Red Hat OpenShift AI Self-Managed 2.6

Installing and uninstalling OpenShift AI Self-Managed

Install and uninstall Red Hat OpenShift AI Self-Managed

Last Updated: 2024-02-06

Red Hat OpenShift AI Self-Managed 2.6 Installing and uninstalling OpenShift AI Self-Managed

Install and uninstall Red Hat OpenShift AI Self-Managed

Legal Notice

Abstract

Install and uninstall Red Hat OpenShift AI Self-Managed on your OpenShift Container Platform cluster.

Table of Contents

PREFACE 3

CHAPTER 1. ARCHITECTURE OF OPENSHIFT AI SELF-MANAGED 4

CHAPTER 2. OVERVIEW OF INSTALLING AND DEPLOYING OPENSHIFT AI 6

CHAPTER 3. REQUIREMENTS FOR OPENSHIFT AI SELF-MANAGED 7

CHAPTER 4. ADDING ADMINISTRATIVE USERS FOR OPENSHIFT CONTAINER PLATFORM 9

CHAPTER 5. INSTALLING THE RED HAT OPENSHIFT AI OPERATOR 10

5.1. INSTALLING THE RED HAT OPENSHIFT AI OPERATOR BY USING THE CLI 10

5.2. INSTALLING THE RED HAT OPENSHIFT AI OPERATOR BY USING THE WEB CONSOLE 13

CHAPTER 6. INSTALLING AND MANAGING RED HAT OPENSHIFT AI COMPONENTS 17

6.1. INSTALLING RED HAT OPENSHIFT AI COMPONENTS BY USING THE CLI 17

6.2. INSTALLING RED HAT OPENSHIFT AI COMPONENTS BY USING THE WEB CONSOLE 19

6.3. UPDATING THE INSTALLATION STATUS OF RED HAT OPENSHIFT AI COMPONENTS BY USING THE WEB CONSOLE 21

CHAPTER 7. UNINSTALLING RED HAT OPENSHIFT AI SELF-MANAGED 23

7.1. UNINSTALLING RED HAT OPENSHIFT AI SELF-MANAGED BY USING THE CLI 23

CHAPTER 8. ACCESSING THE DASHBOARD 25

CHAPTER 9. ENABLING GPU SUPPORT IN OPENSHIFT AI 26

PREFACE

Learn how to use both the OpenShift command-line interface and web console to install Red Hat OpenShift AI Self-Managed on your OpenShift Container Platform cluster. To uninstall the product, learn how to use the recommended command-line interface (CLI) method.

NOTE

Red Hat recommends that you install only one instance of OpenShift AI on your cluster.

Installing the Red Hat OpenShift AI Operator on the same cluster as the OpenShift Data Science Add-on is not recommended or supported.

CHAPTER 1. ARCHITECTURE OF OPENSHIFT AI SELF-MANAGED

Red Hat OpenShift AI Self-Managed is an Operator that is available on a self-managed environment, such as Red Hat OpenShift Container Platform.

OpenShift AI integrates the following components and services:

- At the service layer:

  OpenShift AI dashboard

  A customer-facing dashboard that shows available and installed applications for the OpenShift AI environment as well as learning resources such as tutorials, quick starts, and documentation. Administrative users can access functionality to manage users, clusters, notebook images, accelerator profiles, and model-serving runtimes. Data scientists can use the dashboard to create projects to organize their data science work.

  Model serving

  Data scientists can deploy trained machine-learning models to serve intelligent applications in production. After deployment, applications can send requests to the model using its deployed API endpoint.

  Data science pipelines

  Data scientists can build portable machine learning (ML) workflows with data science pipelines, using Docker containers. This enables your data scientists to automate workflows as they develop their data science models.

  Jupyter (self-managed)

  A self-managed application that allows data scientists to configure their own notebook server environment and develop machine learning models in JupyterLab.

  Distributed workloads

  Data scientists can use multiple nodes in parallel to train machine-learning models or process data more quickly. This approach significantly reduces the task completion time, and enables the use of larger datasets and more complex models.

  TrustyAI

  Data scientists can review local, global, and time-series explainers for predictive models in both enterprise and data science applications. They can also use tools to detect bias and drift. These capabilities help organizations to deliver higher quality, unbiased AI-based applications.

> **IMPORTANT**
>
> The distributed workloads feature is currently available in Red Hat OpenShift AI 2.6 as Technology Preview feature only. Technology Preview features are not supported with Red Hat production service level agreements (SLAs) and might not be functionally complete. Red Hat does not recommend using them in production. These features provide early access to upcoming product features, enabling customers to test functionality and provide feedback during the development process.
>
> For more information about the support scope of Red Hat Technology Preview features, see Technology Preview Features Support Scope .

- At the management layer:

  The Red Hat OpenShift AI Operator

The Red Hat OpenShift AI Operator

A meta-operator that deploys and maintains all components and sub-operators that are part of OpenShift AI.

Monitoring services

Prometheus gathers metrics from OpenShift AI for monitoring purposes.

When you install the Red Hat OpenShift AI Operator in the OpenShift Container Platform cluster, the following new projects are created:

- The **redhat-ods-operator** project contains the Red Hat OpenShift AI Operator.
- The **redhat-ods-applications** project installs the dashboard and other required components of OpenShift AI.
- The **redhat-ods-monitoring** project contains services for monitoring.
- The **rhods-notebooks** project is where notebook environments are deployed by default.

You or your data scientists must create additional projects for the applications that will use your machine learning models.

Do not install independent software vendor (ISV) applications in namespaces associated with OpenShift AI.

CHAPTER 2. OVERVIEW OF INSTALLING AND DEPLOYING OPENSHIFT AI

Red Hat OpenShift AI is a platform for data scientists and developers of artificial intelligence (AI) applications. It provides a fully supported environment that lets you rapidly develop, train, test, and deploy machine learning models on-premises and/or in the public cloud.

OpenShift AI is provided as a managed cloud service add-on for Red Hat OpenShift or as self-managed software that you can install on-premise or in the public cloud on OpenShift.

For information about installing OpenShift AI as a managed cloud service add-on, see Installing OpenShift AI.

Installing OpenShift AI involves the following high-level tasks:

1. Confirm that your OpenShift Container Platform cluster meets all requirements. See Requirements for OpenShift AI Self-Managed.

2. Add administrative users for OpenShift Container Platform. See Adding administrative users for OpenShift Container Platform.

3. Install the Red Hat OpenShift AI Operator. See Installing the Red Hat OpenShift AI Operator.

4. Install OpenShift AI components. See Installing and managing Red Hat OpenShift AI components.

5. Configure user and administrator groups to provide user access to OpenShift AI. See Adding users.

6. Access the OpenShift AI dashboard. See Accessing the OpenShift AI dashboard.

7. Optionally, enable graphics processing units (GPUs) in OpenShift AI to ensure that your data scientists can use compute-heavy workloads in their models. See Enabling GPU support in OpenShift AI.

CHAPTER 3. REQUIREMENTS FOR OPENSHIFT AI SELF-MANAGED

Your environment must meet certain requirements to receive support for Red Hat OpenShift AI.

Installation requirements

You must meet the following requirements before you are able to install OpenShift AI on your Red Hat OpenShift Container Platform cluster.

- **Product subscriptions**
    - A subscription for Red Hat OpenShift AI Self-Managed
      Contact your Red Hat account manager to purchase new subscriptions. If you do not yet have an account manager, complete the form at https://www.redhat.com/en/contact to request one.
- **An OpenShift Container Platform cluster 4.12 or greater**
    - Use an existing cluster or create a new cluster by following the OpenShift Container Platform documentation: OpenShift Container Platform installation overview .
      Your cluster must have at least 2 worker nodes with at least 8 CPUs and 32 GiB RAM available for OpenShift AI to use when you install the Operator. To ensure that OpenShift AI is usable, additional cluster resources are required beyond the minimum requirements.
    - A default storage class that can be dynamically provisioned must be configured.
      Confirm that a default storage class is configured by running the **oc get storageclass** command. If no storage classes are noted with **(default)** beside the name, follow the OpenShift Container Platform documentation to configure a default storage class: Changing the default storage class . For more information about dynamic provisioning, see Dynamic provisioning.
    - Open Data Hub must not be installed on the cluster.
      For more information about managing the machines that make up an OpenShift cluster, see Overview of machine management.
- **An identity provider configured for OpenShift Container Platform**
  Access to the cluster as a user with the **cluster-admin** role; the **kubeadmin** user is not allowed.

  Red Hat OpenShift AI supports the same authentication systems as Red Hat OpenShift Container Platform. See Understanding identity provider configuration for more information on configuring identity providers.
- **Internet access**
  Along with Internet access, the following domains must be accessible during the installation of OpenShift AI Self-Managed:
    - cdn.redhat.com
    - subscription.rhn.redhat.com
    - registry.access.redhat.com
    - registry.redhat.io
    - quay.io

For CUDA-based images, the following domains must be accessible:

  - ngc.download.nvidia.cn
  - developer.download.nvidia.com

- **OpenShift Pipelines operator installation**
  - The Red Hat OpenShift Pipelines operator enables support for installation of pipelines in a self-managed environment.
    Before you use data science pipelines in OpenShift AI, you must install the Red Hat OpenShift Pipelines Operator. For more information, see Installing OpenShift Pipelines. If your deployment is in a disconnected self-managed environment, see Red Hat OpenShift Pipelines Operator in a restricted environment.
  - Before you can execute a pipeline in a disconnected environment, you must mirror any images used by your pipelines to a private registry.
  - You can store your pipeline artifacts in an Amazon Web Services (AWS) Simple Storage Service (S3) bucket to ensure that you do not consume local storage. To do this, you must first configure write access to your S3 bucket on your AWS account.
    If you do not have access to Amazon S3 storage, you must configure your own storage solution for use with pipelines.
- **Install KServe dependencies**
  To support KServe components, you must install dependent Operators, including the Red Hat OpenShift Serverless and Red Hat OpenShift Service Mesh Operators. For more information, see Serving large language models .

CHAPTER 4. ADDING ADMINISTRATIVE USERS FOR OPENSHIFT CONTAINER PLATFORM

Before you can install and configure OpenShift AI for your data scientist users, you must define administrative users. Only users with the **cluster-admin** role can install and configure OpenShift AI.

For more information about creating a cluster admin user, see Creating a cluster admin .

CHAPTER 5. INSTALLING THE RED HAT OPENSHIFT AI OPERATOR

This section shows how to install the Red Hat OpenShift AI Operator on your OpenShift Container Platform cluster using the command-line interface (CLI) and the OpenShift web console.

NOTE

If you want to upgrade from a previous version of OpenShift AI rather than performing a new installation, see Upgrading OpenShift AI .

NOTE

If your OpenShift cluster uses a proxy to access the Internet, you can configure the proxy settings for the Red Hat OpenShift AI Operator. See Overriding proxy settings of an Operator for more information.

5.1. INSTALLING THE RED HAT OPENSHIFT AI OPERATOR BY USING THE CLI

The following procedure shows how to use the OpenShift command-line interface (CLI) to install the Red Hat OpenShift AI Operator on your OpenShift Container Platform cluster. You must install the Operator before you can install OpenShift AI components on the cluster.

Prerequisites

- You have a running OpenShift Container Platform cluster, version 4.12 or greater, configured with a default storage class that can be dynamically provisioned.
- You have cluster administrator privileges for your OpenShift Container Platform cluster.
- You have downloaded and installed the OpenShift command-line interface (CLI). See Installing the OpenShift CLI.
- To support KServe components, you installed the dependent Operators, including the Red Hat OpenShift Serverless and Red Hat OpenShift Service Mesh Operators. For more information, see Serving large language models .

Procedure

1. Open a new terminal window.
2. In the OpenShift command-line interface (CLI), log in to your OpenShift Container Platform cluster as a cluster administrator, as shown in the following example:

```
$ oc login <openshift_cluster_url> -u <admin_username> -p <password>
```

3. Create a namespace for installation of the Operator by performing the following actions:

   a. Create a namespace YAML file, for example, **rhods-operator-namespace.yaml**.

```
apiVersion: v1
kind: Namespace
```

```
metadata:
  name: redhat-ods-operator ①
```

① **redhat-ods-operator** is the recommended namespace for the Operator.

b. Create the namespace in your OpenShift Container Platform cluster.

```
$ oc create -f rhods-operator-namespace.yaml
```

You see output similar to the following:

```
namespace/redhat-ods-operator created
```

4. Create an operator group for installation of the Operator by performing the following actions:

a. Create an **OperatorGroup** object custom resource (CR) file, for example, **rhods-operator-group.yaml**.

```
apiVersion: operators.coreos.com/v1
kind: OperatorGroup
metadata:
  name: rhods-operator
  namespace: redhat-ods-operator ①
```

① You must specify the same namespace that you created earlier in this procedure.

b. Create the **OperatorGroup** object in your OpenShift Container Platform cluster.

```
$ oc create -f rhods-operator-group.yaml
```

You see output similar to the following:

```
operatorgroup.operators.coreos.com/rhods-operator created
```

5. Create a subscription for installation of the Operator by performing the following actions:

a. Create a **Subscription** object CR file, for example, **rhods-operator-subscription.yaml**.

```
apiVersion: operators.coreos.com/v1alpha1
kind: Subscription
metadata:
  name: rhods-operator
  namespace: redhat-ods-operator ①
spec:
  name: rhods-operator
  channel: stable ②
  source: redhat-operators
  sourceNamespace: openshift-marketplace
```

① You must specify the same namespace that you created earlier in this procedure.

②

For **channel**, select a value of **fast**, **stable**, **embedded**, or **alpha**. These subscription channels are described as follows:

fast

In the **fast** channel, Red Hat provides updates for the Operator approximately every three weeks. The fast channel is intended for production use and provides functionally complete, generally available features (in addition to early-access features where noted in the documentation) that are supported with Red Hat production service level agreements (SLAs).

stable

In the **stable** channel, Red Hat provides updates for the Operator approximately every three months. The **stable** channel is intended for production use and provides functionally complete, generally available features (in addition to early-access features where noted in the documentation) that are supported with Red Hat production service level agreements (SLAs).

embedded

The **embedded** channel provides updates for products that integrate Red Hat OpenShift AI. This includes IBM watsonx.ai. If this specific use case does not apply to your organization, select **fast** or **stable**. The **embedded** channel is intended for production use and provides functionally complete, generally available features (in addition to early-access features where noted in the documentation) that are supported with Red Hat production service level agreements (SLAs).

alpha

The **alpha** channel is intended for development use only. The channel provides development builds and early-access features.

NOTE

The development builds and early-access features that the **alpha** channel provides are not supported with Red Hat production service level agreements (SLAs) and might not be functionally complete. Red Hat does not recommend using them in production. Early-access features enable customers to test functionality and provide feedback during the development process. For production environments, select **fast**, **embedded**, or **stable**, based on the preceding descriptions.

For more information about the support scope of Red Hat Technology Preview features, see Technology Preview Features Support Scope . For more information about the support scope of Red Hat Developer Preview features, see Developer Preview Features Support Scope .

NOTE

The **beta** channel is a legacy channel that will be removed in a future release. Do not select the **beta** channel for a new installation of the Operator.

For more information about the lifecycle associated with each of the available subscription channels, see Red Hat OpenShift AI Self-Managed Life Cycle.

b. As described in the preceding step, ensure that the subscription channel you specify is appropriate for your organization’s requirements.

c. Create the **Subscription** object in your OpenShift Container Platform cluster to install the Operator.

```
$ oc create -f rhods-operator-subscription.yaml
```

You see output similar to the following:

```
subscription.operators.coreos.com/rhods-operator created
```

Verification

- In the OpenShift Container Platform web console, click **Operators → Installed Operators** and confirm that the Red Hat OpenShift AI Operator shows one of the following statuses:
    - **Installing** - installation is in progress; wait for this to change to **Succeeded**. This might take several minutes.
    - **Succeeded** - installation is successful.
- In the web console, click **Home → Projects** and confirm that the following project namespaces are visible and listed as **Active**:
    - **redhat-ods-applications**
    - **redhat-ods-monitoring**
    - **redhat-ods-operator**

Additional resources

- Installing and managing Red Hat OpenShift AI components
- Adding users
- Adding Operators to a cluster

5.2. INSTALLING THE RED HAT OPENSHIFT AI OPERATOR BY USING THE WEB CONSOLE

The following procedure shows how to use the OpenShift Container Platform web console to install the Red Hat OpenShift AI Operator on your cluster. You must install the Operator before you can install OpenShift AI components on the cluster.

Prerequisites

- You have a running OpenShift Container Platform cluster, version 4.12 or greater, configured with a default storage class that can be dynamically provisioned.
- You have cluster administrator privileges for your OpenShift Container Platform cluster.
- To support KServe components, you installed the dependent Operators, including the Red Hat OpenShift Serverless and Red Hat OpenShift Service Mesh Operators. For more information, see Serving large language models .

Procedure

1. Log in to the OpenShift Container Platform web console as a cluster administrator.

2. In the web console, click **Operators** → **OperatorHub**.

3. On the **OperatorHub** page, locate the Red Hat OpenShift AI Operator.

    a. Scroll through available Operators or type *Red Hat OpenShift AI* into the **Filter by keyword** box to find the Red Hat OpenShift AI Operator.

4. Select the Operator to display additional information.

5. Read the information about the Operator and click **Install**.

6. For **Update channel**, select **fast**, **stable**, **embedded**, or **alpha**. These subscription channels are described as follows:

    fast

    In the **fast** channel, Red Hat provides updates for the Operator approximately every three weeks. The fast channel is intended for production use and provides functionally complete, generally available features (in addition to early-access features where noted in the documentation) that are supported with Red Hat production service level agreements (SLAs).

    stable

    In the **stable** channel, Red Hat provides updates for the Operator approximately every three months. The **stable** channel is intended for production use and provides functionally complete, generally available features (in addition to early-access features where noted in the documentation) that are supported with Red Hat production service level agreements (SLAs).

    embedded

    The **embedded** channel provides updates for products that integrate Red Hat OpenShift AI. This includes IBM watsonx.ai. If this specific use case does not apply to your organization, select **fast** or **stable**. The **embedded** channel is intended for production use and provides functionally complete, generally available features (in addition to early-access features where noted in the documentation) that are supported with Red Hat production service level agreements (SLAs).

    alpha

    The **alpha** channel is intended for development use only. The channel provides development builds and early-access features.

NOTE

The development builds and early-access features that the **alpha** channel provides are not supported with Red Hat production service level agreements (SLAs) and might not be functionally complete. Red Hat does not recommend using them in production. Early-access features enable customers to test functionality and provide feedback during the development process. For production environments, select **fast**, **embedded**, or **stable**, based on the preceding descriptions.

For more information about the support scope of Red Hat Technology Preview features, see Technology Preview Features Support Scope. For more information about the support scope of Red Hat Developer Preview features, see Developer Preview Features Support Scope.

NOTE

The **beta** channel is a legacy channel that will be removed in a future release. Do not select the **beta** channel for a new installation of the Operator.

For more information about the lifecycle associated with each of the available subscription channels, see Red Hat OpenShift AI Self-Managed Life Cycle.

7. For **Installation mode**, observe that the only available value is **All namespaces on the cluster (default)**. This installation mode makes the Operator available to all namespaces in the cluster.

8. For **Installed Namespace**, select **redhat-ods-operator (Operator recommended)**.

9. Under **Update approval**, select either **Automatic** or **Manual**.

10. Click **Install**.
    An installation pane opens. When the installation finishes, a checkmark appears beside the Operator name in the installation pane.

Verification

- In the OpenShift Container Platform web console, click **Operators** → **Installed Operators** and confirm that the Red Hat OpenShift AI Operator shows one of the following statuses:
    - **Installing** - installation is in progress; wait for this to change to **Succeeded**. This might take several minutes.
    - **Succeeded** - installation is successful.
- In the web console, click **Home** → **Projects** and confirm that the following project namespaces are visible and listed as **Active**:
    - **redhat-ods-applications**
    - **redhat-ods-monitoring**
    - **redhat-ods-operator**

Additional resources

- Installing and managing Red Hat OpenShift AI components

- Adding users
- Adding Operators to a cluster

CHAPTER 6. INSTALLING AND MANAGING RED HAT OPENSHIFT AI COMPONENTS

The following procedures show how to use the command-line interface (CLI) and OpenShift Container Platform web console to install and manage components of Red Hat OpenShift AI on your OpenShift Container Platform cluster.

6.1. INSTALLING RED HAT OPENSHIFT AI COMPONENTS BY USING THE CLI

The following procedure shows how to use the OpenShift command-line interface (CLI) to install specific components of Red Hat OpenShift AI on your OpenShift Container Platform cluster.

IMPORTANT

The following procedure describes how to create and configure a **DataScienceCluster** object to install Red Hat OpenShift AI components as part of a *new* installation. However, if you upgraded from version 1 of OpenShift AI (previously OpenShift Data Science), the upgrade process automatically created a default **DataScienceCluster** object. If you upgraded from version 2.4 to 2.5, the upgrade process uses the settings from the 2.4 version’s **DataScienceCluster** object. To inspect the default **DataScienceCluster** object and change the installation status of Red Hat OpenShift AI components, see Updating the installation status of Red Hat OpenShift AI components by using the web console.

Prerequisites

- To support the KServe component, you installed dependent Operators, including the Red Hat OpenShift Serverless and Red Hat OpenShift Service Mesh Operators. For more information, see Serving large language models .
- The Red Hat OpenShift AI Operator is installed on your OpenShift Container Platform cluster. See Installing the Red Hat OpenShift AI Operator .
- You have cluster administrator privileges for your OpenShift Container Platform cluster.
- You have downloaded and installed the OpenShift command-line interface (CLI). See Installing the OpenShift CLI.

Procedure

1. Open a new terminal window.
2. In the OpenShift command-line interface (CLI), log in to your on your OpenShift Container Platform cluster as a cluster administrator, as shown in the following example:

```
$ oc login <openshift_cluster_url> -u <admin_username> -p <password>
```

3. Create a **DataScienceCluster** object custom resource (CR) file, for example, **rhods-operator-dsc.yaml**.

```
apiVersion: datasciencecluster.opendatahub.io/v1
kind: DataScienceCluster
metadata:
  name: default-dsc
spec:
  components:
    codeflare:
      managementState: "Removed"
    dashboard:
      managementState: "Removed"
    datasciencepipelines:
      managementState: "Removed"
    modelmeshserving:
      managementState: "Removed"
    ray:
      managementState: "Removed"
    trustyai:
      managementState: "Removed"
    workbenches:
      managementState: "Removed"
```

4. In the **spec.components** section of the CR, for each OpenShift AI component shown, set the value of the **managementState** field to either **Managed** or **Removed**. These values are defined as follows:

   Managed

   The Operator actively manages the component, installs it, and tries to keep it active. The Operator will upgrade the component only if it is safe to do so.

   Removed

   The Operator actively manages the component but does not install it. If the component is already installed, the Operator will try to remove it.

   IMPORTANT

   - To learn how to install the KServe component, which is used by the single model serving platform to serve large language models, see Serving large language models.

   - The CodeFlare and KubeRay components are Technology Preview features only. Technology Preview features are not supported with Red Hat production service level agreements (SLAs) and might not be functionally complete. Red Hat does not recommend using them in production. These features provide early access to upcoming product features, enabling customers to test functionality and provide feedback during the development process. For more information about the support scope of Red Hat Technology Preview features, see Technology Preview Features Support Scope.

   - To learn how to configure the distributed workloads feature that uses the CodeFlare and KubeRay components, see Configuring distributed workloads.

5. Create the **DataScienceCluster** object in your OpenShift Container Platform cluster to install the specified OpenShift AI components.

   ```
   $ oc create -f rhods-operator-dsc.yaml
   ```

   You see output similar to the following:

```
datasciencecluster.datasciencecluster.opendatahub.io/default created
```

Verification

- In the OpenShift Container Platform web console, click **Workloads** → **Pods**. In the **Project** list at the top of the page, select **redhat-ods-applications**. In the applications namespace, confirm that there are running pods for each of the OpenShift AI components that you installed.
- In the web console, click **Operators** → **Installed Operators** and then perform the following actions:
  - Click the Red Hat OpenShift AI Operator.
  - Click the **Data Science Cluster** tab and select the **DataScienceCluster** object called **default-dsc**.
  - Select the **YAML** tab.
  - In the **installedComponents** section, confirm that the components you installed have a status value of **true**.

6.2. INSTALLING RED HAT OPENSHIFT AI COMPONENTS BY USING THE WEB CONSOLE

The following procedure shows how to use the OpenShift Container Platform web console to install specific components of Red Hat OpenShift AI on your cluster.

IMPORTANT

The following procedure describes how to create and configure a **DataScienceCluster** object to install Red Hat OpenShift AI components as part of a *new* installation. However, if you upgraded from version 1 of OpenShift AI (previously OpenShift Data Science), the upgrade process automatically created a default **DataScienceCluster** object. If you upgraded from a previous minor version, the upgrade process used the settings from the previous version's **DataScienceCluster** object. To inspect the **DataScienceCluster** object and change the installation status of Red Hat OpenShift AI components, see Updating the installation status of Red Hat OpenShift AI components by using the web console.

Prerequisites

- To support the KServe component, you installed dependent Operators, including the Red Hat OpenShift Serverless and Red Hat OpenShift Service Mesh Operators. For more information, see Serving large language models.
- The Red Hat OpenShift AI Operator is installed on your OpenShift Container Platform cluster. See Installing the Red Hat OpenShift AI Operator.
- You have cluster administrator privileges for your OpenShift Container Platform cluster.

Procedure

1. Log in to the OpenShift Container Platform web console as a cluster administrator.

2. In the web console, click **Operators** → **Installed Operators** and then click the Red Hat OpenShift AI Operator.

3. Create a **DataScienceCluster** object to install OpenShift AI components by performing the following actions:

   a. Click the **Data Science Cluster** tab.

   b. Click **Create DataScienceCluster**.

   c. For **Configure via**, select **YAML view**.
   An embedded YAML editor opens showing a default custom resource (CR) for the **DataScienceCluster** object.

   d. In the **spec.components** section of the CR, for each OpenShift AI component shown, set the value of the **managementState** field to either **Managed** or **Removed**. These values are defined as follows:

   Managed

   The Operator actively manages the component, installs it, and tries to keep it active. The Operator will upgrade the component only if it is safe to do so.

   Removed

   The Operator actively manages the component but does not install it. If the component is already installed, the Operator will try to remove it.

   IMPORTANT

   - To learn how to install the KServe component, which is used by the single model serving platform to serve large language models, see Serving large language models.

   - The CodeFlare and KubeRay components are Technology Preview features only. Technology Preview features are not supported with Red Hat production service level agreements (SLAs) and might not be functionally complete. Red Hat does not recommend using them in production. These features provide early access to upcoming product features, enabling customers to test functionality and provide feedback during the development process. For more information about the support scope of Red Hat Technology Preview features, see Technology Preview Features Support Scope.

   - To learn how to configure the distributed workloads feature that uses the CodeFlare and KubeRay components, see Configuring distributed workloads.

4. Click **Create**.

Verification

- On the **DataScienceClusters** page, click the **default-dsc** object and then perform the following actions:

  ○ Select the **YAML** tab.

  - In the **installedComponents** section, confirm that the components you installed have a status value of **true**.

- In the OpenShift Container Platform web console, click **Workloads** → **Pods** and then perform the following actions:
  - In the **Project** list at the top of the page, select the **redhat-ods-applications** project.
  - In the project, confirm that there are running pods for each of the OpenShift AI components that you installed.

6.3. UPDATING THE INSTALLATION STATUS OF RED HAT OPENSHIFT AI COMPONENTS BY USING THE WEB CONSOLE

The following procedure shows how to use the OpenShift Container Platform web console to update the installation status of components of Red Hat OpenShift AI on your OpenShift Container Platform cluster.

> **IMPORTANT**
>
> If you upgraded from version 1 to version 2 of OpenShift AI, the upgrade process automatically created a default **DataScienceCluster** object and enabled several components of OpenShift AI. If you upgraded from a previous minor version, the upgrade process used the settings from the previous version's **DataScienceCluster** object.
>
> The following procedure describes how to edit the **DataScienceCluster** object:
>
> - Change the installation status of the existing Red Hat OpenShift AI components
> - Add additional components to the **DataScienceCluster** object that were not available in the previous version of OpenShift AI.

Prerequisites

- To support the KServe component, you installed dependent Operators, including the Red Hat OpenShift Serverless and Red Hat OpenShift Service Mesh Operators. For more information, see Serving large language models .
- The Red Hat OpenShift AI Operator is installed on your OpenShift Container Platform cluster.
- You have cluster administrator privileges for your OpenShift Container Platform cluster.

Procedure

1. Log in to the OpenShift Container Platform web console as a cluster administrator.
2. In the web console, click **Operators** → **Installed Operators** and then click the Red Hat OpenShift AI Operator.
3. Click the **Data Science Cluster** tab.
4. On the **DataScienceClusters** page, click the **default** object.
5. Click the **YAML** tab.

An embedded YAML editor opens showing the custom resource (CR) file for the **DataScienceCluster** object.

6. In the **spec.components** section of the CR, for each OpenShift AI component shown, set the value of the **managementState** field to either **Managed** or **Removed**. These values are defined as follows:

   Managed

   The Operator actively manages the component, installs it, and tries to keep it active. The Operator will upgrade the component only if it is safe to do so.

   Removed

   The Operator actively manages the component but does not install it. If the component is already installed, the Operator will try to remove it.

   IMPORTANT

   - To learn how to install the KServe component, which is used by the single model serving platform to serve large language models, see Serving large language models.
   - If they are not already present in the CR file, you can install the CodeFlare, KubeRay, and TrustyAI features by adding components called **codeflare**, **ray**, and **trustyai** to the **spec.components** section of the CR and setting the **managementState** field for the components to **Managed**.
   - The CodeFlare and KubeRay components are Technology Preview features only. Technology Preview features are not supported with Red Hat production service level agreements (SLAs) and might not be functionally complete. Red Hat does not recommend using them in production. These features provide early access to upcoming product features, enabling customers to test functionality and provide feedback during the development process. For more information about the support scope of Red Hat Technology Preview features, see Technology Preview Features Support Scope.
   - To learn how to configure the distributed workloads feature that uses the CodeFlare and KubeRay components, see Configuring distributed workloads.

7. Click **Save**.
   For any components that you updated, OpenShift AI initiates a rollout that affects all pods to use the updated image.

Verification

- On the **DataScienceClusters** page, click the **default-dsc** object and then perform the following actions:
  - Select the **YAML** tab.
  - In the **installedComponents** section, confirm that the components you installed have a status value of **true**.
- In the OpenShift Container Platform web console, click **Workloads** → **Pods**. In the **Project** list at the top of the page, select **redhat-ods-applications**. In the applications namespace, confirm that there are running pods for each of the OpenShift AI components that you have installed.

CHAPTER 7. UNINSTALLING RED HAT OPENSHIFT AI SELF-MANAGED

This section shows how to use the OpenShift command-line interface (CLI) to uninstall the Red Hat OpenShift AI Operator and any OpenShift AI components installed and managed by the Operator.

> **NOTE**
>
> Using the CLI is the recommended way to uninstall the Operator. Depending on your version of OpenShift Container Platform, using the web console to perform the uninstallation might not prompt you to uninstall all associated components. This could leave you unclear about the final state of your cluster.

7.1. UNINSTALLING RED HAT OPENSHIFT AI SELF-MANAGED BY USING THE CLI

The following procedure shows how to use the OpenShift command-line interface (CLI) to uninstall the Red Hat OpenShift AI Operator and any OpenShift AI components installed and managed by the Operator.

Prerequisites

- You have cluster administrator privileges for your OpenShift Container Platform cluster.
- You have downloaded and installed the OpenShift command-line interface (CLI). See Installing the OpenShift CLI.
- You have backed up the persistent disks or volumes used by your persistent volume claims (PVCs).

Procedure

1. Open a new terminal window.
2. In the OpenShift command-line interface (CLI), log in to your OpenShift Container Platform cluster as a cluster administrator, as shown in the following example:

   ```
   $ oc login <openshift_cluster_url> -u system:admin
   ```

3. Create a **ConfigMap** object for deletion of the Red Hat OpenShift AI Operator.

   ```
   $ oc create configmap delete-self-managed-odh -n redhat-ods-operator
   ```

4. To delete the **rhods-operator**, set the **addon-managed-odh-delete** label to **true**.

   ```
   $ oc label configmap/delete-self-managed-odh api.openshift.com/addon-managed-odh-
   delete=true -n redhat-ods-operator
   ```

5. When all objects associated with the Operator are removed, delete the **redhat-ods-operator** project.

   a. Set an environment variable for the **redhat-ods-applications** project.

```
$ PROJECT_NAME=redhat-ods-applications
```

b. Wait until the **redhat-ods-applications** project has been deleted.

```
$ while oc get project $PROJECT_NAME &> /dev/null; do
echo "The $PROJECT_NAME project still exists"
sleep 1
done
echo "The $PROJECT_NAME project no longer exists"
```

When the **redhat-ods-applications** project has been deleted, you see the following output.

```
The redhat-ods-applications project no longer exists
```

c. When the **redhat-ods-applications** project has been deleted, delete the **redhat-ods-operator** project.

```
$ oc delete namespace redhat-ods-operator
```

Verification

- Confirm that the **rhods-operator** subscription no longer exists.

  ```
  $ oc get subscriptions --all-namespaces | grep rhods-operator
  ```

- Confirm that the following projects no longer exist.
  - **redhat-ods-applications**
  - **redhat-ods-monitoring**
  - **redhat-ods-operator**
  - **rhods-notebooks**

  ```
  $ oc get namespaces | grep -e redhat-ods* -e rhods*
  ```

> NOTE
>
> The **rhods-notebooks** project was created only if you installed the workbenches component of OpenShift AI. See Installing and managing Red Hat OpenShift AI components.

CHAPTER 8. ACCESSING THE DASHBOARD

After you have installed OpenShift AI and added users, you can access the URL for your OpenShift AI console and share the URL with the users to let them log in and work on their models.

Prerequisites

- You have installed OpenShift AI on your OpenShift Container Platform cluster.
- You have added at least one user to the user group for OpenShift AI.

Procedure

1. Log in to OpenShift Container Platform web console.
2. Click the application launcher (⊞).
3. Right-click on **Red Hat OpenShift AI** and copy the URL for your OpenShift AI instance.
4. Provide this instance URL to your data scientists to let them log in to OpenShift AI.

Verification

- Confirm that you and your users can log in to OpenShift AI by using the instance URL.

Additional resources

- Logging in to OpenShift AI
- Adding users

CHAPTER 9. ENABLING GPU SUPPORT IN OPENSHIFT AI

Optionally, to ensure that your data scientists can use compute-heavy workloads in their models, you can enable graphics processing units (GPUs) in OpenShift AI.

IMPORTANT

If you are using OpenShift AI in a disconnected self-managed environment, see Enabling GPU support in OpenShift AI instead.

Prerequisites

- You have logged in to your OpenShift Container Platform cluster.
- You have the **cluster-admin** role in your OpenShift Container Platform cluster.

Procedure

1. To enable GPU support on an OpenShift cluster, follow the instructions here: NVIDIA GPU Operator on Red Hat OpenShift Container Platform in the NVIDIA documentation.
2. Delete the **migration-gpu-status** ConfigMap.
   a. In the OpenShift Container Platform web console, switch to the **Administrator** perspective.
   b. Set the **Project** to **All Projects** or **redhat-ods-applications** to ensure you can see the appropriate ConfigMap.
   c. Search for the **migration-gpu-status** ConfigMap.
   d. Click the action menu (⋮) and select **Delete ConfigMap** from the list.
   The **Delete ConfigMap** dialog appears.
   e. Inspect the dialog and confirm that you are deleting the correct ConfigMap.
   f. Click **Delete**.
3. Restart the dashboard replicaset.
   a. In the OpenShift Container Platform web console, switch to the **Administrator** perspective.
   b. Click **Workloads → Deployments**.
   c. Set the **Project** to **All Projects** or **redhat-ods-applications** to ensure you can see the appropriate deployment.
   d. Search for the **rhods-dashboard** deployment.
   e. Click the action menu (⋮) and select **Restart Rollout** from the list.
   f. Wait until the **Status** column indicates that all pods in the rollout have fully restarted.

Verification

- The NVIDIA GPU Operator appears on the **Operators → Installed Operators** page in the OpenShift Container Platform web console.

- The reset **migration-gpu-status** instance is present in the **Instances** tab on the **AcceleratorProfile** custom resource definition (CRD) details page.

After installing the NVIDIA GPU Operator, create an accelerator profile as described in Working with accelerator profiles.