configuring.

GPU Type	Value
NVIDIA vGPU	1. NVIDIA AI Enterprise automatically selects the correct type of license based on the vGPU type.
Physical GPU	The feature type of a GPU in pass-through mode or a bare-metal deployment: ▶ 0: NVIDIA Virtual Applications ▶ 2: NVIDIA RTX Virtual Workstation ▶ 4: NVIDIA Virtual Compute Server

Note: You can also perform this step from **NVIDIA X Server Settings**. Before using **NVIDIA X Server Settings** to perform this step, ensure that this option has been enabled as explained in #unique_52*NVIDIA AI Enterprise Client Licensing User Guide*.

This example shows how to configure a licensed Linux client for .

```
# /etc/nvidia/gridd.conf.template - Configuration file for NVIDIA Grid Daemon
…
# Description: Set Feature to be enabled
# Data type: integer
# Possible values:
# 0 => for unlicensed state
# 1 => for NVIDIA vGPU
# 2 => for NVIDIA RTX Virtual Workstation
# 4 => for NVIDIA Virtual Compute Server
FeatureType=
...
```

3. Copy the client configuration token to the `/etc/nvidia/ClientConfigToken` directory.
4. Ensure that the file access modes of the client configuration token allow the owner to read, write, and execute the token, and the group and others only to read the token.
   a). Determine the current file access modes of the client configuration token.
   ```
   # ls -l client-configuration-token-directory
   ```
   b). If necessary, change the mode of the client configuration token to 744.
   ```
   # chmod 744 client-configuration-token-directory/client_configuration_token_*.tok
   ```
   client-configuration-token-directory
   The directory to which you copied the client configuration token in the previous step.
5. Save your changes to the `/etc/nvidia/gridd.conf` file and close the file.
6. Restart the `nvidia-gridd` service.

The NVIDIA service on the client should now automatically obtain a license from the CLS or DLS instance.

5.2.2. Verifying the NVIDIA AI Enterprise License Status of a Licensed Client

After configuring a client with an NVIDIA AI Enterprise license, verify the license status by displaying the licensed product name and status.

To verify the license status of a licensed client, run `nvidia-smi` with the `-q` or `--query` option from the licensed client, **not** the hypervisor host. If the product is licensed, the expiration date is shown in the license status.

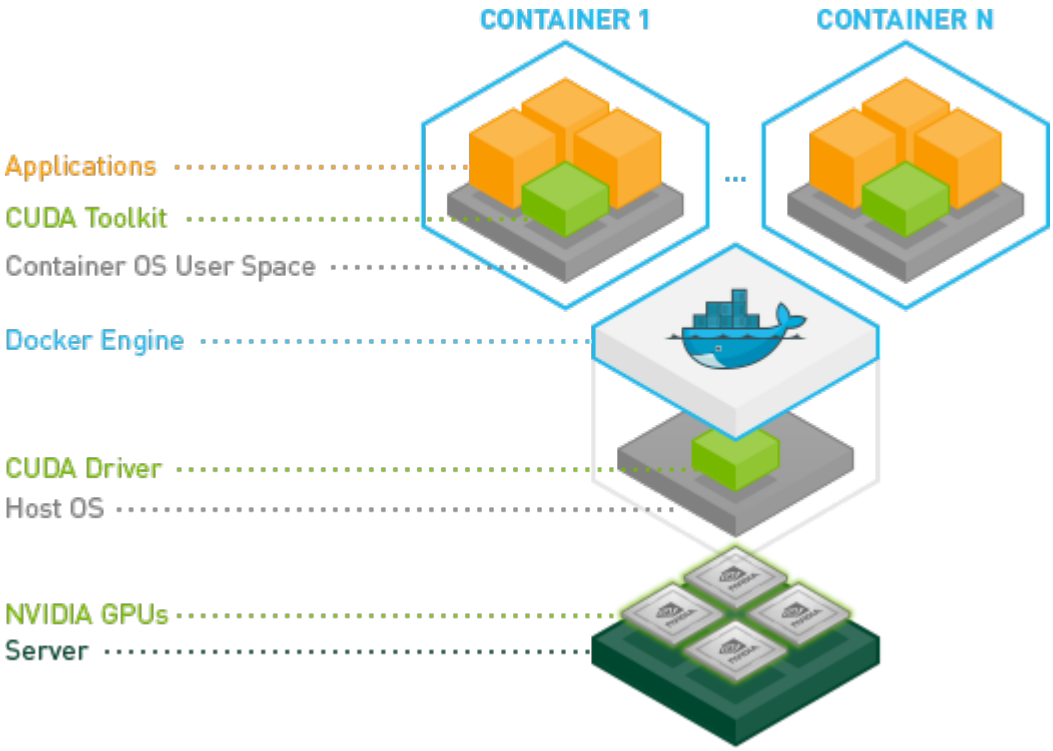
```
nvidia-smi -q
==============NVSMI LOG==============

Timestamp                           : Wed Mar 31 01:49:28 2020
Driver Version                      : 440.88
CUDA Version                        : 10.0

Attached GPUs                       : 1
GPU 00000000:00:08.0
    Product Name                    : Tesla T4
    Product Brand                   : Grid
    Display Mode                    : Enabled
    Display Active                  : Disabled
    Persistence Mode                : N/A
    Accounting Mode                 : Disabled
    Accounting Mode Buffer Size     : 4000
    Driver Model
        Current                     : WDDM
        Pending                     : WDDM
    Serial Number                   : 0334018000638
    GPU UUID                        : GPU-ba2310b6-95d1-802b-f96f-5865410fe517
    Minor Number                    : N/A
    VBIOS Version                   : 90.04.21.00.01
    MultiGPU Board                  : No
    Board ID                        : 0x8
    GPU Part Number                 : 699-2G183-0200-100
    Inforom Version
        Image Version               : G183.0200.00.02
        OEM Object                  : 1.1
        ECC Object                  : 5.0
        Power Management Object     : N/A
    GPU Operation Mode
        Current                     : N/A
        Pending                     : N/A
    GPU Virtualization Mode
        Virtualization mode         : Pass-Through
    vGPU Software Licensed Product
        Product Name                : NVIDIA Virtual Compute ServerGRID vGaming
        License Status              : Licensed (Expiry: 2021-11-13 18:29:59 GMT)
    …
    …
```

5.3. Installing NVIDIA Container Toolkit

Use NVIDIA Container Toolkit to build and run GPU accelerated Docker containers. The toolkit includes a container runtime library and utilities to configure containers to use NVIDIA GPUs automatically.



Ensure that the following software is installed in the guest VM:

- Docker 20.10 for your Linux distribution. For instructions, refer to Install Docker Engine on Ubuntu in the Docker product manuals.
- The NVIDIA AI Enterprise graphics driver. For instructions, refer to Installing the NVIDIA AI Enterprise Graphics Driver on Linux from a Debian Package.

Note: You do **not** need to install NVIDIA CUDA Toolkit on the hypervisor host.

1. Set up the GPG key and configure `apt` to use NVIDIA Container Toolkit packages in the file `/etc/apt/sources.list.d/nvidia-docker.list`.

```
$ distribution=$(. /etc/os-release;echo $ID$VERSION_ID)
$ curl -s -L https://nvidia.github.io/nvidia-docker/gpgkey | sudo apt-key add -
$ curl -s -L https://nvidia.github.io/nvidia-docker/$distribution/nvidia-
docker.list | sudo tee /etc/apt/sources.list.d/nvidia-docker.list
```

2. Download information from all configured sources about the latest versions of the packages and install the `nvidia-container-toolkit` package.

```
$ sudo apt-get update && sudo apt-get install -y nvidia-container-toolkit
```

3. Restart the Docker service.

```
$ sudo systemctl restart docker
```

5.4. Verifying the Installation of NVIDIA Container Toolkit

1. Run the `nvidia-smi` command contained in the latest official NVIDIA CUDA Toolkit image.

```
$ docker run --gpus all nvidia/cuda:11.0-base nvidia-smi
```

2. Start a GPU-enabled container on any two available GPUs.

```
$ docker run --gpus 2 nvidia/cuda:11.0-base nvidia-smi
```

3. Start a GPU-enabled container on two specific GPUs identified by their index numbers.

```
$ docker run --gpus '"device=1,2"' nvidia/cuda:10.0-base nvidia-smi
```

4. Start a GPU-enabled container on two specific GPUs with one GPU identified by its UUID and the other GPU identified by its index number.

```
$ docker run --gpus '"device=UUID-ABCDEF,1"' nvidia/cuda:11.0-base nvidia-smi
```

5. Specify a GPU capability for the container.

```
$ docker run --gpus all,capabilities=utility nvidia/cuda:11.0-base nvidia-smi
```

5.5. Installing Software Distributed as Container Images

The NGC container images accessed through the NVIDIA Enterprise Catalog includes the AI and data science applications, frameworks, and software in the infrastructure optimization and cloud native deployment layers. Each container image for an AI and data science application or framework contains the entire user-space software stack that is required to run the application or framework; namely, the CUDA libraries, cuDNN, any required Magnum IO components, TensorRT, and the framework.

Ensure that you have completed the following tasks in *NGC Private Registry User Guide*:

- Generating Your NGC API Key
- Accessing the NGC Container Registry

Perform this task from the VM.

For each AI or data science application that you are interested in, load the container as explained in Uploading an NVIDIA Container Image onto Your System in *NGC Private Registry User Guide*.

The following table lists the Docker pull command for downloading the container for each application or framework.

Application or Framework	Docker `pull` Command
NVIDIA TensorRT	`docker pull nvcr.io/nvaie/tensorflow-1-1:21.08-nvaie1.2-tf1-py3`
NVIDIA Triton Inference Server	`docker pull nvcr.io/nvaie/tritonserver:21.08-py3-sdk`
NVIDIA Triton Inference Server	`docker pull nvcr.io/nvaie/tritonserver-1-1:21.08-py3-min`
NVIDIA Triton Inference Server	`docker pull nvcr.io/nvaie/tritonserver-1-1:21.08-py3`
PyTorch	`docker pull nvcr.io/nvaie/pytorch-1-1:21.08-py3`
RAPIDS	`docker pull nvcr.io/nvaie/nvidia-rapids-1-1:21.08-cuda11.4-ubuntu20.04-py3.8-lws0.1.1`
TensorFlow 1	`docker pull nvcr.io/nvaie/tensorflow-1-1:21.08-tf1-py3`
TensorFlow 2	`docker pull nvcr.io/nvaie/tensorflow-1-1:21.08-tf2-py3`

Other Software	Docker `pull` Command
GPU Operator	`docker pull nvcr.io/nvaie/gpu-operator-1-1:v22.9.0`
Network Operator	`docker pull nvcr.io/nvaie/network-operator-1-1:v1.1.0`
vGPU Guest Driver, Ubuntu	`docker pull nvcr.io/nvaie/vgpu-guest-driver-1-1:470.129.06-ubuntu20.04`

5.6. Running ResNet-50 with TensorRT

This test verifies correct operation of NVIDIA Virtual Compute Server by running the ResNet-50 convolutional neural network with the TensorRT container from the NVIDIA GPU Cloud (NGC) container registry.

Note: This test does not require results to be reported for review. A `PASSED` result reported by the test is sufficient for the test to pass.

To complete this test, you need a Linux VM that is configured with a C-series vGPU and in which Docker CE 19.03 or later and the NVIDIA CUDA Toolkit are installed.

1. Pull the TensorRT NGC Container from the NGC container registry.
   a). Copy the **Pull Command** provided in the listing for this container image on the NGC website.
   b). Run the command that you copied with `sudo` privileges.

   For example, to pull version 20.03 of the container image, run the following command:

```
$ sudo docker pull nvcr.io/nvidia/tensorrt:20.03-py3
```

2. Launch the container image that you pulled in the previous step on all vGPUs in interactive mode, specifying that the container will be deleted when it is stopped.

   For example, if you pulled version 20.03 of the container image, run the following command to launch it:

```
$ sudo docker run --gpus all -it --rm nvcr.io/nvidia/tensorrt:20.03-py3
```

3. Launch the NVIDIA TensorRT container image on all vGPUs in interactive mode, specifying that the container will be deleted when it is stopped.

```
$ sudo docker run --gpus all -it --rm nvcr.io/nvaie/tensorrt:21.07-py3
```

4. From within the container runtime, change to the directory that contains test data for the ResNet-50 convolutional neural network.

```
# cd /workspace/tensorrt/data/resnet50
```

5. Run the ResNet-50 convolutional neural network with FP32, FP16, and INT8 precision and confirm that each test is completed with the result `PASSED`.

   a). To run ResNet-50 with the default FP32 precision, run this command:

```
# trtexec --duration=90 --workspace=1024 --percentile=99 --avgRuns=100 \
--deploy=ResNet50_N2.prototxt --batch=1 --output=prob
```

   b). To run ResNet-50 with FP16 precision, add the `--fp16` option:

```
# trtexec --duration=90 --workspace=1024 --percentile=99 --avgRuns=100 \
--deploy=ResNet50_N2.prototxt --batch=1 --output=prob --fp16
```

   c). To run ResNet-50 with INT8 precision, add the `--int8` option:

```
# trtexec --duration=90 --workspace=1024 --percentile=99 --avgRuns=100 \
--deploy=ResNet50_N2.prototxt --batch=1 --output=prob --int8
```

6. Press **Ctrl**+**P**, **Ctrl**+**Q** to exit the container runtime and return to the Linux command shell.

5.7. Running ResNet-50 with TensorFlow

This test verifies correct operation of NVIDIA Virtual Compute Server by running the ResNet-50 convolutional neural network with the **TensorFlow 1** container from the NVIDIA GPU Cloud (NGC) container registry.

Note: This test does not require results to be reported for review. Any set of results reported by the test is sufficient for the test to pass.

To complete this test, you need a Linux VM that is configured with a C-series vGPU and in which Docker CE 19.03 or later and the NVIDIA CUDA Toolkit are installed.

1. From the NGC container registry, pull a container image release of the TensorRT NGC Container tagged `tf1`.

   Note: Ensure that you do **not** pull a container image release that is tagged `tf2`. This test runs **only** with container image releases that are tagged `tf1`.

   a). In the listing for this container image on the NGC website, click the **Tags** tab and locate the most recent container image release that is tagged `tf1`.

b). Click the ellipsis (**...**) for the container image release and click **Pull Tag** to copy the command to pull this container image release.

c). Run the command that you copied with `sudo` privileges.

For example, to pull version 20.03 of the container image, run the following command:

```
$ sudo docker pull nvcr.io/nvidia/tensorflow:20.03-tf1-py3
```

2. Launch the container image that you pulled in the previous step on all vGPUs in interactive mode, specifying that the container will be deleted when it is stopped.

   For example, if you pulled version 20.03 of the container image, run the following command to launch it:

```
$ sudo docker run --gpus all -it --rm \
nvcr.io/nvidia/tensorflow:20.03-tf1-py3
```

3. Launch the **TensorFlow 1** container image on all vGPUs in interactive mode, specifying that the container will be deleted when it is stopped.

```
$ sudo docker run --gpus all -it --rm \
nvcr.io/nvaie/tensorflow:21.07-tf1-py3
```

4. From within the container runtime, change to the directory that contains test data for `cnn` example.

```
# cd /workspace/nvidia-examples/cnn
```

5. Run the ResNet-50 training test with FP16 precision.

```
# python resnet.py --layers 50 -b 64 -i 200 -u batch --precision fp16
```

6. Confirm that all operations on the application are performed correctly and that a set of results is reported when the test is completed.
7. Press **Ctrl**+**P**, **Ctrl**+**Q** to exit the container runtime and return to the Linux command shell.

Chapter 6. Additional Information

The following table provides links to additional information about each application or framework in NVIDIA AI Enterprise.

Application or Framework	Additional Information
TensorFlow	▶ TensorFlow Release Notes ▶ TensorFlow User Guide
PyTorch	PyTorch Release Notes
NVIDIA Triton Inference Server	Triton Inference Server Documentation on Github
NVIDIA TensorRT	NVIDIA TensorRT Documentation
RAPIDS	RAPIDS Docs on the RAPIDS project site

Other Software	Additional Information
NVIDIA GPU Operator	NVIDIA GPU Operator Documentation
NVIDIA Network Operator	NVIDIA Network Operator Documentation

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.

VESA DisplayPort

DisplayPort and DisplayPort Compliance Logo, DisplayPort Compliance Logo for Dual-mode Sources, and DisplayPort Compliance Logo for Active Cables are trademarks owned by the Video Electronics Standards Association in the United States and other countries.

HDMI

HDMI, the HDMI logo, and High-Definition Multimedia Interface are trademarks or registered trademarks of HDMI Licensing LLC.

OpenCL

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

Trademarks

NVIDIA, the NVIDIA logo, NVIDIA Maxwell, NVIDIA Pascal, NVIDIA Turing, NVIDIA Volta, Quadro, and Tesla are trademarks 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.

Copyright

© 2024 NVIDIA Corporation & affiliates. All rights reserved.

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