

Part V: Installing



Tables of Contents

Part V: Installing	1
Pre-installation tasks	4
Installing Red Hat OpenShift	4
Setting up shared persistent storage	5
Setting up Portworx storage classes	6
Setting up NFS storage	11
Creating projects (namespaces)	13
Obtaining your IBM entitlement API key	16
Mirroring images to your private container registry	16
Mirroring images with a bastion node	18
Mirroring images with an intermediary container registry	24
Configuring your cluster to pull images	33
Installing IBM Cloud Pak foundational services	45
Creating operator subscriptions	46
Custom SCCs for services	109
Creating custom SCCs for Watson Knowledge Catalog	109
SCCs for Db2	110
Changing required node settings	111
Setting up the scheduling service	114
Installing the scheduling service	115
Installing Cloud Pak for Data	116
Post-installation tasks	119
Integrating with the IAM Service	119
Creating a custom route to the platform	121
Configuring an external route to the Flight Server	121
Securing communication ports	122
Setting up the web client	122
Using a custom TLS certificate to connect to the platform	123
Using a CA certificate to connect to internal servers	124
Configuring SSO	125
Changing shared credentials settings	126
Setting the idle session timeout	127
Restricting the list of storage classes	128
Displaying a terms and conditions prompt	128
Enabling email notifications	129
Enabling users to access the web client from emails	130
Enabling users to create NFS storage volumes	130
Enabling users to connect to external SMB storage volumes	131
Uninstalling the common core services	132
Uninstalling Cloud Pak for Data	133

Installing IBM Cloud Pak for Data

A Red Hat® OpenShift® Container Platform cluster administrator and project administrator can work together to prepare the cluster and install IBM® Cloud Pak for Data.

Before you begin

1. Ensure that you review the following information before you install Cloud Pak for Data:
 - [Planning](#)
 - [System requirements](#)Ensure that you install the software on a system that has sufficient resources and that aligns with the guidance in the [System requirements](#). For example, if you do not follow the specified [disk requirements](#), you can run into out of memory errors.
2. Determine which [services](#) you want to install.
Some of the pre-installation tasks, such as creating catalog source and operator subscriptions, include steps for the services as well as the Cloud Pak for Data platform. If you know which services you plan to install, you can streamline your installation by batching these tasks.
3. Use the following information to ensure that you complete the appropriate tasks for your environment.

1. Do you have an existing Red Hat OpenShift Container Platform cluster?

Cloud Pak for Data is installed on a Red Hat OpenShift Container Platform Version 4.6 or Version 4.8 cluster.

Options	What to do
You already have an OpenShift 4.6 or 4.8 cluster	1. Go to 3. Do you already have supported persistent storage on your cluster?
You have an older version of OpenShift	1. Upgrade your cluster. For details, see the Red Hat OpenShift Container Platform documentation . 2. Then, go to 3. Do you already have supported persistent storage on your cluster?
You don't have an OpenShift cluster	1. Decide where you want to host your Cloud Pak for Data. Go to 2. Where do you want to host your Cloud Pak for Data installation?

2. Where do you want to host your Cloud Pak for Data installation?

You can deploy Cloud Pak for Data on-premises or on the cloud. Your deployment environment determines how you can install Red Hat OpenShift Container Platform:

Options	What to do
You want to deploy Cloud Pak for Data on-premises	1. Follow the Red Hat OpenShift Container Platform 4.6 documentation to install OpenShift. Additional guidance on setting up OpenShift is available in the IBM Cloud Paks documentation . Alternative: If you don't have existing hardware, you can purchase IBM Cloud Pak for Data System, which comes with Red Hat OpenShift Container Platform and Cloud Pak for Data already installed. 2. Go to 3. Do you already have supported persistent storage on your cluster?
You want to deploy Cloud Pak for Data on cloud	1. Decide which cloud provider you want to use. 2. Decide how you want to install and manage Red Hat OpenShift Container Platform. For details, see Installing Red Hat OpenShift Container Platform . 3. Go to 3. Do you already have supported persistent storage on your cluster?

3. Do you already have supported persistent storage on your cluster?

The Cloud Pak for Data platform supports the following storage:

Red Hat OpenShift Container Storage

Version: 4.6 or later fixes

Available in the IBM Storage Suite for IBM Cloud Paks.

IBM Spectrum® Scale Container Native

IBM Spectrum Scale Container Native Storage Access Version: 5.1.1.3 or later fixes

Container Storage Interface Version: 2.3.0 or later fixes

Available in the IBM Storage Suite for IBM Cloud Paks.

Network File System (NFS)

Version: 4

Portworx

Version: 2.7.0 or later fixes

IBM Cloud File Storage

Version: Not applicable

Ensure that you have storage that works with the [Services](#) that you plan to install.

Options	What to do
You have the supported storage	1. Review Setting up shared persistent storage to determine whether you need to complete any additional tasks to configure the storage for Cloud Pak for Data. 2. Go to 4. Do you have the required OpenShift projects on your cluster?

Options	What to do
You don't have supported storage	<ol style="list-style-type: none"> 1. Decide which storage you want to use. Ensure that you choose storage that works with the services that you plan to install. 2. Follow the guidance in Setting up shared persistent storage for installing and configuring the storage. 3. Go to 4. Do you have the required OpenShift projects on your cluster?

4. Do you have the required OpenShift projects on your cluster?

At a minimum, you must have a project where you will install the Cloud Pak for Data operators and service operators and a project where you will install an instance of Cloud Pak for Data. You might need additional projects depending on whether you want to:

- Separate the IBM Cloud Pak® foundational services operators from the Cloud Pak for Data operators
 - Install multiple instances of Cloud Pak for Data on the cluster
1. Review the guidance in [Creating projects \(namespaces\) on Red Hat OpenShift Container Platform](#) to determine whether:
 - You have the necessary projects on your cluster
 - You need to create operator groups for the projects
 2. Go to [5. Do you have your API key?](#)

5. Do you have your API key?

The Cloud Pak for Data software images are hosted on the IBM Entitled Registry. To access the images, you must have your IBM entitlement API key.

Options	What to do
You have your API key	1. Go to 6. How are you going to access the required software images?
You don't have your API key	<ol style="list-style-type: none"> 1. Follow the guidance in Obtaining your IBM entitlement API key. 2. Go to 6. How are you going to access the required software images?

6. How are you going to access the required software images?

Cloud Pak for Data images are accessible from the IBM Entitled Registry. In most situations, it is strongly recommended that you mirror the necessary software images from the IBM Entitled Registry to a private container registry.

The only situation in which you might consider pulling images directly from the IBM Entitled Registry is when your cluster is not air-gapped, your network is extremely reliable, and latency is not a concern. However, for predictable and reliable performance, you should mirror the images to a private container registry.

Important: You must mirror the necessary images to your private container registry in the following situations:

- Your cluster is air-gapped (also called an offline or disconnected cluster)
- Your cluster uses an *allowlist* to permit direct access by specific sites and the allowlist does not include the IBM Entitled Registry
- Your cluster uses a *blocklist* to prevent direct access by specific sites and the blocklist includes the IBM Entitled Registry

Options	What to do
You are pulling images from the IBM Entitled Registry	1. Go to 7. Configuring your cluster to pull software images.
You are mirroring images to a private container registry	<ol style="list-style-type: none"> 1. Review the guidance in Mirroring images to your private container registry to ensure you have a private container registry that meets the minimum requirements. 2. Determine how you will mirror the images and complete the appropriate task: <ul style="list-style-type: none"> • Mirroring images with a bastion node • Mirroring images with an intermediary container registry 3. Go to 7. Configuring your cluster to pull software images.

7. Configuring your cluster to pull software images

You must ensure that your cluster is configured to pull the software images from the appropriate location.

1. Complete the appropriate steps for your environment in [Configuring your cluster to pull Cloud Pak for Data images.](#)
2. Go to [8. Are the IBM Cloud Pak foundational services already installed on your cluster?](#)

8. Are the IBM Cloud Pak foundational services already installed on your cluster?

The IBM Cloud Pak foundational services are a prerequisite for Cloud Pak for Data. However, in some situations the IBM Cloud Pak for Data platform operator can automatically install the IBM Cloud Pak foundational services operators and services on the cluster.

For information about supported versions of IBM Cloud Pak foundational services, see the [Cloud Pak for Data platform software requirements](#).

Options	What to do
IBM Cloud Pak foundational services is already installed	1. Go to 9. Creating operator subscriptions
IBM Cloud Pak foundational services is not installed and you are using the express installation method	<p>With the express installation method, all of the operators are in the same OpenShift project and the IBM Cloud Pak for Data platform operator can automatically install IBM Cloud Pak foundational services.</p> <ol style="list-style-type: none"> 1. Go to 9. Creating operator subscriptions.

Options	What to do
IBM Cloud Pak foundational services is not installed and you are using the specialized installation method	<p>With the specialized installation method, the IBM Cloud Pak foundational services operators and the Cloud Pak for Data operators are in separate OpenShift projects. To ensure IBM Cloud Pak foundational services is installed in the correct project, you must manually install it.</p> <ol style="list-style-type: none"> 1. Follow the guidance in Installing IBM Cloud Pak foundational services. 2. Go to 9. Creating operator subscriptions.

9. Creating operator subscriptions

An operator subscription tells the cluster where to install a given operator and gives information about the operator to Operator Lifecycle Manager (OLM).

1. Complete the appropriate steps for your environment in [Creating operator subscriptions](#).
2. Go to [10. Do you plan to install services that require custom SCCs?](#)

10. Do you plan to install services that require custom SCCs?

The following services require custom security context constraints:

- Data Virtualization
- Db2®
- Db2 Big SQL
- Db2 Warehouse
- OpenPages®
- Watson™ Knowledge Catalog

Options	What to do
You plan to install one or more of these services	<ol style="list-style-type: none"> 1. Create the appropriate SCCs for your environment. For details, see Custom security context constraints for services. 2. Go to 11. Do you plan to install services that require specific node settings?
You don't plan to install any of these services	<ol style="list-style-type: none"> 1. Go to 11. Do you plan to install services that require specific node settings?

11. Do you plan to install services that require specific node settings?

The following services require specific node settings:

- Data Virtualization
- DataStage®
- Db2
- Db2 Big SQL
- Db2 Warehouse
- Jupyter Notebooks with Python 3.7 for GPU
- OpenPages
- Watson Knowledge Catalog
- Watson Machine Learning Accelerator
- Watson Studio

You might also need to adjust some node settings if you are working with large data sets or you have slower network speeds.

Options	What to do
You plan to install one or more of these services	<ol style="list-style-type: none"> 1. Change the appropriate node settings. For details, see Changing required node settings. 2. Go to 12. Do you need to install the scheduling service?
You don't plan to install any of these services	<ol style="list-style-type: none"> 1. Go to 12. Do you need to install the scheduling service?

12. Do you need to install the scheduling service?

The scheduling service is required if you plan to install Watson Machine Learning Accelerator.

However, it is strongly recommended that you install the scheduling service so that you can programmatically enforce the [quotas](#) that you set on the platform and on individual services.

Options	What to do
You need to install the scheduling service	<ol style="list-style-type: none"> 1. Follow the guidance in Installing the scheduling service. 2. Go to 13. Installing Cloud Pak for Data.
You don't plan to install the scheduling service	<ol style="list-style-type: none"> 1. Go to 13. Installing Cloud Pak for Data.

13. Installing Cloud Pak for Data

Depending on the number of OpenShift projects you created, you can install one or more instances of Cloud Pak for Data on your cluster.

1. [Install Cloud Pak for Data](#).
2. Go to [14. Completing post-installation tasks](#).

14. Completing post-installation tasks

After you install Cloud Pak for Data, make sure your cluster is secure and complete tasks that will impact how users interact with Cloud Pak for Data, such as configuring SSO or changing the route to the platform.

1. Complete the appropriate tasks for your environment in [Post-installation tasks](#).
2. Go to [15. Installing services](#).

15. Installing services

You are ready to install services on your cluster.

1. Instructions for installing IBM services are available in [Services](#).

- [Pre-installation tasks](#)
Before you install Cloud Pak for Data, complete the following tasks.
- [Installing Cloud Pak for Data](#)
When you install IBM Cloud Pak for Data, you update the IBM Cloud Pak for Data platform operator and the IBM Cloud Pak foundational services operator to watch the project where you will install IBM Cloud Pak for Data. Then, you create a custom resource to install Cloud Pak for Data in that project.
- [Post-installation tasks](#)
After you install Cloud Pak for Data, complete the following tasks.
- [Uninstalling the common core services](#)
A project administrator can uninstall the common core services from IBM Cloud Pak for Data.
- [Uninstalling Cloud Pak for Data](#)
A project administrator can uninstall the Cloud Pak for Data control plane.

Pre-installation tasks

Before you install Cloud Pak for Data, complete the following tasks.

Tip: See [Installing IBM Cloud Pak for Data](#) for guidance about which tasks you need to complete based on your environment.

1. [Installing Red Hat OpenShift Container Platform](#)
IBM Cloud Pak for Data is deployed on a Red Hat OpenShift Container Platform cluster. If you don't have an existing cluster, complete the appropriate steps to install Red Hat OpenShift on your environment.
2. [Setting up shared persistent storage](#)
Before you can install Cloud Pak for Data, you must set up shared persistent storage on your Red Hat OpenShift cluster.
3. [Creating projects \(namespaces\) on Red Hat OpenShift Container Platform](#)
Before you install IBM Cloud Pak for Data on Red Hat OpenShift Container Platform, a cluster administrator should create the OpenShift projects (Kubernetes namespaces) where you plan to deploy the Cloud Pak for Data software.
4. [Obtaining your IBM entitlement API key](#)
The IBM entitlement API key enables you to pull software images from the IBM Entitled Registry, either for installation or for mirroring.
5. [Mirroring images to your private container registry](#)
IBM Cloud Pak for Data images are accessible from the IBM Entitled Registry. In most situations, it is strongly recommended that you mirror the necessary software images from the IBM Entitled Registry to a private container registry.
6. [Configuring your cluster to pull Cloud Pak for Data images](#)
To ensure that your cluster can pull Cloud Pak for Data software images, you must update your cluster configuration.
7. [Installing IBM Cloud Pak foundational services](#)
IBM Cloud Pak foundational services is a prerequisite for IBM Cloud Pak for Data. IBM Cloud Pak foundational services is installed one time on the cluster and is used by any instances of Cloud Pak for Data or other IBM Cloud Paks that are installed on the cluster.
8. [Creating operator subscriptions](#)
An operator subscription tells the cluster where to install a given operator and gives information about the operator to Operator Lifecycle Manager (OLM).
9. [Custom security context constraints for services](#)
Most Cloud Pak for Data services use the **restricted** security context constraint (SCC) that is provided by Red Hat OpenShift Container Platform. However, if you plan to install certain Cloud Pak for Data services, you might need to use some custom SCCs.
10. [Changing required node settings](#)
Some services that run on IBM Cloud Pak for Data require specific settings on the nodes in the cluster. To ensure that the cluster has the required settings for these services, an operating system administrator with `root` privileges must review and adjust the settings on the appropriate nodes in the cluster.

Related reference

- [Installing Cloud Pak for Data](#)
- [Post-installation tasks](#)
- [Uninstalling the common core services](#)
- [Uninstalling Cloud Pak for Data](#)

Installing Red Hat OpenShift Container Platform

IBM® Cloud Pak for Data is deployed on a Red Hat® OpenShift® Container Platform cluster. If you don't have an existing cluster, complete the appropriate steps to install Red Hat OpenShift on your environment.

Tip: After you install Red Hat OpenShift Container Platform on your cluster, see [Installing IBM Cloud Pak for Data](#) for an overview of the installation flow.

Supported deployment environments

You can deploy Cloud Pak for Data on-premises or on the cloud. Your deployment environment determines how you can install Red Hat OpenShift Container Platform:

- If you deploy Cloud Pak for Data on-premises, you must install a self-managed Red Hat OpenShift Container Platform cluster.
- If you deploy Cloud Pak for Data on cloud, you can choose whether to use a managed or self-managed Red Hat OpenShift Container Platform cluster. However, **managed** OpenShift is not supported on all clouds.

Cloud provider	Managed OpenShift	Self-managed OpenShift
IBM Cloud	Supported (recommended)	Supported
Amazon Web Services (AWS)	Supported	Supported
Microsoft Azure	Not supported	Supported
Google Cloud	Not supported	Supported

Go to the appropriate section for your deployment environment:

- [On-premises](#)
- [IBM Cloud](#)
- [AWS](#)
- [Azure](#)
- [Google Cloud](#)

On-premises

You can install a **self-managed** OpenShift cluster on-premises.

Follow the [Red Hat OpenShift Container Platform 4.6 documentation](#) to install OpenShift.

Additional guidance on setting up OpenShift is available in the [IBM Cloud Paks documentation](#).

Alternative: If you don't have existing hardware, you can purchase IBM Cloud Pak for Data System, which comes with Red Hat OpenShift Container Platform and Cloud Pak for Data already installed.

IBM Cloud

Managed OpenShift

To install managed OpenShift, you can:

- Install [Red Hat OpenShift Container Platform on IBM Cloud](#).

Self-managed OpenShift

To install self-managed OpenShift, contact IBM Software Support.

AWS

Managed OpenShift

To install managed OpenShift, see [Red Hat OpenShift Service on AWS \(ROSA\)](#).

Self-managed OpenShift

To install self-managed OpenShift, you can:

- Install [Red Hat OpenShift Container Platform on AWS](#)

Azure

Self-managed OpenShift

To install self-managed OpenShift, you can:

- Install [Red Hat OpenShift Container Platform on Azure](#).

Google Cloud

You can install a **self-managed** OpenShift cluster Google Cloud.

For details, see [Red Hat OpenShift Container Platform on Google Cloud](#).

Next topic: [Setting up shared persistent storage](#)

Setting up shared persistent storage

Before you can install Cloud Pak for Data, you must set up shared persistent storage on your Red Hat® OpenShift® cluster.

Tip: For information about supported storage providers, see [Storage considerations](#).

Ensure that the services that you plan to install on Cloud Pak for Data can use the storage that you use. For details, see [Storage requirements](#).

Select your storage type and complete the steps to set up storage.

Storage type	What to do
Red Hat OpenShift Container Storage	<p>Installation</p> <p>To install OpenShift Container Storage, see the Red Hat OpenShift Container Storage documentation.</p> <p>Post-installation set up</p> <p>No additional set up is required.</p>
IBM Spectrum® Scale Container Native	<p>Installation</p> <p>For IBM Spectrum Scale and IBM Spectrum Scale Container Storage Interface, see Installation in the IBM Spectrum Scale Container Native documentation.</p> <p>Post-installation set up</p> <p>Edit the IBM Spectrum Scale storage class. The permissions field should be set to 777 for read, write, and execute on the local file system.</p> <p>Independent fileset storage class example:</p> <pre>apiVersion: storage.k8s.io/v1 kind: StorageClass metadata: name: ibm-spectrum-scale-sc provisioner: spectrumscale.csi.ibm.com parameters: volBackendFs: "remote-sample" clusterId: "10724741469548137692" permissions: "777" reclaimPolicy: Delete</pre> <p>There are additional parameters you can set with the recommended Independent fileset storage class, such as:</p> <ul style="list-style-type: none"> • <code>volBackendFs</code> • <code>clusterId</code> • <code>name</code> <p>For details, see Storage class in the IBM Spectrum Scale Container Storage Interface Driver documentation.</p>
Portworx	<p>Installation</p> <p>For details on Portworx Enterprise for IBM, see Install Portworx on OpenShift</p> <p>Post-installation set up</p> <p>You must configure the required storage classes. For details, see Creating Portworx storage classes</p>
NFS	<p>Installation</p> <p>Refer to the installation documentation for your NFS storage provider.</p> <p>Post-installation set up</p> <p>You must set up dynamic storage and configure your storage. For details, see Setting up NFS storage</p>
IBM Cloud File Storage	<p>Installation</p> <p>When you configure your Red Hat OpenShift cluster, ensure that you select one of the following IBM Cloud File Storage storage classes:</p> <ul style="list-style-type: none"> • <code>ibmc-file-gold-gid</code> • <code>ibm-file-custom-gold-gid</code> <p>Post-installation set up</p> <p>No additional configuration is required to use IBM Cloud File Storage. However, you might need to adjust your I/O and storage size settings for production workloads, as indicated in the Storage comparison table.</p>

- [Setting up Portworx storage classes](#)

If you decide to use Portworx as your storage option, Cloud Pak for Data requires the following storage classes. You can set them up manually.

- [Setting up NFS storage](#)

By default, NFS does not support dynamic storage provisioning. If you plan to use Cloud Pak for Data for persistent storage, you must set up your NFS storage before you install Cloud Pak for Data.

Previous topic: [Installing Red Hat OpenShift Container Platform](#)

Next topic: [Creating projects \(namespaces\) on Red Hat OpenShift Container Platform](#)

Setting up Portworx storage classes

If you decide to use Portworx as your storage option, Cloud Pak for Data requires the following storage classes. You can set them up manually.

Before you begin

Required role: To complete this task, you must be a cluster administrator.

Ensure that you have a minimum of 1 TB of raw, unformatted disk on every compute node that is designated for storage. The raw disk must have the same device name on all of the worker nodes.

About this task

You must manually create the following Portworx storage classes that are required for Cloud Pak for Data:

Storage class	Storage type	Storage class definitions
portworx-couchdb-sc	CouchDB	<pre># CouchDB (Implemented application-level redundancy) cat <<EOF oc create -f - kind: StorageClass apiVersion: storage.k8s.io/v1 metadata: name: portworx-couchdb-sc provisioner: kubernetes.io/portworx-volume parameters: repl: "3" priority_io: "high" io_profile: "db_remote" disable_io_profile_protection: "1" allowVolumeExpansion: true reclaimPolicy: Retain volumeBindingMode: Immediate EOF</pre>
portworx-elastic-sc	Elastic Search	<pre># ElasticSearch (Implemented application-level redundancy) cat <<EOF oc create -f - kind: StorageClass apiVersion: storage.k8s.io/v1 metadata: name: portworx-elastic-sc provisioner: kubernetes.io/portworx-volume parameters: repl: "2" priority_io: "high" io_profile: "db_remote" disable_io_profile_protection: "1" allowVolumeExpansion: true reclaimPolicy: Retain volumeBindingMode: Immediate EOF</pre>
portworx-solr-sc	Solr	<pre># Solr cat <<EOF oc create -f - kind: StorageClass apiVersion: storage.k8s.io/v1 metadata: name: portworx-solr-sc provisioner: kubernetes.io/portworx-volume parameters: repl: "3" priority_io: "high" io_profile: "db_remote" disable_io_profile_protection: "1" allowVolumeExpansion: true reclaimPolicy: Retain volumeBindingMode: Immediate EOF</pre>
portworx-cassandra-sc	Cassandra	<pre># Cassandra cat <<EOF oc create -f - kind: StorageClass apiVersion: storage.k8s.io/v1 metadata: name: portworx-cassandra-sc provisioner: kubernetes.io/portworx-volume parameters: repl: "3" priority_io: "high" io_profile: "db_remote" disable_io_profile_protection: "1" allowVolumeExpansion: true reclaimPolicy: Retain volumeBindingMode: Immediate EOF</pre>
portworx-kafka-sc	Apache Kafka	<pre># Kafka cat <<EOF oc create -f - kind: StorageClass apiVersion: storage.k8s.io/v1 metadata: name: portworx-kafka-sc provisioner: kubernetes.io/portworx-volume parameters: repl: "3" priority_io: "high" io_profile: "db_remote" disable_io_profile_protection: "1" allowVolumeExpansion: true reclaimPolicy: Retain volumeBindingMode: Immediate EOF</pre>

Storage class	Storage type	Storage class definitions
portworx-metastoredb-sc	Metastore	<pre># metastoredb: cat <<EOF oc create -f - apiVersion: storage.k8s.io/v1 kind: StorageClass metadata: name: portworx-metastoredb-sc parameters: priority_io: high io_profile: db_remote repl: "3" disable_io_profile_protection: "1" allowVolumeExpansion: true provisioner: kubernetes.io/portworx-volume reclaimPolicy: Retain volumeBindingMode: Immediate EOF</pre>
portworx-rwx-gp3-sc	GP3 replica 3	<pre># General Purpose, 3 Replicas - Default SC for other applications # without specific SC defined and with RWX volume access mode - New Install cat <<EOF oc create -f - apiVersion: storage.k8s.io/v1 kind: StorageClass metadata: name: portworx-rwx-gp3-sc parameters: priority_io: high repl: "3" sharedv4: "true" io_profile: db_remote disable_io_profile_protection: "1" allowVolumeExpansion: true provisioner: kubernetes.io/portworx-volume reclaimPolicy: Retain volumeBindingMode: Immediate EOF</pre>
portworx-shared-gp3	GP3 replica 3	<pre># General Purpose, 3 Replicas [Default for other applications without # specific SC defined and with RWX volume access mode] - SC portworx- shared-gp3 for upgrade purposes cat <<EOF oc create -f - apiVersion: storage.k8s.io/v1 kind: StorageClass metadata: name: portworx-shared-gp3 parameters: priority_io: high repl: "3" sharedv4: "true" io_profile: db_remote disable_io_profile_protection: "1" allowVolumeExpansion: true provisioner: kubernetes.io/portworx-volume reclaimPolicy: Retain volumeBindingMode: Immediate EOF</pre>
portworx-rwx-gp2-sc	GP2 replica 2	<pre># General Purpose, 2 Replicas RWX volumes cat <<EOF oc create -f - apiVersion: storage.k8s.io/v1 kind: StorageClass metadata: name: portworx-rwx-gp2-sc parameters: priority_io: high repl: "2" sharedv4: "true" io_profile: db_remote disable_io_profile_protection: "1" allowVolumeExpansion: true provisioner: kubernetes.io/portworx-volume reclaimPolicy: Retain volumeBindingMode: Immediate EOF</pre>
portworx-dv-shared-gp	Shared DV replica 1	<pre># DV - Single replica cat <<EOF oc create -f - allowVolumeExpansion: true apiVersion: storage.k8s.io/v1 kind: StorageClass metadata: name: portworx-dv-shared-gp parameters: block_size: 4096b priority_io: high repl: "1" shared: "true" provisioner: kubernetes.io/portworx-volume reclaimPolicy: Retain volumeBindingMode: Immediate EOF</pre>

Storage class	Storage type	Storage class definitions
portworx-dv-shared-gp3	Shared DV GP3 replica 3	<pre># DV - three replicas cat <<EOF oc create -f - allowVolumeExpansion: true apiVersion: storage.k8s.io/v1 kind: StorageClass metadata: name: portworx-dv-shared-gp3 parameters: block_size: 4096b priority_io: high repl: "3" shared: "true" provisioner: kubernetes.io/portworx-volume reclaimPolicy: Retain volumeBindingMode: Immediate EOF</pre>
portworx-shared-gp-allow	Streams	<pre># Streams cat <<EOF oc create -f - allowVolumeExpansion: true apiVersion: storage.k8s.io/v1 kind: StorageClass metadata: name: portworx-shared-gp-allow parameters: priority_io: high repl: "3" io_profile: "cms" provisioner: kubernetes.io/portworx-volume reclaimPolicy: Delete volumeBindingMode: Immediate EOF</pre>
portworx-rwx-gp-sc	GP replica 1	<pre># General Purpose, 1 Replica - RWX volumes for TESTING ONLY. cat <<EOF oc create -f - kind: StorageClass apiVersion: storage.k8s.io/v1 metadata: name: portworx-rwx-gp-sc provisioner: kubernetes.io/portworx-volume parameters: repl: "1" priority_io: "high" sharedv4: "true" io_profile: db remote disable_io_profile_protection: "1" allowVolumeExpansion: true volumeBindingMode: Immediate reclaimPolicy: Delete EOF</pre>
portworx-shared-gp	Shared GP high IOPS	<pre># General Purpose, 3 Replicas - RWX volumes - placeholder SC portworx- shared-gp for upgrade purposes cat <<EOF oc create -f - apiVersion: storage.k8s.io/v1 kind: StorageClass metadata: name: portworx-shared-gp parameters: priority_io: high repl: "3" sharedv4: "true" io_profile: db remote disable_io_profile_protection: "1" allowVolumeExpansion: true provisioner: kubernetes.io/portworx-volume reclaimPolicy: Retain volumeBindingMode: Immediate EOF</pre>
portworx-gp3-sc	GP3 replica 3	<pre># General Purpose, 3 Replicas RWO volumes rabbitmq and redis-ha - New Install cat <<EOF oc create -f - apiVersion: storage.k8s.io/v1 kind: StorageClass metadata: name: portworx-gp3-sc parameters: priority_io: high repl: "3" io_profile: "db remote" disable_io_profile_protection: "1" allowVolumeExpansion: true provisioner: kubernetes.io/portworx-volume reclaimPolicy: Retain volumeBindingMode: Immediate EOF</pre>

Storage class	Storage type	Storage class definitions
portworx-nonshared-gp2	GP2 nonshared throughput optimized	# General Purpose, 3 Replicas RWO volumes rabbitmq and redis-ha - placeholder SC portworx-nonshared-gp2 for upgrade purposes cat <<EOF oc create -f - apiVersion: storage.k8s.io/v1 kind: StorageClass metadata: name: portworx-nonshared-gp2 parameters: priority_io: high repl: "3" io_profile: "db_remote" disable_io_profile_protection: "1" allowVolumeExpansion: true provisioner: kubernetes.io/portworx-volume reclaimPolicy: Retain volumeBindingMode: Immediate EOF
portworx-shared-gp1	Shared GP high iops	#Shared gp high iops: cat <<EOF oc create -f - apiVersion: storage.k8s.io/v1 kind: StorageClass metadata: name: portworx-shared-gp1 parameters: priority_io: high repl: "1" sharedv4: "true" allowVolumeExpansion: true provisioner: kubernetes.io/portworx-volume reclaimPolicy: Retain volumeBindingMode: Immediate EOF
January 2021 release or later portworx-db-gp	DB GP replica 1 for MongoDB	# gp db cat <<EOF oc create -f - apiVersion: storage.k8s.io/v1 kind: StorageClass metadata: name: portworx-db-gp parameters: io_profile: "db_remote" repl: "1" disable_io_profile_protection: "1" allowVolumeExpansion: true provisioner: kubernetes.io/portworx-volume reclaimPolicy: Retain volumeBindingMode: Immediate EOF
portworx-db-gp2-sc	DB GP2 replica 3	# General Purpose for Databases, 2 Replicas - MongoDB - (Implemented application-level redundancy) cat <<EOF oc create -f - apiVersion: storage.k8s.io/v1 kind: StorageClass metadata: name: portworx-db-gp2-sc parameters: priority_io: "high" io_profile: "db_remote" repl: "2" disable_io_profile_protection: "1" allowVolumeExpansion: true provisioner: kubernetes.io/portworx-volume reclaimPolicy: Retain volumeBindingMode: Immediate EOF
portworx-db-gp3-sc	DB GP3 replica 3	# General Purpose for Databases, 3 Replicas cat <<EOF oc create -f - apiVersion: storage.k8s.io/v1 kind: StorageClass metadata: name: portworx-db-gp3-sc parameters: io_profile: "db_remote" repl: "3" priority_io: "high" disable_io_profile_protection: "1" allowVolumeExpansion: true provisioner: kubernetes.io/portworx-volume reclaimPolicy: Retain volumeBindingMode: Immediate EOF

Storage class	Storage type	Storage class definitions
portworx-db2-rwx-sc	Db2® and Db2 Warehouse (System and Backup Storage) and Data Virtualization	# DB2 RWX shared volumes for System Storage, backup storage, future load storage, and future diagnostic logs storage cat <<EOF oc create -f - allowVolumeExpansion: true apiVersion: storage.k8s.io/v1 kind: StorageClass metadata: name: portworx-db2-rwx-sc parameters: io_profile: cms block_size: 4096b nfs_v4: "true" repl: "3" sharedv4: "true" priority_io: high provisioner: kubernetes.io/portworx-volume reclaimPolicy: Retain volumeBindingMode: Immediate EOF
portworx-db2-rwo-sc	Db2 and Db2 Warehouse (User Storage) Watson™ Knowledge Catalog Db2 Metastore	# Db2 RWO volumes SC for user storage, future transaction logs storage, future archive/mirrors logs storage. This is also used for WKC DB2 Metastore cat <<EOF oc create -f - allowVolumeExpansion: true apiVersion: storage.k8s.io/v1 kind: StorageClass metadata: name: portworx-db2-rwo-sc parameters: block_size: 4096b io_profile: db_remote priority_io: high repl: "3" sharedv4: "false" disable_io_profile_protection: "1" provisioner: kubernetes.io/portworx-volume reclaimPolicy: Retain volumeBindingMode: Immediate EOF
portworx-db2-sc	Watson Knowledge Catalog Db2 Metastore (Upgrade)	# WKC DB2 Metastore - SC portworx-db2-sc for upgrade purposes cat <<EOF oc create -f - allowVolumeExpansion: true apiVersion: storage.k8s.io/v1 kind: StorageClass metadata: name: portworx-db2-sc parameters: io_profile: "db_remote" priority_io: high repl: "3" disable_io_profile_protection: "1" provisioner: kubernetes.io/portworx-volume reclaimPolicy: Retain volumeBindingMode: Immediate EOF
portworx-watson-assistant-sc	Watson Assistant	# Watson Assitant - This was previously named portworx-assitant cat <<EOF oc create -f - apiVersion: storage.k8s.io/v1 kind: StorageClass metadata: name: portworx-watson-assistant-sc parameters: repl: "3" priority_io: "high" io_profile: "db_remote" block_size: "64k" disable_io_profile_protection: "1" allowVolumeExpansion: true provisioner: kubernetes.io/portworx-volume reclaimPolicy: Retain volumeBindingMode: Immediate EOF
portworx-informix-sc	Informix®	# Informix cat <<EOF oc create -f - apiVersion: storage.k8s.io/v1 kind: StorageClass metadata: name: portworx-informix-sc provisioner: kubernetes.io/portworx-volume parameters: repl: "3" priority_io: high io_profile: db block_size: 2048b allowVolumeExpansion: true EOF

By default, NFS does not support dynamic storage provisioning. If you plan to use Cloud Pak for Data for persistent storage, you must set up your NFS storage before you install Cloud Pak for Data.

Supported storage topology

If you use NFS storage, you can use one of following cluster configurations:

- NFS on a dedicated node in the same VLAN as the cluster (recommended)
- An external NFS server
If you select this option, configure the server based on your availability requirements and ensure that you have a sufficiently fast network connection (at least 1 GB) to reduce latency and ensure performance.

Configuration requirements

Ensure that the following statements are true:

- All of the nodes in the cluster must have access to mount the NFS server.
- All of the nodes in the cluster must have read/write access to the NFS server.
- Containerized processes must have read/write access to the NFS server.
Important: Containerized processes create files that are owned by various UIDs. (In Cloud Pak for Data, most services use long UIDs based on the Red Hat® OpenShift® Container Platform project where they are installed.) If you restrict access to the NFS served to specific UIDs, you might encounter errors when installing or running Cloud Pak for Data.
For information on determining which UIDs are used, see [Service UIDs](#).
- If you use NFS as the storage for a database service, ensure that the storage has sufficient throughput. For details, see the appropriate topic for your environment:
 - Db2®: [Requirements for Db2 on SELinux](#)
 - Db2 Warehouse: [Requirements for Db2 Warehouse on SELinux](#)

Setting the NFS export

Ensure that the NFS export is set to `no_root_squash`.

Note: If you are installing Cloud Pak for Data from the [IBM® Cloud catalog](#), the NFS export is automatically set to `no_root_squash`.

However, if you are *manually* installing Cloud Pak for Data on IBM Cloud, you must follow the guidance in [Implementing no_root_squash for NFS](#).

Configuring dynamic storage

By default, Red Hat OpenShift does not include a [provisioner plug-in](#) to create an NFS storage class. To dynamically provision NFS storage, use the [Kubernetes NFS-Client Provisioner](#), which is available from the [Kubernetes SIGs](#) organization on GitHub.

Permissions you need for this task

You must be a cluster administrator.

Important: The following steps assume you have an existing NFS server. Ensure that you know how to connect to your NFS server. At a minimum, you must have the hostname of the server.

To configure dynamic storage:

1. Ensure that your NFS server is accessible from your Red Hat OpenShift Container Platform cluster.
2. Clone the <https://github.com/kubernetes-sigs/nfs-subdir-external-provisioner> repository.
3. Open a `bash` shell and change to the deploy directory of the repository.
4. Log in to your Red Hat OpenShift Container Platform cluster as a user with sufficient permissions to complete the task:

```
oc login OpenShift_URL:port
```

5. Authorize the provisioner by running the following commands.

- a. Create the required role based access control.

Important: If you plan to deploy the NFS provisioner to a project other than the `default` project, you must replace each instance of `default` in the `rbac.yaml` file before you run this command.

```
oc create -f rbac.yaml
```

- b. Add the `nfs-client-provisioner` security context constraint to the `system` service account.

If you plan to deploy the NFS provisioner to a project other than the `default` project, replace `default` in the following command.

```
oc adm policy add-scc-to-user hostmount-anyuid system:serviceaccount:default:nfs-client-provisioner
```

6. Edit the `deployment.yaml` file in the `deploy` directory to specify the following information:

- The project (namespace) where the NFS provisioner is deployed.
- The image that corresponds to your Red Hat OpenShift Container Platform architecture:
 - x86-64: `gcr.io/k8s-staging-sig-storage/nfs-subdir-external-provisioner:amd64-linux-v4.0.2`
 - Power®: `gcr.io/k8s-staging-sig-storage/nfs-subdir-external-provisioner:ppc64le-linux-v4.0.2`
 - s390x: `gcr.io/k8s-staging-sig-storage/nfs-subdir-external-provisioner:s390x-linux-v4.0.2`
- The hostname of your NFS server.
- The path where you want to dynamically provision storage on your NFS server.

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nfs-client-provisioner
  labels:
    app: nfs-client-provisioner
  namespace: default # Specify the namespace where the NFS provisioner is deployed
spec:
```

```

replicas: 1
strategy:
  type: Recreate
selector:
  matchLabels:
    app: nfs-client-provisioner
template:
  metadata:
    labels:
      app: nfs-client-provisioner
spec:
  serviceAccountName: nfs-client-provisioner
  containers:
  - name: nfs-client-provisioner
    image: nfs-provisioner-image # Specify the appropriate image based on your OpenShift architecture
    volumeMounts:
    - name: nfs-client-root
      mountPath: /persistentvolumes
    env:
    - name: PROVISIONER_NAME
      value: nfs-storage
    - name: NFS_SERVER
      value: MyNFSHostname # Specify the host name of your NFS server
    - name: NFS_PATH
      value: /nfs/cpshare/ # Specify the path where you want to provision storage
  volumes:
  - name: nfs-client-root
    nfs:
      server: MyNFSHostname # Specify the host name of your NFS server
      path: /nfs/cpshare/ # Specify the path where you want to provision storage

```

7. Deploy the NFS provisioner:

```
oc create -f deployment.yaml
```

8. Edit the class.yaml file to specify the names of the storage classes that you want to create. The following example includes the recommended `managed-nfs-storage` storage class:

```

apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: managed-nfs-storage # Recommended storage class name
  provisioner: nfs-client-provisioner # This name must match the value you specified in the deployment.yaml
parameters:
  archiveOnDelete: "false"

```

For a complete list of parameters, see [Deploying your storage class](#) in the NFS provisioner documentation.

9. Create the storage class:

```
oc create -f class.yaml
```

10. Verify that the NFS provisioner is running correctly:

a. Create a test persistent volume claim (PVC).

Note: The test-claim.yaml file uses the `managed-nfs-storage` storage class.

```
oc create -f test-claim.yaml -f test-pod.yaml
```

b. On your NFS server, verify that the share directory, which you specified in the deployment.yaml file, contains a file called SUCCESS.

c. Remove the test PVC:

```
oc delete -f test-pod.yaml -f test-claim.yaml
```

Creating projects (namespaces) on Red Hat OpenShift Container Platform

Before you install IBM® Cloud Pak for Data on Red Hat® OpenShift® Container Platform, a cluster administrator should create the OpenShift projects (Kubernetes namespaces) where you plan to deploy the Cloud Pak for Data software.

Permissions you need for this task

You must be a cluster administrator.

When you need to complete this task

You must complete this task the first time you install Cloud Pak for Data.

You might need to complete this task if you decide to install additional instances of Cloud Pak for Data on your cluster or decide to deploy a service in a tethered namespace.

About this task

For information on supported project configurations, see [Deployment architecture](#).

Use the following table to determine which projects (namespaces) you need to create.

Project	Description
---------	-------------

Project	Description
<p>ibm-common-services</p> <p>(Recommended name, used in subsequent commands)</p>	<p>Required for all installations.</p> <p>The project where IBM Cloud Pak® foundational services is installed.</p> <p>If IBM Cloud Pak foundational services is already installed on your cluster, identify the project where it is installed.</p> <p>If IBM Cloud Pak foundational services is not installed on your cluster, ibm-common-services is the recommended project name.</p> <p>ibm-common-services is used in various installation commands.</p> <p>If you want to install IBM Cloud Pak foundational services in a different project, you must create configmap. For details, see Installing IBM Cloud Pak foundational services in a custom namespace. (Using a different project is not recommended for typical installations.)</p> <p>Additional software that might be installed in this project</p> <p>Depending on the software that you plan to install and the installation method that you use, the following software might also be installed in the ibm-common-services project:</p> <ul style="list-style-type: none"> • The IBM Cloud Pak for Data scheduling service If you need to install the scheduling service, it is recommended that you install it in the same project as IBM Cloud Pak foundational services. • The IBM Cloud Pak for Data platform operator If you decide to use the express installation method, the IBM Cloud Pak for Data platform operator will be installed in this project. • IBM Cloud Pak for Data service operators If you decide to use the express installation method, the service operators will be installed in this project.
<p>cpd-operators</p> <p>(Recommended name, used in subsequent commands)</p>	<p>Required for specialized installations.</p> <p>cpd-operators is the recommended name and is used in various installation commands.</p> <p>In a specialized installation, the IBM Cloud Pak foundational services operators are installed in the ibm-common-services project and the Cloud Pak for Data operators are installed in a separate project (typically cpd-operators). Each project has a dedicated:</p> <ul style="list-style-type: none"> • Operator group, which specifies the OwnNamespace installation mode • NamespaceScope Operator, which allows the operators in the project to manage operators and service workloads in specific projects <p>In this way, you can specify different settings for the IBM Cloud Pak foundational services and for the Cloud Pak for Data operators.</p>
<p>cpd-instance</p> <p>(Sample name, used in subsequent commands)</p>	<p>At least one project is required for all installations.</p> <p>The project where the Cloud Pak for Data control plane is installed. (The Cloud Pak for Data control plane is installed in a <i>separate</i> project from the operators.)</p> <p>If you plan to install multiple instances of Cloud Pak for Data, you must create one project for each instance.</p> <p>cpd-instance is an example. You can use any project name. cpd-instance is used as a placeholder in various installation commands.</p> <p>Most services are installed in the same project as the Cloud Pak for Data control plane. Review the documentation for the services that you plan to deploy to determine whether you must create any additional projects. For details, see Services.</p>
<p>cpd-instance-tether</p> <p>(Sample name, used in subsequent commands)</p>	<p>Required or supported for some services.</p> <p>A few services can be installed in tethered projects. A <i>tethered project</i> is managed by the Cloud Pak for Data control plane but is otherwise isolated from Cloud Pak for Data and the other services that are installed in that project.</p> <p>cpd-instance-tether is an example. You can use any project name. cpd-instance-tether is used as a placeholder in various installation commands.</p> <p>For information on which services can be installed in tethered projects, see Multitenancy support.</p> <p>If you want to install a service in a tethered project, you must create the tethered project before you install the service.</p>

After you decide which projects you need to create, review the following information to ensure that you understand the security considerations that you need to take into account:

Project	Security considerations
---------	-------------------------

Project	Security considerations
ibm-common-services	<p>Operator group</p> <p>The ibm-common-services project uses the OwnNamespace installation mode. See the Procedure after this table for information on creating the operator group.</p> <p>Namespace scope</p> <p>The ibm-common-services project needs to be able to watch the project or projects where Cloud Pak for Data is deployed. IBM Cloud Pak foundational services includes the IBM NamespaceScope Operator, which allows the operators in the ibm-common-services project to manage operators and service workloads in specific projects.</p> <p>When you install Cloud Pak for Data or create a tethered namespace, you submit an operand request to grant permission to the operators in the ibm-common-services project to watch over the project (for example cpd-instance or cpd-instance-tether).</p> <p>By default, the IBM NamespaceScope Operator has <i>cluster permissions</i> so that role binding projections can be completed automatically. However, you can optionally remove the cluster permissions from the IBM NamespaceScope Operator and manually authorize the projections. For details, see Authorizing foundational services to perform operations on workloads in a namespace.</p> <p>SCCs</p> <p>Follow the guidance Security context constraints (SCCs) in the IBM Cloud Pak foundational services documentation.</p> <p>Express installations only</p> <p>The Cloud Pak for Data control plane and most Cloud Pak for Data services use the restricted SCC. However, a few services require custom SCCs. For details, see Custom security context constraints for services.</p>
cpd-operators	<p>Operator group</p> <p>The cpd-operators project uses the OwnNamespace installation mode. See the Procedure after this table for information on creating the operator group.</p> <p>Namespace scope</p> <p>The cpd-operators project needs to be able to watch the project or projects where Cloud Pak for Data is deployed. When you prepare your cluster, you create an operator subscription for the IBM NamespaceScope Operator in the cpd-operators project. The IBM NamespaceScope Operator allows the operators in the cpd-operators project to manage operators and service workloads in specific projects.</p> <p>When you install Cloud Pak for Data or create a tethered namespace, you submit an operand request to grant permission to the operators in the cpd-operators project to watch over the project (for example cpd-instance or cpd-instance-tether).</p> <p>By default, the IBM NamespaceScope Operator has <i>cluster permissions</i> so that role binding projections can be completed automatically. However, you can optionally remove the cluster permissions from the IBM NamespaceScope Operator and manually authorize the projections. For details, see Authorizing foundational services to perform operations on workloads in a namespace.</p> <p>SCCs</p> <p>The Cloud Pak for Data control plane and most Cloud Pak for Data services use the restricted SCC. However, a few services require custom SCCs. For details, see Custom security context constraints for services.</p>
cpd-instance	<p>Operator group</p> <p>Not applicable.</p> <p>Namespace scope</p> <p>Not applicable.</p> <p>SCCs</p> <p>The Cloud Pak for Data control plane and most Cloud Pak for Data services use the restricted SCC. However, a few services require custom SCCs. For details, see Custom security context constraints for services.</p>
cpd-instance-tether	<p>Operator group</p> <p>Not applicable.</p> <p>Namespace scope</p> <p>Not applicable.</p> <p>SCCs</p> <p>The Cloud Pak for Data control plane and most Cloud Pak for Data services use the restricted SCC. However, a few services require custom SCCs. For details, see Custom security context constraints for services.</p>

Procedure

To create the necessary projects for your environment:

1. Log in to your Red Hat OpenShift Container Platform as a cluster administrator:

```
oc login OpenShift:port
```

2. Create the appropriate projects for your environment.

Important: Review the guidance in **About this task** to ensure that you create the appropriate projects for your environment.

Project name	Command to create
ibm-common-services	<code>oc new-project ibm-common-services</code>
cpd-operators	<code>oc new-project cpd-operators</code>
cpd-instance	Remember: cpd-instance is a sample name. If you don't want to use this name, replace cpd-instance with the appropriate name for your environment. You must also replace this name in subsequent commands. <code>oc new-project cpd-instance</code>

Project name	Command to create
cpd-instance-tether	Remember: <code>cpd-instance-tether</code> is a sample name. If you don't want to use this name, replace <code>cpd-instance-tether</code> with the appropriate name for your environment. You must also replace this name in subsequent commands. <code>oc new-project cpd-instance-tether</code>

3. Create the appropriate operator groups based on the type of installation method you are using:

- Express installation

- If IBM Cloud Pak foundational services is not installed, create the operator group for the IBM Cloud Pak foundational services project. The following example uses the recommended project name (`ibm-common-services`):

```
cat <<EOF |oc apply -f -
apiVersion: operators.coreos.com/v1alpha2
kind: OperatorGroup
metadata:
  name: operatorgroup
  namespace: ibm-common-services
spec:
  targetNamespaces:
  - ibm-common-services
EOF
```

- Specialized installation

- If IBM Cloud Pak foundational services is not installed, create the operator group for the IBM Cloud Pak foundational services project. The following example uses the recommended project name (`ibm-common-services`):

```
cat <<EOF |oc apply -f -
apiVersion: operators.coreos.com/v1alpha2
kind: OperatorGroup
metadata:
  name: operatorgroup
  namespace: ibm-common-services
spec:
  targetNamespaces:
  - ibm-common-services
EOF
```

- Create the operator group for the IBM Cloud Pak for Data platform operator project. The following example uses the recommended project name (`cpd-operators`):

```
cat <<EOF |oc apply -f -
apiVersion: operators.coreos.com/v1alpha2
kind: OperatorGroup
metadata:
  name: operatorgroup
  namespace: cpd-operators
spec:
  targetNamespaces:
  - cpd-operators
EOF
```

Previous topic: [Setting up shared persistent storage](#)

Next topic: [Obtaining your IBM entitlement API key](#)

Obtaining your IBM entitlement API key

The IBM entitlement API key enables you to pull software images from the IBM® Entitled Registry, either for installation or for mirroring.

All Cloud Pak for Data images are accessible from the IBM Entitled Registry. You must decide whether you will install the images directly from the IBM Entitled Registry or whether you will mirror the images to your local container registry.

IBM entitlement API key

You must have your IBM entitlement API key to access images in the IBM Entitled Registry.

After you purchase Cloud Pak for Data, an entitlement API key for the software is associated with your My IBM account. You need this key to complete the Cloud Pak for Data installation. To obtain the entitlement key, complete the following steps:

- Log in to [Container software library on My IBM](#) with the IBM ID and password that are associated with the entitled software.
- On the Get entitlement key tab, select Copy key to copy the entitlement key to the clipboard.
- Save the API key in a text file.

Previous topic: [Creating projects \(namespaces\) on Red Hat OpenShift Container Platform](#)

Next topic: [Mirroring images to your private container registry](#)

Mirroring images to your private container registry

IBM® Cloud Pak for Data images are accessible from the IBM Entitled Registry. In most situations, it is strongly recommended that you mirror the necessary software images from the IBM Entitled Registry to a private container registry.

Important: You must mirror the necessary images to your private container registry in the following situations:

- Your cluster is air-gapped (also called an offline or disconnected cluster)
- Your cluster uses an *allowlist* to permit direct access by specific sites and the allowlist does not include the IBM Entitled Registry
- Your cluster uses a *blocklist* to prevent direct access by specific sites and the blocklist includes the IBM Entitled Registry

The only situation in which you might consider pulling images directly from the IBM Entitled Registry is when your cluster is not air-gapped, your network is extremely reliable, and latency is not a concern. However, for predictable and reliable performance, you should mirror the images to a private container registry.

Setting up a private container registry

For details about which container registries you can use with Red Hat® OpenShift® Container Platform, see [Registry options](#) in the Red Hat OpenShift Container Platform documentation.

Your private container registry must meet the following requirements:

- Support the [Docker Image Manifest Version 2, Schema 2](#)
- Allow path separators in image names
- Be in close proximity to your Red Hat OpenShift Container Platform cluster

In addition, the private container registry must be accessible from all of the nodes in the cluster and all of the nodes must have permission to push to and pull from the private container registry.

Restriction: You cannot use the integrated OpenShift Container Platform registry. It does not support multi-architecture images and is not compliant with the Docker Image Manifest Version 2, Schema 2.

As part of your private container registry set up, ensure that you follow the guidelines for configuring the registry in [Image configuration resources](#) in the Red Hat OpenShift Container Platform documentation.

Image prefixes

IBM Cloud Pak software uses the following prefixes to identify images:

Tag	Used for
<code>cp.icr.io/cp</code>	Images that are pulled from the IBM Entitled Registry that require an entitlement key to download. Most of the IBM Cloud Pak for Data software uses this tag.
<code>icr.io/cpopen</code>	Publicly available images that are provided by IBM and that don't require an entitlement key to download. The IBM Cloud Pak for Data operators use this tag.
<code>quay.io/openshift</code>	IBM open source images that are available on quay.io . The IBM Cloud Pak® foundational services software uses this tag.

Ensure that:

- Your private container registry is configured to allow these prefixes
- The credentials that you will use to push images to the private container registry can push images with these prefixes

Methods for mirroring images

There are several ways that you can mirror images from the IBM Entitled Registry to your private container registry. Choose the most appropriate method for your environment:

Method	Description	Connected clusters	Air-gapped clusters
Portable compute device	<p>Example: A laptop that you can move behind your firewall is a portable compute device.</p> <p>High-level process using a portable compute device:</p> <ol style="list-style-type: none"> 1. Create an intermediary container registry on a portable compute device that is connected to the internet. 2. From the portable compute device, mirror images from the IBM Entitled Registry to the intermediary container registry. 3. Bring the device behind your firewall and mirror the images from the intermediary container registry to the private container registry that is accessible from the Red Hat OpenShift Container Platform cluster. <p>For the full process, see Mirroring images with an intermediary container registry.</p>		✓
File transfer	<p>Example: You can either use a portable storage device, such as a USB drive, or use <code>scp</code> or <code>sftp</code> to move images behind your firewall.</p> <p>High-level process using a file transfer:</p> <ol style="list-style-type: none"> 1. Create an intermediary container registry. If you are using a portable storage device, create the intermediary container registry on the storage device. 2. From a workstation that can connect to the internet and the intermediary container registry, mirror the images from the IBM Entitled Registry to the intermediary container registry. 3. Move the files and or the storage device behind your firewall. 4. Set up a workstation behind the firewall to mirror the images to the private container registry that is accessible from the Red Hat OpenShift Container Platform cluster. <p>For the full process, see Mirroring images with an intermediary container registry.</p>		✓

Method	Description	Connected clusters	Air-gapped clusters
Bastion node	<p>Example: A server with access to both the public internet and the private container registry that is accessible from the Red Hat OpenShift Container Platform cluster.</p> <p>High-level process using a bastion node:</p> <ol style="list-style-type: none"> 1. From the bastion node, replicate the images from the IBM Entitled Registry to the private container registry that is accessible from the Red Hat OpenShift Container Platform cluster. <p>For the full process, see Mirroring images with a bastion node.</p>	✓	✓

Mirroring images to a private container registry

Complete the appropriate task for your environment:

- [Mirroring images with a bastion node](#)
If your Red Hat OpenShift Container Platform cluster is air-gapped, you must mirror the software images that you need to a private container registry that is accessible from the cluster. You can use a bastion node that is connected to the internet and to the private container registry to mirror the images from the IBM Entitled Registry.
- [Mirroring images with an intermediary container registry](#)
If your Red Hat OpenShift Container Platform cluster is air-gapped, you must mirror the software images that you need to a private container registry that is accessible from the cluster. You can use an intermediary container registry to mirror the images from the IBM Entitled Registry to a private container registry.

Previous topic: [Obtaining your IBM entitlement API key](#)

Next topic: [Configuring your cluster to pull Cloud Pak for Data images](#)

Mirroring images with a bastion node

If your Red Hat® OpenShift® Container Platform cluster is air-gapped, you must mirror the software images that you need to a private container registry that is accessible from the cluster. You can use a bastion node that is connected to the internet and to the private container registry to mirror the images from the IBM® Entitled Registry.

To obtain fixes, security updates, and new functionality, you must install the latest refresh of the software images, such as the control plane, shared cluster components, and services. The steps in this task assume that you are mirroring the latest software images. If you need older versions of the software, see [Operator and operand versions](#).

Important: Use a Linux x86-64 system with Red Hat Enterprise Linux® to mirror the images.

The system must be able to access the following sites:

- [Red Hat Quay.io \(https://quay.io:443\)](https://quay.io:443)
- [GitHub \(https://github.com\)](https://github.com)
If your company does not permit access to GitHub, contact IBM Support for assistance.
- [IBM Entitled Registry \(http://icr.io:443\)](http://icr.io:443)
To validate that you can connect, run the following command:

```
curl -v https://icr.io
```

The command should return the following message:

```
* Connected to icr.io (169.60.98.86) port 443 (#0)
```

Procedure

Complete the following tasks to mirror the images to your container registry:

- [1. Downloading and installing the software needed to mirror images](#)
- [2. Setting up your environment to download CASE packages](#)
- [3. Downloading the Cloud Pak for Data CASE package](#)
- [4. Configuring credentials for mirroring images](#)
- [5. Downloading shared cluster component CASE packages](#)
- [6. Downloading service CASE packages](#)
- [7. Mirroring the images to the private container registry](#)

1. Downloading and installing the software needed to mirror images

To use a connected bastion node, you must install the following software on the system:

Prerequisite	Purpose
OpenShift CLI	Required to interact with your Red Hat OpenShift Container Platform cluster.
IBM Cloud Pak® CLI (cloudctl)	Required to download images from the IBM Entitled Registry.
httpd-tools	Required to run the IBM Cloud Pak CLI (cloudctl).
skopeo Version 1.2.0 or later	Required to run the IBM Cloud Pak CLI (cloudctl).

To install the prerequisite software:

1. To install the OpenShift CLI, see [Getting started with the OpenShift CLI](#) in the Red Hat documentation.

2. To install the IBM Cloud Pak CLI (cloudctl):
 - a. Download the [cloudctl software](#) from the IBM/cloud-pak-cli repository on GitHub. Ensure that you download the appropriate package for your workstation:

```
cloudctl-operating-system-architecture.tar.gz
```

- b. Extract the contents of the archive file:

```
tar -xzf archive-name
```

- c. Change to the directory where you extracted the file and make the file executable:

```
chmod 775 cloudctl-architecture
```

- d. Move the file to the /usr/local/bin directory:

```
mv cloudctl-architecture /usr/local/bin/cloudctl
```

- e. Confirm that the IBM Cloud Pak CLI (cloudctl) is installed:

```
cloudctl --help
```

Tip: Additional guidance for validating the archive file is available in the [IBM/cloud-pak-cli repository](#).

3. To install `httpd-tools`, run the following command:

```
yum install httpd-tools
```

4. To install `skopeo`, see [Installing from packages](#) in the [skopeo repository](#) on GitHub.

2. Setting up your environment to download CASE packages

A Container Application Software for Enterprises (CASE) package is an archive file that describes a containerized component of Cloud Pak for Data.

There are CASE packages for:

- IBM Cloud Pak foundational services
- IBM Cloud Pak for Data control plane
- Each IBM Cloud Pak for Data service
- Software dependencies for the control plane and services

Each CASE package includes:

- Metadata about the component
- An inventory of the container images that are required to deploy the component
- References to any software dependencies
- The scripts needed to mirror the images to a private container registry

Before you can mirror the images a private container registry, you must download the CASE packages for the software that you plan to install.

To set up your environment:

1. Identify or create the directory or directories where you want to store the CASE packages on the system.

Important: Keep the following requirements in mind:

- You must have sufficient storage in the directory.
 - You must use a persistent directory. Using a persistent directory prevents you from transferring files more than once.
- Additionally, if you use a persistent directory, you can run the mirror process multiple times or on a schedule.

Recommended directory name	Notes
<code>\$HOME/offline/cpd</code>	Primary directory for most Cloud Pak for Data software. This directory is required for all installations. To create this directory, run the following command: <pre>mkdir -p \$HOME/offline/cpd</pre>
<code>\$HOME/offline/assistant</code>	Directory for Watson™ Assistant software. This directory is required only if you plan to install the Watson Assistant service. To create this directory, run the following command: <pre>mkdir -p \$HOME/offline/assistant</pre>

Note: If you have multiple IBM Cloud Paks on your cluster, use a separate directory for the Cloud Pak for Data CASE packages (the packages for the Cloud Pak for Data platform and services).

When you run the command to mirror the images, the command mirrors any updated images in the directory. Using a separate directory ensures that you mirror only the images for Cloud Pak for Data.

2. Set the following environment variables:

Variable	Notes
<code>CASE_REPO_PATH</code>	Required for all installations. This environment variable points to the repository where the CASE packages are hosted. To set this environment variable, run: <pre>export CASE_REPO_PATH=https://github.com/IBM/cloud-pak/raw/master/repo/case</pre>

Variable	Notes
OFFLINEDIR	<p>Required for all installations. This is the directory that you created to store most of the IBM Cloud Paks CASE packages.</p> <p>To set this environment variable, run:</p> <pre>export OFFLINEDIR=\$HOME/offline/cpd</pre> <p>Replace <code>\$HOME/offline/cpd</code> with the appropriate value for your environment.</p>
OFFLINEDIR_WA	<p>Required only if you plan to install the Watson Assistant service. This is the directory that you created to store the Watson Assistant software.</p> <p>To set this environment variable, run:</p> <pre>export OFFLINEDIR_WA=\$HOME/offline/assistant</pre> <p>Replace <code>\$HOME/offline/assistant</code> with the appropriate value for your environment.</p>

3. Downloading the Cloud Pak for Data CASE package

1. Run the following command to download the IBM Cloud Pak for Data platform operator package:

```
cloudctl case save \
  --case ${CASE_REPO_PATH}/ibm-cp-datacore-2.0.5.tgz \
  --outputdir ${OFFLINEDIR} \
  --no-dependency
```

4. Configuring credentials for mirroring images

The IBM Cloud Pak CLI (cloudctl) includes an action called `configure-creds-airgap`. Run the appropriate commands to store the credentials that you will need to mirror images to the private container registry. The command stores the credentials to the following file on your local file system: `$HOME/.airgap/secrets`.

To configure the credentials that you need to mirror software images:

1. Store the IBM Entitled Registry credentials by running the following command:

```
cloudctl case launch \
  --case ${OFFLINEDIR}/ibm-cp-datacore-2.0.5.tgz \
  --inventory cpdPlatformOperator \
  --action configure-creds-airgap \
  --args "--registry cp.icr.io --user cp --pass entitlement-key --inputDir ${OFFLINEDIR}"
```

Replace `entitlement-key` with your entitlement key. For details, see [IBM entitlement API key](#).

2. Store the private container registry credentials:

- a. Work with your private container registry administrator to identify the values for the following parameters:

PRIVATE_REGISTRY_USER

The username of a user who has the required privileges to *push* images to the private container registry.

PRIVATE_REGISTRY_PASSWORD

The password of the user who has the required privileges to *push* images to the private container registry.

PRIVATE_REGISTRY

The location of the private container registry.

- b. Set environment variables for the parameters:

```
export PRIVATE_REGISTRY_USER=username
export PRIVATE_REGISTRY_PASSWORD=password
export PRIVATE_REGISTRY=private-registry-location
```

- c. Run the following command to store the credentials:

```
cloudctl case launch \
  --case ${OFFLINEDIR}/ibm-cp-datacore-2.0.5.tgz \
  --inventory cpdPlatformOperator \
  --action configure-creds-airgap \
  --args "--registry ${PRIVATE_REGISTRY} --user ${PRIVATE_REGISTRY_USER} --pass ${PRIVATE_REGISTRY_PASSWORD}"
```

5. Downloading shared cluster component CASE packages

Determine which [shared cluster components](#) you need to install on your cluster and download the appropriate CASE packages.

Shared cluster component	CASE download command
IBM Cloud Pak foundational services Download this package if IBM Cloud Pak foundational services is not installed on the cluster.	<pre>cloudctl case save \ --case \${CASE_REPO_PATH}/ibm-cp-common- services-1.6.0.tgz \ --outputdir \${OFFLINEDIR}</pre>
Scheduling service Download this package if you plan to install Watson Machine Learning Accelerator or if you want to use the quota enforcement feature.	<pre>cloudctl case save \ --case \${CASE_REPO_PATH}/ibm-cpd-scheduling- 1.2.3.tgz \ --outputdir \${OFFLINEDIR}</pre>

6. Downloading service CASE packages

Decide which [services](#) you plan to install on your cluster and download the appropriate CASE packages.

- Anaconda Repository for IBM Cloud Pak for Data

Not applicable. For details, see [Installing Anaconda Repository for IBM Cloud Pak for Data](#).

- Analytics Engine Powered by Apache Spark

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-analyticsengine-4.0.2.tgz \  
--outputdir ${OFFLINEDIR}
```

- Cognos Analytics

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-cognos-analytics-prod-4.0.4.tgz \  
--outputdir ${OFFLINEDIR}
```

- Cognos Dashboards

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-cde-2.0.2.tgz \  
--outputdir ${OFFLINEDIR}
```

- Data Refinery

The Data Refinery CASE package is automatically downloaded when you download the CASE package for either Watson Knowledge Catalog or Watson Studio.

- Data Virtualization

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-dv-case-1.7.2.tgz \  
--outputdir ${OFFLINEDIR}
```

- DataStage

Download the appropriate package based on your license:

DataStage Enterprise

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-datastage-enterprise-4.0.3.tgz \  
--outputdir ${OFFLINEDIR}
```

DataStage Enterprise Plus

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-datastage-4.0.3.tgz \  
--outputdir ${OFFLINEDIR}
```

- Db2

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-db2oltp-4.0.3.tgz \  
--outputdir ${OFFLINEDIR}
```

- Db2 Big SQL

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-bigsql-case-7.2.2.tgz \  
--outputdir ${OFFLINEDIR}
```

- Db2 Data Gate

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-datagate-prod-4.0.2.tgz \  
--outputdir ${OFFLINEDIR}
```

- Db2 Data Management Console

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-dmc-4.0.2.tgz \  
--outputdir ${OFFLINEDIR}
```

- Db2 Event Store

Not applicable. Contact IBM Software support if you plan to install this service.

- Db2 Warehouse

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-db2wh-4.0.3.tgz \  
--outputdir ${OFFLINEDIR}
```



```
--outputdir ${OFFLINEDIR}
```

-

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-dods-4.0.2.tgz \  
--outputdir ${OFFLINEDIR}
```

-

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-cpd-edb-4.0.2.tgz \  
--outputdir ${OFFLINEDIR}
```

-

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-hadoop-4.0.2.tgz \  
--outputdir ${OFFLINEDIR}
```

-

Not applicable. For details, see the [Financial Services Workbench documentation](#).

-

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-mdm-1.0.115.tgz \  
--outputdir ${OFFLINEDIR}
```

-

Download both of the following CASE packages:

- ```
cloudctl case save \
--case ${CASE_REPO_PATH}/ibm-informix-operator-4.0.2.tgz \
--outputdir ${OFFLINEDIR}
```
- ```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-informix-cp4d-operator-4.0.2.tgz \  
--outputdir ${OFFLINEDIR}
```

-

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-cpd-mongodb-4.0.2.tgz \  
--outputdir ${OFFLINEDIR}
```

-

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-openpages-2.0.2+20210910.011848.82030838.tgz \  
--outputdir ${OFFLINEDIR}
```

If you want OpenPages to automatically provision a Db2 database, you must also download the following package :

Db2 as a service

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-db2aaservice-4.0.3.tgz \  
--outputdir ${OFFLINEDIR}
```

-

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-planning-analytics-4.0.2.tgz \  
--outputdir ${OFFLINEDIR}
```

-

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-productmaster-1.0.1.tgz \  
--outputdir ${OFFLINEDIR}
```

-

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-rstudio-1.0.2.tgz \  
--outputdir ${OFFLINEDIR}
```

-

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-spss-1.0.2.tgz \  
--outputdir ${OFFLINEDIR}
```

- >
Virtual Data Pipeline

Not applicable. For details, see [Installing Virtual Data Pipeline for Cloud Pak for Data](#).

- >
Voice Gateway

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-voice-gateway-1.0.3.tgz \  
--outputdir ${OFFLINEDIR}
```

- >
Watson Assistant

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-watson-assistant-4.0.2.tgz \  
--outputdir ${OFFLINEDIR_WA}
```

- >
Watson Assistant for Voice Interaction

Watson Assistant for Voice Interaction is comprised of the following services:

- o Voice Gateway
- o Watson Assistant
- o Watson Speech to Text
- o Watson Text to Speech

Download the appropriate services for your use case.

- >
Watson Discovery

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-watson-discovery-4.0.2.tgz \  
--outputdir ${OFFLINEDIR}
```

- >
Watson Knowledge Catalog

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-wkc-4.0.2.tgz \  
--outputdir ${OFFLINEDIR}
```

- >
Watson Knowledge Studio

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-watson-ks-4.0.2.tgz \  
--outputdir ${OFFLINEDIR}
```

- >
Watson Machine Learning

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-wml-cpd-4.0.3.tgz \  
--outputdir ${OFFLINEDIR}
```

- >
Watson Machine Learning Accelerator

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-wml-accelerator-2.3.2.tgz \  
--outputdir ${OFFLINEDIR}
```

- >
Watson OpenScale

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-watson-openscale-2.2.0.tgz \  
--outputdir ${OFFLINEDIR}
```

- >
Watson Speech to Text

The same package is used for the Watson Speech to Text service and the Watson Text to Speech service. You only need to download the package once.

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-watson-speech-4.0.0.tgz \  
--outputdir ${OFFLINEDIR}
```

- >
Watson Studio

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-ws1-2.0.2.tgz \  
--outputdir ${OFFLINEDIR}
```

- >
Watson Studio Runtimes

```
cloudctl case save \
--case ${CASE_REPO_PATH}/ibm-wsl-runtimes-1.0.2.tgz \
--outputdir ${OFFLINEDIR}
```

- >
Watson Text to Speech

The same package is used for the Watson Speech to Text service and the Watson Text to Speech service. You only need to download the package once.

```
cloudctl case save \
--case ${CASE_REPO_PATH}/ibm-watson-speech-4.0.0.tgz \
--outputdir ${OFFLINEDIR}
```

7. Mirroring the images to the private container registry

To mirror the images:

1. If you did not purchase a license for EDB Postgres Enterprise, run the following command to remove any `edb-postgres-advanced` images from the list of images to mirror. If you did purchase a license for EDB Postgres Enterprise, skip this step.

```
sed -i -e '/edb-postgres-advanced/d' ${OFFLINEDIR}/ibm-cloud-native-postgresql-4.0.*-images.csv
```

2. Set the following environment variable to ensure that the IBM Cloud Pak CLI (cloudctl) uses `skopeo` rather than `oc mirror`. The `oc mirror` command can have trouble pulling images from binary repositories.

```
export USE_SKOPEO=true
```

3. Use the Cloud Pak for Data platform CASE package to mirror all of the images to the private container registry:

```
cloudctl case launch \
--case ${OFFLINEDIR}/ibm-cp-datacore-2.0.5.tgz \
--inventory cpdPlatformOperator \
--action mirror-images \
--args "--registry ${PRIVATE_REGISTRY} --user ${PRIVATE_REGISTRY_USER} --pass ${PRIVATE_REGISTRY_PASSWORD} --inputDir ${OFFLINEDIR}"
```

Remember: The `mirror-images` command mirrors any changed images in the `${OFFLINEDIR}`.

4. If you downloaded the Watson Assistant CASE package, complete the appropriate steps for your environment:
 - a. If you did not purchase a license for EDB Postgres Enterprise, run the following command to remove any `edb-postgres-advanced` images from the list of images to mirror. If you did purchase a license for EDB Postgres Enterprise, skip this step.

```
sed -i -e '/edb-postgres-advanced/d' ${OFFLINEDIR_WA}/ibm-cloud-native-postgresql-4.0.*-images.csv
```

- b. Run the following command to mirror the Watson Assistant images to the private container registry:

```
cloudctl case launch \
--case ${OFFLINEDIR_WA}/ibm-watson-assistant-4.0.2.tgz \
--inventory assistantOperator \
--action mirror-images \
--args "--registry ${PRIVATE_REGISTRY} --user ${PRIVATE_REGISTRY_USER} --pass ${PRIVATE_REGISTRY_PASSWORD} --inputDir ${OFFLINEDIR_WA}"
```

Mirroring images with an intermediary container registry

If your Red Hat® OpenShift® Container Platform cluster is air-gapped, you must mirror the software images that you need to a private container registry that is accessible from the cluster. You can use an intermediary container registry to mirror the images from the IBM® Entitled Registry to a private container registry.

To obtain fixes, security updates, and new functionality, you must install the latest refresh of the software images, such as the control plane, shared cluster components, and services. The steps in this task assume that you are mirroring the latest software images. If you need older versions of the software, see [Operator and operand versions](#).

Important: Use a Linux x86-64 system with Red Hat Enterprise Linux® to mirror the images.

The system must be able to access the following sites:

- [Red Hat Quay.io \(https://quay.io:443\)](https://quay.io:443)
- [GitHub \(https://github.com\)](https://github.com)
If your company does not permit access to GitHub, contact IBM Support for assistance.

- [IBM Entitled Registry \(http://icr.io:443\)](http://icr.io:443)
To validate that you can connect, run the following command:

```
curl -v https://icr.io
```

The command should return the following message:

```
* Connected to icr.io (169.60.98.86) port 443 (#0)
```

Procedure

Complete the following tasks to mirror the images to your private container registry:

- [1. Installing the software needed to mirror images](#)
- [2. Setting up your environment to download CASE packages](#)
- [3. Downloading the Cloud Pak for Data CASE package](#)
- [4. Configuring credentials for mirroring images](#)
- [5. Setting up an intermediary container registry](#)

- [6. Downloading shared cluster component CASE packages](#)
- [7. Downloading service CASE packages](#)
- [8. Mirroring the images to the intermediary container registry](#)
- [9. Setting up a workstation to serve images](#)
- [10. Mirroring images to the private container registry](#)

1. Installing the software needed to mirror images

To use an intermediary container registry, you must install the following software on the system:

Prerequisite	Purpose
One of the following container client tools: <ul style="list-style-type: none"> • Docker • Podman 	Required to set up an intermediary container registry for the images that you download.
OpenSSL Version 1.1.1 or later	Required to generate TLS certificates and keys for use with the intermediary container registry.
OpenShift CLI	Required to interact with your Red Hat OpenShift Container Platform cluster.
IBM Cloud Pak® CLI (cloudctl)	Required to download images from the IBM Entitled Registry.
<code>httpd-tools</code>	Required to run the IBM Cloud Pak CLI (cloudctl).
<code>skopeo</code> Version 1.2.0 or later	Required to run the IBM Cloud Pak CLI (cloudctl).

To install the prerequisite software:

1. Install the container client tool of your choice:

- To install Docker, run the following commands:

```
yum check-update
yum install docker
```

- To install Podman, see the [Podman installation instructions](#) on the Podman site.

2. To install OpenSSL, see [Downloads](#) on the OpenSSL site.

3. To install the OpenShift CLI, see [Getting started with the OpenShift CLI](#) in the Red Hat documentation.

4. To install the IBM Cloud Pak CLI (cloudctl):

- Download the [cloudctl software](#) from the IBM/cloud-pak-cli repository on GitHub. Ensure that you download the appropriate package for your workstation:

```
cloudctl-operating-system-architecture.tar.gz
```

- Extract the contents of the archive file:

```
tar -xzf archive-name
```

- Change to the directory where you extracted the file and make the file executable:

```
chmod 775 cloudctl-architecture
```

- Move the file to the /usr/local/bin directory:

```
mv cloudctl-architecture /usr/local/bin/cloudctl
```

- Confirm that the IBM Cloud Pak CLI (cloudctl) is installed:

```
cloudctl --help
```

Tip: Additional guidance for validating the archive file is available in the [IBM/cloud-pak-cli repository](#).

5. To install `httpd-tools`, run the following command:

```
yum install httpd-tools
```

6. To install `skopeo`, see [Installing from packages](#) in the [skopeo repository](#) on GitHub.

2. Setting up your environment to download CASE packages

A Container Application Software for Enterprises (CASE) package is an archive file that describes a containerized component of Cloud Pak for Data.

There are CASE packages for:

- IBM Cloud Pak foundational services
- IBM Cloud Pak for Data control plane
- Each IBM Cloud Pak for Data service
- Software dependencies for the control plane and services

Each CASE package includes:

- Metadata about the component
- An inventory of the container images that are required to deploy the component
- References to any software dependencies
- The scripts needed to mirror the images to a private container registry

Before you can mirror the images a private container registry, you must download the CASE packages for the software that you plan to install.

To set up your environment:

1. Identify or create the directory or directories where you want to store the CASE packages on the system. Important: Keep the following requirements in mind:

- You must have sufficient storage in the directory. You must have sufficient storage for both the CASE packages and the software images that you need to mirror.
- You must use a persistent directory. Using a persistent directory prevents you from transferring files more than once. Additionally, if you use a persistent directory, you can run the mirror process multiple times or on a schedule.

Recommended directory name	Notes
\$HOME/offline/cpd	<p>Primary directory for most Cloud Pak for Data software. This directory is required for all installations.</p> <p>To create this directory, run the following command:</p> <pre>mkdir -p \$HOME/offline/cpd</pre>
\$HOME/offline/assistant	<p>Directory for Watson™ Assistant software. This directory is required only if you plan to install the Watson Assistant service.</p> <p>To create this directory, run the following command:</p> <pre>mkdir -p \$HOME/offline/assistant</pre>

Note: If you have multiple IBM Cloud Paks on your cluster, use a separate directory for the Cloud Pak for Data CASE packages (the packages for the Cloud Pak for Data platform and services).

When you run the command to mirror the images, the command mirrors any updated images in the directory. Using a separate directory ensures that you mirror only the images for Cloud Pak for Data.

- Set the following environment variables:

Variable	Notes
CASE_REPO_PATH	<p>Required for all installations. This environment variable points to the repository where the CASE packages are hosted.</p> <p>To set this environment variable, run:</p> <pre>export CASE_REPO_PATH=https://github.com/IBM/cloud-pak/raw/master/repo/case</pre>
OFFLINEDIR	<p>Required for all installations. This is the directory that you created to store most of the IBM Cloud Paks CASE packages.</p> <p>To set this environment variable, run:</p> <pre>export OFFLINEDIR=\$HOME/offline/cpd</pre> <p>Replace <code>\$HOME/offline/cpd</code> with the appropriate value for your environment.</p>
OFFLINEDIR_WA	<p>Required only if you plan to install the Watson Assistant service. This is the directory that you created to store the Watson Assistant software.</p> <p>To set this environment variable, run:</p> <pre>export OFFLINEDIR_WA=\$HOME/offline/assistant</pre> <p>Replace <code>\$HOME/offline/assistant</code> with the appropriate value for your environment.</p>

3. Downloading the Cloud Pak for Data CASE package

- Run the following command to download the IBM Cloud Pak for Data platform operator package:

```
cloudctl case save \
--case ${CASE_REPO_PATH}/ibm-cp-datacore-2.0.5.tgz \
--outputdir ${OFFLINEDIR} \
--no-dependency
```

4. Configuring credentials for mirroring images

The IBM Cloud Pak CLI (cloudctl) includes an action called `configure-cred-airgap`. Run the appropriate commands to store the credentials that you will need to mirror images to the private container registry. The command stores the credentials to the following file on your local file system: `$HOME/.airgap/secrets`.

To configure the credentials that you need to mirror software images:

- Store the IBM Entitled Registry credentials by running the following command:

```
cloudctl case launch \
--case ${OFFLINEDIR}/ibm-cp-datacore-2.0.5.tgz \
--inventory cpdPlatformOperator \
--action configure-creds-airgap \
--args "--registry cp.icr.io --user cp --pass entitlement-key --inputDir ${OFFLINEDIR}"
```

Replace `entitlement-key` with your entitlement key. For details, see [IBM entitlement API key](#).

- Store the intermediary container registry credentials:
 - Determine the values that you want to use for the following parameters:

PORTABLE_REGISTRY_USER

The username that you want to use to authenticate to the intermediary container registry.

PORTABLE_REGISTRY_PASS

The password that you want to use to authenticate to the intermediary container registry.

PORTABLE_REGISTRY_HOST

The location of the intermediary container registry on the local host.

PORTABLE_REGISTRY_PORT

Pick a port that is not currently in use. To avoid conflicts with system ports, choose a port greater than 1024.

b. Set the environment variable for the parameters:

```
export PORTABLE_REGISTRY_USER=username
export PORTABLE_REGISTRY_PASSWORD=password
export PORTABLE_REGISTRY_HOST=localhost
export PORTABLE_REGISTRY_PORT=15000
PORTABLE_REGISTRY="${PORTABLE_REGISTRY_HOST}:${PORTABLE_REGISTRY_PORT}"
export PORTABLE_REGISTRY
```

c. Run the following command to store the credentials:

```
cloudctl case launch \
--case ${OFFLINEDIR}/ibm-cp-datacore-2.0.5.tgz \
--inventory cpdPlatformOperator \
--action configure-creds-airgap \
--args "--registry ${PORTABLE_REGISTRY} --user ${PORTABLE_REGISTRY_USER} --pass ${PORTABLE_REGISTRY_PASSWORD}"
```

5. Setting up an intermediary container registry

You must create an intermediary container registry where you can mirror the images before you can mirror them to the private container registry.

To create an intermediary container registry:

1. To initialize the intermediary container registry, run the following command:

```
cloudctl case launch \
--case ${OFFLINEDIR}/ibm-cp-datacore-2.0.5.tgz \
--inventory cpdPlatformOperator \
--action init-registry \
--args "--registry ${PORTABLE_REGISTRY_HOST} --user ${PORTABLE_REGISTRY_USER} --pass ${PORTABLE_REGISTRY_PASSWORD} --dir ${OFFLINEDIR}/imageregistry"
```

2. Set the following environment variable to ensure that the IBM Cloud Pak CLI (cloudctl) uses `skopeo` rather than `oc mirror`. The `oc mirror` command can have trouble pulling images from binary repositories.

```
export USE_SKOPEO=true
```

3. To start the intermediary container registry, run the following command:

```
cloudctl case launch \
--case ${OFFLINEDIR}/ibm-cp-datacore-2.0.5.tgz \
--inventory cpdPlatformOperator \
--action start-registry \
--args "--port ${PORTABLE_REGISTRY_PORT} --dir ${OFFLINEDIR}/imageregistry --image docker.io/library/registry:2"
```

4. To verify that the intermediary container registry is running, run the following commands:

Container client	Commands
Docker	List the containers by running: <code>docker ps</code>
Podman	List the containers by running: <code>podman ps</code>

6. Downloading shared cluster component CASE packages

Determine which [shared cluster components](#) you need to install on your cluster and download the appropriate CASE packages.

Shared cluster component	CASE download command
IBM Cloud Pak foundational services Download this package if IBM Cloud Pak foundational services is not installed on the cluster.	<code>cloudctl case save \</code> <code>--case \${CASE_REPO_PATH}/ibm-cp-common-</code> <code>services-1.6.0.tgz \</code> <code>--outputdir \${OFFLINEDIR}</code>
Scheduling service Download this package if you plan to install Watson Machine Learning Accelerator or if you want to use the quota enforcement feature.	<code>cloudctl case save \</code> <code>--case \${CASE_REPO_PATH}/ibm-cpd-scheduling-</code> <code>1.2.3.tgz \</code> <code>--outputdir \${OFFLINEDIR}</code>

7. Downloading service CASE packages

Decide which [services](#) you plan to install on your cluster and download the appropriate CASE packages.

- >

 Anaconda Repository for IBM Cloud Pak for Data
 Not applicable. For details, see [Installing Anaconda Repository for IBM Cloud Pak for Data](#).
- >

 Analytics Engine Powered by Apache Spark

```
cloudctl case save \
--case ${CASE_REPO_PATH}/ibm-analyticsengine-4.0.2.tgz \
--outputdir ${OFFLINEDIR}
```
- >

 Cognos Analytics

```
cloudctl case save \
--case ${CASE_REPO_PATH}/ibm-cognos-analytics-prod-4.0.4.tgz \
```

```
--outputdir ${OFFLINEDIR}
```

- [Cognos Dashboards](#)

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-cde-2.0.2.tgz \  
--outputdir ${OFFLINEDIR}
```

- [Data Refinery](#)

The Data Refinery CASE package is automatically downloaded when you download the CASE package for either Watson Knowledge Catalog or Watson Studio.

- [Data Virtualization](#)

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-dv-case-1.7.2.tgz \  
--outputdir ${OFFLINEDIR}
```

- [DataStage](#)

Download the appropriate package based on your license:

DataStage Enterprise

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-datastage-enterprise-4.0.3.tgz \  
--outputdir ${OFFLINEDIR}
```

DataStage Enterprise Plus

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-datastage-4.0.3.tgz \  
--outputdir ${OFFLINEDIR}
```

- [Db2](#)

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-db2oltp-4.0.3.tgz \  
--outputdir ${OFFLINEDIR}
```

- [Db2 Big SQL](#)

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-bigsql-case-7.2.2.tgz \  
--outputdir ${OFFLINEDIR}
```

- [Db2 Data Gate](#)

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-datagate-prod-4.0.2.tgz \  
--outputdir ${OFFLINEDIR}
```

- [Db2 Data Management Console](#)

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-dmc-4.0.2.tgz \  
--outputdir ${OFFLINEDIR}
```

- [Db2 Event Store](#)

Not applicable. Contact IBM Software support if you plan to install this service.

- [Db2 Warehouse](#)

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-db2wh-4.0.3.tgz \  
--outputdir ${OFFLINEDIR}
```

- [Decision Optimization](#)

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-dods-4.0.2.tgz \  
--outputdir ${OFFLINEDIR}
```

- [EDB Postgres](#)

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-cpd-edb-4.0.2.tgz \  
--outputdir ${OFFLINEDIR}
```

- [Execution Engine for Apache Hadoop](#)

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-hadoop-4.0.2.tgz \  
--outputdir ${OFFLINEDIR}
```

- [Financial Services Workbench](#)

Not applicable. For details, see the [Financial Services Workbench documentation](#).

- [IBM Match 360 with Watson](#)

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-mdm-1.0.115.tgz \  
--outputdir ${OFFLINEDIR}
```

- [Informix](#)

Download both of the following CASE packages:

- ```
cloudctl case save \
--case ${CASE_REPO_PATH}/ibm-informix-operator-4.0.2.tgz \
--outputdir ${OFFLINEDIR}
```
- ```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-informix-cp4d-operator-4.0.2.tgz \  
--outputdir ${OFFLINEDIR}
```

- [MongoDB](#)

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-cpd-mongodb-4.0.2.tgz \  
--outputdir ${OFFLINEDIR}
```

- [OpenPages](#)

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-openpages-2.0.2+20210910.011848.82030838.tgz \  
--outputdir ${OFFLINEDIR}
```

If you want OpenPages to automatically provision a Db2 database, you must also download the following package :

Db2 as a service

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-db2aaservice-4.0.3.tgz \  
--outputdir ${OFFLINEDIR}
```

- [Planning Analytics](#)

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-planning-analytics-4.0.2.tgz \  
--outputdir ${OFFLINEDIR}
```

- [Product Master](#)

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-productmaster-1.0.1.tgz \  
--outputdir ${OFFLINEDIR}
```

- [RStudio Server with R 3.6](#)

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-rstudio-1.0.2.tgz \  
--outputdir ${OFFLINEDIR}
```

- [SPSS Modeler](#)

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-spss-1.0.2.tgz \  
--outputdir ${OFFLINEDIR}
```

- [Virtual Data Pipeline](#)

Not applicable. For details, see [Installing Virtual Data Pipeline for Cloud Pak for Data](#).

- [Voice Gateway](#)

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-voice-gateway-1.0.3.tgz \  
--outputdir ${OFFLINEDIR}
```

- [Watson Assistant](#)


```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-watson-assistant-4.0.2.tgz \  
--outputdir ${OFFLINEDIR_WA}
```

- [Watson Assistant for Voice Interaction](#)

Watson Assistant for Voice Interaction is comprised of the following services:

- Voice Gateway
- Watson Assistant
- Watson Speech to Text
- Watson Text to Speech

Download the appropriate services for your use case.

- [Watson Discovery](#)

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-watson-discovery-4.0.2.tgz \  
--outputdir ${OFFLINEDIR}
```

- [Watson Knowledge Catalog](#)

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-wkc-4.0.2.tgz \  
--outputdir ${OFFLINEDIR}
```

- [Watson Knowledge Studio](#)

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-watson-ks-4.0.2.tgz \  
--outputdir ${OFFLINEDIR}
```

- [Watson Machine Learning](#)

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-wml-cpd-4.0.3.tgz \  
--outputdir ${OFFLINEDIR}
```

- [Watson Machine Learning Accelerator](#)

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-wml-accelerator-2.3.2.tgz \  
--outputdir ${OFFLINEDIR}
```

- [Watson OpenScale](#)

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-watson-openscale-2.2.0.tgz \  
--outputdir ${OFFLINEDIR}
```

- [Watson Speech to Text](#)

The same package is used for the Watson Speech to Text service and the Watson Text to Speech service. You only need to download the package once.

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-watson-speech-4.0.0.tgz \  
--outputdir ${OFFLINEDIR}
```

- [Watson Studio](#)

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-wsl-2.0.2.tgz \  
--outputdir ${OFFLINEDIR}
```

- [Watson Studio Runtimes](#)

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-wsl-runtimes-1.0.2.tgz \  
--outputdir ${OFFLINEDIR}
```

- [Watson Text to Speech](#)

The same package is used for the Watson Speech to Text service and the Watson Text to Speech service. You only need to download the package once.

```
cloudctl case save \  
--case ${CASE_REPO_PATH}/ibm-watson-speech-4.0.0.tgz \  
--outputdir ${OFFLINEDIR}
```

8. Mirroring the images to the intermediary container registry

To mirror the images:

1. If you did not purchase a license for EDB Postgres Enterprise, run the following command to remove any `edb-postgres-advanced` images from the list of images to mirror. If you did purchase a license for EDB Postgres Enterprise, skip this step.

```
sed -i -e '/edb-postgres-advanced/d' ${OFFLINEDIR}/ibm-cloud-native-postgresql-4.0.*-images.csv
```

2. Use the Cloud Pak for Data platform CASE package to mirror all of the images from the OFFLINEDIR directory to the intermediary container registry:

```
cloudctl case launch \
--case ${OFFLINEDIR}/ibm-cp-datacore-2.0.5.tgz \
--inventory cpdPlatformOperator \
--action mirror-images \
--args "--registry ${PORTABLE_REGISTRY} --inputDir ${OFFLINEDIR}"
```

Remember: The `mirror-images` command mirrors any changed images in the `${OFFLINEDIR}`.

3. If you downloaded the Watson Assistant CASE package, complete the appropriate steps for your environment:

- a. If you did not purchase a license for EDB Postgres Enterprise, run the following command to remove any `edb-postgres-advanced` images from the list of images to mirror. If you did purchase a license for EDB Postgres Enterprise, skip this step.

```
sed -i -e '/edb-postgres-advanced/d' ${OFFLINEDIR_WA}/ibm-cloud-native-postgresql-4.0.*-images.csv
```

- b. Run the following command to mirror the Watson Assistant images to the intermediary container registry:

```
cloudctl case launch \
--case ${OFFLINEDIR_WA}/ibm-watson-assistant-4.0.2.tgz \
--inventory assistantOperator \
--action mirror-images \
--args "--registry ${PORTABLE_REGISTRY} --inputDir ${OFFLINEDIR_WA}"
```

4. Determine the appropriate action based on the mirroring method that you are using:

Method	Next steps
Portable compute device	<ol style="list-style-type: none"> a. Move the portable compute device behind your firewall or connect the device to your private network. b. Complete 10. Mirroring images to the private container registry.
File transfer with a portable storage device	<ol style="list-style-type: none"> a. Save a copy of the software that you downloaded when you completed 1. Installing the software needed to mirror images to the <code>\${OFFLINEDIR}</code> on the portable storage device. b. Save the Docker registry image: <ul style="list-style-type: none"> • For Docker, run: <pre>docker save docker.io/library/registry:2 -o \${OFFLINEDIR}/registry.tar</pre> • For Podman, run: <pre>podman save docker.io/library/registry:2 -o \${OFFLINEDIR}/registry.tar</pre> c. Disconnect the portable storage device and move the device behind your firewall. d. Identify a workstation from which you can access the private container registry, where you can start the intermediary container registry, and where you can attach the portable storage device. e. Set an environment variable for the directory where the portable storage device is attached. For example: <pre>export OFFLINEDIR=\$HOME/offline</pre> f. Complete 9. Setting up a workstation to serve images.
File transfer with <code>scp</code> or <code>sftp</code>	<ol style="list-style-type: none"> a. Save a copy of the software that you downloaded when you completed 1. Installing the software needed to mirror images to the <code>\${OFFLINEDIR}</code>. b. Save the Docker registry image: <ul style="list-style-type: none"> • For Docker, run: <pre>docker save docker.io/library/registry:2 -o \${OFFLINEDIR}/registry.tar</pre> • For Podman, run: <pre>podman save docker.io/library/registry:2 -o \${OFFLINEDIR}/registry.tar</pre> c. Archive the contents off the <code>\$OFFLINE</code> directory: <pre>tar -cvzf archive-file-name -c \${OFFLINEDIR}</pre> d. Identify a workstation from which you can access the private container registry and where you can start the intermediary container registry. e. Copy the archive file to the workstation using <code>scp</code> or <code>sftp</code>. f. Identify or create the directory where you want to serve the images. For example: <pre>mkdir -p \$HOME/offline</pre> g. Set an environment variable for this directory: <pre>export OFFLINEDIR=\$HOME/offline</pre> h. Extract the contents of the archive: <pre>tar -xvf archive-file-name -c \${OFFLINEDIR}</pre> i. Complete 9. Setting up a workstation to serve images.

9. Setting up a workstation to serve images

Skip this step if you are using a portable compute device.

Note: If you have an existing intermediary container registry on this workstation that you set up using the IBM Cloud Pak CLI (`cloudctl`), you can re-use your setup and go directly to [10. Mirroring images to the private container registry](#).

If you are transferring the files inside your firewall, you must set up a workstation to serve the images so that you can transfer them to the private container registry.

To set up the workstation:

1. Install the software that you installed in [1. Installing the software needed to mirror images](#).
Remember: For any software that you downloaded, you should have a copy of the software in the `$(OFFLINEDIR)`.
2. Use the IBM Cloud Pak for Data platform operator package to set up the intermediary container registry on the cluster node:
 - a. Set the following environment variables for the IBM Cloud Pak for Data platform operator:

```
export CASE_ARCHIVE=ibm-cp-datacore-2.0.5.tgz
export CASE_INVENTORY_SETUP=cpdPlatformOperator
```

- b. Set environment variables for the intermediary container registry credentials. For example, reuse the credentials that you used when you set up the intermediary container registry:

```
export PORTABLE_REGISTRY_USER=username
export PORTABLE_REGISTRY_PASSWORD=password
export PORTABLE_REGISTRY=localhost:port_number
```

- c. Initialize the intermediary container registry:

```
cloudctl case launch \
  --case ${OFFLINEDIR}/ibm-cp-datacore-2.0.5.tgz \
  --inventory cpdPlatformOperator \
  --action init-registry \
  --args "--registry ${PORTABLE_REGISTRY_HOST} --user ${PORTABLE_REGISTRY_USER} --pass ${PORTABLE_REGISTRY_PASSWORD}"
--dir ${OFFLINEDIR}/imageregistry"
```

- d. Set the following environment variable to ensure that the IBM Cloud Pak CLI (cloudctl) uses `skopeo` rather than `oc mirror`. The `oc mirror` command can have trouble pulling images from binary repositories.

```
export USE_SKOPEO=true
```

- e. Start the intermediary container registry:

```
cloudctl case launch \
  --case ${OFFLINEDIR}/ibm-cp-datacore-2.0.5.tgz \
  --inventory cpdPlatformOperator \
  --action start-registry \
  --args "--port ${PORTABLE_REGISTRY_PORT} --dir ${OFFLINEDIR}/imageregistry --image docker.io/library/registry:2"
```

- f. To verify that the intermediary container registry is running, run the following commands:

Container client	Commands
Docker	List the containers by running: <code>docker ps</code>
Podman	List the containers by running: <code>podman ps</code>

10. Mirroring images to the private container registry

To mirror the images:

1. Store the intermediary container registry credentials:
 - a. Determine the values that you want to use for the following parameters:

```
PORTABLE_REGISTRY_USER
  The username that you want to use to authenticate to the intermediary container registry.
PORTABLE_REGISTRY_PASS
  The password that you want to use to authenticate to the intermediary container registry.
PORTABLE_REGISTRY_HOST
  The location of the intermediary container registry on the local host.
PORTABLE_REGISTRY_PORT
  Pick a port that is not currently in use. To avoid conflicts with system ports, choose a port greater than 1024.
```

- b. Set the environment variable for the parameters:

```
export PORTABLE_REGISTRY_USER=username
export PORTABLE_REGISTRY_PASSWORD=password
export PORTABLE_REGISTRY_HOST=localhost
export PORTABLE_REGISTRY_PORT=15000
PORTABLE_REGISTRY="${PORTABLE_REGISTRY_HOST}:${PORTABLE_REGISTRY_PORT}"
export PORTABLE_REGISTRY
```

- c. Run the following command to store the credentials:

```
cloudctl case launch \
  --case ${OFFLINEDIR}/ibm-cp-datacore-2.0.5.tgz \
  --inventory cpdPlatformOperator \
  --action configure-creds-airgap \
  --args "--registry ${PORTABLE_REGISTRY} --user ${PORTABLE_REGISTRY_USER} --pass ${PORTABLE_REGISTRY_PASSWORD}"
```

2. Store the private container registry credentials:
 - a. Work with your private container registry administrator to identify the values for the following parameters:

```
PRIVATE_REGISTRY_USER
  The username of a user who has the required privileges to push images to the private container registry.
PRIVATE_REGISTRY_PASSWORD
  The password of the user who has the required privileges to push images to the private container registry.
```

PRIVATE_REGISTRY

The location of the private container registry.

- b. Set environment variables for the parameters:

```
export PRIVATE_REGISTRY_USER=username
export PRIVATE_REGISTRY_PASSWORD=password
export PRIVATE_REGISTRY=private-registry-location
```

- c. Run the following command to store the credentials:

```
cloudctl case launch \
  --case ${OFFLINEDIR}/ibm-cp-datacore-2.0.5.tgz \
  --inventory cpdPlatformOperator \
  --action configure-creds-airgap \
  --args "--registry ${PRIVATE_REGISTRY} --user ${PRIVATE_REGISTRY_USER} --pass ${PRIVATE_REGISTRY_PASSWORD}"
```

3. Use the Cloud Pak for Data platform CASE package to mirror all of the images to the private container registry:

```
cloudctl case launch \
  --case ${OFFLINEDIR}/ibm-cp-datacore-2.0.5.tgz \
  --inventory cpdPlatformOperator \
  --action mirror-images \
  --args "--fromRegistry ${PORTABLE_REGISTRY} --registry ${PRIVATE_REGISTRY} --user ${PRIVATE_REGISTRY_USER} --pass ${PRIVATE_REGISTRY_PASSWORD} --inputDir ${OFFLINEDIR}"
```

4. If you mirrored the Watson Assistant images to the intermediary container registry, run the following command to mirror the images to the private container registry:

```
cloudctl case launch \
  --case ${OFFLINEDIR_WA}/ibm-watson-assistant-4.0.2.tgz \
  --inventory assistantOperator \
  --action mirror-images \
  --args "--fromRegistry ${PORTABLE_REGISTRY} --registry ${PRIVATE_REGISTRY} --user ${PRIVATE_REGISTRY_USER} --pass ${PRIVATE_REGISTRY_PASSWORD} --inputDir ${OFFLINEDIR_WA}"
```

Configuring your cluster to pull Cloud Pak for Data images

To ensure that your cluster can pull Cloud Pak for Data software images, you must update your cluster configuration.

Permissions you need for this task

You must be a cluster administrator.

When you need to complete this task

You must complete this task the first time you install Cloud Pak for Data.

The tasks that you must complete depend on whether your cluster pulls images directly from the IBM® Entitled Registry or from a private container registry.

Task	IBM Entitled Registry	Private container registry
1. Configuring the global image pull secret	Required	Required
2. Configuring an image content source policy	Not applicable	Required
3. Creating the catalog source	Required	Required

1. Configuring the global image pull secret

The global image pull secret ensures that your cluster has the necessary credentials to pull images.

The credentials that you need to specify depend on where you want to pull images from:

IBM Entitled Registry

If you are pulling images from the IBM Entitled Registry, the global image pull secret must contain your [IBM entitlement API key](#).

Private container registry

If you are pulling images from a private container registry, the global image pull secret must contain the credentials of an account that can pull images from the registry.

If you have already configured the global image pull secret with the necessary credentials, you can skip this task.

Important: When you change the global image pull secret, each node in the cluster is *automatically* restarted so that the Machine Config Operator can apply the changes. This restart process happens one node at a time. The cluster will wait for a node to restart before starting the process on the next node. In some situations, it takes more than 30 minutes for all of the nodes to be restarted. During this process, you might notice that resources are temporarily unavailable.

If your deployment is on IBM Cloud, you must *manually* reload the worker nodes in your cluster for the changes to take effect.

To configure the global image pull secret:

1. Determine whether there is an existing global image pull secret:

```
oc extract secret/pull-secret -n openshift-config
```

This command generates a JSON file called `.dockerconfigjson` in the current directory.

2. Take the appropriate action based on the contents of the `.dockerconfigjson` file:

Pull secret status	Image content source policy
--------------------	-----------------------------

Pull secret status	Image content source policy
The file is empty	<p>a. Set the following environment variables based on the container registry that OpenShift® is going to pull from:</p> <p>IBM Entitled Registry</p> <pre>export REGISTRY_USER=cp export REGISTRY_PASSWORD=entitlement-key export REGISTRY_SERVER=cp.icr.io</pre> <p>Replace <i>entitlement-key</i> with your entitlement key. For details, see IBM entitlement API key.</p> <p>Private container registry</p> <pre>export REGISTRY_USER=username export REGISTRY_PASSWORD=password export REGISTRY_SERVER=registry-location</pre> <p>Replace the following values:</p> <p><i>username</i> The username of a user that can pull images from the private container registry</p> <p><i>password</i> The password for the specified user.</p> <p><i>registry-location</i> The location of the private container registry. For example, <code>private-registry.example.com</code>.</p> <p>b. Run the following command to create the pull secret:</p> <pre>oc create secret docker-registry \ --docker-server=\${REGISTRY_SERVER} \ --docker-username=\${REGISTRY_USER} \ --docker-password=\${REGISTRY_PASSWORD} \ --docker-email=\${REGISTRY_USER} \ -n openshift-config pull-secret</pre>
There is an existing pull secret	<p>a. Encode the username and password using Base64 encoding:</p> <p>IBM Entitled Registry</p> <pre>echo -n "cp:entitlement-key" base64 -w0</pre> <p>Replace <i>entitlement-key</i> with your entitlement key. For details, see IBM entitlement API key.</p> <p>Private container registry</p> <pre>echo -n "username:password" base64 -w0</pre> <p>Replace the following values:</p> <p><i>username</i> The username of a user that can pull images from the private container registry</p> <p><i>password</i> The password for the specified user.</p> <p>b. Add an entry for the container registry to the <code>auths</code> section in the JSON file. In the following example, 1 is the new entry and 2 is the existing entry:</p> <pre>{ "auths": { 1 "registry-location": { "auth": "base64-encoded-credentials", "email": "not-used" }, 2 "myregistry.example.com": { "auth": "b3Blb=", "email": "not-used" } } }</pre> <p>Replace the following values:</p> <p><i>registry-location</i> If you are pulling images from the IBM Entitled Registry, the value is <code>cp.icr.io</code>. If you are pulling images from a private container registry, specify the location of the private container registry. For example, <code>private-registry.example.com</code>.</p> <p><i>base64-encoded-credentials</i> The encoded credentials that you generated in the previous step. For example, <code>cmVnX3VzZXJ1Ym11OnJlZ19wYXNkd29yZAo=</code>.</p> <p>c. Apply the new configuration:</p> <pre>oc set data secret/pull-secret -n openshift-config --from-file=.dockerconfigjson=.dockerconfigjson</pre>

Important: For deployments on IBM Cloud, you must reload the worker nodes in your cluster for the changes to take effect. For details, see [Adding a private registry to the global pull secret](#).

If have a VPC Gen2 cluster and you use Portworx storage, see [Portworx storage limitations](#) before you reload your worker nodes.

3. Get the status of the nodes:

```
oc get node
```

Wait until all the nodes are **Ready** before you proceed to the next step. For example, if you see **Ready, SchedulingDisabled**, wait for the process to complete:

NAME	STATUS	ROLES	AGE	VERSION
master0	Ready	master	5h57m	v1.20.0
master1	Ready	master	5h57m	v1.20.0
master2	Ready	master	5h57m	v1.20.0
worker0	Ready, SchedulingDisabled	worker	5h48m	v1.20.0
worker1	Ready	worker	5h48m	v1.20.0
worker2	Ready	worker	5h48m	v1.20.0

2. Configuring an image content source policy

If you are pulling images directly from the IBM Entitled Registry on a connected cluster, you can skip this step.

If you mirrored images to a private container registry, you must tell your cluster where to find the software images. (For more information how Red Hat® OpenShift Container Platform locates images from an mirrored repository, see [Configuring image registry repository mirroring](#) in the Red Hat OpenShift Container Platform documentation.)

Important: This process will temporarily disable scheduling on each node in the cluster, so you might notice that resources are temporarily unavailable. However, this process happens on one node at a time. The cluster will temporarily disable scheduling on a node, apply the configuration change, and then re-enable scheduling before starting the process on the next node.

To configure an image content source policy:

1. Set the following environment variable to point to the location of the private container registry:

```
export PRIVATE_REGISTRY=private-registry-location
```

2. Create an image content source policy. The contents of the policy depend on whether you have an existing policy for IBM Cloud Pak® foundational services.

Options	Image content source policy
IBM Cloud Pak foundational services is already installed on the cluster	<p>If IBM Cloud Pak foundational services is already installed, it is likely that you already have an image content source policy for <code>quay.io/openshiftio</code>. Therefore, you do not need to create a mirroring policy for those images.</p> <pre>cat <<EOF oc apply -f - apiVersion: operator.openshift.io/v1alpha1 kind: ImageContentSourcePolicy metadata: name: cloud-pak-for-data-mirror spec: repositoryDigestMirrors: - mirrors: - \${PRIVATE_REGISTRY}/cp source: cp.icr.io/cp - mirrors: - \${PRIVATE_REGISTRY}/cp/cpd source: cp.icr.io/cp/cpd - mirrors: - \${PRIVATE_REGISTRY}/copen source: icr.io/copen EOF</pre>
IBM Cloud Pak foundational services is not installed on the cluster	<p>If IBM Cloud Pak foundational services is not installed, it is unlikely that you have an image content source policy for <code>quay.io/openshiftio</code>, so you should create a mirroring policy for those images.</p> <pre>cat <<EOF oc apply -f - apiVersion: operator.openshift.io/v1alpha1 kind: ImageContentSourcePolicy metadata: name: cloud-pak-for-data-mirror spec: repositoryDigestMirrors: - mirrors: - \${PRIVATE_REGISTRY}/openshiftio source: quay.io/openshiftio - mirrors: - \${PRIVATE_REGISTRY}/cp source: cp.icr.io/cp - mirrors: - \${PRIVATE_REGISTRY}/cp/cpd source: cp.icr.io/cp/cpd - mirrors: - \${PRIVATE_REGISTRY}/copen source: icr.io/copen EOF</pre>

3. Verify that the image content source policy was created:

```
oc get imageContentSourcePolicy
```

4. Get the status of the nodes:

```
oc get node
```

Wait until all the nodes are **Ready** before you proceed to the next step. For example, if you see **Ready, SchedulingDisabled**, wait for the process to complete:

NAME	STATUS	ROLES	AGE	VERSION
master0	Ready	master	5h57m	v1.20.0
master1	Ready	master	5h57m	v1.20.0
master2	Ready	master	5h57m	v1.20.0
worker0	Ready, SchedulingDisabled	worker	5h48m	v1.20.0
worker1	Ready	worker	5h48m	v1.20.0
worker2	Ready	worker	5h48m	v1.20.0

3. Creating the catalog source

Operator Lifecycle Manager (OLM) uses an [Operator catalog](#) to discover and install Operators and their dependencies.

A *catalog source* is a repository of cluster service versions (CSVs), custom resource definitions (CRDs), and packages that comprise an application. To ensure that OLM can use the Cloud Pak for Data operators to install the software, you must create the appropriate catalog sources for your environment. (For more information about these terms, see the [Operator Framework glossary of common terms](#) in the Red Hat OpenShift Container Platform documentation.)

Important: If you are using a private container registry, the steps in this task assume that you are using the latest CASE package to create the catalog source. Older CASE packages are listed in [Operator and operand versions](#).

To create the catalog source, complete the appropriate steps for your environment:

Image location	Required catalog source
IBM Entitled Registry	<p>If you are pulling images from the IBM Entitled Registry, create the following catalog sources:</p> <ol style="list-style-type: none"> 1. Create the catalog source for the IBM Operator Catalog: <p>This catalog source is used for :</p> <ul style="list-style-type: none"> • IBM Cloud Pak foundational services • IBM Cloud Pak for Data platform operator • Service operators <p>Important: If you are pulling images from the IBM Entitled Registry, all of the services are included in the IBM Operator Catalog. This means that you do not need to create catalog source objects for each service that you plan to install.</p> <ol style="list-style-type: none"> a. Check whether the IBM Operator Catalog already exists on your cluster: <pre>oc get catalogsource -n openshift-marketplace</pre> <p>Review the output to determine whether there is an entry called <code>ibm-operator-catalog</code>.</p> b. If the IBM Operator Catalog does not exist, create it: <pre>cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: CatalogSource metadata: name: ibm-operator-catalog namespace: openshift-marketplace spec: displayName: "IBM Operator Catalog" publisher: IBM sourceType: grpc image: icr.io/cpopen/ibm-operator-catalog:latest updateStrategy: registryPoll: interval: 45m EOF</pre> c. Verify that the IBM Operator Catalog was successfully created: <pre>oc get catalogsource -n openshift-marketplace</pre> <p>Review the output to ensure that there is an entry called <code>ibm-operator-catalog</code>.</p> 2. Create the Db2U catalog source if you plan to install one of the following services: <ul style="list-style-type: none"> • Data Virtualization • Db2® • Db2 Big SQL • Db2 Warehouse • OpenPages® (required only if you want OpenPages to automatically provision a Db2 database) <ol style="list-style-type: none"> a. Check whether the IBM Db2U Catalog already exists on your cluster: <pre>oc get catalogsource -n openshift-marketplace</pre> <p>Review the output to determine whether there is an entry called <code>ibm-db2uoperator-catalog</code>.</p> b. If the IBM Db2U Catalog does not exist, create it: <pre>cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: CatalogSource metadata: name: ibm-db2uoperator-catalog namespace: openshift-marketplace spec: sourceType: grpc image: docker.io/ibmcom/ibm-db2uoperator-catalog:latest imagePullPolicy: Always displayName: IBM Db2U Catalog publisher: IBM updateStrategy: registryPoll: interval: 45m EOF</pre> c. Verify that the IBM Db2U Catalog was successfully created: <pre>oc get catalogsource -n openshift-marketplace</pre> <p>Review the output to ensure that there is an entry called <code>ibm-db2uoperator-catalog</code>.</p>
Private	The following steps assume that you have the CASE packages on you local file system from mirroring the images to your private container registry.

Catalog location	Required catalog source
registry	<p>If you are running the commands on a different machine, you must download the necessary packages before you create the catalog source:</p> <ul style="list-style-type: none"> • IBM Cloud Pak foundational services CASE package (Skip this download if you already have a catalog source for IBM Cloud Pak foundational services) • Scheduling service CASE package (Skip this download if you are not installing the scheduling service.) • IBM Cloud Pak for Data CASE package • Service CASE packages <p>If you are pulling images from a private container registry, create the following catalog sources:</p> <ol style="list-style-type: none"> 1. Create the IBM Cloud Pak foundational services catalog source: Skip this step if a supported version of IBM Cloud Pak foundational services is already installed on your cluster. <ol style="list-style-type: none"> a. Run the following command to create the IBM Cloud Pak foundational services catalog source: <pre>cloudctl case launch \ --case \${OFFLINEDIR}/ibm-cp-common-services-1.6.0.tgz \ --inventory ibmCommonServiceOperatorSetup \ --namespace openshift-marketplace \ --action install-catalog \ --args "--registry \${PRIVATE_REGISTRY} --inputDir \${OFFLINEDIR} --recursive"</pre> b. Verify that <code>opencloud-operators</code> is READY: <pre>oc get catalogsource -n openshift-marketplace opencloud-operators \ -o jsonpath='{.status.connectionState.lastObservedState} {"\n"}'</pre> 2. Create the scheduling service catalog source: Skip this step if you are not installing the scheduling service. <ol style="list-style-type: none"> a. Run the following command to create the scheduling service catalog source: <pre>cloudctl case launch \ --case \${OFFLINEDIR}/ibm-cpd-scheduling-1.2.3.tgz \ --inventory schedulerSetup \ --namespace openshift-marketplace \ --action install-catalog \ --args "--inputDir \${OFFLINEDIR} --recursive"</pre> b. Verify that <code>ibm-cpd-scheduling-catalog</code> is READY: <pre>oc get catalogsource -n openshift-marketplace ibm-cpd-scheduling-catalog \ -o jsonpath='{.status.connectionState.lastObservedState} {"\n"}'</pre> 3. Create the IBM Cloud Pak for Data catalog source: <ol style="list-style-type: none"> a. Run the following command to create the IBM Cloud Pak for Data catalog source: <pre>cloudctl case launch \ --case \${OFFLINEDIR}/ibm-cp-datacore-2.0.5.tgz \ --inventory cpdPlatformOperator \ --namespace openshift-marketplace \ --action install-catalog \ --args "--inputDir \${OFFLINEDIR} --recursive"</pre> b. Verify that <code>cpd-platform</code> is READY: <pre>oc get catalogsource -n openshift-marketplace cpd-platform \ -o jsonpath='{.status.connectionState.lastObservedState} {"\n"}'</pre> 4. Create the Db2U catalog source if you plan to install one of the following services: <ul style="list-style-type: none"> • Data Virtualization • Db2 • Db2 Big SQL • Db2 Warehouse • OpenPages (required only if you want OpenPages to automatically provision a Db2 database) <ol style="list-style-type: none"> a. Install the following Python software on the system where you are running the <code>cloudctl</code> commands: <ol style="list-style-type: none"> i. Python 2 To install Python 2, run the following command: <pre>yum install -y python2 alternatives --set python /usr/bin/python2</pre> ii. <code>pyyaml</code> To install <code>pyyaml</code>, run the following command: <pre>pip2 install pyyaml</pre> b. Run the following command to create the Db2U catalog source: <pre>cloudctl case launch \ --case \${OFFLINEDIR}/ibm-db2uoperator-4.0.4.tgz \ --inventory db2uOperatorSetup \ --namespace openshift-marketplace \ --action install-catalog \ --args "--inputDir \${OFFLINEDIR} --recursive"</pre> c. Verify that <code>ibm-db2uoperator-catalog</code> is READY: <pre>oc get catalogsource -n openshift-marketplace ibm-db2uoperator-catalog \ -o jsonpath='{.status.connectionState.lastObservedState} {"\n"}'</pre>

Image location	Required catalog source
	5. Create the catalog source for <i>each</i> service that you mirrored to the private container registry. For details, see Service catalog source .

Service catalog source

If you are using a *private container registry*, create the catalog source for each service that you plan to install.

Remember: The steps assume that you are using the latest CASE package to create the catalog source. Older CASE packages are listed in [Operator and operand versions](#).

- >

Anaconda Repository for IBM Cloud Pak for Data

Not applicable. For details, see [Installing Anaconda Repository for IBM Cloud Pak for Data](#).
- >

Analytics Engine Powered by Apache Spark

 - Run the following command to create the Analytics Engine Powered by Apache Spark catalog source:

```
cloudctl case launch \
--case ${OFFLINEDIR}/ibm-analyticsengine-4.0.2.tgz \
--inventory analyticsengineOperatorSetup \
--namespace openshift-marketplace \
--action install-catalog \
--args "--inputDir ${OFFLINEDIR} --recursive"
```
 - Verify that `ibm-cpd-ae-operator-catalog` is **READY**:

```
oc get catalogsource -n openshift-marketplace ibm-cpd-ae-operator-catalog \
-o jsonpath='{.status.connectionState.lastObservedState} {"\n"}'
```
- >

Cognos Analytics

 - Run the following command to create the Cognos Analytics catalog source:

```
cloudctl case launch \
--case ${OFFLINEDIR}/ibm-cognos-analytics-prod-4.0.4.tgz \
--inventory ibmCaOperatorSetup \
--namespace openshift-marketplace \
--action install-catalog \
--args "--inputDir ${OFFLINEDIR} --recursive"
```
 - Verify that `ibm-ca-operator-catalog` is **READY**:

```
oc get catalogsource -n openshift-marketplace ibm-ca-operator-catalog \
-o jsonpath='{.status.connectionState.lastObservedState} {"\n"}'
```
- >

Cognos Dashboards

 - Run the following command to create the Cognos Dashboards catalog source:

```
cloudctl case launch \
--case ${OFFLINEDIR}/ibm-cde-2.0.2.tgz \
--inventory cdeOperatorSetup \
--namespace openshift-marketplace \
--action install-catalog \
--args "--inputDir ${OFFLINEDIR} --recursive"
```
 - Verify that `ibm-cde-operator-catalog` is **READY**:

```
oc get catalogsource -n openshift-marketplace ibm-cde-operator-catalog \
-o jsonpath='{.status.connectionState.lastObservedState} {"\n"}'
```
- >

Data Refinery

The catalog source for Data Refinery is automatically created when you create the catalog source for either Watson™ Knowledge Catalog or Watson Studio.
- >

Data Virtualization

 - This service has a dependency on Db2U. Verify that the Db2U catalog source (`ibm-db2uoperator-catalog`) is **READY**:

```
oc get catalogsource -n openshift-marketplace ibm-db2uoperator-catalog \
-o jsonpath='{.status.connectionState.lastObservedState} {"\n"}'
```
 - Run the following command to create the Data Virtualization catalog source:

```
cloudctl case launch \
--case ${OFFLINEDIR}/ibm-dv-case-1.7.2.tgz \
--inventory dv \
--namespace openshift-marketplace \
--action install-catalog \
--args "--inputDir ${OFFLINEDIR} --recursive"
```
 - Verify that `ibm-dv-operator-catalog` is **READY**:

```
oc get catalogsource -n openshift-marketplace ibm-dv-operator-catalog \
-o jsonpath='{.status.connectionState.lastObservedState} {"\n"}'
```
- >

DataStage

Create the appropriate catalog source for your environment:

DataStage Enterprise

1. Run the following command to create the DataStage Enterprise catalog source:

```
cloudctl case launch \  
--case ${OFFLINEDIR}/ibm-datastage-enterprise-4.0.3.tgz \  
--inventory dsOperatorSetup \  
--namespace openshift-marketplace \  
--action install-catalog \  
--args "--inputDir ${OFFLINEDIR} --recursive"
```

2. Verify that `ibm-datastage-operator-catalog` is `READY`:

```
oc get catalogsource -n openshift-marketplace ibm-datastage-operator-catalog \  
-o jsonpath='{.status.connectionState.lastObservedState} {"\n"}'
```

DataStage Enterprise Plus

1. Run the following command to create the DataStage Enterprise Plus catalog source:

```
cloudctl case launch \  
--case ${OFFLINEDIR}/ibm-datastage-4.0.3.tgz \  
--inventory dsOperatorSetup \  
--namespace openshift-marketplace \  
--action install-catalog \  
--args "--inputDir ${OFFLINEDIR} --recursive"
```

2. Verify that `ibm-datastage-operator-catalog` is `READY`:

```
oc get catalogsource -n openshift-marketplace ibm-datastage-operator-catalog \  
-o jsonpath='{.status.connectionState.lastObservedState} {"\n"}'
```

>
Db2

1. This service has a dependency on Db2U. Verify that the Db2U catalog source (`ibm-db2uoperator-catalog`) is `READY`:

```
oc get catalogsource -n openshift-marketplace ibm-db2uoperator-catalog \  
-o jsonpath='{.status.connectionState.lastObservedState} {"\n"}'
```

2. Run the following command to create the Db2 catalog source:

```
cloudctl case launch \  
--case ${OFFLINEDIR}/ibm-db2oltp-4.0.3.tgz \  
--inventory db2oltpOperatorSetup \  
--namespace openshift-marketplace \  
--action install-catalog \  
--args "--inputDir ${OFFLINEDIR} --recursive"
```

3. Verify that `ibm-db2oltp-cp4d-operator-catalog` is `READY`:

```
oc get catalogsource -n openshift-marketplace ibm-db2oltp-cp4d-operator-catalog \  
-o jsonpath='{.status.connectionState.lastObservedState} {"\n"}'
```

>
Db2 Big SQL

1. This service has a dependency on Db2U. Verify that the Db2U catalog source (`ibm-db2uoperator-catalog`) is `READY`:

```
oc get catalogsource -n openshift-marketplace ibm-db2uoperator-catalog \  
-o jsonpath='{.status.connectionState.lastObservedState} {"\n"}'
```

2. Run the following command to create the Db2 Big SQL catalog source:

```
cloudctl case launch \  
--case ${OFFLINEDIR}/ibm-bigsql-case-7.2.2.tgz \  
--inventory bigsql \  
--namespace openshift-marketplace \  
--action install-catalog \  
--args "--inputDir ${OFFLINEDIR} --recursive"
```

3. Verify that `ibm-bigsql-operator-catalog` is `READY`:

```
oc get catalogsource -n openshift-marketplace ibm-bigsql-operator-catalog \  
-o jsonpath='{.status.connectionState.lastObservedState} {"\n"}'
```

>
Db2 Data Gate

1. Run the following command to create the Db2 Data Gate catalog source:

```
cloudctl case launch \  
--case ${OFFLINEDIR}/ibm-datagate-prod-4.0.2.tgz \  
--inventory datagateOperatorSetup \  
--namespace openshift-marketplace \  
--action install-catalog \  
--args "--inputDir ${OFFLINEDIR} --recursive"
```

2. Verify that `ibm-datagate-operator-catalog` is `READY`:

```
oc get catalogsource -n openshift-marketplace ibm-datagate-operator-catalog \  
-o jsonpath='{.status.connectionState.lastObservedState} {"\n"}'
```

>
Db2 Data Management Console

1. Run the following command to create the Db2 Data Management Console catalog source:

```
cloudctl case launch \
--case ${OFFLINEDIR}/ibm-dmc-4.0.2.tgz \
--inventory dmcOperatorSetup \
--namespace openshift-marketplace \
--action install-catalog \
--args "--inputDir ${OFFLINEDIR} --recursive"
```

2. Verify that `ibm-dmc-operator-catalog` is `READY`:

```
oc get catalogsource -n openshift-marketplace ibm-dmc-operator-catalog \
-o jsonpath='{.status.connectionState.lastObservedState}' {"\n"}
```

- >
Db2 Event Store

Not applicable. Contact IBM Software support if you plan to install this service.

- >
Db2 Warehouse

1. This service has a dependency on Db2U. Verify that the Db2U catalog source (`ibm-db2uoperator-catalog`) is `READY`:

```
oc get catalogsource -n openshift-marketplace ibm-db2uoperator-catalog \
-o jsonpath='{.status.connectionState.lastObservedState}' {"\n"}
```

2. Run the following command to create the Db2 Warehouse catalog source:

```
cloudctl case launch \
--case ${OFFLINEDIR}/ibm-db2wh-4.0.3.tgz \
--inventory db2whOperatorSetup \
--namespace openshift-marketplace \
--action install-catalog \
--args "--inputDir ${OFFLINEDIR} --recursive"
```

3. Verify that `ibm-db2wh-cp4d-operator-catalog` is `READY`:

```
oc get catalogsource -n openshift-marketplace ibm-db2wh-cp4d-operator-catalog \
-o jsonpath='{.status.connectionState.lastObservedState}' {"\n"}
```

- >
Decision Optimization

1. Run the following command to create the Decision Optimization catalog source:

```
cloudctl case launch \
--case ${OFFLINEDIR}/ibm-dods-4.0.2.tgz \
--inventory dodsOperatorSetup \
--namespace openshift-marketplace \
--action install-catalog \
--args "--inputDir ${OFFLINEDIR} --recursive"
```

2. Verify that `ibm-cpd-dods-operator-catalog` is `READY`:

```
oc get catalogsource -n openshift-marketplace ibm-cpd-dods-operator-catalog \
-o jsonpath='{.status.connectionState.lastObservedState}' {"\n"}
```

- >
EDB Postgres

1. Run the following command to create the EDB Postgres catalog source:

```
cloudctl case launch \
--case ${OFFLINEDIR}/ibm-cpd-edb-4.0.2.tgz \
--inventory ibmCPDEDBSetup \
--namespace openshift-marketplace \
--action install-catalog \
--args "--inputDir ${OFFLINEDIR} --recursive"
```

2. Verify that `ibm-cpd-edb-operator-catalog` is `READY`:

```
oc get catalogsource -n openshift-marketplace ibm-cpd-edb-operator-catalog \
-o jsonpath='{.status.connectionState.lastObservedState}' {"\n"}
```

- >
Execution Engine for Apache Hadoop

1. Run the following command to create the Execution Engine for Apache Hadoop catalog source:

```
cloudctl case launch \
--case ${OFFLINEDIR}/ibm-hadoop-4.0.2.tgz \
--inventory hadoopSetup \
--namespace openshift-marketplace \
--action install-catalog \
--args "--inputDir ${OFFLINEDIR} --recursive"
```

2. Verify that `ibm-cpd-hadoop-operator-catalog` is `READY`:

```
oc get catalogsource -n openshift-marketplace ibm-cpd-hadoop-operator-catalog \
-o jsonpath='{.status.connectionState.lastObservedState}' {"\n"}
```

- >
Financial Services Workbench

Not applicable. For details, see the [Financial Services Workbench documentation](#).

- >
IBM Match 360 with Watson

1. Run the following command to create the IBM Match 360 with Watson catalog source:

```
cloudctl case launch \
--case ${OFFLINEDIR}/ibm-mdm-1.0.115.tgz \
--inventory mdmOperator \
--namespace openshift-marketplace \
--action install-catalog \
--args "--inputDir ${OFFLINEDIR} --recursive"
```

2. Verify that `ibm-mdm-operator-catalog` is `READY`:

```
oc get catalogsource -n openshift-marketplace ibm-mdm-operator-catalog \
-o jsonpath='{.status.connectionState.lastObservedState} {"\n"}'
```

- > Informix

1. Run the following command to create the catalog source for the Informix install operator:

```
cloudctl case launch \
--case ${OFFLINEDIR}/ibmInformixOperatorSetup \
--inventory ibmInformixOperatorSetup \
--namespace openshift-marketplace \
--action install-catalog \
--args "--inputDir ${OFFLINEDIR} --recursive"
```

2. Verify that the `ibm-informix-operator-catalog` is `READY`:

```
oc get catalogsource -n openshift-marketplace ibm-informix-operator-catalog \
-o jsonpath='{.status.connectionState.lastObservedState} {"\n"}'
```

3. Run the following command to create the catalog source for the Informix deployment operator:

```
cloudctl case launch \
--case ${OFFLINEDIR}/ibm-informix-cp4d-operator-4.0.2.tgz \
--inventory ibmInformixCp4dOperatorSetup \
--namespace openshift-marketplace \
--action install-catalog \
--args "--inputDir ${OFFLINEDIR} --recursive"
```

4. Verify that `ibm-informix-cp4d-operator-catalog` is `READY`:

```
oc get catalogsource -n openshift-marketplace ibm-informix-cp4d-operator-catalog \
-o jsonpath='{.status.connectionState.lastObservedState} {"\n"}'
```

- > MongoDB

1. Run the following command to create the MongoDB catalog source:

```
cloudctl case launch \
--case ${OFFLINEDIR}/ibm-cpd-mongodb-4.0.2.tgz \
--inventory ibmCPDMongodbSetup \
--namespace openshift-marketplace \
--action install-catalog \
--args "--inputDir ${OFFLINEDIR} --recursive"
```

2. Verify that `ibm-cpd-mongodb-catalog` is `READY`:

```
oc get catalogsource -n openshift-marketplace ibm-cpd-mongodb-catalog \
-o jsonpath='{.status.connectionState.lastObservedState} {"\n"}'
```

- > OpenPages

1. Run the following command to create the OpenPages catalog source:

```
cloudctl case launch \
--case ${OFFLINEDIR}/ibm-openpages-2.0.2+20210910.011848.82030838.tgz \
--inventory operatorSetup \
--namespace openshift-marketplace \
--action install-catalog \
--args "--inputDir ${OFFLINEDIR} --recursive"
```

2. Verify that `ibm-cpd-openpages-operator-catalog` is `READY`:

```
oc get catalogsource -n openshift-marketplace ibm-cpd-openpages-operator-catalog \
-o jsonpath='{.status.connectionState.lastObservedState} {"\n"}'
```

If you want OpenPages to automatically provision a Db2 database, you must also create the following catalog sources:

Db2U

Verify that the Db2U catalog source (`ibm-db2uoperator-catalog`) is `READY`:

```
oc get catalogsource -n openshift-marketplace ibm-db2uoperator-catalog \
-o jsonpath='{.status.connectionState.lastObservedState} {"\n"}'
```

Db2 as a service

1. Run the following command to create the Db2 as a service catalog source:

```
cloudctl case launch \
--case ${OFFLINEDIR}/ibm-db2aaservice-4.0.3.tgz \
--inventory db2aaserviceOperatorSetup \
--namespace openshift-marketplace \
--action install-catalog \
--args "--inputDir ${OFFLINEDIR} --recursive"
```

2. Verify that `ibm-db2aaservice-cp4d-operator-catalog` is `READY`:

```
oc get catalogsource -n openshift-marketplace ibm-db2aaservice-cp4d-operator-catalog \
-o jsonpath='{.status.connectionState.lastObservedState} {"\n"}'
```

>
Planning Analytics

1. Run the following command to create the Planning Analytics catalog source:

```
cloudctl case launch \
--case ${OFFLINEDIR}/ibm-planning-analytics-4.0.2.tgz \
--inventory ibmPlanningAnalyticsOperatorSetup \
--namespace openshift-marketplace \
--action install-catalog \
--args "--inputDir ${OFFLINEDIR} --recursive"
```

2. Verify that `ibm-planning-analytics-operator-catalog` is `READY`:

```
oc get catalogsource -n openshift-marketplace ibm-planning-analytics-operator-catalog \
-o jsonpath='{.status.connectionState.lastObservedState} {"\n"}'
```

>
Product Master

1. Run the following command to create the Product Master catalog source:

```
cloudctl case launch \
--case ${OFFLINEDIR}/ibm-productmaster-1.0.1.tgz \
--inventory productmasterOperatorSetup \
--namespace openshift-marketplace \
--action install-catalog \
--args "--inputDir ${OFFLINEDIR} --recursive"
```

```
cloudctl case launch \
--case ${OFFLINEDIR}/ibm-productmaster-1.0.1.tgz \
--inventory productmasterOperatorSetup \
--namespace openshift-marketplace \
--action install-catalog \
--args "--inputDir ${OFFLINEDIR} --recursive"
```

2. Verify that `ibm-productmaster-catalog` is `READY`:

```
oc get catalogsource -n openshift-marketplace ibm-productmaster-catalog \
-o jsonpath='{.status.connectionState.lastObservedState} {"\n"}'
```

>
RStudio Server with R 3.6

1. Run the following command to create the RStudio Server with R 3.6 catalog source:

```
cloudctl case launch \
--case ${OFFLINEDIR}/ibm-rstudio-1.0.2.tgz \
--inventory rstudioSetup \
--namespace openshift-marketplace \
--action install-catalog \
--args "--inputDir ${OFFLINEDIR} --recursive"
```

2. Verify that `ibm-cpd-rstudio-operator-catalog` is `READY`:

```
oc get catalogsource -n openshift-marketplace ibm-cpd-rstudio-operator-catalog \
-o jsonpath='{.status.connectionState.lastObservedState} {"\n"}'
```

>
SPSS Modeler

1. Run the following command to create the SPSS Modeler catalog source:

```
cloudctl case launch \
--case ${OFFLINEDIR}/ibm-spss-1.0.2.tgz \
--inventory spssSetup \
--namespace openshift-marketplace \
--action install-catalog \
--args "--inputDir ${OFFLINEDIR} --recursive"
```

2. Verify that `ibm-cpd-spss-operator-catalog` is `READY`:

```
oc get catalogsource -n openshift-marketplace ibm-cpd-spss-operator-catalog \
-o jsonpath='{.status.connectionState.lastObservedState} {"\n"}'
```

>
Virtual Data Pipeline

Not applicable. For details, see [Installing Virtual Data Pipeline for Cloud Pak for Data](#).

>
Voice Gateway

1. Run the following command to create the Voice Gateway catalog source:

```
cloudctl case launch \
--case ${OFFLINEDIR}/ibm-voice-gateway-1.0.3.tgz \
--inventory voiceGatewayOperatorSetup \
--namespace openshift-marketplace \
--action install-catalog \
--args "--inputDir ${OFFLINEDIR} --recursive"
```

2. Verify that `ibm-voice-gateway-operator-catalog` is `READY`:

```
oc get catalogsource -n openshift-marketplace ibm-voice-gateway-operator-catalog \
-o jsonpath='{.status.connectionState.lastObservedState} {"\n"}'
```

>
Watson Assistant

1. Run the following command to create the Watson Assistant catalog source:

```
cloudctl case launch \  
  --case ${OFFLINEDIR_WA}/ibm-watson-assistant-4.0.2.tgz \  
  --inventory assistantOperator \  
  --namespace openshift-marketplace \  
  --action install-catalog \  
  --args "--inputDir ${OFFLINEDIR_WA} --recursive"
```

2. Verify that `ibm-watson-assistant-operator-catalog` is `READY`:

```
oc get catalogsource -n openshift-marketplace ibm-watson-assistant-operator-catalog \  
-o jsonpath='{.status.connectionState.lastObservedState} {"\n"}'
```

>
Watson Assistant for Voice Interaction

Watson Assistant for Voice Interaction is comprised of the following services:

- o Voice Gateway
- o Watson Assistant
- o Watson Speech to Text
- o Watson Text to Speech

>
Watson Discovery

1. Run the following command to create the Watson Discovery catalog source:

```
cloudctl case launch \  
  --case ${OFFLINEDIR}/ibm-watson-discovery-4.0.2.tgz \  
  --inventory discoveryOperators \  
  --namespace openshift-marketplace \  
  --action install-catalog \  
  --args "--inputDir ${OFFLINEDIR} --recursive"
```

2. Verify that `ibm-watson-discovery-operator-catalog` is `READY`:

```
oc get catalogsource -n openshift-marketplace ibm-watson-discovery-operator-catalog \  
-o jsonpath='{.status.connectionState.lastObservedState} {"\n"}'
```

Important: If you plan to install both Watson Discovery and Watson Assistant, you must create the catalog source for Watson Discovery first, and then create the catalog source for Watson Assistant. The order matters.

>
Watson Knowledge Catalog

1. Run the following command to create the Watson Knowledge Catalog catalog source:

```
cloudctl case launch \  
  --case ${OFFLINEDIR}/ibm-wkc-4.0.2.tgz \  
  --inventory wkcOperatorSetup \  
  --namespace openshift-marketplace \  
  --action install-catalog \  
  --args "--inputDir ${OFFLINEDIR} --recursive"
```

2. Verify that `ibm-cpd-wkc-operator-catalog` is `READY`:

```
oc get catalogsource -n openshift-marketplace ibm-cpd-wkc-operator-catalog \  
-o jsonpath='{.status.connectionState.lastObservedState} {"\n"}'
```

>
Watson Knowledge Studio

1. Run the following command to create the Watson Knowledge Studio catalog source:

```
cloudctl case launch \  
  --case ${OFFLINEDIR}/ibm-watson-ks-4.0.2.tgz \  
  --inventory wksOperatorSetup \  
  --namespace openshift-marketplace \  
  --action install-catalog \  
  --args "--inputDir ${OFFLINEDIR} --recursive"
```

2. Verify that `ibm-watson-ks-operator-catalog` is `READY`:

```
oc get catalogsource -n openshift-marketplace ibm-watson-ks-operator-catalog \  
-o jsonpath='{.status.connectionState.lastObservedState} {"\n"}'
```

>
Watson Machine Learning

1. Run the following command to create the Watson Machine Learning catalog source:

```
cloudctl case launch \  
  --case ${OFFLINEDIR}/ibm-wml-cpd-4.0.3.tgz \  
  --inventory wmlOperatorSetup \  
  --namespace openshift-marketplace \  
  --action install-catalog \  
  --args "--inputDir ${OFFLINEDIR} --recursive"
```

2. Verify that `ibm-cpd-wml-operator-catalog` is `READY`:

```
oc get catalogsource -n openshift-marketplace ibm-cpd-wml-operator-catalog \  
-o jsonpath='{.status.connectionState.lastObservedState} {"\n"}'
```

>
Watson Machine Learning Accelerator

1. Run the following command to create the Watson Machine Learning Accelerator catalog source:

```
cloudctl case launch \  
  --case ${OFFLINEDIR}/ibm-wml-accelerator-2.3.2.tgz \  
  --inventory wmla_operator_deploy \  
  --namespace openshift-marketplace \  
  --action install-catalog \  
  --args "--inputDir ${OFFLINEDIR} --recursive"
```

2. Verify that `ibm-cpd-wml-accelerator-catalog` is `READY`:

```
oc get catalogsource -n openshift-marketplace ibm-cpd-wml-accelerator-catalog \  
-o jsonpath='{.status.connectionState.lastObservedState} {"\n"}'
```

> Watson OpenScale

1. Run the following command to create the Watson OpenScale catalog source:

```
cloudctl case launch \  
  --case ${OFFLINEDIR}/ibm-watson-openscale-2.2.0.tgz \  
  --inventory ibmWatsonOpenscaleOperatorSetup \  
  --namespace openshift-marketplace \  
  --action install-catalog \  
  --args "--inputDir ${OFFLINEDIR} --recursive"
```

2. Verify that `ibm-openscale-operator-catalog` is `READY`:

```
oc get catalogsource -n openshift-marketplace ibm-openscale-operator-catalog \  
-o jsonpath='{.status.connectionState.lastObservedState} {"\n"}'
```

> Watson Speech to Text

The same operator is used for the Watson Speech to Text service and the Watson Text to Speech service. You only need to create the catalog source once.

1. Run the following command to create the Watson Speech to Text catalog source:

```
cloudctl case launch \  
  --case ${OFFLINEDIR}/ibm-watson-speech-4.0.0.tgz \  
  --inventory speechOperatorSetup \  
  --namespace openshift-marketplace \  
  --action install-catalog \  
  --args "--registry ${PRIVATE_REGISTRY} --inputDir ${OFFLINEDIR} --recursive"
```

2. Verify that `ibm-watson-speech-operator-catalog` is `READY`:

```
oc get catalogsource -n openshift-marketplace ibm-watson-speech-operator-catalog \  
-o jsonpath='{.status.connectionState.lastObservedState} {"\n"}'
```

> Watson Studio

1. Run the following command to create the Watson Studio catalog source:

```
cloudctl case launch \  
  --case ${OFFLINEDIR}/ibm-wsl-2.0.2.tgz \  
  --inventory wslSetup \  
  --namespace openshift-marketplace \  
  --action install-catalog \  
  --args "--inputDir ${OFFLINEDIR} --recursive"
```

2. Verify that `ibm-cpd-ws-operator-catalog` is `READY`:

```
oc get catalogsource -n openshift-marketplace ibm-cpd-ws-operator-catalog \  
-o jsonpath='{.status.connectionState.lastObservedState} {"\n"}'
```

> Watson Studio Runtimes

1. Run the following command to create the Watson Studio Runtimes catalog source:

```
cloudctl case launch \  
  --case ${OFFLINEDIR}/ibm-wsl-runtimes-1.0.2.tgz \  
  --inventory runtimesOperatorSetup \  
  --namespace openshift-marketplace \  
  --action install-catalog \  
  --args "--inputDir ${OFFLINEDIR} --recursive"
```

2. Verify that `ibm-cpd-ws-runtimes-operator-catalog` is `READY`:

```
oc get catalogsource -n openshift-marketplace ibm-cpd-ws-runtimes-operator-catalog \  
-o jsonpath='{.status.connectionState.lastObservedState} {"\n"}'
```

> Watson Text to Speech

The same operator is used for the Watson Speech to Text service and the Watson Text to Speech service. You only need to create the catalog source once.

1. Run the following command to create the Watson Speech to Text catalog source:

```
cloudctl case launch \  
  --case ${OFFLINEDIR}/ibm-watson-speech-4.0.0.tgz \  
  --inventory speechOperatorSetup \  
  --namespace openshift-marketplace \  
  --action install-catalog \  
  --args "--registry ${PRIVATE_REGISTRY} --inputDir ${OFFLINEDIR} --recursive"
```

2. Verify that `ibm-watson-speech-operator-catalog` is `READY`:

```
oc get catalogsource -n openshift-marketplace ibm-watson-speech-operator-catalog \
-o jsonpath='{.status.connectionState.lastObservedState}' {"\n"}
```

Previous topic: [Mirroring images to your private container registry](#)

Next topic: [Installing IBM Cloud Pak foundational services](#)

Installing IBM Cloud Pak foundational services

IBM Cloud Pak® foundational services is a prerequisite for IBM® Cloud Pak for Data. IBM Cloud Pak foundational services is installed one time on the cluster and is used by any instances of Cloud Pak for Data or other IBM Cloud Paks that are installed on the cluster.

Permissions you need for this task

You must be a cluster administrator.

When you need to complete this task

Use the following guidance to determine if you need to complete this task:

- If IBM Cloud Pak foundational services is already installed, you can skip this task.
- If you are running an [express installation](#) of IBM Cloud Pak for Data, you can skip this task. The IBM Cloud Pak for Data platform operator will automatically install IBM Cloud Pak foundational services.
- If you are running a [specialized installation](#) of IBM Cloud Pak for Data and IBM Cloud Pak foundational services is not installed, you must complete this task.

Before you begin

Verify that you completed these tasks before you install IBM Cloud Pak foundational services:

1. For environments that use a private container registry, such as air-gapped environments, the IBM Cloud Pak foundational services images are mirrored to the private container registry. For details, see [Mirroring images to your private container registry](#).
2. The cluster is configured to pull the software images. For details, see [Configuring your cluster to pull Cloud Pak for Data images](#).

If you do not complete these steps, the IBM Cloud Pak foundational services installation will fail.

Procedure

To install IBM Cloud Pak foundational services:

1. Create the appropriate operator subscription for your environment:

Image location	Required catalog source
Publicly available registry (<code>quay.io/openshift</code>)	If you are pulling images from the publicly available registry, create the following operator subscription: <pre>cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-common-service-operator namespace: ibm-common-services spec: channel: v3 installPlanApproval: Automatic name: ibm-common-service-operator source: ibm-operator-catalog sourceNamespace: openshift-marketplace EOF</pre>
Private container registry	If you are pulling images from a private container registry, create the following operator subscription: <pre>cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-common-service-operator namespace: ibm-common-services spec: channel: v3 installPlanApproval: Automatic name: ibm-common-service-operator source: openshift-operators sourceNamespace: openshift-marketplace EOF</pre>

When you create the operator subscription:

- The IBM Cloud Pak foundational services operator installs the `Operand Deployment Lifecycle Manager` operator and the `IBM NamespaceScope` operator in the `ibm-common-services` project.
- The IBM Cloud Pak foundational services operator creates the `CommonService` custom resource.
- The `Operand Deployment Lifecycle Manager` operator creates the `OperandRegistry`, `OperandConfig`, and the `OperatorBindInfo` instances in the `ibm-common-services` project.

Verifying the installation

Verify the status of the operators by running the following commands:

1. Verify the status of `ibm-common-service-operator`:


```
oc --namespace ibm-common-services get csv
```

If you installed IBM Cloud Pak foundational services in a different project, replace `ibm-common-services` with the correct project name.

The command should return output with the following format:

NAME	DISPLAY	VERSION	REPLACES
ibm-common-service-operator.v3.10.0	IBM Cloud Pak foundational services	3.10.0	ibm-common-service-opera

Note: The preceding output is a sample and is intended to show the format of the output. Your installation might show a newer version of the IBM Cloud Pak foundational services. Ensure that the version of IBM Cloud Pak foundational services that is displayed in the output matches the version that is listed in [Operator and operand versions](#).

2. Verify that the custom resource definitions were created:

```
oc get crd | grep operandrequest
```

The command should return output with the following format:

NAME	CREATED AT
operandrequests.operator.ibm.com	2021-06-23T10:10:22Z

3. Confirm that IBM Cloud Pak foundational services API resources are available:

```
oc api-resources --api-group operator.ibm.com
```

The command should return output similar to the following:

NAME	SHORTNAMES	APIGROUP	NAMESPACED	KIND
commonservices		operator.ibm.com	true	CommonService
namespacescopes	nss	operator.ibm.com	true	NamespaceScope
operandbindinfos	opbi	operator.ibm.com	true	OperandBindInfo
operandconfigs	opcon	operator.ibm.com	true	OperandConfig
operandregistries	opreg	operator.ibm.com	true	OperandRegistry
operandrequests	opreq	operator.ibm.com	true	OperandRequest
podpresets		operator.ibm.com	true	PodPreset

Installing individual foundational services

The IBM Cloud Pak for Data platform operator automatically installs the following foundational services:

Certificate management service

The IBM Cloud Pak for Data platform operator requires the Certificate management service (`ibm-cert-manager-operator`). If the Certificate management service is not installed, the IBM Cloud Pak for Data platform operator automatically installs the service.

Identity and Access Management Service (IAM Service)

The IAM Service (`ibm-iam-operator`) is required if you [decide to integrate with the IAM Service](#). If the IAM Service is not installed, the IBM Cloud Pak for Data platform operator automatically installs the service.

Administration hub

The Administration hub (`ibm-commonui-operator`) is required if you [decide to integrate with the IAM Service](#). If the Administration hub is not installed, the IBM Cloud Pak for Data platform operator automatically installs the service.

If you want to install additional foundational services, such as the License Service, you must manually install them. For details, see [Installing foundational services in your cluster](#) in the IBM Cloud Pak foundational services documentation.

Additional considerations

By default, the **IBM NamespaceScope Operator** that is installed with IBM Cloud Pak foundational services has *cluster permissions* so that role binding projections can be completed automatically.

You can optionally remove the cluster permissions from the **IBM NamespaceScope Operator** and manually authorize the projections. For details, see [Authorizing foundational services to perform operations on workloads in a namespace](#).

Previous topic: [Configuring your cluster to pull Cloud Pak for Data images](#)

Next topic: [Creating operator subscriptions](#)

Creating operator subscriptions

An operator subscription tells the cluster where to install a given operator and gives information about the operator to Operator Lifecycle Manager (OLM).

When you create an operator subscription, OLM gets the cluster service version (CSV) for the operator. The CSV describes the operator, and OLM uses the CSV to:

- Introduce the custom resource definition (CRD) if it doesn't exist
- Set up the operator's service accounts
- Start up the operator deployment

Procedure

Complete the following tasks to create the relevant operator subscriptions on your cluster:

1. [Choosing an install plan](#)
2. [Creating an operator subscription for the scheduling service](#)
3. [Creating an operator subscription for the IBM Cloud Pak for Data platform operator](#)
4. [Enabling services to use namespace scoping with third-party operators](#)

- [5. Creating an operator subscription for services](#)

1. Choosing an install plan

When you create an operator subscription, you specify the install plan for the operator; you can specify whether you want to manually or automatically upgrade the operator. Read the following descriptions carefully to determine the appropriate install plan for your environment.

Automatic

If you specify `installPlanApproval: Automatic`, Red Hat® OpenShift® Container Platform will automatically load newer versions of the operator if they are available. For example, if you mirror images to a private container registry and you set the install plan to automatic, Red Hat OpenShift Container Platform will automatically use the latest version of the operator that is available in the private container registry.

Best practice: It is recommended that you use the automatic install plan to simplify the process of managing your environment, especially if you use a private container registry to host your images. However, it might not be appropriate for all environments, such as environments that pull images from the IBM® Entitled Registry and where you need to ensure that the software remains at the same version.

An operator *manages* the software. Upgrading the operator does not necessarily impact the version of the software that is running on the cluster. In many cases, upgrading the operator updates only the operator. However, if you remove the `version` entry from the Cloud Pak for Data custom resource or from a service's custom resource, the software is also updated when the operator is updated. If you don't want the software to be automatically updated when the operator is updated, ensure that you specify the version in the custom resource objects.

Some services, such as Db2® and the common core services do not include versions in their custom resource objects, so these services are automatically upgraded when a newer version of the operator is installed.

For an in-depth description of operators, see the [Red Hat OpenShift: Operators Framework video](#) from Red Hat.

Manual

If you specify `installPlanApproval: Manual`, OLM creates an update request when a newer version of an operator is available. A cluster administrator must manually approve the update request to update the operator to the newer version.

Important: If you pull images from the IBM Entitled Registry and you want to ensure that your software remains at the same version, use the manual install plan. The IBM Operator Catalog is refreshed any time any IBM operator is released. OLM handles any updates that it finds in the IBM Operator Catalog (`ibm-operator-catalog`) automatically.

All of the operator subscriptions in this topic specify the automatic install plan (`installPlanApproval: Automatic`). If you want to use the manual install plan, update the subscription to use `installPlanApproval: Manual` before you create the operator subscription on your environment.

2. Creating an operator subscription for the scheduling service

If you don't plan to install the scheduling service, you can skip this step.

Create the appropriate operator subscription for your environment:

Image location	Required subscriptions
IBM Entitled Registry	<p>If you are pulling images from the publicly available registry:</p> <ol style="list-style-type: none"> Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre> cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-cpd-scheduling-catalog-subscription namespace: ibm-common-services # Specify the project that contains the Cloud Pak foundational services operators spec: channel: v1.2 installPlanApproval: Automatic name: ibm-cpd-scheduling-operator source: ibm-operator-catalog sourceNamespace: openshift-marketplace EOF </pre> Validate that the operator was successfully created. <ol style="list-style-type: none"> Run the following command to confirm that the subscription was triggered: <pre> oc get sub -n ibm-common-services ibm-cpd-scheduling-catalog-subscription \ -o jsonpath='{.status.installedCSV} {"\n"}' </pre> Verify that the command returns <code>ibm-cpd-scheduling-operator.v1.2.3</code>. Run the following command to confirm that the cluster service version (CSV) is ready: <pre> oc get csv -n ibm-common-services ibm-cpd-scheduling-operator.v1.2.3 \ -o jsonpath='{.status.phase} : { .status.message} {"\n"}' </pre> Verify that the command returns <code>Succeeded : install strategy completed with no errors.</code> Run the following command to confirm that the operator is ready: <pre> oc get deployments -n ibm-common-services -l olm.owner="ibm-cpd-scheduling-operator.v1.2.3" \ -o jsonpath='{.items[0].status.availableReplicas} {"\n"}' </pre> Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.

Image location	Required subscriptions
Private container registry	<p>If you are pulling images from a private container registry,</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre data-bbox="414 233 1468 510"> cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-cpd-scheduling-catalog-subscription namespace: ibm-common-services # Specify the project that contains the Cloud Pak foundational services operators spec: channel: v1.2 installPlanApproval: Automatic name: ibm-cpd-scheduling-operator source: ibm-cpd-scheduling-catalog sourceNamespace: openshift-marketplace EOF </pre> 2. Validate that the operator was successfully created. <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre data-bbox="467 596 1263 636"> oc get sub -n ibm-common-services ibm-cpd-scheduling-catalog-subscription \ -o jsonpath='{.status.installedCSV} {"\n"}' </pre> <p>Verify that the command returns <code>ibm-cpd-scheduling-operator.v1.2.3</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre data-bbox="467 737 1211 777"> oc get csv -n ibm-common-services ibm-cpd-scheduling-operator.v1.2.3 \ -o jsonpath='{.status.phase} : { .status.message} {"\n"}' </pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre data-bbox="467 877 1458 917"> oc get deployments -n ibm-common-services -l olm.owner="ibm-cpd-scheduling-operator.v1.2.3" \ -o jsonpath='{.items[0].status.availableReplicas} {"\n"}' </pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

3. Creating an operator subscription for the IBM Cloud Pak for Data platform operator

Create the appropriate operator subscription for your environment:

Image location	Required subscriptions

Image location	Required subscriptions
IBM Entitled Registry	<p>If you are pulling images from the publicly available registry:</p> <ol style="list-style-type: none"> 1. Create the Cloud Pak for Data operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre> cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: cpd-operator namespace: ibm-common-services cpd-operators # Pick the project where you want to install the Cloud Pak for Data platform operator spec: channel: v2.0 installPlanApproval: Automatic name: cpd-platform-operator source: ibm-operator-catalog sourceNamespace: openshift-marketplace EOF </pre> 2. If you are running a specialized installation (installing the IBM Cloud Pak® for Data platform operator and the IBM Cloud Pak foundational services in separate projects), create an operator subscription for the <code>IBM NamespaceScope Operator</code> in the IBM Cloud Pak for Data platform operator project: <pre> cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-namespace-scope-operator namespace: cpd-operators spec: channel: v3 installPlanApproval: Automatic name: ibm-namespace-scope-operator source: ibm-operator-catalog sourceNamespace: openshift-marketplace EOF </pre> 3. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre> oc get sub -n operator-project cpd-operator \ -o jsonpath='{.status.installedCSV}' {"\n"} </pre> <p>Verify that the command returns <code>cpd-platform-operator.v2.0.4</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre> oc get csv -n operator-project cpd-platform-operator.v2.0.4 \ -o jsonpath='{.status.phase} : { .status.message}' {"\n"} </pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre> oc get deployments -n operator-project -l olm.owner="cpd-platform-operator.v2.0.4" \ -o jsonpath='{.items[0].status.availableReplicas}' {'\n'} </pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

Image location	Required subscriptions
Private container registry	<p>If you are pulling images from a private container registry:</p> <ol style="list-style-type: none"> 1. Create the Cloud Pak for Data operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre>cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: cpd-operator namespace: ibm-common-services cpd-operators # Pick the project where you want to install the Cloud Pak for Data platform operator spec: channel: v2.0 installPlanApproval: Automatic name: cpd-platform-operator source: cpd-platform sourceNamespace: openshift-marketplace EOF</pre> 2. If you are running a specialized installation (installing the IBM Cloud Pak for Data platform operator and the IBM Cloud Pak foundational services in separate projects), create an operator subscription for the IBM NamespaceScope Operator in the IBM Cloud Pak for Data platform operator project: <pre>cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-namespace-scope-operator namespace: cpd-operators spec: channel: v3 installPlanApproval: Automatic name: ibm-namespace-scope-operator source: opencloud-operators sourceNamespace: openshift-marketplace EOF</pre> 3. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre>oc get sub -n operator-project cpd-operator \ -o jsonpath='{.status.installedCSV}' {"\n"}</pre> <p>Verify that the command returns <code>cpd-platform-operator.v2.0.4</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre>oc get csv -n operator-project cpd-platform-operator.v2.0.4 \ -o jsonpath='{.status.phase} : { .status.message}' {"\n"}</pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre>oc get deployments -n operator-project -l olm.owner="cpd-platform-operator.v2.0.4" \ -o jsonpath='{.items[0].status.availableReplicas}' {"\n"}</pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

4. Enabling services to use namespace scoping with third-party operators

Some services have dependencies on third-party operators, such as open source operators. By default, the service has no control over the namespace scope of these third-party operators.

However, you can enable the `csvInjector` option in the **IBM NamespaceScope Operator** to ensure that the required third-party operators use the same namespace scope as the service that requires them. (Setting the `csvInjector` option enables a service to augment the cluster service version (CSV) of a third-party operator.)

To ensure that the service can update the CSV, you must enable the `csvInjector` option before you create the operator subscription for the service.

This setting is required for the following services:

- Data Virtualization
- Db2 Data Management Console
- IBM Match 360 with Watson™
- Watson Assistant
- Watson Assistant for Voice Interaction
- Watson Discovery
- Watson Speech to Text
- Watson Text to Speech

To update the **IBM NamespaceScope Operator**, run the appropriate command for your environment:

- Express installations

Run the following command to update the **IBM NamespaceScope Operator** in the **ibm-common-services** project:

```
oc patch NamespaceScope common-service \
-n ibm-common-services \
--type=merge \
--patch='{"spec": {"csvInjector": {"enable": true} } }'
```

- Specialized installations

Run the following command to update the **IBM NamespaceScope Operator** in the **cpd-operators** project:

```
cat <<EOF |oc apply -f -
apiVersion: operator.ibm.com/v1
kind: NamespaceScope
metadata:
  name: cpd-operators
  namespace: cpd-operators # (Default) Replace with the Cloud Pak for Data platform operator project name
spec:
  csvInjector:
    enable: true
  namespaceMembers:
    - cpd-operators # (Default) Replace with the Cloud Pak for Data platform operator project name
EOF
```

5. Creating an operator subscription for services

Create the operator subscription for each service that you plan to install.

- Anaconda Repository for IBM Cloud Pak for Data

Not applicable. For details, see [Installing Anaconda Repository for IBM Cloud Pak for Data](#).

- Analytics Engine Powered by Apache Spark

Create the appropriate operator subscription for your environment:

Image location	Required subscriptions
IBM Entitled Registry	<p>If you are pulling images from the publicly available registry:</p> <ol style="list-style-type: none"> Create the following operator subscription. (Ensure that you update the namespace parameter to specify the correct Red Hat OpenShift project.) <pre>cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: labels: app.kubernetes.io/instance: ibm-cpd-ae-operator-subscription app.kubernetes.io/managed-by: ibm-cpd-ae-operator app.kubernetes.io/name: ibm-cpd-ae-operator-subscription name: ibm-cpd-ae-operator-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: stable-v1 installPlanApproval: Automatic name: analyticsengine-operator source: ibm-operator-catalog sourceNamespace: openshift-marketplace EOF</pre> Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either ibm-common-services or cpd-operators) for the --namespace (-n) argument.</p> <ol style="list-style-type: none"> Run the following command to confirm that the subscription was triggered: <pre>oc get sub -n operator-project ibm-cpd-ae-operator-subscription \ -o jsonpath='{.status.installedCSV} {"\n"}'</pre> <p>Verify that the command returns ibm-cpd-ae.v1.0.2.</p> Run the following command to confirm that the cluster service version (CSV) is ready: <pre>oc get csv -n operator-project ibm-cpd-ae.v1.0.2 \ -o jsonpath='{.status.phase} : { .status.message} {"\n"}'</pre> <p>Verify that the command returns Succeeded : install strategy completed with no errors.</p> Run the following command to confirm that the operator is ready: <pre>oc get deployments -n operator-project -l olm.owner="ibm-cpd-ae.v1.0.2" \ -o jsonpath='{.items[0].status.availableReplicas} {"\n"}'</pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

Image location	Required subscriptions
Private container registry	<p>If you are pulling images from a private container registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre data-bbox="440 222 1507 575">cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: labels: app.kubernetes.io/instance: ibm-cpd-ae-operator-subscription app.kubernetes.io/managed-by: ibm-cpd-ae-operator app.kubernetes.io/name: ibm-cpd-ae-operator-subscription name: ibm-cpd-ae-operator-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: stable-v1 installPlanApproval: Automatic name: analyticsengine-operator source: ibm-cpd-ae-operator-catalog sourceNamespace: openshift-marketplace EOF</pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre data-bbox="493 730 1187 768">oc get sub -n operator-project ibm-cpd-ae-operator-subscription \ -o jsonpath='{.status.installedCSV}' {"\n"}</pre> <p>Verify that the command returns <code>ibm-cpd-ae.v1.0.2</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre data-bbox="493 869 1122 907">oc get csv -n operator-project ibm-cpd-ae.v1.0.2 \ -o jsonpath='{.status.phase} : { .status.message}' {"\n"}</pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre data-bbox="493 1008 1271 1050">oc get deployments -n operator-project -l olm.owner="ibm-cpd-ae.v1.0.2" \ -o jsonpath='{.items[0].status.availableReplicas}' {"\n"}</pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

>
Cognos Analytics

Create the appropriate operator subscription for your environment:

Image location	Required subscriptions
----------------	------------------------

Image location	Required subscriptions
IBM Entitled Registry	<p>If you are pulling images from the publicly available registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre data-bbox="440 216 1507 569"> cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-ca-operator-catalog-subscription labels: app.kubernetes.io/instance: ibm-ca-operator app.kubernetes.io/managed-by: ibm-ca-operator app.kubernetes.io/name: ibm-ca-operator namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v4.0 name: ibm-ca-operator installPlanApproval: Automatic source: ibm-operator-catalog sourceNamespace: openshift-marketplace EOF </pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre data-bbox="493 722 1227 762"> oc get sub -n operator-project ibm-ca-operator-catalog-subscription \ -o jsonpath='{.status.installedCSV} {"\n"}' </pre> <p>Verify that the command returns <code>ibm-ca-operator.v4.0.2</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre data-bbox="493 863 1122 903"> oc get csv -n operator-project ibm-ca-operator.v4.0.2 \ -o jsonpath='{.status.phase } : { .status.message} {"\n"}' </pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre data-bbox="493 1003 1325 1043"> oc get deployments -n operator-project -l olm.owner="ibm-ca-operator.v4.0.2" \ -o jsonpath='{.items[0].status.availableReplicas} {"\n"}' </pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

Image location	Required subscriptions
Private container registry	<p>If you are pulling images from a private container registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre data-bbox="440 216 1507 569"> cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-ca-operator-catalog-subscription labels: app.kubernetes.io/instance: ibm-ca-operator app.kubernetes.io/managed-by: ibm-ca-operator app.kubernetes.io/name: ibm-ca-operator namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v4.0 name: ibm-ca-operator installPlanApproval: Automatic source: ibm-ca-operator-catalog sourceNamespace: openshift-marketplace EOF </pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre data-bbox="493 724 1227 762"> oc get sub -n operator-project ibm-ca-operator-catalog-subscription \ -o jsonpath='{.status.installedCSV} {"\n"}' </pre> <p>Verify that the command returns <code>ibm-ca-operator.v4.0.2</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre data-bbox="493 863 1122 903"> oc get csv -n operator-project ibm-ca-operator.v4.0.2 \ -o jsonpath='{.status.phase } : { .status.message} {"\n"}' </pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre data-bbox="493 1003 1325 1043"> oc get deployments -n operator-project -l olm.owner="ibm-ca-operator.v4.0.2" \ -o jsonpath='{.items[0].status.availableReplicas} {"\n"}' </pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

>
Cognos Dashboards

Create the appropriate operator subscription for your environment:

Image location	Required subscriptions
----------------	------------------------

Image location	Required subscriptions
IBM Entitled Registry	<p>If you are pulling images from the publicly available registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre data-bbox="440 216 1507 569"> cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: labels: app.kubernetes.io/instance: ibm-cde-operator-subscription app.kubernetes.io/managed-by: ibm-cde-operator app.kubernetes.io/name: ibm-cde-operator-subscription name: ibm-cde-operator-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v1.0 installPlanApproval: Automatic name: ibm-cde-operator source: ibm-operator-catalog sourceNamespace: openshift-marketplace EOF </pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre data-bbox="493 722 1154 762"> oc get sub -n operator-project ibm-cde-operator-subscription \ -o jsonpath='{.status.installedCSV}' {"\n"} </pre> <p>Verify that the command returns <code>ibm-cpd-cde.v1.0.2</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre data-bbox="493 863 1122 903"> oc get csv -n operator-project ibm-cpd-cde.v1.0.2 \ -o jsonpath='{.status.phase } : { .status.message } {"\n"} </pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre data-bbox="493 1003 1281 1043"> oc get deployments -n operator-project -l olm.owner="ibm-cpd-cde.v1.0.2" \ -o jsonpath="{.items[0].status.availableReplicas}" {"\n"} </pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

Image location	Required subscriptions
Private container registry	<p>If you are pulling images from a private container registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre data-bbox="440 216 1507 569"> cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: labels: app.kubernetes.io/instance: ibm-cde-operator-subscription app.kubernetes.io/managed-by: ibm-cde-operator app.kubernetes.io/name: ibm-cde-operator-subscription name: ibm-cde-operator-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v1.0 installPlanApproval: Automatic name: ibm-cde-operator source: ibm-cde-operator-catalog sourceNamespace: openshift-marketplace EOF </pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre data-bbox="493 722 1154 762"> oc get sub -n operator-project ibm-cde-operator-subscription \ -o jsonpath='{.status.installedCSV}' {"\n"} </pre> <p>Verify that the command returns <code>ibm-cpd-cde.v1.0.2</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre data-bbox="493 863 1122 903"> oc get csv -n operator-project ibm-cpd-cde.v1.0.2 \ -o jsonpath='{.status.phase } : { .status.message } {"\n"} </pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre data-bbox="493 1003 1281 1043"> oc get deployments -n operator-project -l olm.owner="ibm-cpd-cde.v1.0.2" \ -o jsonpath="{.items[0].status.availableReplicas}" {"\n"} </pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

- > Data Refinery

The operator subscription for Data Refinery is automatically created when you create the operator subscription for either Watson Knowledge Catalog or Watson Studio.

- > Data Virtualization

Create the appropriate operator subscription for your environment:

Image location	Required subscriptions
----------------	------------------------

Image location	Required subscriptions
IBM Entitled Registry	<p>If you are pulling images from the publicly available registry:</p> <ol style="list-style-type: none"> <p>Create the Db2U operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.)</p> <pre>cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-db2u-operator namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v1.1 name: db2u-operator installPlanApproval: Automatic source: ibm-operator-catalog sourceNamespace: openshift-marketplace EOF</pre> <p>Validate that the operator was successfully created.</p> <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> <p>Run the following command to confirm that the subscription was triggered:</p> <pre>oc get sub -n operator-project ibm-db2u-operator \ -o jsonpath='{.status.installedCSV} {"\n"}'</pre> <p>Verify that the command returns <code>db2u-operator.v1.1.6</code>.</p> <p>Run the following command to confirm that the cluster service version (CSV) is ready:</p> <pre>oc get csv -n operator-project db2u-operator.v1.1.6 \ -o jsonpath='{.status.phase} : { .status.message} {"\n"}'</pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> <p>Run the following command to confirm that the operator is ready:</p> <pre>oc get deployments -n operator-project -l olm.owner="db2u-operator.v1.1.6" \ -o jsonpath="{.items[0].status.availableReplicas} {"\n"}"</pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p> <p>Create the Data Virtualization operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.)</p> <pre>cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-dv-operator-catalog-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v1.7 installPlanApproval: Automatic name: ibm-dv-operator source: ibm-operator-catalog sourceNamespace: openshift-marketplace EOF</pre> <p>Validate that the operator was successfully created.</p> <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> <p>Run the following command to confirm that the subscription was triggered:</p> <pre>oc get sub -n operator-project ibm-dv-operator-catalog-subscription \ -o jsonpath='{.status.installedCSV} {"\n"}'</pre> <p>Verify that the command returns <code>ibm-dv-operator.v1.7.2</code>.</p> <p>Run the following command to confirm that the cluster service version (CSV) is ready:</p> <pre>oc get csv -n operator-project ibm-dv-operator.v1.7.2 \ -o jsonpath='{.status.phase} : { .status.message} {"\n"}'</pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> <p>Run the following command to confirm that the operator is ready:</p> <pre>oc get deployments -n operator-project -l olm.owner="ibm-dv-operator.v1.7.2" \ -o jsonpath="{.items[0].status.availableReplicas} {"\n"}"</pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

Image location	Required subscriptions
Private container registry	<p>If you are pulling images from a private container registry:</p> <ol style="list-style-type: none"> 1. Create the Db2U operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre> cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-db2u-operator namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v1.1 name: db2u-operator installPlanApproval: Automatic source: ibm-db2uoperator-catalog sourceNamespace: openshift-marketplace EOF </pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre> oc get sub -n operator-project ibm-db2u-operator \ -o jsonpath='{.status.installedCSV} {"\n"}' </pre> <p>Verify that the command returns <code>db2u-operator.v1.1.6</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre> oc get csv -n operator-project db2u-operator.v1.1.6 \ -o jsonpath='{.status.phase} : { .status.message} {"\n"}' </pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre> oc get deployments -n operator-project -l olm.owner="db2u-operator.v1.1.6" \ -o jsonpath="{.items[0].status.availableReplicas} {'\n'}" </pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p> 3. Create the Data Virtualization operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre> cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-dv-operator-catalog-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v1.7 installPlanApproval: Automatic name: ibm-dv-operator source: ibm-dv-operator-catalog sourceNamespace: openshift-marketplace EOF </pre> 4. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre> oc get sub -n operator-project ibm-dv-operator-catalog-subscription \ -o jsonpath='{.status.installedCSV} {"\n"}' </pre> <p>Verify that the command returns <code>ibm-dv-operator.v1.7.2</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre> oc get csv -n operator-project ibm-dv-operator.v1.7.2 \ -o jsonpath='{.status.phase} : { .status.message} {"\n"}' </pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre> oc get deployments -n operator-project -l olm.owner="ibm-dv-operator.v1.7.2" \ -o jsonpath="{.items[0].status.availableReplicas} {'\n'}" </pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

>
DataStage

The operator subscription is the same for DataStage Enterprise or DataStage Enterprise Plus.

Create the appropriate operator subscription for your environment:

Image location	Required subscriptions
IBM Entitled Registry	<p>If you are pulling images from the publicly available registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre data-bbox="440 344 1507 625">cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-cpd-datastage-operator-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v1.0 installPlanApproval: Automatic name: ibm-cpd-datastage-operator source: ibm-operator-catalog sourceNamespace: openshift-marketplace EOF</pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre data-bbox="493 772 1260 814">oc get sub -n operator-project ibm-cpd-datastage-operator-subscription \ -o jsonpath='{.status.installedCSV} {"\n"}'</pre> <p>Verify that the command returns <code>ibm-cpd-datastage-operator.v1.0.0</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre data-bbox="493 915 1195 957">oc get csv -n operator-project ibm-cpd-datastage-operator.v1.0.0 \ -o jsonpath='{.status.phase} : { .status.message} {"\n"}'</pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre data-bbox="493 1058 1442 1100">oc get deployments -n operator-project -l olm.owner="ibm-cpd-datastage-operator.v1.0.0" \ -o jsonpath='{.items[0].status.availableReplicas} {"\n"}'</pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

Image location	Required subscriptions
Private container registry	<p>If you are pulling images from a private container registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre data-bbox="440 216 1507 491"> cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-cpd-datastage-operator-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v1.0 installPlanApproval: Automatic name: ibm-cpd-datastage-operator source: ibm-cpd-datastage-operator-catalog sourceNamespace: openshift-marketplace EOF </pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre data-bbox="493 642 1260 680"> oc get sub -n operator-project ibm-cpd-datastage-operator-subscription \ -o jsonpath='{.status.installedCSV}' {"\n"} </pre> <p>Verify that the command returns <code>ibm-cpd-datastage-operator.v1.0.0</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre data-bbox="493 783 1195 821"> oc get csv -n operator-project ibm-cpd-datastage-operator.v1.0.0 \ -o jsonpath='{.status.phase} : { .status.message}' {"\n"} </pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre data-bbox="493 924 1442 961"> oc get deployments -n operator-project -l olm.owner="ibm-cpd-datastage-operator.v1.0.0" \ -o jsonpath='{.items[0].status.availableReplicas}' {"\n"} </pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

>
Db2

Create the appropriate operator subscriptions for your environment:

Image location	Required subscriptions
----------------	------------------------

Image location	Required subscriptions
IBM Entitled Registry	<p>If you are pulling images from the publicly available registry:</p> <ol style="list-style-type: none"> Specialized installations only. By default, IBM Cloud Pak foundational services expects that the Db2U operator (<code>ibm-db2u-operator</code>) is installed in the <code>ibm-common-services</code> project. However, if you are using the <i>specialized</i> installation architecture (where the IBM Cloud Pak foundational services operators and Cloud Pak for Data operators are in separate OpenShift projects), you must edit the IBM Cloud Pak foundational services operand registry to point to the project where the Cloud Pak for Data operators are installed: <ol style="list-style-type: none"> Run the following command to edit the IBM Cloud Pak foundational services operand registry: <pre>oc edit operandRegistry common-service</pre> Change the value of the <code>namespace</code> entry for the <code>ibm-db2u-operator</code>. Specify the OpenShift project where the Cloud Pak for Data operators are installed, for example, <code>cpd-operators</code>: <pre>- channel: v1.1 installPlanApproval: Automatic name: ibm-db2u-operator namespace: cpd-operators # Specify the project where the Cloud Pak for Data operators are installed packageName: db2u-operator scope: public sourceName: ibm-db2uoperator-catalog sourceNamespace: openshift-marketplace</pre> Save your changes and exit. For example, if you are using <code>vi</code>, hit <code>esc</code> and enter <code>:wq</code>. Create the Db2 operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre>cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-db2oltp-cp4d-operator-catalog-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v1.0 name: ibm-db2oltp-cp4d-operator installPlanApproval: Automatic source: ibm-operator-catalog sourceNamespace: openshift-marketplace EOF</pre> Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> Run the following command to confirm that the subscription was triggered: <pre>oc get sub -n operator-project ibm-db2oltp-cp4d-operator-catalog-subscription \ -o jsonpath='{.status.installedCSV}' {"\n"}</pre> <p>Verify that the command returns <code>ibm-db2oltp-cp4d-operator.v1.0.3</code>.</p> Run the following command to confirm that the cluster service version (CSV) is ready: <pre>oc get csv -n operator-project ibm-db2oltp-cp4d-operator.v1.0.3 \ -o jsonpath='{.status.phase} : { .status.message}' {"\n"}</pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> Run the following command to confirm that the operator is ready: <pre>oc get deployments -n operator-project -l olm.owner="ibm-db2oltp-cp4d-operator.v1.0.3" \ -o jsonpath='{.items[0].status.availableReplicas}' {"\n"}</pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

Image location	Required subscriptions
Private container registry	<p>If you are pulling images from a private container registry:</p> <ol style="list-style-type: none"> Specialized installations only. By default, IBM Cloud Pak foundational services expects that the Db2U operator (<code>ibm-db2u-operator</code>) is installed in the <code>ibm-common-services</code> project. However, if you are using the <i>specialized</i> installation architecture (where the IBM Cloud Pak foundational services operators and Cloud Pak for Data operators are in separate OpenShift projects), you must edit the IBM Cloud Pak foundational services operand registry to point to the project where the Cloud Pak for Data operators are installed: <ol style="list-style-type: none"> Run the following command to edit the IBM Cloud Pak foundational services operand registry: <pre>oc edit operandRegistry common-service</pre> Change the value of the <code>namespace</code> entry for the <code>ibm-db2u-operator</code>. Specify the OpenShift project where the Cloud Pak for Data operators are installed, for example, <code>cpd-operators</code>: <pre>- channel: v1.1 installPlanApproval: Automatic name: ibm-db2u-operator namespace: cpd-operators # Specify the project where the Cloud Pak for Data operators are installed packageName: db2u-operator scope: public sourceName: ibm-db2uoperator-catalog sourceNamespace: openshift-marketplace</pre> Save your changes and exit. For example, if you are using <code>vi</code>, hit <code>esc</code> and enter <code>:wq</code>. Create the Db2 operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre>cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-db2oltp-cp4d-operator-catalog-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v1.0 name: ibm-db2oltp-cp4d-operator installPlanApproval: Automatic source: ibm-db2oltp-cp4d-operator-catalog sourceNamespace: openshift-marketplace EOF</pre> Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> Run the following command to confirm that the subscription was triggered: <pre>oc get sub -n operator-project ibm-db2oltp-cp4d-operator-catalog-subscription \ -o jsonpath='{.status.installedCSV}' {"\n"}</pre> <p>Verify that the command returns <code>ibm-db2oltp-cp4d-operator.v1.0.3</code>.</p> Run the following command to confirm that the cluster service version (CSV) is ready: <pre>oc get csv -n operator-project ibm-db2oltp-cp4d-operator.v1.0.3 \ -o jsonpath='{.status.phase} : { .status.message}' {"\n"}</pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> Run the following command to confirm that the operator is ready: <pre>oc get deployments -n operator-project -l olm.owner="ibm-db2oltp-cp4d-operator.v1.0.3" \ -o jsonpath="{.items[0].status.availableReplicas}" {"\n"}</pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

>
Db2 Big SQL

Create the appropriate operator subscription for your environment:

Image location	Required subscriptions
----------------	------------------------

Image location	Required subscriptions
IBM Entitled Registry	<p>If you are pulling images from the publicly available registry:</p> <ol style="list-style-type: none"> 1. Create the Db2U operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre>cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-db2u-operator namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v1.1 name: db2u-operator installPlanApproval: Automatic source: ibm-operator-catalog sourceNamespace: openshift-marketplace EOF</pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre>oc get sub -n operator-project ibm-db2u-operator \ -o jsonpath='{.status.installedCSV} {"\n"}'</pre> <p>Verify that the command returns <code>db2u-operator.v1.1.6</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre>oc get csv -n operator-project db2u-operator.v1.1.6 \ -o jsonpath='{.status.phase} : { .status.message} {"\n"}'</pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre>oc get deployments -n operator-project -l olm.owner="db2u-operator.v1.1.6" \ -o jsonpath="{.items[0].status.availableReplicas} {'\n'}"</pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p> 3. Create the Db2 Big SQL operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre>cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-bigsql-operator-catalog-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v7.2 installPlanApproval: Automatic name: ibm-bigsql-operator source: ibm-operator-catalog sourceNamespace: openshift-marketplace EOF</pre> 4. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre>oc get sub -n operator-project ibm-bigsql-operator-catalog-subscription \ -o jsonpath='{.status.installedCSV} {"\n"}'</pre> <p>Verify that the command returns <code>ibm-bigsql-operator.v7.2.2</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre>oc get csv -n operator-project ibm-bigsql-operator.v7.2.2 \ -o jsonpath='{.status.phase} : { .status.message} {"\n"}'</pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre>oc get deployments -n operator-project -l olm.owner="ibm-bigsql-operator.v7.2.2" \ -o jsonpath="{.items[0].status.availableReplicas} {'\n'}"</pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

Image location	Required subscriptions
Private container registry	<p>If you are pulling images from a private container registry:</p> <ol style="list-style-type: none"> 1. Create the Db2U operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre data-bbox="440 218 1479 495">cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-db2u-operator namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v1.1 name: db2u-operator installPlanApproval: Automatic source: ibm-db2uoperator-catalog sourceNamespace: openshift-marketplace EOF</pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre data-bbox="493 646 1024 684">oc get sub -n operator-project ibm-db2u-operator \ -o jsonpath='{.status.installedCSV} {"\n"}'</pre> <p>Verify that the command returns <code>db2u-operator.v1.1.6</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre data-bbox="493 789 1122 827">oc get csv -n operator-project db2u-operator.v1.1.6 \ -o jsonpath='{.status.phase} : { .status.message} {"\n"}'</pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre data-bbox="493 926 1300 963">oc get deployments -n operator-project -l olm.owner="db2u-operator.v1.1.6" \ -o jsonpath="{.items[0].status.availableReplicas} {"\n"}"</pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p> 3. Create the Db2 Big SQL operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre data-bbox="440 1115 1507 1392">cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-bigsql-operator-catalog-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v7.2 installPlanApproval: Automatic name: ibm-bigsql-operator source: ibm-bigsql-operator-catalog sourceNamespace: openshift-marketplace EOF</pre> 4. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre data-bbox="493 1541 1268 1579">oc get sub -n operator-project ibm-bigsql-operator-catalog-subscription \ -o jsonpath='{.status.installedCSV} {"\n"}'</pre> <p>Verify that the command returns <code>ibm-bigsql-operator.v7.2.2</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre data-bbox="493 1682 1122 1719">oc get csv -n operator-project ibm-bigsql-operator.v7.2.2 \ -o jsonpath='{.status.phase} : { .status.message} {"\n"}'</pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre data-bbox="493 1818 1365 1856">oc get deployments -n operator-project -l olm.owner="ibm-bigsql-operator.v7.2.2" \ -o jsonpath="{.items[0].status.availableReplicas} {"\n"}"</pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

>
Db2 Data Gate

Create the appropriate operator subscription for your environment:

Image location	Required subscriptions
IBM Entitled Registry	<p>If you are pulling images from the publicly available registry:</p> <ol style="list-style-type: none">1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre>cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: labels: app.kubernetes.io/instance: ibm-datagate-operator-subscription app.kubernetes.io/managed-by: ibm-datagate-operator app.kubernetes.io/name: ibm-datagate-operator-subscription name: ibm-datagate-operator-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v2.0 installPlanApproval: Automatic name: ibm-datagate-operator source: ibm-operator-catalog sourceNamespace: openshift-marketplace EOF</pre>2. Validate that the operator was successfully created. For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.<ol style="list-style-type: none">a. Run the following command to confirm that the subscription was triggered: <pre>oc get sub -n operator-project ibm-datagate-operator-subscription \ -o jsonpath='{.status.installedCSV}' {"\n"}</pre><p>Verify that the command returns <code>ibm-datagate-operator.v2.0.2</code>.</p>b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre>oc get csv -n operator-project ibm-datagate-operator.v2.0.2 \ -o jsonpath='{.status.phase } : { .status.message}' {"\n"}</pre><p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p>c. Run the following command to confirm that the operator is ready: <pre>oc get deployments -n operator-project -l olm.owner="ibm-datagate-operator.v2.0.2" \ -o jsonpath='{.items[0].status.availableReplicas}' {"\n"}</pre><p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

Image location	Required subscriptions
Private container registry	<p>If you are pulling images from a private container registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre data-bbox="440 216 1500 569"> cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: labels: app.kubernetes.io/instance: ibm-datagate-operator-subscription app.kubernetes.io/managed-by: ibm-datagate-operator app.kubernetes.io/name: ibm-datagate-operator-subscription name: ibm-datagate-operator-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v2.0 installPlanApproval: Automatic name: ibm-datagate-operator source: ibm-datagate-operator-catalog sourceNamespace: openshift-marketplace EOF </pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre data-bbox="493 722 1206 762"> oc get sub -n operator-project ibm-datagate-operator-subscription \ -o jsonpath='{.status.installedCSV}' {"\n"} </pre> <p>Verify that the command returns <code>ibm-datagate-operator.v2.0.2</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre data-bbox="493 863 1143 903"> oc get csv -n operator-project ibm-datagate-operator.v2.0.2 \ -o jsonpath='{.status.phase} : { .status.message}' {"\n"} </pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre data-bbox="493 1031 1386 1071"> oc get deployments -n operator-project -l olm.owner="ibm-datagate-operator.v2.0.2" \ -o jsonpath='{.items[0].status.availableReplicas}' {"\n"} </pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

>
 Db2 Data Management Console

Create the appropriate operator subscription for your environment:

Image location	Required subscriptions
----------------	------------------------

Image location	Required subscriptions
IBM Entitled Registry	<p>If you are pulling images from the publicly available registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre data-bbox="440 216 1507 491">cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-dmc-operator-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v1.0 installPlanApproval: Automatic name: ibm-dmc-operator source: ibm-operator-catalog sourceNamespace: openshift-marketplace EOF</pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre data-bbox="493 642 1154 680">oc get sub -n operator-project ibm-dmc-operator-subscription \ -o jsonpath='{.status.installedCSV}' {"\n"}'</pre> <p>Verify that the command returns <code>ibm-databases-dmc.v1.0.2</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre data-bbox="493 785 1122 823">oc get csv -n operator-project ibm-databases-dmc.v1.0.2 \ -o jsonpath='{.status.phase} : { .status.message}' {"\n"}'</pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre data-bbox="493 951 1344 989">oc get deployments -n operator-project -l olm.owner="ibm-databases-dmc.v1.0.2" \ -o jsonpath='{.items[0].status.availableReplicas}' {"\n"}'</pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>
Private container registry	<p>If you are pulling images from a private container registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre data-bbox="440 1150 1507 1425">cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-dmc-operator-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v1.0 installPlanApproval: Automatic name: ibm-dmc-operator source: ibm-dmc-operator-catalog sourceNamespace: openshift-marketplace EOF</pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre data-bbox="493 1577 1154 1614">oc get sub -n operator-project ibm-dmc-operator-subscription \ -o jsonpath='{.status.installedCSV}' {"\n"}'</pre> <p>Verify that the command returns <code>ibm-databases-dmc.v1.0.2</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre data-bbox="493 1719 1122 1757">oc get csv -n operator-project ibm-databases-dmc.v1.0.2 \ -o jsonpath='{.status.phase} : { .status.message}' {"\n"}'</pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre data-bbox="493 1864 1344 1902">oc get deployments -n operator-project -l olm.owner="ibm-databases-dmc.v1.0.2" \ -o jsonpath='{.items[0].status.availableReplicas}' {"\n"}'</pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

- >
Db2 Event Store

Not applicable. Contact IBM Software support if you plan to install this service.

- >
Db2 Warehouse

Create the appropriate operator subscription for your environment:

Image location	Required subscriptions
IBM Entitled Registry	<p>If you are pulling images from the publicly available registry:</p> <ol style="list-style-type: none"> Specialized installations only. By default, IBM Cloud Pak foundational services expects that the Db2U operator (<code>ibm-db2u-operator</code>) is installed in the <code>ibm-common-services</code> project. However, if you are using the <i>specialized</i> installation architecture (where the IBM Cloud Pak foundational services operators and Cloud Pak for Data operators are in separate OpenShift projects), you must edit the IBM Cloud Pak foundational services operand registry to point to the project where the Cloud Pak for Data operators are installed: <ol style="list-style-type: none"> Run the following command to edit the IBM Cloud Pak foundational services operand registry: <pre>oc edit operandRegistry common-service</pre> Change the value of the <code>namespace</code> entry for the <code>ibm-db2u-operator</code>. Specify the OpenShift project where the Cloud Pak for Data operators are installed, for example, <code>cpd-operators</code>: <pre>- channel: v1.1 installPlanApproval: Automatic name: ibm-db2u-operator namespace: cpd-operators # Specify the project where the Cloud Pak for Data operators are installed packageName: db2u-operator scope: public sourceName: ibm-db2uoperator-catalog sourceNamespace: openshift-marketplace</pre> Save your changes and exit. For example, if you are using <code>vi</code>, hit <code>esc</code> and enter <code>:wq</code>. Create the Db2 Warehouse operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre>cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-db2wh-cp4d-operator-catalog-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v1.0 name: ibm-db2wh-cp4d-operator installPlanApproval: Automatic source: ibm-operator-catalog sourceNamespace: openshift-marketplace EOF</pre> Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> Run the following command to confirm that the subscription was triggered: <pre>oc get sub -n operator-project ibm-db2wh-cp4d-operator-catalog-subscription \ -o jsonpath='{.status.installedCSV} {"\n"}'</pre> <p>Verify that the command returns <code>ibm-db2wh-cp4d-operator.v1.0.3</code>.</p> Run the following command to confirm that the cluster service version (CSV) is ready: <pre>oc get csv -n operator-project ibm-db2wh-cp4d-operator.v1.0.3 \ -o jsonpath='{.status.phase} : { .status.message} {"\n"}'</pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> Run the following command to confirm that the operator is ready: <pre>oc get deployments -n operator-project -l olm.owner="ibm-db2wh-cp4d-operator.v1.0.3" \ -o jsonpath="{.items[0].status.availableReplicas} {"\n"}"</pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

Image location	Required subscriptions
Private container registry	<p>If you are pulling images from a private container registry:</p> <ol style="list-style-type: none"> Specialized installations only. By default, IBM Cloud Pak foundational services expects that the Db2U operator (<code>ibm-db2u-operator</code>) is installed in the <code>ibm-common-services</code> project. However, if you are using the <i>specialized</i> installation architecture (where the IBM Cloud Pak foundational services operators and Cloud Pak for Data operators are in separate OpenShift projects), you must edit the IBM Cloud Pak foundational services operand registry to point to the project where the Cloud Pak for Data operators are installed: <ol style="list-style-type: none"> Run the following command to edit the IBM Cloud Pak foundational services operand registry: <pre>oc edit operandRegistry common-service</pre> Change the value of the <code>namespace</code> entry for the <code>ibm-db2u-operator</code>. Specify the OpenShift project where the Cloud Pak for Data operators are installed, for example, <code>cpd-operators</code>: <pre>- channel: v1.1 installPlanApproval: Automatic name: ibm-db2u-operator namespace: cpd-operators # Specify the project where the Cloud Pak for Data operators are installed packageName: db2u-operator scope: public sourceName: ibm-db2uoperator-catalog sourceNamespace: openshift-marketplace</pre> Save your changes and exit. For example, if you are using <code>vi</code>, hit <code>esc</code> and enter <code>:wq</code>. Create the Db2 Warehouse operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre>cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-db2wh-cp4d-operator-catalog-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v1.0 name: ibm-db2wh-cp4d-operator installPlanApproval: Automatic source: ibm-db2wh-cp4d-operator-catalog sourceNamespace: openshift-marketplace EOF</pre> Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> Run the following command to confirm that the subscription was triggered: <pre>oc get sub -n operator-project ibm-db2wh-cp4d-operator-catalog-subscription \ -o jsonpath='{.status.installedCSV}' {"\n"}'</pre> <p>Verify that the command returns <code>ibm-db2wh-cp4d-operator.v1.0.3</code>.</p> Run the following command to confirm that the cluster service version (CSV) is ready: <pre>oc get csv -n operator-project ibm-db2wh-cp4d-operator.v1.0.3 \ -o jsonpath='{.status.phase} : { .status.message}' {"\n"}'</pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> Run the following command to confirm that the operator is ready: <pre>oc get deployments -n operator-project -l olm.owner="ibm-db2wh-cp4d-operator.v1.0.3" \ -o jsonpath="{.items[0].status.availableReplicas}" {"\n"}"</pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

>
Decision Optimization

Create the appropriate operator subscription for your environment:

Image location	Required subscriptions
----------------	------------------------

Image location	Required subscriptions
IBM Entitled Registry	<p>If you are pulling images from the publicly available registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre data-bbox="440 216 1507 569"> cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: labels: app.kubernetes.io/instance: ibm-cpd-dods-operator-catalog-subscription app.kubernetes.io/managed-by: ibm-cpd-dods-operator app.kubernetes.io/name: ibm-cpd-dods-operator-catalog-subscription name: ibm-cpd-dods-operator-catalog-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v4.0 installPlanApproval: Automatic name: ibm-cpd-dods source: ibm-operator-catalog sourceNamespace: openshift-marketplace EOF </pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre data-bbox="493 722 1292 762"> oc get sub -n operator-project ibm-cpd-dods-operator-catalog-subscription \ -o jsonpath='{.status.installedCSV} {"\n"}' </pre> <p>Verify that the command returns <code>ibm-cpd-dods.v4.0.2</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre data-bbox="493 863 1122 903"> oc get csv -n operator-project ibm-cpd-dods.v4.0.2 \ -o jsonpath='{.status.phase } : { .status.message} {"\n"}' </pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre data-bbox="493 1003 1292 1043"> oc get deployments -n operator-project -l olm.owner="ibm-cpd-dods.v4.0.2" \ -o jsonpath="{.items[0].status.availableReplicas} {"\n"}" </pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

Image location	Required subscriptions
Private container registry	<p>If you are pulling images from a private container registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre data-bbox="440 216 1507 569"> cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: labels: app.kubernetes.io/instance: ibm-cpd-dods-operator-catalog-subscription app.kubernetes.io/managed-by: ibm-cpd-dods-operator app.kubernetes.io/name: ibm-cpd-dods-operator-catalog-subscription name: ibm-cpd-dods-operator-catalog-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v4.0 installPlanApproval: Automatic name: ibm-cpd-dods source: ibm-cpd-dods-operator-catalog sourceNamespace: openshift-marketplace EOF </pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre data-bbox="493 722 1292 762"> oc get sub -n operator-project ibm-cpd-dods-operator-catalog-subscription \ -o jsonpath='{.status.installedCSV} {"\n"}' </pre> <p>Verify that the command returns <code>ibm-cpd-dods.v4.0.2</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre data-bbox="493 863 1122 903"> oc get csv -n operator-project ibm-cpd-dods.v4.0.2 \ -o jsonpath='{.status.phase } : { .status.message} {"\n"}' </pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre data-bbox="493 1003 1292 1043"> oc get deployments -n operator-project -l olm.owner="ibm-cpd-dods.v4.0.2" \ -o jsonpath="{.items[0].status.availableReplicas} {"\n"}" </pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

>
EDB Postgres

Create the appropriate operator subscription for your environment:

Image location	Required subscriptions
----------------	------------------------

Image location	Required subscriptions
IBM Entitled Registry	<p>If you are pulling images from the publicly available registry:</p> <ol style="list-style-type: none"> Create the PostgreSQL operator subscription. Important: This operator is installed in the <code>openshift-operators</code> project. The operator does not support namespace scoping and must be installed as cluster-wide resource. <pre>cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: cloud-native-postgresql-catalog-subscription namespace: openshift-operators spec: channel: stable name: cloud-native-postgresql installPlanApproval: Automatic source: ibm-operator-catalog sourceNamespace: openshift-marketplace EOF</pre> Validate that the operator was successfully created. <ol style="list-style-type: none"> Run the following command to confirm that the subscription was triggered: <pre>oc get sub -n openshift-operators cloud-native-postgresql-catalog-subscription \ -o jsonpath='{.status.installedCSV} {"\n"}'</pre> Verify that the command returns <code>cloud-native-postgresql.v1.8.0</code>. Run the following command to confirm that the cluster service version (CSV) is ready: <pre>oc get csv -n openshift-operators cloud-native-postgresql.v1.8.0 \ -o jsonpath='{.status.phase} : { .status.message} {"\n"}'</pre> Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>. Run the following command to confirm that the operator is ready: <pre>oc get deployments -n openshift-operators -l olm.owner="cloud-native-postgresql.v1.8.0" \ -o jsonpath="{.items[0].status.availableReplicas} {'\n'}"</pre> Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available. Create the EDB Postgres operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre>cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-cpd-edb-operator-catalog-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data spec: installPlanApproval: Automatic channel: v4.0 name: ibm-cpd-edb source: ibm-operator-catalog sourceNamespace: openshift-marketplace EOF</pre> Validate that the operator was successfully created. For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument. <ol style="list-style-type: none"> Run the following command to confirm that the subscription was triggered: <pre>oc get sub -n operator-project ibm-cpd-edb-operator-catalog-subscription \ -o jsonpath='{.status.installedCSV} {"\n"}'</pre> Verify that the command returns <code>ibm-cpd-edb.v4.0.2</code>. Run the following command to confirm that the cluster service version (CSV) is ready: <pre>oc get csv -n operator-project ibm-cpd-edb.v4.0.2 \ -o jsonpath='{.status.phase} : { .status.message} {"\n"}'</pre> Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>. Run the following command to confirm that the operator is ready: <pre>oc get deployments -n operator-project -l olm.owner="ibm-cpd-edb.v4.0.2" \ -o jsonpath="{.items[0].status.availableReplicas} {'\n'}"</pre> Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.

Image location	Required subscriptions
Private container registry	<p>If you are pulling images from a private container registry:</p> <ol style="list-style-type: none"> 1. Create the PostgreSQL operator subscription. Important: This operator is installed in the openshift-operators project. The operator does not support namespace scoping and must be installed as cluster-wide resource. <pre>cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: cloud-native-postgresql-catalog-subscription namespace: openshift-operators spec: channel: stable name: cloud-native-postgresql installPlanApproval: Automatic source: cloud-native-postgresql-catalog sourceNamespace: openshift-marketplace EOF</pre> <ol style="list-style-type: none"> 2. Validate that the operator was successfully created. <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre>oc get sub -n openshift-operators cloud-native-postgresql-catalog-subscription \ -o jsonpath='{.status.installedCSV} {"\n"}'</pre> Verify that the command returns cloud-native-postgresql.v1.8.0. b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre>oc get csv -n openshift-operators cloud-native-postgresql.v1.8.0 \ -o jsonpath='{.status.phase} : { .status.message} {"\n"}'</pre> Verify that the command returns Succeeded : install strategy completed with no errors. c. Run the following command to confirm that the operator is ready: <pre>oc get deployments -n openshift-operators -l olm.owner="cloud-native-postgresql.v1.8.0" \ -o jsonpath="{.items[0].status.availableReplicas} {"\n"}"</pre> Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available. 3. Create the EDB Postgres operator subscription. (Ensure that you update the namespace parameter to specify the correct Red Hat OpenShift project.) <pre>cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-cpd-edb-operator-catalog-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data spec: installPlanApproval: Automatic channel: v4.0 name: ibm-cpd-edb source: ibm-cpd-edb-operator-catalog sourceNamespace: openshift-marketplace EOF</pre> 4. Validate that the operator was successfully created. For each command, ensure that you specify the appropriate Red Hat OpenShift project (either ibm-common-services or cpd-operators) for the --namespace (-n) argument. <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre>oc get sub -n operator-project ibm-cpd-edb-operator-catalog-subscription \ -o jsonpath='{.status.installedCSV} {"\n"}'</pre> Verify that the command returns ibm-cpd-edb.v4.0.2. b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre>oc get csv -n operator-project ibm-cpd-edb.v4.0.2 \ -o jsonpath='{.status.phase} : { .status.message} {"\n"}'</pre> Verify that the command returns Succeeded : install strategy completed with no errors. c. Run the following command to confirm that the operator is ready: <pre>oc get deployments -n operator-project -l olm.owner="ibm-cpd-edb.v4.0.2" \ -o jsonpath="{.items[0].status.availableReplicas} {"\n"}"</pre> Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.

>
Execution Engine for Apache Hadoop

Create the appropriate operator subscription for your environment:

Image location	Required subscriptions
----------------	------------------------

Image location	Required subscriptions
IBM Entitled Registry	<p>If you are pulling images from the publicly available registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre data-bbox="440 216 1507 569"> cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: labels: app.kubernetes.io/instance: ibm-cpd-hadoop-operator-catalog-subscription app.kubernetes.io/managed-by: ibm-cpd-hadoop-operator app.kubernetes.io/name: ibm-cpd-hadoop-operator-catalog-subscription name: ibm-cpd-hadoop-operator-catalog-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v1.0 installPlanApproval: Automatic name: ibm-cpd-hadoop source: ibm-operator-catalog sourceNamespace: openshift-marketplace EOF </pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre data-bbox="493 722 1312 762"> oc get sub -n operator-project ibm-cpd-hadoop-operator-catalog-subscription \ -o jsonpath='{.status.installedCSV}' {"\n"} </pre> <p>Verify that the command returns <code>ibm-cpd-hadoop.v1.0.2</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre data-bbox="493 863 1120 903"> oc get csv -n operator-project ibm-cpd-hadoop.v1.0.2 \ -o jsonpath='{.status.phase} : { .status.message}' {"\n"} </pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre data-bbox="493 1003 1312 1043"> oc get deployments -n operator-project -l olm.owner="ibm-cpd-hadoop.v1.0.2" \ -o jsonpath="{.items[0].status.availableReplicas}" {"\n"} </pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

Image location	Required subscriptions
Private container registry	<p>If you are pulling images from a private container registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre data-bbox="440 216 1507 569"> cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: labels: app.kubernetes.io/instance: ibm-cpd-hadoop-operator-catalog-subscription app.kubernetes.io/managed-by: ibm-cpd-hadoop-operator app.kubernetes.io/name: ibm-cpd-hadoop-operator-catalog-subscription name: ibm-cpd-hadoop-operator-catalog-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v1.0 installPlanApproval: Automatic name: ibm-cpd-hadoop source: ibm-cpd-hadoop-operator-catalog sourceNamespace: openshift-marketplace EOF </pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre data-bbox="493 722 1312 762"> oc get sub -n operator-project ibm-cpd-hadoop-operator-catalog-subscription \ -o jsonpath='{.status.installedCSV}' {"\n"} </pre> <p>Verify that the command returns <code>ibm-cpd-hadoop.v1.0.2</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre data-bbox="493 863 1122 903"> oc get csv -n operator-project ibm-cpd-hadoop.v1.0.2 \ -o jsonpath='{.status.phase} : { .status.message}' {"\n"} </pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre data-bbox="493 1003 1312 1043"> oc get deployments -n operator-project -l olm.owner="ibm-cpd-hadoop.v1.0.2" \ -o jsonpath="{.items[0].status.availableReplicas}" {"\n"} </pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

- >

Financial Services Workbench

Not applicable. For details, see the [Financial Services Workbench documentation](#).

- >

IBM Match 360 with Watson

Create the appropriate operator subscription for your environment:

Image location	Required subscriptions
----------------	------------------------

Image location	Required subscriptions
IBM Entitled Registry	<p>If you are pulling images from the publicly available registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre data-bbox="440 216 1507 569"> cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: labels: app.kubernetes.io/instance: ibm-mdm-operator-subscription app.kubernetes.io/managed-by: ibm-mdm-operator app.kubernetes.io/name: ibm-mdm-operator-subscription name: ibm-mdm-operator-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v1.1 installPlanApproval: Automatic name: ibm-mdm source: ibm-operator-catalog sourceNamespace: openshift-marketplace EOF </pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre data-bbox="493 722 1154 762"> oc get sub -n operator-project ibm-mdm-operator-subscription \ -o jsonpath='{.status.installedCSV} {"\n"}' </pre> <p>Verify that the command returns <code>ibm-mdm.v1.1.90</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre data-bbox="493 890 1122 930"> oc get csv -n operator-project ibm-mdm.v1.1.90 \ -o jsonpath='{.status.phase} : { .status.message} {"\n"}' </pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre data-bbox="493 1031 1252 1071"> oc get deployments -n operator-project -l olm.owner="ibm-mdm.v1.1.90" \ -o jsonpath='{.items[0].status.availableReplicas} {"\n"}' </pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

Image location	Required subscriptions
Private container registry	<p>If you are pulling images from a private container registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre data-bbox="440 216 1507 569"> cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: labels: app.kubernetes.io/instance: ibm-mdm-operator-subscription app.kubernetes.io/managed-by: ibm-mdm-operator app.kubernetes.io/name: ibm-mdm-operator-subscription name: ibm-mdm-operator-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v1.1 installPlanApproval: Automatic name: ibm-mdm source: ibm-mdm-operator-catalog sourceNamespace: openshift-marketplace EOF </pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre data-bbox="493 722 1154 762"> oc get sub -n operator-project ibm-mdm-operator-subscription \ -o jsonpath='{.status.installedCSV}' {"\n"} </pre> <p>Verify that the command returns <code>ibm-mdm.v1.1.90</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre data-bbox="493 863 1122 903"> oc get csv -n operator-project ibm-mdm.v1.1.90 \ -o jsonpath='{.status.phase} : { .status.message}' {"\n"} </pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre data-bbox="493 1031 1248 1071"> oc get deployments -n operator-project -l olm.owner="ibm-mdm.v1.1.90" \ -o jsonpath='{.items[0].status.availableReplicas}' {"\n"} </pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

>
Informix

Create the appropriate operator subscription for your environment:

Image location	Required subscriptions
----------------	------------------------

Image location	Required subscriptions
IBM Entitled Registry	<p>If you are pulling images from the publicly available registry:</p> <ol style="list-style-type: none"> Create the following operator subscription for the install operator. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre>cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-informix-operator-catalog-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: stable installPlanApproval: Automatic name: ibm-informix-operator source: ibm-operator-catalog sourceNamespace: openshift-marketplace EOF</pre> Validate that the install operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> Run the following command to confirm that the subscription was triggered: <pre>oc get sub -n operator-project ibm-informix-operator-catalog-subscription \ -o jsonpath='{.status.installedCSV} {"\n"}'</pre> <p>Verify that the command returns <code>ibm-informix-operator.v4.0.0</code>.</p> Run the following command to confirm that the cluster service version (CSV) is ready: <pre>oc get csv -n operator-project ibm-informix-operator.v4.0.0 \ -o jsonpath='{.status.phase} : { .status.message} {"\n"}'</pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> Run the following command to confirm that the operator is ready: <pre>oc get deployments -n operator-project -l olm.owner="ibm-informix-operator.v4.0.0" \ -o jsonpath='{.items[0].status.availableReplicas} {"\n"}'</pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p> Create the following operator subscription for the deployment operator. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre>cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-informix-cp4d-operator-catalog-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v1.0 installPlanApproval: Automatic name: ibm-informix-cp4d-operator source: ibm-operator-catalog sourceNamespace: openshift-marketplace EOF</pre> Validate that the deployment operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> Run the following command to confirm that the subscription was triggered: <pre>oc get sub -n operator-project ibm-informix-cp4d-operator-catalog-subscription \ -o jsonpath='{.status.installedCSV} {"\n"}'</pre> <p>Verify that the command returns <code>ibm-informix-cp4d-operator.v4.0.0</code>.</p> Run the following command to confirm that the cluster service version (CSV) is ready: <pre>oc get csv -n operator-project ibm-informix-cp4d-operator.v4.0.0 \ -o jsonpath='{.status.phase} : { .status.message} {"\n"}'</pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> Run the following command to confirm that the operator is ready: <pre>oc get deployments -n operator-project -l olm.owner="ibm-informix-cp4d-operator.v4.0.0" \ -o jsonpath='{.items[0].status.availableReplicas} {"\n"}'</pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

Image location	Required subscriptions
Private container registry	<p>If you are pulling images from a private container registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription for the install operator. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre>cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-informix-operator-catalog-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: stable installPlanApproval: Automatic name: ibm-informix-operator source: ibm-informix-operator-catalog sourceNamespace: openshift-marketplace EOF</pre> 2. Validate that the install operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre>oc get sub -n operator-project ibm-informix-operator-catalog-subscription \ -o jsonpath='{.status.installedCSV} {"\n"}'</pre> <p>Verify that the command returns <code>ibm-informix-operator.v4.0.0</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre>oc get csv -n operator-project ibm-informix-operator.v4.0.0 \ -o jsonpath='{.status.phase} : { .status.message} {"\n"}'</pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre>oc get deployments -n operator-project -l olm.owner="ibm-informix-operator.v4.0.0" \ -o jsonpath="{.items[0].status.availableReplicas} {"\n"}"</pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p> 3. Create the following operator subscription for the deployment operator. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre>cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-informix-cp4d-operator-catalog-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v1.0 installPlanApproval: Automatic name: ibm-informix-cp4d-operator source: ibm-informix-cp4d-operator-catalog sourceNamespace: openshift-marketplace EOF</pre> 4. Validate that the deployment operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre>oc get sub -n operator-project ibm-informix-cp4d-operator-catalog-subscription \ -o jsonpath='{.status.installedCSV} {"\n"}'</pre> <p>Verify that the command returns <code>ibm-informix-cp4d-operator.v4.0.0</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre>oc get csv -n operator-project ibm-informix-cp4d-operator.v4.0.0 \ -o jsonpath='{.status.phase} : { .status.message} {"\n"}'</pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre>oc get deployments -n operator-project -l olm.owner="ibm-informix-cp4d-operator.v4.0.0" \ -o jsonpath="{.items[0].status.availableReplicas} {"\n"}"</pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

>
MongoDB

Create the appropriate operator subscription for your environment:

Image location	Required subscriptions
IBM Entitled Registry	<p>If you are pulling images from the publicly available registry:</p> <ol style="list-style-type: none"> 1. Create the MongoDB Enterprise (third-party) operator subscription. Important: This operator is installed in the openshift-operators project. The operator does not support namespace scoping and must be installed as cluster-wide resource. <pre>cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-mongodb-enterprise-catalog-subscription namespace: openshift-operators spec: channel: stable name: mongodb-enterprise installPlanApproval: Automatic source: ibm-operator-catalog sourceNamespace: openshift-marketplace EOF</pre> <ol style="list-style-type: none"> 2. Validate that the operator was successfully created. <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre>oc get sub -n openshift-operators ibm-mongodb-enterprise-catalog-subscription \ -o jsonpath='{.status.installedCSV} {"\n"}'</pre> <p>Verify that the command returns mongodb-enterprise.v1.12.0.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre>oc get csv -n openshift-operators mongodb-enterprise.v1.12.0 \ -o jsonpath='{.status.phase} : { .status.message} {"\n"}'</pre> <p>Verify that the command returns Succeeded : install strategy completed with no errors.</p> c. Run the following command to confirm that the operator is ready: <pre>oc get deployments -n openshift-operators -l olm.owner="mongodb-enterprise.v1.12.0" \ -o jsonpath="{.items[0].status.availableReplicas} {"\n"}"</pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p> 3. Create the MongoDB (Cloud Pak for Data) operator subscription. (Ensure that you update the namespace parameter to specify the correct Red Hat OpenShift project.) <pre>cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-cpd-mongodb-catalog-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v4.0 name: ibm-cpd-mongodb installPlanApproval: Automatic source: ibm-operator-catalog sourceNamespace: openshift-marketplace EOF</pre> <ol style="list-style-type: none"> 4. Validate that the operator was successfully created. For each command, ensure that you specify the appropriate Red Hat OpenShift project (either ibm-common-services or cpd-operators) for the --namespace (-n) argument. <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre>oc get sub -n operator-project ibm-cpd-mongodb-catalog-subscription \ -o jsonpath='{.status.installedCSV} {"\n"}'</pre> <p>Verify that the command returns ibm-cpd-mongodb.v4.0.2.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre>oc get csv -n operator-project ibm-cpd-mongodb.v4.0.2 \ -o jsonpath='{.status.phase} : { .status.message} {"\n"}'</pre> <p>Verify that the command returns Succeeded : install strategy completed with no errors.</p> c. Run the following command to confirm that the operator is ready: <pre>oc get deployments -n operator-project -l olm.owner="ibm-cpd-mongodb.v4.0.2" \ -o jsonpath="{.items[0].status.availableReplicas} {"\n"}"</pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

Image location	Required subscriptions
Private container registry	<p>If you are pulling images from a private container registry:</p> <ol style="list-style-type: none"> 1. Create the MongoDB Enterprise (third-party) operator subscription. Important: This operator is installed in the openshift-operators project. The operator does not support namespace scoping and must be installed as cluster-wide resource. <pre>cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-mongodb-enterprise-catalog-subscription namespace: openshift-operators spec: channel: stable name: mongodb-enterprise installPlanApproval: Automatic source: ibm-mongodb-enterprise-catalog sourceNamespace: openshift-marketplace EOF</pre> <ol style="list-style-type: none"> 2. Validate that the operator was successfully created. <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre>oc get sub -n openshift-operators ibm-mongodb-enterprise-catalog-subscription \ -o jsonpath='{.status.installedCSV} {"\n"}'</pre> <p>Verify that the command returns mongodb-enterprise.v1.12.0.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre>oc get csv -n openshift-operators mongodb-enterprise.v1.12.0 \ -o jsonpath='{.status.phase} : { .status.message} {"\n"}'</pre> <p>Verify that the command returns Succeeded : install strategy completed with no errors.</p> c. Run the following command to confirm that the operator is ready: <pre>oc get deployments -n openshift-operators -l olm.owner="mongodb-enterprise.v1.12.0" \ -o jsonpath="{.items[0].status.availableReplicas} {"\n"}"</pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p> 3. Create the MongoDB (Cloud Pak for Data) operator subscription. (Ensure that you update the namespace parameter to specify the correct Red Hat OpenShift project.) <pre>cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-cpd-mongodb-catalog-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v4.0 name: ibm-cpd-mongodb installPlanApproval: Automatic source: ibm-cpd-mongodb-catalog sourceNamespace: openshift-marketplace EOF</pre> <ol style="list-style-type: none"> 4. Validate that the operator was successfully created. For each command, ensure that you specify the appropriate Red Hat OpenShift project (either ibm-common-services or cpd-operators) for the --namespace (-n) argument. <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre>oc get sub -n operator-project ibm-cpd-mongodb-catalog-subscription \ -o jsonpath='{.status.installedCSV} {"\n"}'</pre> <p>Verify that the command returns ibm-cpd-mongodb.v4.0.2.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre>oc get csv -n operator-project ibm-cpd-mongodb.v4.0.2 \ -o jsonpath='{.status.phase} : { .status.message} {"\n"}'</pre> <p>Verify that the command returns Succeeded : install strategy completed with no errors.</p> c. Run the following command to confirm that the operator is ready: <pre>oc get deployments -n operator-project -l olm.owner="ibm-cpd-mongodb.v4.0.2" \ -o jsonpath="{.items[0].status.availableReplicas} {"\n"}"</pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

>
OpenPages

Create the appropriate operator subscription for your environment:

Image location	Required subscriptions
----------------	------------------------

Image location	Required subscriptions
IBM Entitled Registry	<p>If you are pulling images from the publicly available registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre>cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-cpd-openpages-operator namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v1.0 installPlanApproval: Automatic name: ibm-cpd-openpages-operator source: ibm-operator-catalog sourceNamespace: openshift-marketplace EOF</pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre>oc get sub -n operator-project ibm-cpd-openpages-operator \ -o jsonpath='{.status.installedCSV} {"\n"}'</pre> <p>Verify that the command returns <code>ibm-cpd-openpages-operator.v8.203.2</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre>oc get csv -n operator-project ibm-cpd-openpages-operator.v8.203.2 \ -o jsonpath='{.status.phase} : { .status.message} {"\n"}'</pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre>oc get deployments -n operator-project -l olm.owner="ibm-cpd-openpages-operator.v8.203.2" \ -o jsonpath="{.items[0].status.availableReplicas} {"\n"}"</pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>
Private container registry	<p>If you are pulling images from a private container registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre>cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-cpd-openpages-operator namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v1.0 installPlanApproval: Automatic name: ibm-cpd-openpages-operator source: ibm-cpd-openpages-operator-catalog sourceNamespace: openshift-marketplace EOF</pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre>oc get sub -n operator-project ibm-cpd-openpages-operator \ -o jsonpath='{.status.installedCSV} {"\n"}'</pre> <p>Verify that the command returns <code>ibm-cpd-openpages-operator.v8.203.2</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre>oc get csv -n operator-project ibm-cpd-openpages-operator.v8.203.2 \ -o jsonpath='{.status.phase} : { .status.message} {"\n"}'</pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre>oc get deployments -n operator-project -l olm.owner="ibm-cpd-openpages-operator.v8.203.2" \ -o jsonpath="{.items[0].status.availableReplicas} {"\n"}"</pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

Create the appropriate operator subscription for your environment:

Image location	Required subscriptions
IBM Entitled Registry	<p>If you are pulling images from the publicly available registry:</p> <ol style="list-style-type: none"> Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre>cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-planning-analytics-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v4.0 name: ibm-planning-analytics-operator installPlanApproval: Automatic source: ibm-operator-catalog sourceNamespace: openshift-marketplace EOF</pre> Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> Run the following command to confirm that the subscription was triggered: <pre>oc get sub -n operator-project ibm-planning-analytics-subscription \ -o jsonpath='{.status.installedCSV} {"\n"}'</pre> <p>Verify that the command returns <code>ibm-planning-analytics-operator.v4.0.2</code>.</p> Run the following command to confirm that the cluster service version (CSV) is ready: <pre>oc get csv -n operator-project ibm-planning-analytics-operator.v4.0.2 \ -o jsonpath='{.status.phase} : { .status.message} {"\n"}'</pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> Run the following command to confirm that the operator is ready: <pre>oc get deployments -n operator-project -l olm.owner="ibm-planning-analytics-operator.v4.0.2" \ -o jsonpath="{.items[0].status.availableReplicas} {'\n'}"</pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>
Private container registry	<p>If you are pulling images from a private container registry:</p> <ol style="list-style-type: none"> Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre>cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-planning-analytics-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v4.0 name: ibm-planning-analytics-operator installPlanApproval: Automatic source: ibm-planning-analytics-operator-catalog sourceNamespace: openshift-marketplace EOF</pre> Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> Run the following command to confirm that the subscription was triggered: <pre>oc get sub -n operator-project ibm-planning-analytics-subscription \ -o jsonpath='{.status.installedCSV} {"\n"}'</pre> <p>Verify that the command returns <code>ibm-planning-analytics-operator.v4.0.2</code>.</p> Run the following command to confirm that the cluster service version (CSV) is ready: <pre>oc get csv -n operator-project ibm-planning-analytics-operator.v4.0.2 \ -o jsonpath='{.status.phase} : { .status.message} {"\n"}'</pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> Run the following command to confirm that the operator is ready: <pre>oc get deployments -n operator-project -l olm.owner="ibm-planning-analytics-operator.v4.0.2" \ -o jsonpath="{.items[0].status.availableReplicas} {'\n'}"</pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

Create the appropriate operator subscription for your environment:

Image location	Required subscriptions
IBM Entitled Registry	<p>If you are pulling images from the publicly available registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre>cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-productmaster-catalog-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: alpha name: ibm-cpd-productmaster installPlanApproval: Automatic source: ibm-operator-catalog sourceNamespace: openshift-marketplace EOF</pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre>oc get sub --namespace ibm-common-services ibm-productmaster-catalog-subscription \ -o jsonpath='{.status.installedCSV} {"\n"}'</pre> <p>Verify that the command returns <code>ibm-cpd-productmaster.v1.0.1</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre>oc get csv -n operator-project ibm-cpd-productmaster.v1.0.1 \ -o jsonpath='{.status.phase} : { .status.message} {"\n"}'</pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre>oc get deployments -n operator-project -l olm.owner="ibm-cpd-productmaster.v1.0.1" \ -o jsonpath='{.items[0].status.availableReplicas} {"\n"}'</pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

Image location	Required subscriptions
Private container registry	<p>If you are pulling images from a private container registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre data-bbox="440 216 1507 491">cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-productmaster-catalog-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: alpha name: ibm-cpd-productmaster installPlanApproval: Automatic source: ibm-productmaster-catalog sourceNamespace: openshift-marketplace EOF</pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre data-bbox="493 642 1377 680">oc get sub --namespace ibm-common-services ibm-productmaster-catalog-subscription \ -o jsonpath='{.status.installedCSV}' {"\n"}</pre> <p>Verify that the command returns <code>ibm-cpd-productmaster.v1.0.1</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre data-bbox="493 785 1143 823">oc get csv -n operator-project ibm-cpd-productmaster.v1.0.1 \ -o jsonpath='{.status.phase } : { .status.message } {"\n"}'</pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre data-bbox="493 926 1386 963">oc get deployments -n operator-project -l olm.owner="ibm-cpd-productmaster.v1.0.1" \ -o jsonpath="{.items[0].status.availableReplicas} {"\n"}"</pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

- > RStudio Server with R 3.6

Create the appropriate operator subscription for your environment:

Image location	Required subscriptions
----------------	------------------------

Image location	Required subscriptions
IBM Entitled Registry	<p>If you are pulling images from the publicly available registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre data-bbox="440 216 1507 569"> cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: labels: app.kubernetes.io/instance: ibm-cpd-rstudio-operator-catalog-subscription app.kubernetes.io/managed-by: ibm-cpd-rstudio-operator app.kubernetes.io/name: ibm-cpd-rstudio-operator-catalog-subscription name: ibm-cpd-rstudio-operator-catalog-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v1.0 installPlanApproval: Automatic name: ibm-cpd-rstudio source: ibm-operator-catalog sourceNamespace: openshift-marketplace EOF </pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre data-bbox="493 724 1325 762"> oc get sub -n operator-project ibm-cpd-rstudio-operator-catalog-subscription \ -o jsonpath='{.status.installedCSV}' {"\n"} </pre> <p>Verify that the command returns <code>ibm-cpd-rstudio.v1.0.2</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre data-bbox="493 863 1122 903"> oc get csv -n operator-project ibm-cpd-rstudio.v1.0.2 \ -o jsonpath='{.status.phase } : { .status.message } {"\n"}' </pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre data-bbox="493 1003 1325 1045"> oc get deployments -n operator-project -l olm.owner="ibm-cpd-rstudio.v1.0.2" \ -o jsonpath='{.items[0].status.availableReplicas}' {"\n"} </pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

Image location	Required subscriptions
Private container registry	<p>If you are pulling images from a private container registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre data-bbox="440 216 1507 569"> cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: labels: app.kubernetes.io/instance: ibm-cpd-rstudio-operator-catalog-subscription app.kubernetes.io/managed-by: ibm-cpd-rstudio-operator app.kubernetes.io/name: ibm-cpd-rstudio-operator-catalog-subscription name: ibm-cpd-rstudio-operator-catalog-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v1.0 installPlanApproval: Automatic name: ibm-cpd-rstudio source: ibm-cpd-rstudio-operator-catalog sourceNamespace: openshift-marketplace EOF </pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre data-bbox="493 724 1323 762"> oc get sub -n operator-project ibm-cpd-rstudio-operator-catalog-subscription \ -o jsonpath='{.status.installedCSV}' {"\n"} </pre> <p>Verify that the command returns <code>ibm-cpd-rstudio.v1.0.2</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre data-bbox="493 863 1120 903"> oc get csv -n operator-project ibm-cpd-rstudio.v1.0.2 \ -o jsonpath='{.status.phase } : { .status.message}' {"\n"} </pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre data-bbox="493 1003 1323 1043"> oc get deployments -n operator-project -l olm.owner="ibm-cpd-rstudio.v1.0.2" \ -o jsonpath="{.items[0].status.availableReplicas}" {"\n"} </pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

>
SPSS Modeler

Create the appropriate operator subscription for your environment:

Image location	Required subscriptions
----------------	------------------------

Image location	Required subscriptions
IBM Entitled Registry	<p>If you are pulling images from the publicly available registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre data-bbox="440 216 1507 569"> cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: labels: app.kubernetes.io/instance: ibm-cpd-spss-operator-catalog-subscription app.kubernetes.io/managed-by: ibm-cpd-spss-operator app.kubernetes.io/name: ibm-cpd-spss-operator-catalog-subscription name: ibm-cpd-spss-operator-catalog-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v1.0 installPlanApproval: Automatic name: ibm-cpd-spss source: ibm-operator-catalog sourceNamespace: openshift-marketplace EOF </pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre data-bbox="493 722 1292 762"> oc get sub -n operator-project ibm-cpd-spss-operator-catalog-subscription \ -o jsonpath='{.status.installedCSV} {"\n"}' </pre> <p>Verify that the command returns <code>ibm-cpd-spss.v4.0.2</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre data-bbox="493 863 1122 903"> oc get csv -n operator-project ibm-cpd-spss.v4.0.2 \ -o jsonpath='{.status.phase } : { .status.message} {"\n"}' </pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre data-bbox="493 1003 1292 1043"> oc get deployments -n operator-project -l olm.owner="ibm-cpd-spss.v4.0.2" \ -o jsonpath="{.items[0].status.availableReplicas} {"\n"}" </pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

Image location	Required subscriptions
Private container registry	<p>If you are pulling images from a private container registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre data-bbox="440 216 1507 569">cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: labels: app.kubernetes.io/instance: ibm-cpd-spss-operator-catalog-subscription app.kubernetes.io/managed-by: ibm-cpd-spss-operator app.kubernetes.io/name: ibm-cpd-spss-operator-catalog-subscription name: ibm-cpd-spss-operator-catalog-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v1.0 installPlanApproval: Automatic name: ibm-cpd-spss source: ibm-cpd-spss-operator-catalog sourceNamespace: openshift-marketplace EOF</pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre data-bbox="493 722 1292 762">oc get sub -n operator-project ibm-cpd-spss-operator-catalog-subscription \ -o jsonpath='{.status.installedCSV}' {"\n"}</pre> <p>Verify that the command returns <code>ibm-cpd-spss.v4.0.2</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre data-bbox="493 863 1122 903">oc get csv -n operator-project ibm-cpd-spss.v4.0.2 \ -o jsonpath='{.status.phase } : { .status.message } {"\n"}</pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre data-bbox="493 1003 1292 1043">oc get deployments -n operator-project -l olm.owner="ibm-cpd-spss.v4.0.2" \ -o jsonpath="{.items[0].status.availableReplicas} {"\n"}"</pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

- > Virtual Data Pipeline

Not applicable. For details, see [Installing Virtual Data Pipeline for Cloud Pak for Data](#).

- > Voice Gateway

Create the appropriate operator subscription for your environment:

Image location	Required subscriptions
----------------	------------------------

Image location	Required subscriptions
IBM Entitled Registry	<p>If you are pulling images from the publicly available registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre> cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-voice-gateway-operator-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v1.0 name: ibm-voice-gateway-operator source: ibm-operator-catalog sourceNamespace: openshift-marketplace installPlanApproval: Automatic EOF </pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre> oc get sub -n operator-project ibm-voice-gateway-operator-subscription \ -o jsonpath='{.status.installedCSV}' {"\n"} </pre> <p>Verify that the command returns <code>ibm-voice-gateway-operator.v1.0.3</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre> oc get csv -n operator-project ibm-voice-gateway-operator.v1.0.3 \ -o jsonpath='{.status.phase} : { .status.message}' {"\n"} </pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre> oc get deployments -n operator-project -l olm.owner="ibm-voice-gateway-operator.v1.0.3" \ -o jsonpath="{.items[0].status.availableReplicas}" {"\n"} </pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>
Private container registry	<p>If you are pulling images from a private container registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre> cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-voice-gateway-operator-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v1.0 name: ibm-voice-gateway-operator source: ibm-voice-gateway-operator-catalog sourceNamespace: openshift-marketplace installPlanApproval: Automatic EOF </pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre> oc get sub -n operator-project ibm-voice-gateway-operator-subscription \ -o jsonpath='{.status.installedCSV}' {"\n"} </pre> <p>Verify that the command returns <code>ibm-voice-gateway-operator.v1.0.3</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre> oc get csv -n operator-project ibm-voice-gateway-operator.v1.0.3 \ -o jsonpath='{.status.phase} : { .status.message}' {"\n"} </pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre> oc get deployments -n operator-project -l olm.owner="ibm-voice-gateway-operator.v1.0.3" \ -o jsonpath="{.items[0].status.availableReplicas}" {"\n"} </pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

Create the appropriate operator subscription for your environment:

Image location	Required subscriptions
IBM Entitled Registry	<p>If you are pulling images from the publicly available registry:</p> <ol style="list-style-type: none"> Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre>cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-watson-assistant-operator-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v4.0 name: ibm-watson-assistant-operator source: ibm-operator-catalog sourceNamespace: openshift-marketplace installPlanApproval: Automatic EOF</pre> Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> Run the following command to confirm that the subscription was triggered: <pre>oc get sub -n operator-project ibm-watson-assistant-operator-subscription \ -o jsonpath='{.status.installedCSV} {"\n"}'</pre> <p>Verify that the command returns <code>ibm-watson-assistant-operator.v4.0.2</code>.</p> Run the following command to confirm that the cluster service version (CSV) is ready: <pre>oc get csv -n operator-project ibm-watson-assistant-operator.v4.0.2 \ -o jsonpath='{.status.phase} : { .status.message} {"\n"}'</pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> Run the following command to confirm that the operator is ready: <pre>oc get deployments -n operator-project -l olm.owner="ibm-watson-assistant-operator.v4.0.2" \ -o jsonpath="{.items[0].status.availableReplicas} {'\n'}"</pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>
Private container registry	<p>If you are pulling images from a private container registry:</p> <ol style="list-style-type: none"> Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre>cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-watson-assistant-operator-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v4.0 name: ibm-watson-assistant-operator source: ibm-watson-assistant-operator-catalog sourceNamespace: openshift-marketplace installPlanApproval: Automatic EOF</pre> Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> Run the following command to confirm that the subscription was triggered: <pre>oc get sub -n operator-project ibm-watson-assistant-operator-subscription \ -o jsonpath='{.status.installedCSV} {"\n"}'</pre> <p>Verify that the command returns <code>ibm-watson-assistant-operator.v4.0.2</code>.</p> Run the following command to confirm that the cluster service version (CSV) is ready: <pre>oc get csv -n operator-project ibm-watson-assistant-operator.v4.0.2 \ -o jsonpath='{.status.phase} : { .status.message} {"\n"}'</pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> Run the following command to confirm that the operator is ready: <pre>oc get deployments -n operator-project -l olm.owner="ibm-watson-assistant-operator.v4.0.2" \ -o jsonpath="{.items[0].status.availableReplicas} {'\n'}"</pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

- >
 - Watson Assistant for Voice Interaction

Watson Assistant for Voice Interaction is comprised of the following services:

- o Voice Gateway
- o Watson Assistant
- o Watson Speech to Text
- o Watson Text to Speech

- >
 - Watson Discovery

Create the appropriate operator subscription for your environment:

Image location	Required subscriptions
IBM Entitled Registry	<p>If you are pulling images from the publicly available registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre data-bbox="440 474 1507 831">cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: labels: app.kubernetes.io/instance: ibm-watson-discovery-operator-subscription app.kubernetes.io/managed-by: ibm-watson-discovery-operator app.kubernetes.io/name: ibm-watson-discovery-operator-subscription name: ibm-watson-discovery-operator-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v4.0 name: ibm-watson-discovery-operator source: ibm-operator-catalog sourceNamespace: openshift-marketplace installPlanApproval: Automatic EOF</pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre data-bbox="493 978 1292 1020">oc get sub -n operator-project ibm-watson-discovery-operator-subscription \ -o jsonpath='{.status.installedCSV}' {"\n"}</pre> <p>Verify that the command returns <code>ibm-watson-discovery-operator.v4.0.2</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre data-bbox="493 1121 1227 1163">oc get csv -n operator-project ibm-watson-discovery-operator.v4.0.2 \ -o jsonpath='{.status.phase} : { .status.message}' {"\n"}</pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre data-bbox="493 1264 1474 1306">oc get deployments -n operator-project -l olm.owner="ibm-watson-discovery-operator.v4.0.2" \ -o jsonpath='{.items[0].status.availableReplicas}' {"\n"}</pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

Image location	Required subscriptions
Private container registry	<p>If you are pulling images from a private container registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre data-bbox="440 216 1507 569"> cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: labels: app.kubernetes.io/instance: ibm-watson-discovery-operator-subscription app.kubernetes.io/managed-by: ibm-watson-discovery-operator app.kubernetes.io/name: ibm-watson-discovery-operator-subscription name: ibm-watson-discovery-operator-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v4.0 name: ibm-watson-discovery-operator source: ibm-watson-discovery-operator-catalog sourceNamespace: openshift-marketplace installPlanApproval: Automatic EOF </pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre data-bbox="493 722 1292 762"> oc get sub -n operator-project ibm-watson-discovery-operator-subscription \ -o jsonpath='{.status.installedCSV}' {"\n"} </pre> <p>Verify that the command returns <code>ibm-watson-discovery-operator.v4.0.2</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre data-bbox="493 863 1227 903"> oc get csv -n operator-project ibm-watson-discovery-operator.v4.0.2 \ -o jsonpath='{.status.phase } : { .status.message}' {"\n"} </pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre data-bbox="493 1003 1474 1043"> oc get deployments -n operator-project -l olm.owner="ibm-watson-discovery-operator.v4.0.2" \ -o jsonpath="{.items[0].status.availableReplicas}" {"\n"} </pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

>
[Watson Knowledge Catalog](#)

Create the appropriate operator subscription for your environment:

Image location	Required subscriptions
----------------	------------------------

Image location	Required subscriptions
IBM Entitled Registry	<p>If you are pulling images from the publicly available registry:</p> <ol style="list-style-type: none"> Specialized installations only. By default, IBM Cloud Pak foundational services expects that the Db2U operator (<code>ibm-db2u-operator</code>) is installed in the <code>ibm-common-services</code> project. However, if you are using the <i>specialized</i> installation architecture (where the IBM Cloud Pak foundational services operators and Cloud Pak for Data operators are in separate OpenShift projects), you must edit the IBM Cloud Pak foundational services operand registry to point to the project where the Cloud Pak for Data operators are installed: <ol style="list-style-type: none"> Run the following command to edit the IBM Cloud Pak foundational services operand registry: <pre>oc edit operandRegistry common-service</pre> Change the value of the <code>namespace</code> entry for the <code>ibm-db2u-operator</code>. Specify the OpenShift project where the Cloud Pak for Data operators are installed, for example, <code>cpd-operators</code>: <pre>- channel: v1.1 installPlanApproval: Automatic name: ibm-db2u-operator namespace: cpd-operators # Specify the project where the Cloud Pak for Data operators are installed packageName: db2u-operator scope: public sourceName: ibm-db2uoperator-catalog sourceNamespace: openshift-marketplace</pre> Save your changes and exit. For example, if you are using <code>vi</code>, hit <code>esc</code> and enter <code>:wq</code>. Create the Watson Knowledge Catalog operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre>cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: labels: app.kubernetes.io/instance: ibm-cpd-wkc-operator-catalog-subscription app.kubernetes.io/managed-by: ibm-cpd-wkc-operator app.kubernetes.io/name: ibm-cpd-wkc-operator-catalog-subscription name: ibm-cpd-wkc-operator-catalog-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v1.0 installPlanApproval: Automatic name: ibm-cpd-wkc source: ibm-operator-catalog sourceNamespace: openshift-marketplace EOF</pre> Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> Run the following command to confirm that the subscription was triggered: <pre>oc get sub -n operator-project ibm-cpd-wkc-operator-catalog-subscription \ -o jsonpath='{.status.installedCSV} {"\n"}'</pre> <p>Verify that the command returns <code>ibm-cpd-wkc.v1.0.2</code>.</p> Run the following command to confirm that the cluster service version (CSV) is ready: <pre>oc get csv -n operator-project ibm-cpd-wkc.v1.0.2 \ -o jsonpath='{.status.phase} : { .status.message} {"\n"}'</pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> Run the following command to confirm that the operator is ready: <pre>oc get deployments -n operator-project -l olm.owner="ibm-cpd-wkc.v1.0.2" \ -o jsonpath='{.items[0].status.availableReplicas} {"\n"}'</pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

Image location	Required subscriptions
Private container registry	<p>If you are pulling images from a private container registry:</p> <ol style="list-style-type: none"> Specialized installations only. By default, IBM Cloud Pak foundational services expects that the Db2U operator (<code>ibm-db2u-operator</code>) is installed in the <code>ibm-common-services</code> project. However, if you are using the <i>specialized</i> installation architecture (where the IBM Cloud Pak foundational services operators and Cloud Pak for Data operators are in separate OpenShift projects), you must edit the IBM Cloud Pak foundational services operand registry to point to the project where the Cloud Pak for Data operators are installed: <ol style="list-style-type: none"> Run the following command to edit the IBM Cloud Pak foundational services operand registry: <pre>oc edit operandRegistry common-service</pre> Change the value of the <code>namespace</code> entry for the <code>ibm-db2u-operator</code>. Specify the OpenShift project where the Cloud Pak for Data operators are installed, for example, <code>cpd-operators</code>: <pre>- channel: v1.1 installPlanApproval: Automatic name: ibm-db2u-operator namespace: cpd-operators # Specify the project where the Cloud Pak for Data operators are installed packageName: db2u-operator scope: public sourceName: ibm-db2uoperator-catalog sourceNamespace: openshift-marketplace</pre> Save your changes and exit. For example, if you are using <code>vi</code>, hit <code>esc</code> and enter <code>:wq</code>. Create the Watson Knowledge Catalog operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre>cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: labels: app.kubernetes.io/instance: ibm-cpd-wkc-operator-catalog-subscription app.kubernetes.io/managed-by: ibm-cpd-wkc-operator app.kubernetes.io/name: ibm-cpd-wkc-operator-catalog-subscription name: ibm-cpd-wkc-operator-catalog-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v1.0 installPlanApproval: Automatic name: ibm-cpd-wkc source: ibm-cpd-wkc-operator-catalog sourceNamespace: openshift-marketplace EOF</pre> Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> Run the following command to confirm that the subscription was triggered: <pre>oc get sub -n operator-project ibm-cpd-wkc-operator-catalog-subscription \ -o jsonpath='{.status.installedCSV}' {"\n"}</pre> <p>Verify that the command returns <code>ibm-cpd-wkc.v1.0.2</code>.</p> Run the following command to confirm that the cluster service version (CSV) is ready: <pre>oc get csv -n operator-project ibm-cpd-wkc.v1.0.2 \ -o jsonpath='{.status.phase} : { .status.message}' {"\n"}</pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> Run the following command to confirm that the operator is ready: <pre>oc get deployments -n operator-project -l olm.owner="ibm-cpd-wkc.v1.0.2" \ -o jsonpath='{.items[0].status.availableReplicas}' {"\n"}</pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

>
Watson Knowledge Studio

Create the appropriate operator subscription for your environment:

Image location	Required subscriptions
----------------	------------------------

Image location	Required subscriptions
IBM Entitled Registry	<p>If you are pulling images from the publicly available registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre data-bbox="440 216 1507 573"> cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: labels: app.kubernetes.io/instance: ibm-watson-ks-operator-subscription app.kubernetes.io/managed-by: ibm-watson-ks-operator app.kubernetes.io/name: ibm-watson-ks-operator-subscription name: ibm-watson-ks-operator-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v4.0 name: ibm-watson-ks-operator source: ibm-operator-catalog sourceNamespace: openshift-marketplace installPlanApproval: Automatic EOF </pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre data-bbox="493 722 1218 762"> oc get sub -n operator-project ibm-watson-ks-operator-subscription \ -o jsonpath='{.status.installedCSV}' {"\n"} </pre> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre data-bbox="493 821 1154 861"> oc get csv -n operator-project ibm-watson-ks-operator.v4.0.2 \ -o jsonpath='{.status.phase } : { .status.message}' {"\n"} </pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors.</code></p> c. Run the following command to confirm that the operator is ready: <pre data-bbox="493 961 1398 1001"> oc get deployments -n operator-project -l olm.owner="ibm-watson-ks-operator.v4.0.2" \ -o jsonpath="{.items[0].status.availableReplicas}" {"\n"} </pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

Image location	Required subscriptions
Private container registry	<p>If you are pulling images from a private container registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre data-bbox="440 216 1507 569">cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: labels: app.kubernetes.io/instance: ibm-watson-ks-operator-subscription app.kubernetes.io/managed-by: ibm-watson-ks-operator app.kubernetes.io/name: ibm-watson-ks-operator-subscription name: ibm-watson-ks-operator-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v4.0 name: ibm-watson-ks-operator source: ibm-watson-ks-operator-catalog sourceNamespace: openshift-marketplace installPlanApproval: Automatic EOF</pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre data-bbox="493 722 1214 762">oc get sub -n operator-project ibm-watson-ks-operator-subscription \ -o jsonpath='{.status.installedCSV}' {"\n"}</pre> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre data-bbox="493 821 1154 861">oc get csv -n operator-project ibm-watson-ks-operator.v4.0.2 \ -o jsonpath='{.status.phase} : { .status.message}' {"\n"}</pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors.</code></p> c. Run the following command to confirm that the operator is ready: <pre data-bbox="493 961 1398 1001">oc get deployments -n operator-project -l olm.owner="ibm-watson-ks-operator.v4.0.2" \ -o jsonpath='{.items[0].status.availableReplicas}' {"\n"}</pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

>
Watson Machine Learning

Create the appropriate operator subscription for your environment:

Image location	Required subscriptions
----------------	------------------------

Image location	Required subscriptions
IBM Entitled Registry	<p>If you are pulling images from the publicly available registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre data-bbox="440 216 1507 569"> cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: labels: app.kubernetes.io/instance: ibm-cpd-wml-operator-subscription app.kubernetes.io/managed-by: ibm-cpd-wml-operator app.kubernetes.io/name: ibm-cpd-wml-operator-subscription name: ibm-cpd-wml-operator-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v1.1 installPlanApproval: Automatic name: ibm-cpd-wml-operator source: ibm-operator-catalog sourceNamespace: openshift-marketplace EOF </pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre data-bbox="493 722 1195 762"> oc get sub -n operator-project ibm-cpd-wml-operator-subscription \ -o jsonpath='{.status.installedCSV}' {"\n"} </pre> <p>Verify that the command returns <code>ibm-cpd-wml-operator.v1.1.1</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre data-bbox="493 863 1133 903"> oc get csv -n operator-project ibm-cpd-wml-operator.v1.1.1 \ -o jsonpath='{.status.phase} : { .status.message}' {"\n"} </pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre data-bbox="493 1003 1377 1043"> oc get deployments -n operator-project -l olm.owner="ibm-cpd-wml-operator.v1.1.1" \ -o jsonpath='{.items[0].status.availableReplicas}' {"\n"} </pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

Image location	Required subscriptions
Private container registry	<p>If you are pulling images from a private container registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre data-bbox="440 216 1507 569"> cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: labels: app.kubernetes.io/instance: ibm-cpd-wml-operator-subscription app.kubernetes.io/managed-by: ibm-cpd-wml-operator app.kubernetes.io/name: ibm-cpd-wml-operator-subscription name: ibm-cpd-wml-operator-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v1.1 installPlanApproval: Automatic name: ibm-cpd-wml-operator source: ibm-cpd-wml-operator-catalog sourceNamespace: openshift-marketplace EOF </pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre data-bbox="493 722 1195 762"> oc get sub -n operator-project ibm-cpd-wml-operator-subscription \ -o jsonpath='{.status.installedCSV}' {"\n"} </pre> <p>Verify that the command returns <code>ibm-cpd-wml-operator.v1.1.1</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre data-bbox="493 863 1130 903"> oc get csv -n operator-project ibm-cpd-wml-operator.v1.1.1 \ -o jsonpath='{.status.phase} : { .status.message}' {"\n"} </pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre data-bbox="493 1003 1377 1043"> oc get deployments -n operator-project -l olm.owner="ibm-cpd-wml-operator.v1.1.1" \ -o jsonpath="{.items[0].status.availableReplicas}" {"\n"} </pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

>
 Watson Machine Learning Accelerator

Create the appropriate operator subscription for your environment:

Image location	Required subscriptions
----------------	------------------------

Image location	Required subscriptions
IBM Entitled Registry	<p>If you are pulling images from the publicly available registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre> cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-cpd-wml-accelerator-operator namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: name: ibm-cpd-wml-accelerator-operator channel: v1.0 installPlanApproval: Automatic source: ibm-operator-catalog sourceNamespace: openshift-marketplace EOF </pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre> oc get sub -n operator-project ibm-cpd-wml-accelerator-operator \ -o jsonpath='{.status.installedCSV} {"\n"}' </pre> <p>Verify that the command returns <code>ibm-cpd-wml-accelerator-operator.v1.0.2</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre> oc get csv -n operator-project ibm-cpd-wml-accelerator-operator.v1.0.2 \ -o jsonpath='{.status.phase} : { .status.message} {"\n"}' </pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre> oc get deployments -n operator-project -l olm.owner="ibm-cpd-wml-accelerator-operator.v1.0.2" \ -o jsonpath='{.items[0].status.availableReplicas} {"\n"}' </pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>
Private container registry	<p>If you are pulling images from a private container registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre> cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-cpd-wml-accelerator-operator namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: name: ibm-cpd-wml-accelerator-operator channel: v1.0 installPlanApproval: Automatic source: ibm-cpd-wml-accelerator-catalog sourceNamespace: openshift-marketplace EOF </pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre> oc get sub -n operator-project ibm-cpd-wml-accelerator-operator \ -o jsonpath='{.status.installedCSV} {"\n"}' </pre> <p>Verify that the command returns <code>ibm-cpd-wml-accelerator-operator.v1.0.2</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre> oc get csv -n operator-project ibm-cpd-wml-accelerator-operator.v1.0.2 \ -o jsonpath='{.status.phase} : { .status.message} {"\n"}' </pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre> oc get deployments -n operator-project -l olm.owner="ibm-cpd-wml-accelerator-operator.v1.0.2" \ -o jsonpath='{.items[0].status.availableReplicas} {"\n"}' </pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

Create the appropriate operator subscription for your environment:

Image location	Required subscriptions
IBM Entitled Registry	<p>If you are pulling images from the publicly available registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre>cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-watson-openscale-operator-subscription labels: app.kubernetes.io/instance: ibm-watson-openscale-operator-subscription app.kubernetes.io/managed-by: ibm-watson-openscale-operator app.kubernetes.io/name: ibm-watson-openscale-operator-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v1 installPlanApproval: Automatic name: ibm-cpd-wos source: ibm-operator-catalog sourceNamespace: openshift-marketplace EOF</pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre>oc get sub -n operator-project ibm-watson-openscale-operator-subscription \ -o jsonpath='{.status.installedCSV} {"\n"}'</pre> <p>Verify that the command returns <code>ibm-cpd-wos.v1.2.0</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre>oc get csv -n operator-project ibm-cpd-wos.v1.2.0 \ -o jsonpath='{.status.phase } : { .status.message} {"\n"}'</pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre>oc get deployments -n operator-project -l olm.owner="ibm-cpd-wos.v1.2.0" \ -o jsonpath='{.items[0].status.availableReplicas} {"\n"}'</pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

Image location	Required subscriptions
Private container registry	<p>If you are pulling images from a private container registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre data-bbox="440 216 1507 569"> cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-watson-openscale-operator-subscription labels: app.kubernetes.io/instance: ibm-watson-openscale-operator-subscription app.kubernetes.io/managed-by: ibm-watson-openscale-operator app.kubernetes.io/name: ibm-watson-openscale-operator-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v1 installPlanApproval: Automatic name: ibm-cpd-wos source: ibm-openscale-operator-catalog sourceNamespace: openshift-marketplace EOF </pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre data-bbox="493 722 1292 762"> oc get sub -n operator-project ibm-watson-openscale-operator-subscription \ -o jsonpath='{.status.installedCSV}' {"\n"} </pre> <p>Verify that the command returns <code>ibm-cpd-wos.v1.2.0</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre data-bbox="493 863 1122 903"> oc get csv -n operator-project ibm-cpd-wos.v1.2.0 \ -o jsonpath='{.status.phase} : { .status.message}' {"\n"} </pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre data-bbox="493 1031 1284 1071"> oc get deployments -n operator-project -l olm.owner="ibm-cpd-wos.v1.2.0" \ -o jsonpath='{.items[0].status.availableReplicas}' {"\n"} </pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

>
Watson Speech to Text

Create the appropriate operator subscription for your environment.

Important: The same operator subscription is used for Watson Speech to Text and Watson Text to Speech. If you plan to install both services, you only need to create the subscription once.

Image location	Required subscriptions
----------------	------------------------

Image location	Required subscriptions
IBM Entitled Registry	<p>If you are pulling images from the publicly available registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre data-bbox="430 216 1490 491"> cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-watson-speech-operator-subscription namespace: ibm-common-services # Pick the project that contains the Cloud Pak for Data operator: ibm-common-services (default) or cpd-operators spec: channel: v4.0 name: ibm-watson-speech-operator source: ibm-operator-catalog sourceNamespace: openshift-marketplace installPlanApproval: Automatic EOF </pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre data-bbox="483 642 1247 680"> oc get sub -n operator-project ibm-watson-speech-operator-subscription \ -o jsonpath='{.status.installedCSV}' {"\n"}' </pre> <p>Verify that the command returns <code>ibm-watson-speech-operator.v4.0.0</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre data-bbox="483 785 1182 823"> oc get csv -n operator-project ibm-watson-speech-operator.v4.0.0 \ -o jsonpath='{.status.phase} : { .status.message}' {"\n"}' </pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre data-bbox="483 953 1429 991"> oc get deployments -n operator-project -l olm.owner="ibm-watson-speech-operator.v4.0.0" \ -o jsonpath="{.items[0].status.availableReplicas}" {"\n"}' </pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>
Private container registry	<p>If you are pulling images from a private container registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre data-bbox="430 1152 1490 1428"> cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-watson-speech-operator-subscription namespace: ibm-common-services # Pick the project that contains the Cloud Pak for Data operator: ibm-common-services (default) or cpd-operators spec: channel: v4.0 name: ibm-watson-speech-operator source: ibm-watson-speech-operator-catalog sourceNamespace: openshift-marketplace installPlanApproval: Automatic EOF </pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre data-bbox="483 1577 1247 1614"> oc get sub -n operator-project ibm-watson-speech-operator-subscription \ -o jsonpath='{.status.installedCSV}' {"\n"}' </pre> <p>Verify that the command returns <code>ibm-watson-speech-operator.v4.0.0</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre data-bbox="483 1719 1182 1757"> oc get csv -n operator-project ibm-watson-speech-operator.v4.0.0 \ -o jsonpath='{.status.phase} : { .status.message}' {"\n"}' </pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre data-bbox="483 1862 1429 1900"> oc get deployments -n operator-project -l olm.owner="ibm-watson-speech-operator.v4.0.0" \ -o jsonpath="{.items[0].status.availableReplicas}" {"\n"}' </pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

Create the appropriate operator subscription for your environment:

Image location	Required subscriptions
IBM Entitled Registry	<p>If you are pulling images from the publicly available registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre>cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: annotations: name: ibm-cpd-ws-operator-catalog-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v2.0 installPlanApproval: Automatic name: ibm-cpd-wsl source: ibm-operator-catalog sourceNamespace: openshift-marketplace EOF</pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre>oc get sub -n operator-project ibm-cpd-ws-operator-catalog-subscription \ -o jsonpath='{.status.installedCSV} {"\n"}'</pre> <p>Verify that the command returns <code>ibm-cpd-wsl.v2.0.2</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre>oc get csv -n operator-project ibm-cpd-wsl.v2.0.2 \ -o jsonpath='{.status.phase } : { .status.message} {"\n"}'</pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre>oc get deployments -n operator-project -l olm.owner="ibm-cpd-wsl.v2.0.2" \ -o jsonpath='{.items[0].status.availableReplicas} {"\n"}'</pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

Image location	Required subscriptions
Private container registry	<p>If you are pulling images from a private container registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre> cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: annotations: name: ibm-cpd-ws-operator-catalog-subscription namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v2.0 installPlanApproval: Automatic name: ibm-cpd-wsl source: ibm-cpd-ws-operator-catalog sourceNamespace: openshift-marketplace EOF </pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre> oc get sub -n operator-project ibm-cpd-ws-operator-catalog-subscription \ -o jsonpath='{.status.installedCSV}' {"\n"} </pre> <p>Verify that the command returns <code>ibm-cpd-wsl.v2.0.2</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre> oc get csv -n operator-project ibm-cpd-wsl.v2.0.2 \ -o jsonpath='{.status.phase} : { .status.message}' {"\n"} </pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre> oc get deployments -n operator-project -l olm.owner="ibm-cpd-wsl.v2.0.2" \ -o jsonpath='{.items[0].status.availableReplicas}' {"\n"} </pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

• [Watson Studio Runtimes](#)

Create the appropriate operator subscription for your environment:

Image location	Required subscriptions
----------------	------------------------

Image location	Required subscriptions
IBM Entitled Registry	<p>If you are pulling images from the publicly available registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre data-bbox="440 216 1507 569"> cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: labels: app.kubernetes.io/instance: ibm-cpd-ws-runtimes-operator-catalog-subscription app.kubernetes.io/managed-by: ibm-cpd-ws-runtimes-operator app.kubernetes.io/name: ibm-cpd-ws-runtimes-operator name: ibm-cpd-ws-runtimes-operator namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v1.0 installPlanApproval: Automatic name: ibm-cpd-ws-runtimes source: ibm-operator-catalog sourceNamespace: openshift-marketplace EOF </pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre data-bbox="493 722 1143 762"> oc get sub -n operator-project ibm-cpd-ws-runtimes-operator \ -o jsonpath='{.status.installedCSV}' {"\n"} </pre> <p>Verify that the command returns <code>ibm-cpd-ws-runtimes.v1.0.2</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre data-bbox="493 863 1122 903"> oc get csv -n operator-project ibm-cpd-ws-runtimes.v1.0.2 \ -o jsonpath='{.status.phase } : { .status.message}' {"\n"} </pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre data-bbox="493 1031 1365 1071"> oc get deployments -n operator-project -l olm.owner="ibm-cpd-ws-runtimes.v1.0.2" \ -o jsonpath='{.items[0].status.availableReplicas}' {"\n"} </pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

Image location	Required subscriptions
Private container registry	<p>If you are pulling images from a private container registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre data-bbox="440 216 1507 573">cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: labels: app.kubernetes.io/instance: ibm-cpd-ws-runtimes-operator-catalog-subscription app.kubernetes.io/managed-by: ibm-cpd-ws-runtimes-operator app.kubernetes.io/name: ibm-cpd-ws-runtimes-operator name: ibm-cpd-ws-runtimes-operator namespace: ibm-common-services cpd-operators # Pick the project that contains the Cloud Pak for Data operator spec: channel: v1.0 installPlanApproval: Automatic name: ibm-cpd-ws-runtimes source: ibm-cpd-ws-runtimes-operator-catalog sourceNamespace: openshift-marketplace EOF</pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre data-bbox="493 722 1143 764">oc get sub -n operator-project ibm-cpd-ws-runtimes-operator \ -o jsonpath='{.status.installedCSV}' {"\n"}</pre> <p>Verify that the command returns <code>ibm-cpd-ws-runtimes.v1.0.2</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre data-bbox="493 863 1122 905">oc get csv -n operator-project ibm-cpd-ws-runtimes.v1.0.2 \ -o jsonpath='{.status.phase} : { .status.message}' {"\n"}</pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre data-bbox="493 1031 1365 1073">oc get deployments -n operator-project -l olm.owner="ibm-cpd-ws-runtimes.v1.0.2" \ -o jsonpath='{.items[0].status.availableReplicas}' {"\n"}</pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

>
Watson Text to Speech

Create the appropriate operator subscription for your environment.

Important: The same operator subscription is used for Watson Speech to Text and Watson Text to Speech. If you plan to install both services, you only need to create the subscription once.

Image location	Required subscriptions
----------------	------------------------

Image location	Required subscriptions
IBM Entitled Registry	<p>If you are pulling images from the publicly available registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre data-bbox="428 216 1490 491"> cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-watson-speech-operator-subscription namespace: ibm-common-services # Pick the project that contains the Cloud Pak for Data operator: ibm-common-services (default) or cpd-operators spec: channel: v4.0 name: ibm-watson-speech-operator source: ibm-operator-catalog sourceNamespace: openshift-marketplace installPlanApproval: Automatic EOF </pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre data-bbox="483 642 1247 680"> oc get sub -n operator-project ibm-watson-speech-operator-subscription \ -o jsonpath='{.status.installedCSV}' {"\n"} </pre> <p>Verify that the command returns <code>ibm-watson-speech-operator.v4.0.0</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre data-bbox="483 785 1182 823"> oc get csv -n operator-project ibm-watson-speech-operator.v4.0.0 \ -o jsonpath='{.status.phase} : { .status.message}' {"\n"} </pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre data-bbox="483 928 1429 966"> oc get deployments -n operator-project -l olm.owner="ibm-watson-speech-operator.v4.0.0" \ -o jsonpath="{.items[0].status.availableReplicas}" {"\n"} </pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>
Private container registry	<p>If you are pulling images from a private container registry:</p> <ol style="list-style-type: none"> 1. Create the following operator subscription. (Ensure that you update the <code>namespace</code> parameter to specify the correct Red Hat OpenShift project.) <pre data-bbox="428 1125 1490 1400"> cat <<EOF oc apply -f - apiVersion: operators.coreos.com/v1alpha1 kind: Subscription metadata: name: ibm-watson-speech-operator-subscription namespace: ibm-common-services # Pick the project that contains the Cloud Pak for Data operator: ibm-common-services (default) or cpd-operators spec: channel: v4.0 name: ibm-watson-speech-operator source: ibm-watson-speech-operator-catalog sourceNamespace: openshift-marketplace installPlanApproval: Automatic EOF </pre> 2. Validate that the operator was successfully created. <p>For each command, ensure that you specify the appropriate Red Hat OpenShift project (either <code>ibm-common-services</code> or <code>cpd-operators</code>) for the <code>--namespace (-n)</code> argument.</p> <ol style="list-style-type: none"> a. Run the following command to confirm that the subscription was triggered: <pre data-bbox="483 1549 1247 1587"> oc get sub -n operator-project ibm-watson-speech-operator-subscription \ -o jsonpath='{.status.installedCSV}' {"\n"} </pre> <p>Verify that the command returns <code>ibm-watson-speech-operator.v4.0.0</code>.</p> b. Run the following command to confirm that the cluster service version (CSV) is ready: <pre data-bbox="483 1692 1182 1730"> oc get csv -n operator-project ibm-watson-speech-operator.v4.0.0 \ -o jsonpath='{.status.phase} : { .status.message}' {"\n"} </pre> <p>Verify that the command returns <code>Succeeded : install strategy completed with no errors</code>.</p> c. Run the following command to confirm that the operator is ready: <pre data-bbox="483 1835 1429 1873"> oc get deployments -n operator-project -l olm.owner="ibm-watson-speech-operator.v4.0.0" \ -o jsonpath="{.items[0].status.availableReplicas}" {"\n"} </pre> <p>Verify that the command returns an integer greater than or equal to 1. If the command returns 0, wait for the deployment to become available.</p>

Previous topic: [Installing IBM Cloud Pak foundational services](#)

Custom security context constraints for services

Most Cloud Pak for Data services use the **restricted** security context constraint (SCC) that is provided by Red Hat® OpenShift® Container Platform. However, if you plan to install certain Cloud Pak for Data services, you might need to use some custom SCCs.

OpenShift provides a set of predefined SCCs that control the actions that a pod can perform and what it can access. These SCCs can be used, modified, or extended by any administrator. By default, the execution of any container is granted access to the restricted SCC and only the capabilities that are defined by that SCC. For more information, see [Managing security context constraints](#) in the Red Hat OpenShift Container Platform documentation.

When you install Cloud Pak for Data services, the default service account is associated with the restricted SCC. Cloud Pak for Data does not support the use of privileged SCCs in OpenShift. However, some Cloud Pak for Data services might require custom SCCs, for example to support IPCs. For more information, see [Security context constraints](#) in the IBM® Cloud Platform Common Services documentation.

The following Cloud Pak for Data services use custom SCCs:

- Data Virtualization
- Db2®
- Db2 Big SQL
- Db2 Warehouse
- OpenPages®
- Watson™ Knowledge Catalog

If you plan to install Watson Knowledge Catalog, you must create the custom SCCs manually. If you have multiple copies of Cloud Pak for Data installed in different namespaces, you must create these SCCs one time for the cluster.

If you plan to install Db2 or Db2 Warehouse, the Db2 operator creates the custom SCC, service accounts, roles, and role bindings.

The Db2 Big SQL and Data Virtualization services embed Db2 and use the Db2 custom SCC.

The OpenPages service embeds Db2, but the custom SCC is used only if the OpenPages service instance is provisioned by using the internal database option. When the OpenPages service is provisioned by using a database outside the cluster, the custom SCC is not required.

For more information about basic security features in Cloud Pak for Data, see [Basic security features on Red Hat OpenShift Container Platform](#).

- [Creating custom security context constraints for Watson Knowledge Catalog](#)
The Watson Knowledge Catalog service requires the use of a custom security context constraint (SCC).
- [Security context constraints for Db2](#)
Db2 requires the use of custom security context constraints (SCCs), which are created automatically when you install Db2.

Previous topic: [Creating operator subscriptions](#)

Next topic: [Changing required node settings](#)

Creating custom security context constraints for Watson Knowledge Catalog

The Watson Knowledge Catalog service requires the use of a custom security context constraint (SCC).

Custom SCCs for Watson Knowledge Catalog

Administrators can use security context constraints to control permissions for pods on their Red Hat OpenShift cluster. These permissions include actions that a pod can perform and what resources it can access. For Watson Knowledge Catalog, you must create a custom SCC.

Creating custom SCCs for Watson Knowledge Catalog

To create the SCC, complete the following steps:

1. Define the SCC in the file `wkc-iis-scc.yaml`, as follows:

```
allowHostDirVolumePlugin: false
allowHostIPC: false
allowHostNetwork: false
allowHostPID: false
allowHostPorts: false
allowPrivilegeEscalation: true
allowPrivilegedContainer: false
allowedCapabilities: null
apiVersion: security.openshift.io/v1
defaultAddCapabilities: null
fsGroup:
  type: RunAsAny
kind: SecurityContextConstraints
metadata:
  annotations:
    kubernetes.io/description: WKC/IIS provides all features of the restricted SCC
    but runs as user 10032.
  name: wkc-iis-scc
readOnlyRootFilesystem: false
requiredDropCapabilities:
- KILL
```



```

- MKNOD
- SETUID
- SETGID
runAsUser:
  type: MustRunAs
  uid: 10032
seLinuxContext:
  type: MustRunAs
supplementalGroups:
  type: RunAsAny
volumes:
- configMap
- downwardAPI
- emptyDir
- persistentVolumeClaim
- projected
- secret
users:
- system:serviceaccount:{{ namespace }}:wkc-iis-sa

```

- Replace `{{ namespace }}` with the name of the Red Hat® OpenShift® project where you plan to install Watson Knowledge Catalog. For example, if you plan to install Watson Knowledge Catalog in the `cpd-instance` project, the system:service account entry would be:

```
- system:serviceaccount:cpd-instance:wkc-iis-sa
```

- If the custom SCC (`wkc-iis-scc`) exists in the environment, delete the custom SCC that already exists and create a new custom SCC by using the YAML file from this step. Use the following command to delete the custom SCC: `oc delete scc wkc-iis-scc`

2. Run `oc create` to create the file:

```
$ oc create -f <yaml_file_name.yaml>
```

3. Run the following command to verify that the SCC was created:

```
$ oc get scc wkc-iis-scc
```

4. Create the SCC cluster role for `wkc-iis-scc`:

```
oc create clusterrole system:openshift:scc:wkc-iis-scc --verb=use --resource=scc --resource-name=wkc-iis-scc
```

5. Assign the `wkc-iis-sa` service account to the SCC cluster role:

```
oc create rolebinding wkc-iis-scc-rb --clusterrole=system:openshift:scc:wkc-iis-scc --serviceaccount={{ namespace }}:wkc-iis-sa
```

6. Confirm that the `wkc-iis-sa` service account can use the `wkc-iis-scc` SCC:

```
oc adm policy who-can use scc wkc-iis-scc -n {{ namespace }} | grep "wkc-iis-sa"
```

For more information about SCCs, see [Red Hat - Managing Security Context Constraints](#).

Security context constraints for Db2

Db2 requires the use of custom security context constraints (SCCs), which are created automatically when you install Db2.

SYS_RESOURCE

Allows manipulation of reservations, memory allocations, and resource limits. Maximum memory allocation is still constrained by the memory cgroup (memcg) limit, which cannot be overridden by this sys-capability. The Db2 database engine needs this sys-capability to increase the resource limits (IE.ulimits).

IPC_OWNER

Bypasses permission checks for operations on IPC objects. Even when the IPC kernel parameters are set to maximum values on the hosts/worker nodes, the Db2 engine still tries to dynamically throttle those values. This system capability is provided in addition to sharing IPC namespace with the host.

SYS_NICE

Allows changing process priorities. Because each container has its own PID namespace, this capability applies to that container only. The Db2 database engine relies on process thread prioritization to ensure that Work Load Management (WLM) and Fast Communications Manager (FCM) processing is prioritized over generic agent work.

CHOWN

Necessary to run `chown` to change ownership of files/directories in persistent volumes.

DAC_OVERRIDE

Bypasses permission checks for file read, write, and execute.

FSETID

Prevents the clearing of the `setuid` and `setgid` mode bits when a file is modified.

FOWNER

Bypasses permission checks on operations that normally require the file system UID of the process to match the UID of the file (for example, `chmod(2)`, `utime(2)`), excluding those operations that are covered by `CAP_DAC_OVERRIDE` and `CAP_DAC_READ_SEARCH`.

SETGID

Necessary to run Db2 engine processes with escalated group privileges.

SETUID

Necessary to run Db2 engine processes with escalated user privileges.

SETFCAP

Used to set capabilities on files.

SETPCAP

Used to set capabilities on processes.

SYS_CHROOT

Necessary to use the `chroot` command.

KILL

Bypasses permission checks for sending signals. Necessary for signal handling during process management.

Changing required node settings

Some services that run on IBM® Cloud Pak for Data require specific settings on the nodes in the cluster. To ensure that the cluster has the required settings for these services, an operating system administrator with `root` privileges must review and adjust the settings on the appropriate nodes in the cluster.

The Machine Config Operator is a cluster-level operator that you can use to manage the operating system and keep the cluster up to date and configured. For more information, see [Using MachineConfig objects to configure nodes](#).

Node settings for services

The following table shows the node settings that require changes for some services, with links to instructions for changing the settings.

Node settings	Services that require changes to the setting	Environments	Instructions
HAProxy timeout settings for the load balancer	<ul style="list-style-type: none"> OpenPages® Watson™ Discovery Watson Knowledge Catalog Also recommended if you are working with large data sets or you have slower network speeds. 	All environments	<ul style="list-style-type: none"> Load balancer timeout settings
CRI-O container settings	<ul style="list-style-type: none"> Data Virtualization Db2® Db2 Big SQL Db2 Warehouse Watson Discovery Watson Knowledge Catalog Watson Studio 	All environments except IBM Cloud	<ul style="list-style-type: none"> CRI-O container settings
Kernel parameter settings	<ul style="list-style-type: none"> Data Virtualization Db2 Db2 Big SQL Db2 Warehouse Watson Discovery Watson Knowledge Catalog Watson Studio 	All environments	<ul style="list-style-type: none"> Kernel parameter settings Deploying Db2 with limited privileges Deploying Db2 Warehouse with limited privileges Updating kernel semaphore settings - Db2 Big SQL Preparing to install the service (Data Virtualization)
Power settings			<ul style="list-style-type: none"> Power settings
GPU settings	<ul style="list-style-type: none"> Jupyter Notebooks with Python 3.7 for GPU Watson Machine Learning Accelerator (requires that the NVIDIA GPU Operator is installed) 	All environments	<ul style="list-style-type: none"> GPU node settings Installing the NVIDIA GPU Operator

Load balancer timeout settings

To prevent connections from being closed before processes complete, you might need to adjust the timeout settings on your load balancer node. If you are using HAProxy, the load balancer node is usually the OpenShift® cluster public node.

The recommended timeout is at least 5 minutes (300 seconds).

This setting is required if you plan to install the Watson Knowledge Catalog service or the OpenPages service. For more information about timeout settings in Watson Knowledge Catalog, see [Processes time out before completing](#), and for timeout settings in OpenPages, see [Setting up the cluster for OpenPages](#).

This setting is also recommended if you are working with large data sets or you have slower network speeds.

The following steps assume that you are using HAProxy. If you are using a load balancer other than HAProxy, see the documentation for your load balancer for information about how to configure the timeout.

Changing timeout settings on premises or private cloud

- On the load balancer node, check the HAProxy timeout settings in the `/etc/haproxy/haproxy.cfg` file. The recommended values are at least:

```
timeout client 300s
timeout server 300s
```

- If the timeout values are less than 300 seconds (5 minutes), update the values:

- To change the `timeout client` setting, enter the following command:

```
sed -i -e "/timeout client/s/ [0-9].*/ 5m/" /etc/haproxy/haproxy.cfg
```

- To change the `timeout server` setting, enter the following command:

```
sed -i -e "/timeout server/s/ [0-9].*/ 5m/" /etc/haproxy/haproxy.cfg
```

- Run the following command to apply the changes that you made to the HAProxy configuration:

```
systemctl restart haproxy
```

Changing timeout settings on IBM Cloud

If you are setting HAProxy timeout settings for Cloud Pak for Data on IBM Cloud, you can configure route timeouts by using the `oc annotate` command.

1. Use the following command to set the server-side timeout for the HAProxy route to 360 seconds:

```
oc annotate route zen-cpd --overwrite haproxy.router.openshift.io/timeout=360s
```

If you don't provide the units, `ms` is the default.

2. Optionally, customize other route-specific settings. For more information, see [Route-specific annotations](#).

Note: On a Virtual Private Cloud (VPC) Gen2 cluster, the load balancer timeout is set to 30s by default. If you use the `annotate` command to set the timeout value greater than 50s, it will be set to 50s. You cannot customize the timeout value to be greater than 50s. The server might time out during long running transactions. For more information, see [Connection timeouts](#).

CRI-O container settings

To ensure that services can run correctly, you must adjust values in the CRI-O container settings to specify the maximum number of processes and the maximum number of open descriptor files.

These settings are required for the CRI-O CRI-O container runtime on the OpenShift Container Platform.

Note: If you install Cloud Pak for Data on IBM Cloud, the CRI-O container settings are automatically applied to your cluster as part of the installation. You do not need to manually change these settings.

To change CRI-O settings, you modify the contents of the `crio.conf` file and pass those updates to your nodes as a machine config.

1. Obtain a copy of the existing `crio.conf` file from a worker node. For example, run the following command, replacing `$node` with one of the worker nodes. You can obtain the worker nodes by using the `oc get nodes` command.

```
scp core@$node:/etc/crio/crio.conf /tmp/crio.conf
```

If the `crio.conf` file doesn't exist in the path `/etc/crio/crio.conf`, use the path `/etc/crio/crio.conf.d/00-default` instead.

If you don't have access by using the `scp` command, ask your cluster administrator for the `crio.conf` file.

Make sure that you obtain the latest version of the `crio.conf` file.

2. In the `crio.conf` file, make the following changes in the `[crio.runtime]` section (uncomment the lines if necessary):
 - To set the maximum number of open files, change the `default_ulimits` setting to at least **66560**, as follows:

```
.....
[crio.runtime]
default_ulimits = [
  "nofile=66560:66560"
]
.....
```

- To set the maximum number of processes, change the `pids_limit` setting to at least **12288**, as follows:

```
.....
# Maximum number of processes allowed in a container.
pids_limit = 12288
.....
```

3. Create a `machineconfig` object YAML file, as follows, and apply it.

```
cat << EOF | oc apply -f -
apiVersion: machineconfiguration.openshift.io/v1
kind: MachineConfig
metadata:
  labels:
    machineconfiguration.openshift.io/role: worker
  name: 99-worker-cp4d-crio-conf
spec:
  config:
    ignition:
      version: 3.1.0
    storage:
      files:
      - contents:
          source: data:text/plain;charset=utf-8;base64,$(cat /tmp/crio.conf | base64 -w0)
          filesystem: root
          mode: 0644
          path: /etc/crio/crio.conf
EOF
```

4. Monitor all of the nodes to ensure that the changes are applied, by using the following command:

```
watch oc get nodes
```

You can also use the following command to confirm that the MachineConfig sync is complete:

```
watch oc get mcp
```

Kernel parameter settings

Enabling unsafe sysctls

Installing the scheduling service

A Red Hat® OpenShift® cluster administrator can install the scheduling service in the IBM Cloud Pak® foundational services operator namespace.

Permissions you need for this task

You must be a cluster administrator.

When you need to complete this task

If you plan to install the Watson™ Machine Learning Accelerator service, you must install the scheduling service.

In general, it is strongly recommended that you install the scheduling service, because it enables you to programmatically enforce the quotas that you set for the Cloud Pak for Data control plane or various Cloud Pak for Data services. For details on quota enforcement, see [Monitoring the platform](#).

Information you need to complete this task

- The scheduling service is installed in the same project as the IBM Cloud Pak foundational services, typically `ibm-common-services`.
- The scheduling service uses the following storage classes. If you don't use these storage classes on your cluster, ensure that you have a storage class with an equivalent definition:
 - **OpenShift Container Storage:** `ocs-storagecluster-cephfs`
 - **IBM Spectrum® Scale Container Native:** `ibm-spectrum-scale-sc`
 - **NFS:** `managed-nfs-storage`
 - **Portworx:** `portworx-shared-gp3`
 - **IBM® Cloud File Storage:** `ibmc-file-gold-gid` or `ibm-file-custom-gold-gid`

Before you begin

Ensure that the cluster meets the minimum requirements for installing the scheduling service. For details, see [System requirements](#).

Verify that you completed the required [Pre-installation tasks](#) for your environment. Specifically, verify that:

1. The `ibm-common-services` namespace exists and has the required operator group. For details, see [Creating projects \(namespaces\) on Red Hat OpenShift Container Platform](#).
2. For environments that use a private container registry, such as air-gapped environments, the scheduling service images are mirrored to the private container registry. For details, see [Mirroring images to your container registry](#).
3. The cluster is configured to pull the software images. For details, see [Configuring your cluster to pull Cloud Pak for Data images](#).
4. The scheduling service operator subscription exists. For details, see [Creating operator subscriptions](#).

If these tasks are not complete, the scheduling service installation will fail.

Procedure

Complete the following tasks to install the scheduling service:

1. [Installing the scheduling service](#)
2. [Verifying the installation](#)
3. [What to do next](#)

Installing the scheduling service

To install the scheduling service:

1. Log in to Red Hat OpenShift Container Platform as a user with sufficient permissions to complete the task:

```
oc login OpenShift_URL:port
```

2. Create a `Scheduling` custom resource to install the scheduling service.

The recommended storage class names are described in [Setting up shared persistent storage](#).

Create a custom resource with the following format.

```
cat <<EOF |oc apply -f -
apiVersion: scheduler.spectrumcomputing.ibm.com/v1
kind: Scheduling
metadata:
  labels:
    release: cpd-scheduler
    velero.io/exclude-from-backup: "true"
  name: ibm-cpd-scheduler
  namespace: ibm-common-services # Specify the namespace where IBM Cloud Pak foundational services is installed.
spec:
  version: 1.2.3
  cluster:
    pvc:
      dynamicStorage: true
      size: 10G
  license:
    accept: true
  registry: cp.icr.io/cp/cpd
  releasename: ibm-cpd-scheduler
  storageClass: storage-class-name # See the guidance in "Information you need to complete this task"
EOF
```

When you create the custom resource, the scheduling service operator installs the scheduling service.

Verifying the installation

When you create the custom resource, the scheduling service operator processes the contents of the custom resource and starts up the microservices that comprise the scheduling service, including **Scheduling**. (The **Scheduling** microservice is defined by the **ibm-cpd-scheduler** custom resource.) The scheduling service is installed when the **Scheduling** status is **Completed**.

To check the status of the installation:

1. Change to the project where you installed the scheduling service:

```
oc project ibm-common-services
```

2. Get the status of the scheduling service (**ibm-cpd-scheduler**):

```
oc get scheduling -o jsonpath='{.items[0].status.cpd-schedulingStatus}' {"\n"}
```

The scheduling service is ready when the command returns **Completed**.

What to do next

The scheduling service is ready to use. The Cloud Pak for Data control plane and Watson Machine Learning Accelerator will automatically integrate with the scheduling service when they are installed.

You can now [install Cloud Pak for Data](#).

Installing Cloud Pak for Data

When you install IBM® Cloud Pak for Data, you update the IBM Cloud Pak® for Data platform operator and the IBM Cloud Pak foundational services operator to watch the project where you will install IBM Cloud Pak for Data. Then, you create a custom resource to install Cloud Pak for Data in that project.

Permissions you need for this task

You must be either:

- A cluster administrator
- An administrator of the following projects:
 - The IBM Cloud Pak foundational services project (**ibm-common-services**)
 - The IBM Cloud Pak for Data platform operator project (**cpd-operators** or **ibm-common-services**)
 - The project where you plan to install Cloud Pak for Data

When you need to complete this task

You must complete this task each time you want to install an instance of Cloud Pak for Data on your cluster.

Information you need to complete this task

- The Cloud Pak for Data control plane needs only the **restricted** security context constraint (SCC).
- The Cloud Pak for Data control plane uses the following storage classes. If you don't use these storage classes on your cluster, ensure that you have a storage class with an equivalent definition:
 - **OpenShift® Container Storage: ocs-storagecluster-cephfs**
 - **IBM Spectrum® Scale Container Native: ibm-spectrum-scale-sc**
 - **NFS: managed-nfs-storage**
 - **Portworx: portworx-shared-gp3**
 - **IBM Cloud File Storage: ibmc-file-gold-gid** or **ibm-file-custom-gold-gid**

Important: The steps in this task assume that you are installing the [latest software images](#).

Before you begin

Ensure that a cluster administrator completed the required [pre-installation tasks](#) for your environment. Specifically, verify that a cluster administrator completed the following tasks:

1. If you are using the specialized installation method, ensure that IBM Cloud Pak foundational services is installed. For details, see [Installing IBM Cloud Pak foundational services](#).
2. For environments that use a private container registry, such as air-gapped environments, the Cloud Pak for Data software images are mirrored to the private container registry. For details, see [Mirroring images to your private container registry](#).
3. The cluster is configured to pull the software images. For details, see [Configuring your cluster to pull Cloud Pak for Data images](#).
4. The Cloud Pak for Data operator subscription and the **IBM Namespace Scope Operator** subscription exist. For details, see [Creating operator subscriptions](#).

If you do not complete these steps, the Cloud Pak for Data installation will fail.

Procedure

To install Cloud Pak for Data:

1. Log in to the Red Hat® OpenShift Container Platform as a user with sufficient permissions to complete the task:

```
oc login OpenShift_URL:port
```

2. Enable the IBM Cloud Pak for Data platform operator and the IBM Cloud Pak foundational services operator to watch the project where you will install IBM Cloud Pak for Data:

- >
Express installations

Create an operand request to grant permission to the IBM Cloud Pak for Data platform operator and the IBM Cloud Pak foundational services operator to manage the project where you plan to install Cloud Pak for Data:

```
cat <<EOF |oc apply -f -
apiVersion: operator.ibm.com/v1alpha1
kind: OperandRequest
metadata:
  name: empty-request
  namespace: cpd-instance # Replace with the project where you will install Cloud Pak for Data
spec:
  requests: []
EOF
```

- >
Specialized installations

Update the `IBM NamespaceScope Operator` in the Cloud Pak for Data operators project to watch the project where you plan to install Cloud Pak for Data.

Edit the `namespaceMembers` list to add the project where you plan to install Cloud Pak for Data. For example, if you plan to install Cloud Pak for Data in the `cpd-instance` project, add that project to the list:

```
cat <<EOF |oc apply -f -
apiVersion: operator.ibm.com/v1
kind: NamespaceScope
metadata:
  name: cpd-operators
  namespace: cpd-operators # (Default) Replace with the Cloud Pak for Data platform operator project name
spec:
  csvInjector: # This setting is required for some services. Do not delete this line if you
    enable: true # This setting is required for some services. Do not delete this line if you
  namespaceMembers:
    - cpd-operators # (Default) Replace with the Cloud Pak for Data platform operator project name
    - cpd-instance # Replace with the project where you will install Cloud Pak for Data
EOF
```

3. Create a custom resource to install Cloud Pak for Data. Follow the appropriate guidance for your environment:

- >
The cluster uses the recommended storage class names on Red Hat OpenShift Container Storage

The recommended storage class names are described in [Setting up shared persistent storage](#).

Create a custom resource with the following format:

```
cat <<EOF |oc apply -f -
apiVersion: cpd.ibm.com/v1
kind: Ibmcpcd
metadata:
  name: ibmcpcd-cr # This is the recommended name, but you can change it
  namespace: cpd-instance # Replace with the project where you will install Cloud Pak for Data
spec:
  license:
    accept: true
    license: Enterprise|Standard # Specify the Cloud Pak for Data license you purchased
  storageVendor: ocs
EOF
```

- >
The cluster uses the recommended storage class names on Portworx

The recommended storage class names are described in [Setting up shared persistent storage](#).

Create a custom resource with the following format:

```
cat <<EOF |oc apply -f -
apiVersion: cpd.ibm.com/v1
kind: Ibmcpcd
metadata:
  name: ibmcpcd-cr # This is the recommended name, but you can change it
  namespace: cpd-instance # Replace with the project where you will install Cloud Pak for Data
spec:
  license:
    accept: true
    license: Enterprise|Standard # Specify the Cloud Pak for Data license you purchased
  storageVendor: portworx
EOF
```

- >
The cluster uses IBM Spectrum Scale Container Native storage

The recommended storage class names are described in [Setting up shared persistent storage](#).

Create a custom resource with the following format:

```
cat <<EOF |oc apply -f -
apiVersion: cpd.ibm.com/v1
kind: Ibmcpcd
metadata:
  name: ibmcpcd-cr # This is the recommended name, but you can change it
  namespace: cpd-instance # Replace with the project where you will install Cloud Pak for
Data
spec:
  license:
```



```

    accept: true
    license: Enterprise|Standard # Specify the Cloud Pak for Data license you purchased
    storageClass: ibm-spectrum-scale-sc
EOF

```

- `>`
The cluster uses NFS storage

The recommended storage class names are described in [Setting up shared persistent storage](#).

```

cat <<EOF |oc apply -f -
apiVersion: cpd.ibm.com/v1
kind: Ibmcpcd
metadata:
  name: ibmcpcd-cr # This is the recommended name, but you can change it
  namespace: cpd-instance # Replace with the project where you will install Cloud Pak for
Data
spec:
  license:
    accept: true
    license: Enterprise|Standard # Specify the Cloud Pak for Data license you purchased
    storageClass: RWX-storage-class # Replace with the name of a RWX storage class, such as
managed-nfs-storage
EOF

```

- `>`
The cluster uses custom storage classes

If your cluster uses storage class names other than those described in [Setting up shared persistent storage](#), you must tell Cloud Pak for Data what storage class names to use.

```

cat <<EOF |oc apply -f -
apiVersion: cpd.ibm.com/v1
kind: Ibmcpcd
metadata:
  name: ibmcpcd-cr # This is the recommended name, but you can change it
  namespace: cpd-instance # Replace with the project where you will install Cloud Pak for
Data
spec:
  license:
    accept: true
    license: Enterprise|Standard # Specify the Cloud Pak for Data license you purchased
    storageClass: RWX-storage-class # Replace with the name of a RWX storage class
    zenCoreMetadbStorageClass: RWO-storage-class # Replace with the name of a RWO storage class that points to
block storage
EOF

```

Important:

Configuring metadata storage

If your cluster does not use the recommended storage classes, you must configure metadata storage for Cloud Pak for Data by specifying a storage class for the `zenCoreMetadbStorageClass` setting. The storage class must be ReadWriteOnce (RWO) and point to block storage, such as:

- `ocs-storagecluster-ceph-rbd` on Red Hat OpenShift Container Storage
- `portworx-metastoredb-sc` on Portworx

Verifying the installation

When you create the custom resource, the IBM Cloud Pak for Data platform operator processes the contents of the custom resource and starts up the microservices that comprise the Cloud Pak for Data control plane, including the `zenservice`, which is defined by the `lite-cr` custom resource definition.

To check the status of the installation:

1. Change to the project where you installed Cloud Pak for Data. For example:

```
oc project cpd-instance
```

2. Get the status of the control plane:

- a. Run the following command to determine whether the `ibmcpcd-cr` has been created:

```
oc get Ibmcpcd ibmcpcd-cr -o jsonpath="{.status.controlPlaneStatus}"{'\n'}
```

Output	What to do next
InProgress	Wait a few minutes. Then, run the command again.
Completed	Go to step 2b.
Failed	Go to step 2b.

- b. Run the following command to determine whether the `lite-cr` is ready:

```
oc get ZenService lite-cr -o jsonpath="{.status.zenStatus}"{'\n'}
```

Output	What to do next
InProgress	Wait a few minutes. Then, run the command again. It can take up to 90 minutes for the command to return <code>Completed</code> . If the command still has not returned <code>Completed</code> after 90 minutes, contact IBM Software Support.
Completed	The Cloud Pak for Data control plane is ready when the command returns <code>Completed</code> . Go to step 3.
Failed	Contact IBM Software Support.

3. Get the URL of the Cloud Pak for Data web client:

```
oc get ZenService lite-cr -o jsonpath="{.status.url}"{'\n'}
```

The URL has the following format:

```
https://cpd-namespace.apps.OCP-default-domain
```

4. Get the initial password for the **admin** user:

```
oc extract secret/admin-user-details --keys=initial_admin_password --to=-
```

Important: Save the output of this command so that you can log in to the web client. It is strongly recommended that you change the initial password the first time that you log in to the web client.

Choosing an upgrade plan for the Cloud Pak for Data control plane

Automatic upgrade (recommended)

By default, the Cloud Pak for Data control plane will be automatically upgraded when you install a newer version of the IBM Cloud Pak for Data platform operator or IBM Cloud Pak foundational services on the cluster.

If you want to continue using the automatic upgrade plan, no additional action is required.

Note: For this release of Cloud Pak for Data, the default version of the control plane is 4.2.0. If you install the minimum required version of IBM Cloud Pak foundational services (Version 3.11), the control plane will remain at 4.2.0. If you install a later release of IBM Cloud Pak foundational services, you might see a later version of the control plane on your cluster.

Manual upgrade

If you want to manually upgrade the Cloud Pak for Data control plane when you install a newer version of the IBM Cloud Pak for Data platform operator, you can optionally *pin* the installation to a specific version.

To change to the manual upgrade plan:

1. Update the **ZenService** custom resource.

For a list of operand versions supported by the **zen** operator, see [Cloud Pak for Data operator and operand versions](#).

For example, to pin the installation at 4.2.0, run the following command:

```
oc patch ZenService lite-cr \
--namespace <cpd-instance> \
--type=merge \
--patch '{"spec": {"version": "4.2.0"}}'
```

Related reference

- [Pre-installation tasks](#)
- [Post-installation tasks](#)
- [Uninstalling the common core services](#)
- [Uninstalling Cloud Pak for Data](#)

Post-installation tasks

After you install Cloud Pak for Data, complete the following tasks.

- [Integrating with the IAM Service](#)

By default, IBM Cloud Pak for Data user records are stored in an internal repository database. However, it is strongly recommended that you use an enterprise-grade password management solution, such as single sign-on (SSO) or LDAP.

- [Creating a custom route to the platform](#)

After you install IBM Cloud Pak for Data, you can optionally customize the route by which users access the Cloud Pak for Data web client.

- [Configuring an external route to the Flight Server](#)

The Flight Server is a data connection service that enables assets, such as notebooks, to interact with various data sources without calling the REST APIs for the data sources. By default, the Flight Server is only available to the IBM Cloud Pak for Data instance where the Flight Server is running. However, a Red Hat OpenShift Container Platform project administrator can create an external route to the Flight Server to enable other applications to interact with it.

- [Securing communication ports](#)

To ensure secure transmission of network traffic to and from the Cloud Pak for Data cluster, you need to configure the communication ports used by the network.

- [Setting up the Cloud Pak for Data web client](#)

After you install Cloud Pak for Data, you can configure the web client to add users and set up email notifications.

Related reference

- [Pre-installation tasks](#)
- [Installing Cloud Pak for Data](#)
- [Uninstalling the common core services](#)
- [Uninstalling Cloud Pak for Data](#)

Integrating with the IAM Service

By default, IBM® Cloud Pak for Data user records are stored in an internal repository database. However, it is strongly recommended that you use an enterprise-grade password management solution, such as single sign-on (SSO) or LDAP.

If you use LDAP, you can choose between the following options:

Mechanism	Benefits	Drawbacks
-----------	----------	-----------

Mechanism	Benefits	Drawbacks
LDAP integration provided by Cloud Pak for Data	<p>You can use LDAP with or without SAML SSO. You can choose the level of integration with the LDAP server. You can use LDAP to:</p> <ul style="list-style-type: none"> Validate users' credentials Manage access to the platform 	<p>You can connect to a single LDAP server from each instance of Cloud Pak for Data. The LDAP configuration cannot be shared across Cloud Pak for Data instances or used by any other IBM Cloud Paks on the cluster.</p>
LDAP integration provided by the Identity and Access Management Service (IAM Service) in IBM Cloud Pak® foundational services	<p>You can connect to multiple LDAP servers, and the connections can be used by multiple instances of Cloud Pak for Data or other IBM Cloud Paks on the cluster.</p>	<p>Do not use this method if you have multiple LDAP servers that must be isolated from each other. For example, you maintain two instances of Cloud Pak for Data for different groups of users. Each group of users is managed by a different LDAP server, and you don't want the users to be able to see information about users in the other LDAP server.</p>

To use the LDAP integration provided by Cloud Pak for Data, see [Connecting to your identity provider](#).

Permissions you need for this task

You must be either:

- A cluster administrator
- An administrator of the following projects:
 - The project where IBM Cloud Pak foundational services is installed (`ibm-common-services`)
 - The project where the IBM Cloud Pak for Data platform operator is installed (either `ibm-common-services` or `cpd-operators`)
 - The project where Cloud Pak for Data is installed

When you need to complete this task

If you want to use the LDAP integration provided by the IAM Service, you must integrate Cloud Pak for Data with the IAM Service before you onboard users or create user groups.

When you integrate with the IAM Service, you delegate all authentication to the IAM Service. If you onboard users before you integrate with the IAM Service, existing users might not be able to log in to Cloud Pak for Data.

About this task

Important: Integrating with the IAM Service is irreversible.

Contact IBM Software support to reset Cloud Pak for Data to the previous state.

Procedure

1. Log in to Red Hat® OpenShift® Container Platform as a user with sufficient permissions to complete the task:

```
oc login OpenShift_URL:port
```

2. Modify the `ZenService` custom resource to add the `iamIntegration`:

`true` entry:

- a. Run the following command to get the name of the platform custom resource:

```
oc get Ibmcpcd -n Cloud-Pak-for-Data-project
```

By default, the custom resource name is `ibmcpcd-cr`.

- b. Run the following command to edit the platform custom resource:

```
oc edit Ibmcpcd custom-resource-name
```

- c. Add the `iamIntegration: true` entry to the custom resource:

```
apiVersion: cpd.ibm.com/v1
kind: Ibmcpcd
metadata:
  name: ibmcpcd-cr
  namespace: cpd-instance # The project where Cloud Pak for Data is installed
spec:
  csNamespace: ibm-common-services
  version: 4.0.0
  license:
    accept: true
    license: Enterprise
  storageClass: RWX-storage-class # The RWX storage class you specified during installation
  zenCoreMetaDbStorageClass: RWO-storage-class # The RWO storage class you specified during installation
  cloudpakfordata: true
  iamIntegration: true
```

Note: In the preceding example, the cluster uses custom storage class names. Your custom resource file might use the `storageVendor` setting instead of the `storageClass` and `zenCoreMetaDbStorageClass` settings.

- d. Save your changes to the `ZenService` custom resource. For example, if you are using `vi`, hit `esc` and enter `:wq`

3. Check the status of the `ZenService` custom resource:

```
oc get ZenService custom-resource-name -o jsonpath="{.status}"
```

The command triggers a reconciliation in the `Zen` operator:

```
{"conditions":[{"lastTransitionTime":"2021-06-20T01:05:55Z","message":"Running reconciliation","reason":"Running","status":"True","type":"Running"}],"url":"cloud-pak-for-data-URL","zenOperatorBuildNumber":"zen operator build 305","zenStatus":"InProgress"}
```

It might take up to 20 minutes for the process to complete if the IAM Service needs to be started and configured.

Tip: You can follow the logs generated by the **Zen** operator pod.

The location of the **Zen** operator pod depends on whether IBM Cloud Pak foundational services and the IBM Cloud Pak for Data platform operator are installed in the same project (**ibm-common-services**) or separate projects (**cpd-operators**).

```
oc logs -n project-name $(oc get pod -n project-name -l name=ibm-zen-operator -o jsonpath='{.items[0].metadata.name}') -f
```

4. Wait for the **ZenService** custom resource to return the following status:

```
{"conditions":[{"ansibleResult":{"changed":22,"completion":"2021-06-12T06:57:56.861621","failures":0,"ok":288,"skipped":324},"lastTransitionTime":"2021-06-20T01:05:55Z","message":"Awaiting next reconciliation","reason":"Successful","status":"True","type":"Running"}],"url":"cloud-pak-for-data-URL","zenOperatorBuildNumber":"zen operator build 305","zenStatus":"Completed"}
```

5. Confirm that the IAM Service is set up:

- a. Go to the Cloud Pak for Data web client.

Tip: If you don't know the URL, you can run the following command to get the route to the web client:

```
oc get ZenService lite-cr -o jsonpath="{.status.url}"
```

- b. Verify that the login page includes the following options:

- Enterprise LDAP
- OpenShift authentication
- IBM provided credentials (admin only)

6. Get the initial password for the **admin** user from the IAM Service:

```
oc extract -n ibm-common-services secret/platform-auth-idp-credentials --keys=admin_password --to=-
```

Creating a custom route to the platform

After you install IBM® Cloud Pak for Data, you can optionally customize the route by which users access the Cloud Pak for Data web client.

Before you begin

Required permissions: To complete this task, you must be an administrator of the project (namespace) where Cloud Pak for Data is installed.

About this task

In Red Hat® OpenShift®, a route is how you expose a service. A route is an externally reachable hostname, such as <https://www.ibm.com>.

By default, the route to the Cloud Pak for Data web client has the following format:

```
https://cpd-namespace.apps.OCP-default-domain/zen/
```

You can create a custom route to override the default URL.

Procedure

1. Log in to your Red Hat OpenShift cluster as a project administrator:

```
oc login OpenShift_URL:Port
```

2. Change to the project where Cloud Pak for Data is installed:

```
oc project project_name
```

3. Run the following command to create the new route:

```
oc create route passthrough route_name --hostname hostname --service=ibm-nginx-svc
```

Replace *route_name* with the label you want to use for the route, and replace *hostname* with the FQDN you want to assign to the route. The following example command creates a route called **mycompany** with a FQDN of **mycompany.com**:

```
oc create route passthrough mycompany --hostname mycompany.com --service=ibm-nginx-svc
```

4. Run the following command to annotate the route:

```
oc annotate route route_name haproxy.router.openshift.io/balance=roundrobin
```

Replace *route_name* with the value that you specified in the preceding step.

Users can now access the Cloud Pak for Data through the route that you created.

5. Run the following command to ensure that any HTTP requests are automatically redirected to HTTPS:

```
oc patch route route_name -p '{"spec":{"tls":{"insecureEdgeTerminationPolicy":"Redirect"}}}'
```

Configuring an external route to the Flight Server

The Flight Server is a data connection service that enables assets, such as notebooks, to interact with various data sources without calling the REST APIs for the data sources. By default, the Flight Server is only available to the IBM® Cloud Pak for Data instance where the Flight Server is running. However, a Red Hat® OpenShift® Container Platform project administrator can create an external route to the Flight Server to enable other applications to interact with it.

Permissions you need for this task
You must be either:

- A cluster administrator
- An administrator of the Red Hat OpenShift project (namespace) where Cloud Pak for Data is installed.

About this task

Common core services The Flight Server is available only if the common core services are installed.

Remember: The common core services are not a separately installable component. They are automatically installed if you install a service that has a dependency on them. For information about which services install the common core services, see [Service software requirements](#).

Procedure

To create an external route to the Flight Server:

1. Log in to your Red Hat OpenShift Container Platform as a user with sufficient permissions to complete the task:

```
oc login OpenShift_URL:port
```

2. Change to the project where Cloud Pak for Data and the Flight Server are installed.

```
oc project project-name
```

3. Run the following command to create the route:

```
oc create route passthrough route-name --service=wdp-connect-flight
```

Replace *route_name* with the label you want to use for the route

If you want to use a specific hostname, you can include the `--hostname` *hostname* argument in the command. For example:

```
oc create route passthrough route-name --hostname hostname --service=wdp-connect-flight
```

4. Run the following command to get the URL:

```
oc get route route-name
```

Use this URL to access the Flight Server from external applications.

Securing communication ports

To ensure secure transmission of network traffic to and from the Cloud Pak for Data cluster, you need to configure the communication ports used by the network.

Cluster ports

The primary port is what the Red Hat® OpenShift® router exposes. See [Configuring and managing cluster networking for Red Hat OpenShift Container Platform 4.6](#) for details.

Ports for services

When you provision a new service or integration on your Cloud Pak for Data cluster, the services might require connections to be made from outside the cluster. For example, you might require connections when you access databases, or run data virtualization through an ODBC/JDBC connection. If the service or integration requires connections to be made to the cluster, locate the port numbers from each service's Details page and open those network ports. Each port is TCP, and randomly allocated between the 30000-32767 range.

Setting up the Cloud Pak for Data web client

After you install Cloud Pak for Data, you can configure the web client to add users and set up email notifications.

Before you begin

You will need the web client URL provided by the Cloud Pak for Data installation.

Tip: The web client URL is an OpenShift® route, which can be viewed by your cluster administrator by entering the `oc get routes` command. See [Temporarily disabling the route to the platform](#) for details on manually disabling or creating one.

Procedure

To configure the Cloud Pak for Data web client:

1. Sign in to the Cloud Pak for Data web client as the default administrator. The default user name is `admin`, and the default password is `password`.
2. Change the password for the `admin` user:

- a. From the menu, click Administer > User management.
- b. On the Users page, select the admin user and select Edit user from the actions menu.
- c. Change the password and specify an email address.

What to do next

It is strongly recommended that you complete the following tasks before you [give users access to the web client](#):

- [Using a custom TLS certificate for HTTPS connections to the platform](#)
- [Configuring single sign-on](#)
- [Connecting to your LDAP server](#)
- [Displaying a terms and conditions prompt](#)
- [Enabling email notifications](#)
- [Enabling users to access the web client from platform-generated emails](#)

- [Using a custom TLS certificate for HTTPS connections to the platform](#)
The Cloud Pak for Data installation includes a self-signed TLS certificate that can be used to enable HTTPS connections. By default, this certificate is untrusted by all HTTPS clients. However, you can replace the default certificate with your own TLS certificate.
- [Using a CA certificate to connect to internal servers from the platform](#)
If you want to enable the IBM Cloud Pak for Data platform to use your company's CA certificate to validate certificates from your internal servers, you must create a secret that contains the CA certificate. Additionally, if your internal servers use an SSL certificate that is signed using your company's CA certificate, you must create this secret to enable the platform to connect to the servers.
- [Configuring single sign-on](#)
You can use Security Assertion Markup Language (SAML) for single sign-on (SSO) to the IBM Cloud Pak for Data web client.
- [Changing shared credentials settings](#)
A Red Hat OpenShift Container Platform project administrator can optionally disable the option to use shared credentials in connections in platform connections, projects, and catalogs.
- [Setting the idle session timeout](#)
You can adjust the idle session timeout for IBM Cloud Pak for Data in accordance with your security and compliance requirements. If a user leaves their session idle in a web browser for the specified length of time, the user is automatically logged out of the web client.
- [Restricting the list of storage classes that are available to an instance of Cloud Pak for Data](#)
After you install IBM Cloud Pak for Data, you can optionally restrict the list of storage classes that end users can see and select in the web client. By default, users can see all of the storage classes that are defined on the cluster. However, you might want to prevent users from selecting certain storage classes when creating new storage volumes or deploying service instances.
- [Displaying a terms and conditions prompt](#)
If you need users to accept terms and conditions before they use the web client, you can enable a dialog that prompts users to accept the terms and conditions before they can log in to the web client. For example, you might need to enable the prompt to comply with the Federal Information Security Management Act (FISMA) regulations.
- [Enabling email notifications](#)
You can configure a connection to your SMTP server so that Cloud Pak for Data can send email to users.
- [Enabling users to access the web client from platform-generated emails](#)
Some services in IBM Cloud Pak for Data generate notifications. For example, collaborators in an analytics project get a notification when assets or new collaborators are added to the project. If you configure a connection to your SMTP server, users can receive these notifications through email. To ensure that these emails include active links to the web client, you must add the `URL_PREFIX` for your deployment to the Cloud Pak for Data product-configmap.
- [Enabling users to create NFS storage volumes](#)
To enable users to connect to remote NFS storage volumes from Cloud Pak for Data, a cluster administrator must create a cluster role and cluster role binding to grant the `meta-api` the appropriate permissions to provision persistent volumes.
- [Enabling users to connect to external SMB storage volumes](#)
To enable users to connect to remote SMB storage volumes from IBM Cloud Pak for Data a cluster administrator must install the SMB CSI Driver for Kubernetes to enable the cluster to communicate with your remote SMB volumes. Then the administrator must create a cluster role and cluster role binding to grant the `meta-api` the appropriate permissions to provision persistent volumes.

Using a custom TLS certificate for HTTPS connections to the platform

The Cloud Pak for Data installation includes a self-signed TLS certificate that can be used to enable HTTPS connections. By default, this certificate is untrusted by all HTTPS clients. However, you can replace the default certificate with your own TLS certificate.

IBM® Cloud Pak for Data exposes one HTTPS port as the primary access point for the web client and for API requests. On Red Hat® OpenShift®, the port is exposed as an OpenShift route.

Before you begin

Required permissions

To complete this task, you must have one of the following roles:

- Red Hat OpenShift cluster administrator
- Red Hat OpenShift project administrator on the project where Cloud Pak for Data is installed

To complete this task, you must have your own certificate and private key file that meet the following requirements:

- Both files are in PEM format.
- The certificate is named cert.crt.
The certificate can be a bundle that contains your server, intermediates, and root certificates concatenated (in the proper order) into one file. The necessary certificates must be enabled as trusted certificates on the clients that connect to the cluster.
- The private key is named cert.key.

Procedure

To replace the default TLS certificate with your custom TLS certificate:

1. Place the cert.crt and cert.key files in the same directory on your local file system.
2. Change to the directory where the files are located.
3. Connect to your OpenShift cluster:

```
oc login OpenShift_URL:port
```

4. Set the context to the project where Cloud Pak for Data is deployed:

```
oc project Project_name
```

5. Create a secret to store your certificate files:

```
oc create secret generic external-tls-secret --from-file=cert.crt=./cert.crt --from-file=cert.key=./cert.key --dry-run -o yaml | oc apply -f -
```

Important: Do not change the name of the secret. You must use the name `external-tls-secret`.

Wait for the command to return a message that the secret was created:

```
secret/external-tls-secret created
```

Then, wait another minute to ensure that `kubelet` has sufficient time to detect where the secret will be used and to mount the secret to the `ibm-nginx` pods.

6. Reload `ibm-nginx`:

```
for i in `oc get pods | grep ibm-nginx | cut -f1 -d\ `; do oc exec ${i} -- /scripts/reload.sh; done
```

The output should be similar to the following output:

```
reloading nginx conf
Setting up ssl certificate files...
Custom ssl certificate files were found. Processing them...
lrwxrwxrwx. 1 1000321000 root 50 DATE-AND-TIME /nginx_data/defaults.d/external-server.conf ->
/nginx_data/defaults.d/external-server.active.conf
nginx: the configuration file /usr/local/openresty/nginx/conf/nginx.conf syntax is ok
nginx: configuration file /usr/local/openresty/nginx/conf/nginx.conf test is successful
TIMESTAMP [notice] 76#76: signal process started
reloading nginx conf
Setting up ssl certificate files...
Custom ssl certificate files were found. Processing them...
lrwxrwxrwx. 1 1000321000 root 50 DATE-AND-TIME /nginx_data/defaults.d/external-server.conf ->
/nginx_data/defaults.d/external-server.active.conf
nginx: the configuration file /usr/local/openresty/nginx/conf/nginx.conf syntax is ok
nginx: configuration file /usr/local/openresty/nginx/conf/nginx.conf test is successful
TIMESTAMP [notice] 76#76: signal process started
reloading nginx conf
Setting up ssl certificate files...
Custom ssl certificate files were found. Processing them...
lrwxrwxrwx. 1 1000321000 root 50 DATE-AND-TIME /nginx_data/defaults.d/external-server.conf ->
/nginx_data/defaults.d/external-server.active.conf
nginx: the configuration file /usr/local/openresty/nginx/conf/nginx.conf syntax is ok
nginx: configuration file /usr/local/openresty/nginx/conf/nginx.conf test is successful
TIMESTAMP [notice] 76#76: signal process started
```

Verify that the certificate files were found.

Using a CA certificate to connect to internal servers from the platform

If you want to enable the IBM® Cloud Pak for Data platform to use your company's CA certificate to validate certificates from your internal servers, you must create a secret that contains the CA certificate. Additionally, if your internal servers use an SSL certificate that is signed using your company's CA certificate, you must create this secret to enable the platform to connect to the servers.

After you create the secret, the CA certificate is used in the following connections:

- Platform connections
- Connections created in an analytics project
- Connections created in a catalog

Before you begin

Required permissions

To complete this task, you must have one of the following roles:

- Red Hat® OpenShift® cluster administrator
- Red Hat OpenShift project administrator on the project where Cloud Pak for Data is installed

To complete this task, you must have your own certificate file in PEM format.

Procedure

To add a CA certificate to the platform:

1. Place the PEM file on your local file system.
2. Change to the directory where the files are located.
3. Log in to Red Hat OpenShift Container Platform as a user with sufficient permissions to complete the task.

```
oc login OpenShift_URL:port
```

4. Set the context to the project where Cloud Pak for Data is deployed, for example `cpd-instance`

```
oc project cpd-instance
```

5. Create a secret named `connection-ca-certs` to store the CA certificate.
Important: The secret must be named `connection-ca-certs`. If you use a different name, the platform will not be able to locate the secret.

```
oc create secret generic connection-ca-certs --from-file=my-cert.pem=./my-cert.crt
```

Wait for the command to return a message that the secret was created:

```
secret/connection-ca-certs created
```

6. If the following pods are running on your cluster, restart them so that they can use the `connection-ca-certs` secret:

```
wdp-connect pods
```

Run the following command to restart any `wdp-connect` pods in the project:

```
for i in `oc get pods | grep wdp-connect- | cut -f1 -d\ `; do oc delete po ${i}; done
```

If there are any `wdp-connect` pods in the project, the command should return output with the following format:

```
pod "wdp-connect-connector-6f968c97ff-4lcxc" deleted
pod "wdp-connect-connection-77879f485d-f7cf1" deleted
pod "wdp-connect-flight-6b75df645b-ts4rg" deleted
```

Configuring single sign-on

You can use Security Assertion Markup Language (SAML) for single sign-on (SSO) to the IBM® Cloud Pak for Data web client.

Before you begin

You must have an existing SAML identity provider (IdP). Work with your IdP administrator to gather the following information:

Parameter	Description	Value
entryPoint	The URL of the login page for your identity provider.	
fieldToAuthenticate	The name of the parameter you use to authenticate with the identity provider, such as <code>emailAddress</code> or <code>username</code> . If you plan to use LDAP and SAML, ensure that you use the same attribute to identify users. This parameter should have the same value as the User search field in your LDAP configuration.	
spCert	The private key used to sign SAML requests to the identity provider. The certificate corresponding to this key needs to be set when you register Cloud Pak for Data with your identity provider so that the SAML requests can be verified by your identity provider. If you do not specify a certificate, the requests won't be signed.	Remove the "BEGIN PRIVATE KEY" and "END PRIVATE KEY" lines and provide the private key as a single line.
idpCert	The certificate provided by the identity provider to verify SAML responses from the identity provider.	Remove the "BEGIN CERTIFICATE" and "END CERTIFICATE" lines and provide the certificate as a single line.
issuer	The name that you want to use to register Cloud Pak for Data with your identity provider. If you do not specify a value, the default (<code>ibm_privatecloud</code>) is used.	
identifierFormat	The format of requests from Cloud Pak for Data to the identity provider. The format must be supported by the identity provider. If you do not specify a format, the default format (<code>urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress</code>) is used	
callbackUrl	An approved URL (that you set with the SAML identity provider) to redirect users to after they successfully authenticate using SSO. For example, to redirect successfully authenticated users to the Cloud Pak for Data landing page, you can specify <code>https://cluster/auth/login/sso/callback</code> .	
disableRequesteAuthnContext	A boolean parameter for AD FS client authentication. If set the true, the authentication context is disabled so that the IDP determines the method of authentication. If you do not specify a value, then the default is false.	

About this task

To configure SSO, you must specify information about your Identity Provider in a configuration file. Use the preceding table to gather the required information that you need to supply in the configuration file.

Important: It is strongly recommended that you complete this task before you add users to Cloud Pak for Data. If you have already added users to Cloud Pak for Data you must re-add the users with their SAML ID to enable them to use SSO.

Procedure

1. Log in to your Red Hat® OpenShift® cluster as a project administrator:


```
oc login OpenShift_URL:port
```

2. Enable SAML by running the following command:

```
oc exec -it -n namespace \  
$(oc get pod -n namespace -l component=usermgmt | tail -1 | cut -f1 -d\ ) \  
-- bash -c "vi /user-home/_global_/config/saml/samlConfig.json"
```

Replace *namespace* with the namespace where Cloud Pak for Data is deployed.

3. In the `samlConfig.json` file, specify the appropriate values for your environment.

```
{  
  "entryPoint": "",  
  "fieldToAuthenticate": "",  
  "spCert": "",  
  "idpCert": "",  
  "issuer": "",  
  "identifierFormat": "",  
  "callbackUrl": ""  
}
```

4. Save your changes to `samlConfig.json`.
 - a. Press `Esc`.
 - b. Press `..`
 - c. Enter `:x`.
 - d. Press `Enter`.
5. Run the following command to delete the `usermgmt` pods:

```
oc delete pods -l component=usermgmt
```

What to do next

Wait several minutes before you attempt to log in to the web client. The instructions restart the `usermgmt` pods. If the pods are not running, you will not be able to log in.

If you previously added users to Cloud Pak for Data, you must re-add the users with their SAML ID to enable them to use SSO. To add users:

1. Go directly to the web client log in page by appending the following path to your Cloud Pak for Data URL: `/auth/login/zen-login.html`.
2. Log in to the web client as the admin user or another administrator with user management permissions.
3. Add users with their SAML IDs. For details, see [Managing users](#).

Disabling SAML

Procedure

1. Disable SAML by running the following command:

```
oc exec -it -n namespace \  
$(oc get pod -n namespace -l component=usermgmt | tail -1 | cut -f1 -d\ ) \  
-- bash -c "rm /user-home/_global_/config/saml/samlConfig.json"
```

Replace *namespace* with the namespace where Cloud Pak for Data is deployed.

2. Run the following command to delete the `usermgmt` pods:

```
oc delete pods -l component=usermgmt
```

Changing shared credentials settings

A Red Hat® OpenShift® Container Platform project administrator can optionally disable the option to use shared credentials in connections in platform connections, projects, and catalogs.

About this task

When a connection is created with shared credentials, all users access the connection with the same credentials. However, because the credentials are shared, it is difficult to audit access to the connection, to identify the source of data loss, or identify the source of a security breach.

Permissions you need for this task

You must be a Red Hat OpenShift Container Platform project or cluster administrator.

When you need to complete this task

If individual accountability is required, especially by industry-specific regulations that your organization must comply with, it is recommended that you disable shared credentials on the platform.

However, this setting is only applicable if the Cloud Pak for Data common core services are installed. If you can see `Data_>Platform connections` in the navigation, you should determine whether you want to allow the use of shared credentials or whether you want to require each user to provide personal credentials.

If you want to require each user to provide personal credentials, disable the option to use shared credentials.

Procedure

1. Log in to your Red Hat OpenShift Container Platform cluster as a user with sufficient permissions to complete the task:

```
oc login OpenShift_URL:port
```

2. Change to the project where Cloud Pak for Data is installed:

```
oc project Project_name
```

3. Run the following command to edit the Cloud Pak for Data config-wdp-connect-connection file:

```
oc edit configmap config-wdp-connect-connection
```

4. Change the value of allow-shared-credentials parameter from `true` to `false`.

```
allow-shared-credentials: false
```

5. Save your changes to the config-wdp-connect-connection file.
For example, if you are using `vi`, hit `esc` and enter:

```
:wq
```

Setting the idle session timeout

You can adjust the idle session timeout for IBM® Cloud Pak for Data in accordance with your security and compliance requirements. If a user leaves their session idle in a web browser for the specified length of time, the user is automatically logged out of the web client.

Before you begin

Required permissions

To complete this task, you must have one of the following roles:

- Red Hat® OpenShift® cluster administrator
- Red Hat OpenShift project administrator on the project where Cloud Pak for Data is installed

About this task

By default, Cloud Pak for Data logs users out after 12 hours. You can edit the Cloud Pak for Data `product-configmap` to adjust:

The length of time until a user's session expires (`TOKEN_EXPIRY_TIME`).

The default is 12 hours.

If you set `TOKEN_EXPIRY_TIME: "1"`, a user's session will expire in after 1 hour of inactivity. If you set `TOKEN_EXPIRY_TIME: "0.5"`, a user's session will expire after 30 minutes of inactivity. When the user leaves their session idle for the specified length of time, the user is automatically logged out of the web client.

It is recommended that you set the value between 0.1 and 1.

The length of time that a user has to refresh their session (`TOKEN_REFRESH_PERIOD`).

The default is 12 hours.

If you set `TOKEN_REFRESH_PERIOD: "1"` and the user's session does not expire, the user's session is automatically refreshed during this 60 minute period. The session is extended based on the value that is set for the `TOKEN_EXPIRY_TIME` parameter. However, after the token refresh period passes, the user must log back into the web client when their current session expires.

It is recommended that you set the value between 1 and 24.

If you don't want to allow users to extend their sessions, set the value of the `TOKEN_REFRESH_PERIOD` parameter to a value less than the value of the `TOKEN_EXPIRY_TIME` parameter.

For example, as an administrator, you configure:

```
TOKEN_EXPIRY_TIME: "0.5"  
TOKEN_REFRESH_PERIOD: "2"
```

If a user starts work at 8 AM and logs in to the web client, the user must be active in the web session within 30 minutes for their token to be refreshed:

- If the user stops using the web client at 8:10 and attempts to use the web client again until 8:41, the user must re-authenticate to the web client because their session expired.
- If the user remains active in their session and their token refreshes at 9:59 AM, their session will last until 10:29 AM. However, when the session expires at 10:29, the user must re-authenticate to the web client because the token refresh period expired.

Procedure

1. Log in to your OpenShift cluster:

```
oc login OpenShift_URL:port
```

2. Change to the project where Cloud Pak for Data is deployed:

```
oc project Project
```

3. Run the following command to edit the Cloud Pak for Data `product-configmap`:

```
oc edit configmap product-configmap
```

4. Add an entry for the `TOKEN_EXPIRY_TIME` parameter to the `data` section of the `product-configmap` file. For example:

```
data:  
  ...
```

```
TOKEN_EXPIRY_TIME: "1"
...
```

5. Add an entry for the `TOKEN_REFRESH_PERIOD` parameter to the `data` section of the `product-configmap` file. For example:

```
data:
...
  TOKEN_REFRESH_PERIOD: "1"
...
```

6. Save your changes to the `product-configmap` file.
For example, if you are using `vi`, hit `esc` and enter:

```
:wq
```

7. You must restart the `usermgmt` pods for the changes to take effect. To restart the pods, run the following command:

```
oc delete pod -l component=usermgmt
```

Restricting the list of storage classes that are available to an instance of Cloud Pak for Data

After you install IBM® Cloud Pak for Data, you can optionally restrict the list of storage classes that end users can see and select in the web client. By default, users can see all of the storage classes that are defined on the cluster. However, you might want to prevent users from selecting certain storage classes when creating new storage volumes or deploying service instances.

Before you begin

Required permissions: To complete this task, you must be an administrator of the project (namespace) where Cloud Pak for Data is installed.

Work with your cluster administrator to determine which storage classes to display in the web client. As part of this discussion, you might need to consider which services you plan to deploy on this instance of Cloud Pak for Data.

About this task

You can restrict the list of storage classes by creating a list of allowed storage classes in the Cloud Pak for Data `product-configmap` file.

Procedure

1. Log in to your Red Hat® OpenShift® cluster as a project administrator:

```
oc login OpenShift_URL:Port
```

2. Change to the project where Cloud Pak for Data is installed:

```
oc project Project_name
```

3. Run the following command to edit the Cloud Pak for Data `product-configmap`:

```
oc edit configmap product-configmap
```

4. Add an entry for the `ALLOWED_STORAGE_CLASSES` parameter to the `data` section of the `product-configmap` file. For example:

```
data:
...
  ALLOWED_STORAGE_CLASSES: "storage-class-1, storage-class-2, storage-class-3"
...
```

Specify multiple storage classes as a comma separated list.

5. Save your changes to the `product-configmap` file.
For example, if you are using `vi`, hit `esc` and enter:

```
:wq
```

Displaying a terms and conditions prompt

If you need users to accept terms and conditions before they use the web client, you can enable a dialog that prompts users to accept the terms and conditions before they can log in to the web client. For example, you might need to enable the prompt to comply with the Federal Information Security Management Act (FISMA) regulations.

About this task

When you configure the web client to display a terms and conditions prompt, you must specify the following information:

- The header text for the dialog
- The terms and conditions that the user must accept
- The prompt that the user must click to acknowledge that they accept the terms and conditions

Procedure

1. Log in to your Red Hat® OpenShift® cluster as a project administrator:

```
oc login OpenShift_URL:port
```

2. Create the login-dialog.json configuration file:

Property	Description
enabled	Set enabled to true to enable the dialog. For example: <pre>"enabled": true,</pre> Valid values: true or false
headerText	Specify the text to display as the header in the dialog. For example: <pre>"headerText": "Terms of use",</pre> Use standard JSON string format.
dialogText	Specify the terms and conditions that the user must agree to before they can access the web client. For example: <pre>"dialogText": "The terms and conditions of use that your user must accept.",</pre> Use standard JSON string format.
acceptText	Specify the text that the user must click to acknowledge that they agree to the terms and conditions. For example: <pre>"acceptText": "I understand and accept the terms",</pre> Use standard JSON string format.

3. Copy the login-dialog.json file into the config directory:

```
oc cp login-dialog.json $(oc get pod -n namespace -l component=usermgmt | tail -1 | cut -f1 -d\ ):/user-home/_global_/config/
```

Replace *namespace* with your project namespace.

Enabling email notifications

You can configure a connection to your SMTP server so that Cloud Pak for Data can send email to users.

About this task

To send emails to users, one of the following services must be installed:

- Watson™ Studio
- Watson Knowledge Catalog

If neither service is installed, the SMTP configuration is not used.

Procedure

To enable Cloud Pak for Data to send email:

1. Log in to the web client as an administrator.
2. From the menu, select **Administer > Configure platform**.
3. On the SMTP settings page, specify the following information:
 - Your SMTP mail server address.
 - The port number of your SMTP server.
Important: If you specify a secure port, you must select Use TLS connection so that Cloud Pak for Data can communicate with your SMTP server.
 - Specify the appropriate SMTP credentials for your environment:

Method of sending communications	SMTP server requires authentication	SMTP server does not require authentication
My SMTP server uses a mailer daemon to send communications	You must specify the following fields: <ul style="list-style-type: none">• SMTP username• SMTP password	You don't need to specify any fields. However, if you want to override the mailer daemon, you can specify a From account.
My SMTP server uses a default account to send communications	You must specify the following fields: <ul style="list-style-type: none">• SMTP username• SMTP password• From account	You must specify the following fields: <ul style="list-style-type: none">• From account

4. Click **Save**. If your SMTP configuration is successful, you will receive a confirmation email.
 - If you specified a From account when you configured the connection to your SMTP server, the confirmation email is sent to the account specified in the From account field.
 - If you did not specify a From account when you configured a connection to your SMTP server, the confirmation email is sent to the account specified in the SMTP username field.

Results

Depending on your configuration, notification emails are sent from one of the following accounts:

- If you specified a From account when you configured the connection to your SMTP server, notifications are sent from the account specified in the From account field.
- If you did not specify a From account when you configured a connection to your SMTP server, notifications are sent from the mailer daemon.

Enabling users to access the web client from platform-generated emails

Some services in IBM® Cloud Pak for Data generate notifications. For example, collaborators in an analytics project get a notification when assets or new collaborators are added to the project. If you configure a connection to your SMTP server, users can receive these notifications through email. To ensure that these emails include active links to the web client, you must add the `URL_PREFIX` for your deployment to the Cloud Pak for Data product-configmap.

About this task

A Red Hat® OpenShift® project (namespace) administrator can edit the Cloud Pak for Data product-configmap to specify the `URL_PREFIX` for your deployment.

The `URL_PREFIX` is the domain name at the beginning of your deployment URL. For example, if your deployment of Cloud Pak for Data is accessible from `https://domain.my.company.com/zen`, your domain name is `domain.my.company.com`. Do not include the protocol in the value that you specify.

If you use the default port, `443`, you do not need to specify the port number in the value for the `URL_PREFIX` parameter. However, if you use a non-standard port, include it in the `URL_PREFIX`. For example, if you use port `31843`, your entry would be:

```
URL_PREFIX: domain.my.company.com:31843
```

Procedure

To enable users to access the web client from platform-generated emails:

1. Log in to your Red Hat OpenShift cluster as a project administrator:

```
oc login OpenShift_URL:port
```

2. Change to the project where you installed Cloud Pak for Data:

```
oc project Project
```

3. Run the following command to edit the Cloud Pak for Data product-configmap:

```
oc edit cm product-configmap
```

4. Add an entry for the `URL_PREFIX` parameter to the `data` section of the product-configmap file. For example:

```
data:
  ...
  SHOW_USER_APPROVAL: "false"
  URL_PREFIX: domain.my.company.com
  ...
```

5. Save your changes to the product-configmap file.
For example, if you are using `vi`, hit `esc` and enter:

```
:wq
```

The changes are automatically applied to the platform.

Enabling users to create NFS storage volumes

To enable users to connect to remote NFS storage volumes from Cloud Pak for Data, a cluster administrator must create a cluster role and cluster role binding to grant the `meta-api` the appropriate permissions to provision persistent volumes.

About this task

Permissions you need for this task

You must be a cluster administrator.

When you need to complete this task

Complete this task only if you want to enable Cloud Pak for Data users to connect to storage volumes on a remote NFS server.

Note: The External NFS option is always visible on the New storage volume page. However, if you do not complete this task, users cannot connect to NFS storage volumes from the New storage volume page or through the Volumes API.

Procedure

To enable users to connect to a remote NFS storage volume:

1. Log on to Red Hat® OpenShift® Container Platform as a user with sufficient permissions to complete this task:

```
oc login OpenShift_URL:port
```

2. Run the following command to create the required cluster role and cluster role binding for the `meta-api`:

Tip: The same role binding is required to use remote SMB storage volumes. If you already completed [Enabling users to connect to external SMB storage volumes](#), this role binding already exists.

```
cat <<EOF | oc apply -f -
---
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
  # "namespace" omitted. ClusterRoles are not scoped to a namespace.
  name: ibm-zen-volumes-cluster-role
rules:
- apiGroups: [""]
  #
  # at the HTTP level, the name of the resource for accessing Secret
  # objects is "secrets"
  resources: ["persistentvolumes"]
  verbs: ["create", "get", "list", "patch", "update", "watch", "delete", "use"]
---
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRoleBinding
metadata:
  name: ibm-zen-volumes-cluster-role-binding
subjects:
- kind: ServiceAccount
  name: ibm-zen-operator-serviceaccount
  namespace: ibm-common-services # The namespace where the IBM Cloud Pak foundational services are installed
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: ClusterRole
  name: ibm-zen-volumes-cluster-role
EOF
```

Enabling users to connect to external SMB storage volumes

To enable users to connect to remote SMB storage volumes from IBM® Cloud Pak for Data a cluster administrator must install the SMB CSI Driver for Kubernetes to enable the cluster to communicate with your remote SMB volumes. Then the administrator must create a cluster role and cluster role binding to grant the `meta-api` the appropriate permissions to provision persistent volumes.

About this task

Permissions you need for this task

You must be a cluster administrator.

When you need to complete this task

Complete this task only if you want to enable Cloud Pak for Data users to connect to storage volumes on an SMB file share server.

Note: The External SMB option is always visible on the New storage volume page. However, if you do not complete this task, users cannot connect to SMB storage volumes from the New storage volume page or through the Volumes API.

Procedure

To enable users to connect to a remote SMB storage volume:

1. Log on to Red Hat® OpenShift® Container Platform as a user with sufficient permissions to complete this task:

```
oc login OpenShift_URL:port
```

2. Install the [SMB CSI Driver for Kubernetes](#) (`csi-smb-driver`) on the cluster. For details, see [the readme](#) for the driver. The `csi-smb-driver` is provided by the Kubernetes CSI organization, which provides Kubernetes specific Constainer-Storage-Interface (CSI) components.

3. Run the following command to create the required cluster role and cluster role binding for the `meta-api`.

Tip: The same role binding is required to use external NFS storage volumes. If you already completed [Enabling users to create NFS storage volumes](#), this role binding already exists.

```
cat <<EOF | oc apply -f -
---
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
  # "namespace" omitted. ClusterRoles are not scoped to a namespace.
  name: ibm-zen-volumes-cluster-role
rules:
- apiGroups: [""]
  #
  # at the HTTP level, the name of the resource for accessing Secret
  # objects is "secrets"
  resources: ["persistentvolumes"]
  verbs: ["create", "get", "list", "patch", "update", "watch", "delete", "use"]
---
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRoleBinding
metadata:
  name: ibm-zen-volumes-cluster-role-binding
subjects:
- kind: ServiceAccount
```

```
name: ibm-zen-operator-serviceaccount
namespace: ibm-common-services # The namespace where the IBM Cloud Pak foundational services are installed
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: ClusterRole
  name: ibm-zen-volumes-cluster-role
EOF
```

Uninstalling the common core services

A project administrator can uninstall the common core services from IBM® Cloud Pak for Data.

When you need to complete this task

You should uninstall the common core services if you plan to uninstall Cloud Pak for Data.

Attention: Do not uninstall the common core services if:

- You are using other services that also have a dependency on the common core services. For details, see [Service software requirements](#).
- You plan to re-install a service that has a dependency on the common core services.

If you uninstall the common core services, the data that is associated with the common core services is deleted and cannot be recovered.

Contact IBM Software Support if you are unsure whether you should complete this task.

Procedure

The common core services are automatically installed by services that require them, such as Watson™ Studio and Watson Knowledge Catalog. However, when you uninstall these services, the common core services are *not* uninstalled. The common core services remain on the cluster to ensure that they are available to any services that use them.

Complete the following tasks to uninstall the common core services:

1. [Uninstalling dependent services](#)
2. [Uninstalling common core services](#)
3. [Uninstalling the operator](#)

Uninstalling dependent services

When you need to complete this task

Complete this task when you want to remove the common core services.

The following services install the common core services. Ensure that you uninstall these services before you uninstall the common core services.

- [Cognos® Analytics](#)
- [Cognos Dashboards](#)
- [Watson Knowledge Catalog](#)
- [Watson Machine Learning](#)
- [Watson Studio](#)

To identify all services that depend on the common core services, run the following command:

```
oc get operandrequest -n project-name | grep ccs
```

Uninstalling the common core services

To uninstall the common core services:

1. Delete the common core services custom resource in the OpenShift® project, where the common core services are installed. Use the following command:

```
oc delete ccs -n project-name ccs-cr
```

2. Confirm that the common core services were deleted by using the following command:

```
oc get ccs -n project-name
```

That command should return the following results:

```
No resources found in project-name namespace.
```

3. Locate the common core services operandrequest by using the following command:

```
oc get operandrequest -n project-name | grep ccs
```

4. Delete the common core services operandrequest by using the following command:

```
oc delete operandrequest -n project-name <ccs oprandrequest name>
```

Uninstalling the operator

When you need to complete this task

Complete this task only if you want to completely remove the common core services.

Complete this task *after* you uninstall the common core services.

Before you begin

Confirm that there are no common core services custom resources and common core services operand requests all namespaces in the cluster.

For each command, ensure that you specify the appropriate Red Hat® OpenShift project, either `ibm-common-services` or `cpd-operators` for the `--namespace` argument.

For the variable, use `operator-project`.

1. Delete the common core services subscription using the following command:

```
oc delete sub ibm-cpd-ccs-operator -n <ccs operator namespace>
```

2. Delete the common core services catalog source.

```
oc delete catsrc ibm-cpd-ccs-operator-catalog -n openshift-marketplace
```

Related reference

- [Pre-installation tasks](#)
- [Installing Cloud Pak for Data](#)
- [Post-installation tasks](#)
- [Uninstalling Cloud Pak for Data](#)

Uninstalling Cloud Pak for Data

A project administrator can uninstall the Cloud Pak for Data control plane.

Permissions you need for this task:

To complete this task, you must be an administrator of the OpenShift® project (Kubernetesnamespace) where:

- The Cloud Pak for Data control plane is installed
- The Cloud Pak for Data operators are installed, either `ibm-common-services` or `cpd-operators`

Procedure

1. [Uninstalling dependent services](#)
2. [Uninstalling Cloud Pak for Data](#)
3. [Uninstalling the operators](#)

Note: Complete this step only if you want to completely remove the Cloud Pak for Data software from your cluster.

Uninstalling dependent services

All services depend on the Cloud Pak for Data control plane.

Ensure that you uninstall any services that are installed in the same project (or tethered projects) as the control plane before you uninstall the control plane.

For details, see the appropriate service documentation in [Services](#).

Uninstalling Cloud Pak for Data

When you need to complete this task

Complete this task when you want to remove a running instance of the Cloud Pak for Data.

If you installed multiple instances of Cloud Pak for Data on the cluster, you must complete this task for each instance of Cloud Pak for Data that you want to uninstall.

If you plan to uninstall the Cloud Pak for Data operators, you must uninstall all instances of Cloud Pak for Data *before* you uninstall the operators.

To remove Cloud Pak for Data:

1. Log in to your Red Hat® OpenShift cluster as a user with sufficient permissions to complete the task:

```
oc login OpenShift_URL:port
```

2. Change to the project where the Cloud Pak for Data control plane is deployed:

```
oc project project_name
```

3. Get the name of the custom resource:

```
oc get Ibmcpd -n project_name
```

4. Delete the `Ibmcpd` custom resource to remove the Cloud Pak for Data platform:

```
oc delete Ibmcpd custom-resource-name -n project_name
```

5. Delete the `zenservice` custom resource to remove the control plane:

```
oc delete zenservice lite-cr -n project_name
```

6. Run the following command to verify that the resources that were created by Cloud Pak for Data and the control plane are deleted:


```
oc get all -l "app.kubernetes.io/name in (0020-zen-base, 0015-setup, 0010-infra)"
```

Uninstalling the operators

When you need to complete this task

Complete this task only if you want to completely remove Cloud Pak for Data from your cluster.

Complete this task *after* you uninstall the Cloud Pak for Data control plane.

To uninstall the Cloud Pak for Data operators:

1. Log in to your Red Hat OpenShift cluster as a user with sufficient permissions to complete the task:

```
oc login OpenShift_URL:port
```

2. Change to the project where Cloud Pak for Data was deployed:

```
oc project project_name
```

3. Remove the **zen** operator:

```
oc delete operandrequest zen-service
```

This command removes the **zen** operator from the instance project and from the project where the IBM Cloud Pak® for Data platform operator is installed (either **ibm-common-services** or **cpd-operators**).

4. Delete the IBM Cloud Pak for Data platform operator:

- a. Delete the subscription to the operator:

```
oc delete subscription cpd-operator -n ibm-common-services|cpd-operators
```

- b. Delete the CSV for the operator:

```
oc delete csv cpd-platform-operator.v2.0.0 -n ibm-common-services|cpd-operators
```

5. Verify that the IBM Cloud Pak for Data platform operator deployment was removed:

```
oc get deploy cpd-platform-operator-manager
```

The command should return the following message:

```
Error from server (NotFound): deployments.apps "cpd-platform-operator-manager" not found
```

Uninstalling IBM Cloud Pak foundational services

If you don't have other IBM® Cloud Paks on your cluster, you might want to uninstall IBM Cloud Pak foundational services after you uninstall Cloud Pak for Data. For details, see:

- For connected clusters, see [Uninstalling foundational services](#).
- For air-gapped clusters, see [Uninstalling IBM Cloud Pak foundational services in an airgap environment](#).

Related reference

- [Pre-installation tasks](#)
- [Installing Cloud Pak for Data](#)
- [Post-installation tasks](#)
- [Uninstalling the common core services](#)