KNIME Business Hub Update Instructions

KNIME AG, Zurich, Switzerland
Version 1.15 (last updated on)

Table of Contents

This guide contains the information needed to update KNIME Business Hub.

Three different types of updates can be performed:

1. Update the Business Hub instance - when a new version of KNIME Business Hub is released you might need to update to a newer version. To update your Business Hub instance follow the instructions here.

2. Update an Executor - when a new version of KNIME Analytics Platform is released you might need to update the executors to a newer version. To update executors follow the instructions here.

3. Update the Kubernetes version - when support is added by KNIME for a newer Kubernetes version you might need to update the underlying Kubernetes version. To update Kubernetes follow the instructions here.

KNIME Business Hub and executor updates happen independently of each other, meaning that updating Hub will not necessitate updating the executors and vice versa. Also, in the case in which your license allows you to have multiple Execution Contexts you can decide to have different executor versions for each Execution Context. This means that you might need to update only some of the executors and not necessarily all of them.

Update the KNIME Business Hub instance

To update the KNIME Business Hub instance go to your KOTS Admin Console.

If there is a new version available it will show up here. Click *Deploy* to initiate the update.

Depending on the services that are going to be updated, as well as the installation type, there might be some downtime. Consult the release notes to learn about possible downtime for each version update. To read more about downtime go to the Downtime section.

We recommend, before the update, doing a backup by using the built-in backup tool Velero.

> **i** Before proceeding with the update make sure to unlock the host packages if they are locked. Refer to OS documentation to learn how to unlock packages.

> **i** For version-specific update pathways refer to the Version-specific update pathways section.

Update airgapped instances

To update airgapped KNIME Business Hub installations, you will first need to download the newest version of the *knime-hub Airgap Bundle* and the *kURL installer* from the Download Portal of Replicated similar as during the airgapped installation process. In case you don't have a link to the Download Portal and/or the required password, please contact your KNIME customer care representative.

The downloaded *kURL installer* should have a size of approximately 4-5 GB. Once downloaded, transfer the *kURL installer* bundle onto the airgapped machine hosting the KNIME Business Hub, for example with a command similar to this:

```
scp -i <path-to-pem-file>/<filename>.pem <path-to-downloaded-bundle>/knime-hub.tar.gz
ubuntu@<host-ip>:/home/ubuntu
```

Once the *kURL installer* is downloaded and transferred to the airgapped machine, follow these steps:

1. Run the following commands before proceeding with updating:

```
kubectl delete pod --field-selector=status.phase==Failed --all-namespaces
kubectl delete pod --field-selector=status.phase==Succeeded --all-namespaces
```

2. Unzip the new installer and install it by running the following commands:

```
tar xvzf knime-hub.tar.gz
cat install.sh | sudo bash -s airgap installer-spec-file="./patch.yaml"
bash -l
```

3. Access the KOTS Admin Console. Under *Application* and *Dashboard*, select *Upload a new version* and upload the *knime-hub Airgap Bundle*.

Update an Execution Context Docker image on airgapped instances

To update an Execution Context Docker image on an airgapped machine, you will need to build the image on a machine with internet access and transfer it to the airgapped KNIME Business Hub instance as described in the KNIME Business Hub Admin guide or directly push it to the registry of your choice (embedded or private) using the `docker push` command. You can use the Execution Context Builder data application to generate a dockerfile or create it manually according to this template.

Downtime for Hub instance updates - FAQ

Is it necessary to schedule a maintenance window during which a Hub update is performed?

- For **single node** installations: yes
  - However, this depends on which services are updated (some updates will work without downtime). We will address this in every release note.
- For **multi node** installations:
  - If High Availability is enabled for all services, one instance of a service can be updated while anotherone remains operational, so no downtime is expected.
  - If High Availability is not enabled, a maintenance window should be scheduled, unless stated otherwise in the release notes.
- If a maintenance window is necessary, we recommend scheduling it for 2 hours.

Will my workflow deployments still run during a Hub update?

- For typical updates, yes. If not, this will be stated explicitly in the release notes.

- Only High Availability installations can ensure that an update does not affect deployments.

Ingress resources consolidation when updating to KNIME Business Hub version 1.13

With KNIME Business Hub version 1.13, Ingress resources are consolidated into a single namespace.

- *For customers using ingress resources managed by KNIME:*
  The release process will attempt to remove the ingresses from their prior namespaces and consolidate them to the `knime` namespace automatically.
  This will result in a few minutes of downtime for KNIME Business Hub while the ingresses are updated. It is recommended to perform the KNIME Business Hub upgrade during a maintenance window.

  > ℹ️ **Troubleshooting**: In some edge-case scenarios, ingresses will not be consolidated automatically and deployment errors may display in KOTS Admin Console such as `Error: UPGRADE FAILED: failed to create resource: admission webhook "validate.nginx.ingress.kubernetes.io" denied the request: host "registry.hub.example.com" and path "/" is already defined in ingress kurl/registry`. If these path conflicts occur, the conflicting ingress(es) can be removed manually and a redeployment can be triggered to resolve the issue.

- *For customers managing their own ingress resources:*
  It is recommended to consolidate ingresses into the `knime` namespace. Please see here for more examples in the `istio-externalservice.yaml` `kurl-externalservice.yaml` files.

Update the executors

To update executors you will need to update the execution context by pointing it to a new Docker image. The Docker image will contain an installation of KNIME Analytics Platform that is configured to be used as an executor, including all needed extensions. Switching to a new image is done separately for each execution context, by the team admin of the respective team.

> **i** Please note that with the release of 5.2.2 KNIME executors will have HTML sanitization of old JavaScript View nodes and Widget nodes turned on by default. This should ensure that no malicious HTML can be output. For more information check the KNIME Analytics Platform 5.2.2 changelog.

To do so go to the *Execution resources* overview page, click the three dots icon corresponding to the execution context you want to modify, and select Edit from the menu that opens. In the side panel that opens you can point the execution context to the new Docker image, as shown here.

Updates of executors are independent of any other parts of the installation so an update of one execution context will not affect any other execution contexts, or the Hub instance itself.

Testing executor updates

- For **Standard** and **Enterprise** editions, we recommend first creating a new execution context to test an update.
  - This execution context exists alongside the "old" version. I.e., this does not interfere with existing operations.
  - In case you have already used up all the licensed vCores, you can shift over some vCores from an existing execution context to be able to start a new one. This is done by editing the existing execution context.
  - After creating this new testing execution context, you can run all the needed workflows and make sure they still produce the expected results.
  - After testing is completed, the new execution context can be deleted, and vCores can be shifted back to the original execution context if necessary.
  - If testing was successful, you can proceed with the update of the original execution context.
- For the **Basic** edition, this is currently not possible since there is only a single execution context available.

- For **all the editions**, the recommended **rollback strategy** is to point the execution context to the previous Docker image again. There is no need to do a restore from a backup, as the executor containers are not expected to hold valuable data.

Downtime for executor updates

There are two basic strategies for updating execution contexts:

- Recreate

  - Executor containers of the old version are terminated when an update prompt is received. New containers are started only when old containers are gone.

    - This strategy involves downtime but is easy on cluster resources since no old and new containers exist in parallel.

    - Downtime depends on the time needed to pull the image from where it is hosted (the image is ~5GB in size) plus ~30s to start a new executor container.

- Rolling Update

  - Executor containers of the old version keep running until the pods of the new version are ready and terminate soon after.

    - This strategy presents no downtime but is heavy on resources as there is a short period where both old and new containers are running in parallel. Also, you will need to provide enough spare resources in the cluster.

Recreate is the default for all Hub installations since version 1.2.0.

However, it is possible to change the default executor update strategy from the Replicated portal. If you want zero downtime during executor updates and can provide the necessary resources, you can switch executor updates to Rolling Update.

Figure 1. Change the update strategy for executors in the Replicated portal

Update Kubernetes

Kubernetes is the infrastructure component inside which KNIME Business Hub runs. There are regular releases from Kubernetes, each with a fixed support lifecycle (see here). Each Business Hub version supports a range of Kubernetes versions such that Kubernetes updates can be performed independently. The Kubernetes versions for the according KNIME Business Hub versions are listed in the KNIME Business Hub Installation Guide.

A Kubernetes update is required only if you want to use a new feature that requires a newer Kubernetes version, or if your current Kubernetes version is not supported by KNIME Business Hub anymore.

The installer will check before each Hub update if the current Kubernetes version is supported during the preflight-check. In case it is not, the update won't start.

In case an update of Kubernetes needs to be performed you will need to run the Kubernetes installer script again.

- For **single node** installations: the update will result in downtime since the node itself is restarted. We recommend scheduling a maintenance window of 30 minutes.

- For **multi node** installations: the downtime, in this case, is reduced since one node at a time is drained and then updated. However, since there might still be moments in which the Kubernetes control plane is temporarily not reachable we still recommend scheduling a maintenance window, unless you are running an installation that has High Availability enabled for all services.

Update KOTS

KOTS is the application managing the Replicated installation itself. A fresh installation of a KNIME Business Hub instance will automatically use the latest KOTS version. However, when updating to a newer version of KNIME Business Hub it might happen, on some occasions, that an update of KOTS needs to be forced. This will be stated explicitly in the release notes together with instructions on how to update KOTS.

To update KOTS use this command:

```
curl -sSL https://kurl.sh/knime-hub | sudo bash
```

Version-specific update pathways

In the following section you can find a collection of version-specific pathways when updating from older versions of KNIME Business Hub.

Update from version before 1.9

Before updating from a version prior to 1.9 to a later version, if the you have a existing ingress deployed you will need to first change the namespace for the exisiting ingress from `hub` to `istio-system` and make sure all the annotations and ingress values are correct. You can find a working template of the required `Ingresses` in this `.yaml` file.

Migration of versioning and compatibility

ℹ️ This is important only if you are migrating from a KNIME Business Hub of version below 1.5.

With the 1.5 release of the KNIME Business Hub we introduced the new item level versioning, migrating from space level versioning that was made available in the previous releases.

This means that users can now create, restore and delete versions of individual items as described in the versioning section of the KNIME Business Hub User Guide.

The KNIME Business Hub version 1.5 is fully compatible with KNIME Analytics Platform version 5.1, which can create versions of items to use for ad hoc execution, create deployments and update components. When upgrading to version 1.5 all items in previously versioned spaces are automatically migrated to be versioned on the single item level. However we recommend to perform a backup of your KNIME Business Hub instance.

General compatibility between KNIME Hub, KNIME Analytics Platform client and executor

- KNIME Business Hub from version 1.5.0:
  - Analytics Platform version 5.1 client is fully compatible as previously stated
  - Executor version 5.1 is fully compatible
  - All the Analytics Platform and Executor versions < 5.1 are also compatible
- KNIME Business Hub version 1.4.2:
  - Executor version 5.1 is compatible
  - New nodes for versioning of Analytics Platform version 5.1 are not usable
- KNIME Business Hub version 1.4.1:
  - Executor version 5.1 is compatible
  - New nodes for versioning of Analytics Platform version 5.1 are not usable
  - Workflow uploads do not work via Analytics Platform version 5.1

Migration from space level versioning to item level versioning

- KNIME Business Hub 1.5.0 (with item level versioning):
  - Analytics Platform version 5.1:

Space Connector node will not have the version selection functionality.
The Space Version Creator node is replaced by the Version Creator node.

- ◦ Analytics Platform version < 4.7:
  Drag and drop of repository items will stop working

- ◦ Analytics Platform version < 5.1:
  Space Version Creator nodes will stop working.
  Space Connector nodes will stop working when the version checkbox is checked.
  They will work if the version checkbox is unchecked.

- ◦ Any Executor from version 4.7:
  Deployments migrated from Business Hub version 1.4.x are compatible although they might require some attention/adjustments for more complex cases.
  Deployments created with Business Hub version 1.5 are fully compatible.

- KNIME Business Hub 1.4.x (with space level versioning):

  - ◦ Analytics Platform version 5.1:
    Space connector nodes will not have the version selection functionality and will default to *latest*.
    Version Creator node will not work.

  - ◦ Executor version 5.1 is compatible for simple deployments although some more complex deployments might require some attention/adjustments.

Troubleshooting

In this section you can find common issues that can be encountered during the update process and the steps to take to solve the issues.

Helm charts are in an unhealthy state

Helm charts are in an unhealthy state after updating KNIME Business Hub instance, with the following error:

```
Error: UPGRADE FAILED: another operation (install/upgrade/rollback) is in progress
```

These errors need to be fixed before attempting to redeploy.

To do so you need to:

1. Have the helm command line installed

2. List all the helm releases and check which ones are healthy by using this command:

   ```
   helm list -A
   ```

   The helm releases that are healthy will have `STATUS = deployed`. If they are not in the deployed status KOTS will not be able to handle them.

3. Do a rollback on each of the unhealthy helm releases with this command:

   ```
   helm rollback -n <namespace> <release-name>
   ```

   This will roll the unhealthy helm releases to the previous version so that KOTS will be able to handle them.

   > ℹ The rollback might not bring the cluster back to a healty state, but the rollback will allow the upgrade process to proceed further when redeploying the application.

4. After the rollback is done, you need to **redeploy** the application.

Find more information about this on Replicated community.