

Switching from Database Flow Persistence Provider to Git Flow Persistence Provider using NiFi CLI Export/Import all from NiFi Registry

Date published: 2019-12-16

Date modified: 2023-06-27



Legal Notice

© Cloudera Inc. 2024. All rights reserved.

The documentation is and contains Cloudera proprietary information protected by copyright and other intellectual property rights. No license under copyright or any other intellectual property right is granted herein.

Unless otherwise noted, scripts and sample code are licensed under the Apache License, Version 2.0.

Copyright information for Cloudera software may be found within the documentation accompanying each component in a particular release.

Cloudera software includes software from various open source or other third party projects, and may be released under the Apache Software License 2.0 (“ASLv2”), the Affero General Public License version 3 (AGPLv3), or other license terms. Other software included may be released under the terms of alternative open source licenses. Please review the license and notice files accompanying the software for additional licensing information.

Please visit the Cloudera software product page for more information on Cloudera software. For more information on Cloudera support services, please visit either the Support or Sales page. Feel free to contact us directly to discuss your specific needs.

Cloudera reserves the right to change any products at any time, and without notice. Cloudera assumes no responsibility nor liability arising from the use of products, except as expressly agreed to in writing by Cloudera.

Cloudera, Cloudera Altus, HUE, Impala, Cloudera Impala, and other Cloudera marks are registered or unregistered trademarks in the United States and other countries. All other trademarks are the property of their respective owners.

Disclaimer: EXCEPT AS EXPRESSLY PROVIDED IN A WRITTEN AGREEMENT WITH CLOUDERA, CLOUDERA DOES NOT MAKE NOR GIVE ANY REPRESENTATION, WARRANTY, NOR COVENANT OF ANY KIND, WHETHER EXPRESS OR IMPLIED, IN CONNECTION WITH CLOUDERA TECHNOLOGY OR RELATED SUPPORT PROVIDED IN CONNECTION THEREWITH. CLOUDERA DOES NOT WARRANT THAT CLOUDERA PRODUCTS NOR SOFTWARE WILL OPERATE UNINTERRUPTED NOR THAT IT WILL BE FREE FROM DEFECTS NOR ERRORS, THAT IT WILL PROTECT YOUR DATA FROM LOSS, CORRUPTION NOR UNAVAILABILITY, NOR THAT IT WILL MEET ALL OF CUSTOMER’S BUSINESS REQUIREMENTS. WITHOUT LIMITING THE FOREGOING, AND TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, CLOUDERA EXPRESSLY DISCLAIMS ANY AND ALL IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, QUALITY, NON-INFRINGEMENT, TITLE, AND FITNESS FOR A PARTICULAR PURPOSE AND ANY REPRESENTATION, WARRANTY, OR COVENANT BASED ON COURSE OF DEALING OR USAGE IN TRADE.

Contents

Use case overview.....	4
Prerequisites.....	4
Switching flow persistence providers.....	5

Use case overview

Persistence providers in NiFi Registry store and manage metadata and versioned flow information. Each type of versioned item, such as a versioned flow or extension bundle, has its own persistence provider. Each persistence provider has its own configuration parameters, which can be configured in an XML file specified in `nifi-registry.properties`. You can use NiFi CLI to export and import flows in NiFi Registry when switching from one persistence provider to another, or for other use cases like syncing two different NiFi Registry instances.

Flow persistence providers store the content of the flows saved to the registry. In CDF for Data Hub, the Flow Management Data Hub cluster is configured to use the database provider as the flow persistence provider by default. `DatabaseFlowPersistenceProvider` uses a relational database for storing and managing the versioned flow information and metadata. `GitFlowPersistenceProvider` stores flow contents in a Git repository, leveraging the capabilities of Git, such as version control, collaboration, and auditing.

The `GitFlowPersistenceProvider` has the following limitations:

- You cannot make changes to your flows in Git and commit these flows for NiFi Registry to pick up. All commits need to be done by NiFi Registry following actions through NiFi or NiFi Registry.
- You cannot have multiple NiFi Registry instances sharing the same Git repository because Registry does not read from Git but only pushes for persistence.
- Git repositories can grow in size over time, especially if they contain large files or a long history of commits. This can lead to increased storage requirements and slower operations, so it is important to manage repository size and consider strategies like using Git LFS (Large File Storage) for large binary files.

In this use case, you can see how to switch from `DatabaseFlowPersistenceProvider` to `GitFlowPersistenceProvider` with the help of NiFi CLI. You can find details on how to:

- Export your flows from NiFi Registry using the CLI
- Clean up the databases
- Change the flow persistence provider
- Import all flows to your local Git repository
- Configure NiFi Registry so that versioned flows in NiFi are automatically saved to the Git repository.

Prerequisites

Use this checklist to make sure that you meet all the requirements before you switch flow persistence providers.

- You have a Flow Management Data Hub cluster in your CDP Public Cloud environment.
- You have the necessary credentials and permissions to perform administrative tasks for NiFi Registry in Cloudera Manager.
- You have the host name of your NiFi Registry instance.
- NiFi CLI interactive shell is launched.
- You have configured Git.
 - You cloned a remote Git repository.
 - You prepared the necessary credentials.



Important:

To switch providers successfully, the flow content directory and the metadata database must be fully empty, including NiFi Registry buckets.

Switching flow persistence providers

This tutorial walks you through how to switch from DatabaseFlowPersistenceProvider to GitFlowPersistenceProvider with the help of NiFi CLI.

Procedure

1. Connect to your NiFi Registry instance through NiFi CLI using `ssh`.
2. Create the target directory where you will save the flows that you are exporting from NiFi Registry.

For example: `sudo mkdir /tmp/flowbkp`



Note: Make sure you have write access to the target directory.

3. Export the flows stored in NiFi Registry and save them in the directory you have created using the following command:

```
sudo -u nifiregistry **[/path/to/cli.sh]** registry export-all-flows --outputDirectory "**[path/to/directory]**"
```

For example:

```
sudo -u nifiregistry /opt/cloudera/parcels/CFM/TOOLKIT/bin/cli.sh registry export-all-flows --outputDirectory "/tmp/flowbkp"
```

Where:

- `sudo -u nifiregistry` executes the subsequent command as the user `nifiregistry`.
 - `/opt/cloudera/parcels/CFM/TOOLKIT/bin/cli.sh` specifies the full path to the NiFi Toolkit CLI script.
 - `registry export-all-flows` instructs the CLI to export all flows and the associated resources from the NiFi Registry.
 - `--outputDirectory "/tmp/flowbkp"` specifies the output directory where the exported flows will be saved.
4. Stop NiFi Registry in Cloudera Manager.

This ensures a clean transition and prevents conflicts or data inconsistencies during switching persistence providers.

- a) Click the Clusters tab in the left-hand navigation.
- b) Select the NiFi Registry service in the list of services to access the service dashboard page.
- c) Stop the service by opening the Actions drop-down menu next to the service name and clicking Stop.



Note: It may take a few moments for the service to stop completely. You can verify that the NiFi Registry service is stopped by checking its status on the Cloudera Manager UI.

5. Back up the existing database.

It is crucial to create a backup of your existing NiFi Registry database before making any changes. This ensures that you have a copy of the data in case any issues occur during switching persistence providers.

- a) Click the Configuration tab of the NiFi service dashboard page in Cloudera Manager.
- b) Locate the configuration property named `nifi.registry.flow.storage.directory`.

This property specifies the directory where the Flow Storage files are stored for the `FileSystemFlowPersistenceProvider`. It is only required if `Enable Flow File Provider` is enabled. Otherwise, both flow content and metadata are stored in the database.

- c) Create a directory as a backup location, where you want to store the Flow Storage files.
- d) Update the flow storage directory to the backup location you created using the `Providers: Default Flow Persistence File Provider Property - Flow Storage Directory` property.

You can search for 'providers' to find this property more easily. Make sure to specify the full absolute path to the new location.

- e) Use file system commands or tools to move the contents of the original Flow Storage directory to the new backup directory following your usual backup procedure.

This involves copying all the files and directories from the original location to the new location while preserving the directory structure.

- f) Delete all data from the original database location.

The NiFi Registry should be completely empty including the buckets.

6. Configure the NiFi Registry flow persistence provider properties in Cloudera Manager.

- a) Click the Configuration tab of the NiFi service dashboard page in Cloudera Manager.

This tab displays the configuration settings for NiFi Registry.

- b) Disable the database provider by unclicking the `Providers: Enable Database Flow Persistence Provider` property.

If `Flow File Provider` was enabled, it needs to be disabled as well using `Enable Flow File Provider`.

- c) Enable the Git-based provider by selecting the `Providers: Enable the git-based flow persistence provider` property.
- d) Configure the `GitFlowPersistenceProvider` provider.

1. Set the path to your cloned Git repository using the `Providers: File system path for a directory where flow contents files are persisted to` property.

This is the file system path for the directory where flow contents files are persisted to. The directory must exist when NiFi Registry starts, and it must be initialized as a Git directory.

2. Define where you want to push the flow changes using the `Providers: Name of the remote repository to push to` property.

When a new flow snapshot is created, files are updated in the specified Git directory. The `GitFlowPersistenceProvider` then creates a commit to the local repository. If `Remote to push` is defined, it also pushes to the specified remote repository.

3. Provide a Git user name using the `Providers: Git username for the remote repository` property.

This username should be used by the NiFi Registry to interact with the repository. It is used to make push requests to the remote repository when `Remote to push` is enabled, and the remote repository is accessed by HTTP protocol. If `SSH` is used, user authentication is done with SSH keys.

4. Provide a Git password using the `Providers: The password for the Remote Access User` property.

- e) Save the configuration changes.

7. Start NiFi Registry to apply the new configuration.

- a) From Cloudera Manager, click the Clusters tab in the left-hand navigation.
- b) Select NiFi Registry.
- c) Start the service by opening the Actions drop-down menu next to the service name and clicking Start.

8. Ensure the user has full access to the Git repository and that the Git working tree is clean.

9. Import all flows from the directory where you saved your exported flows using the following command:

```
sudo -u nifiregistry **[/path/to/cli.sh]** registry import-all-flows --input
 **[/path/to/directory]** --skipExisting
```

For example:

```
sudo -u nifiregistry /opt/cloudera/parcels/CFM/TOOLKIT/bin/cli.sh registry
 import-all-flows --input "/tmp/flowbcp" --skipExisting
```

Where:

- `sudo -u nifiregistry` executes the subsequent command as the user `nifiregistry`.
- `/opt/cloudera/parcels/CFM/TOOLKIT/bin/cli.sh` specifies the full path to the NiFi Toolkit CLI script.
- `registry import-all-flows` instructs the CLI to import all flows and the associated resources to the NiFi Registry.
- `--input "/tmp/flowbcp"` specifies the input directory from which the flows will be imported.
- `--skipExisting` indicates that any existing flows with the same UUIDs in the Registry should be skipped during the import process. It helps to avoid overwriting existing flows.



Note: Depending on your settings, `git push` may be required to finalize the change.

10. Verify and test:

Ensure that NiFi Registry is functioning as expected with the new flow persistence provider. Test the functionality, including creating and versioning flows, accessing metadata, and verifying data integrity.

11. Monitor and troubleshoot:

Monitor the NiFi Registry instance closely after the migration to ensure there are no unexpected issues. If any problems arise, check the documentation and troubleshooting resources provided by NiFi Registry and the chosen flow persistence provider.