

Apache NiFi 3

Versioning a Data Flow

Date of Publish: 2020-04-28



<https://docs.cloudera.com/>

Contents

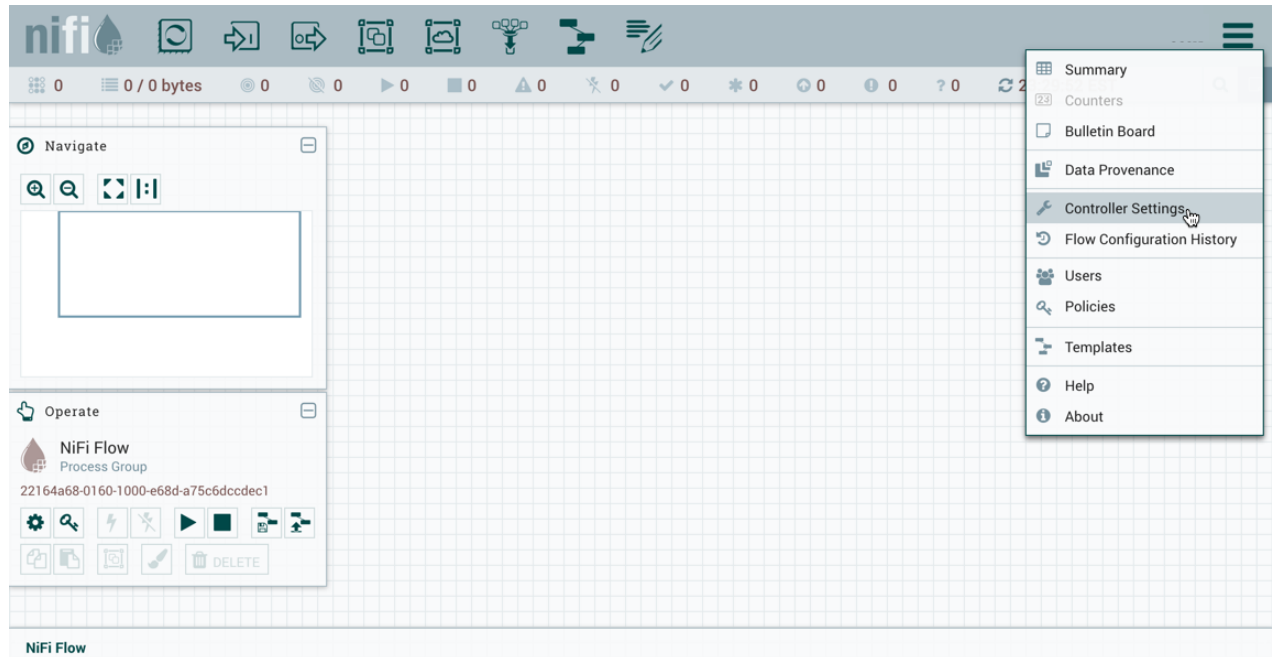
Versioning a DataFlow.....	3
Connecting to a NiFi Registry.....	3
Version States.....	6
Import a Versioned Flow.....	8
Start Version Control.....	10
Managing Local Changes.....	13
Show Local Changes.....	15
Revert Local Changes.....	16
Commit Local Changes.....	17
Change Version.....	18
Stop Version Control.....	21
Nested Versioned Flows.....	23
Parameters in Versioned Flows.....	23
Variables in Versioned Flows.....	25
Restricted Components in Versioned Flows.....	33
Restricted Controller Service Created in Root Process Group.....	35
Restricted Controller Service Created in Process Group.....	41

Versioning a DataFlow

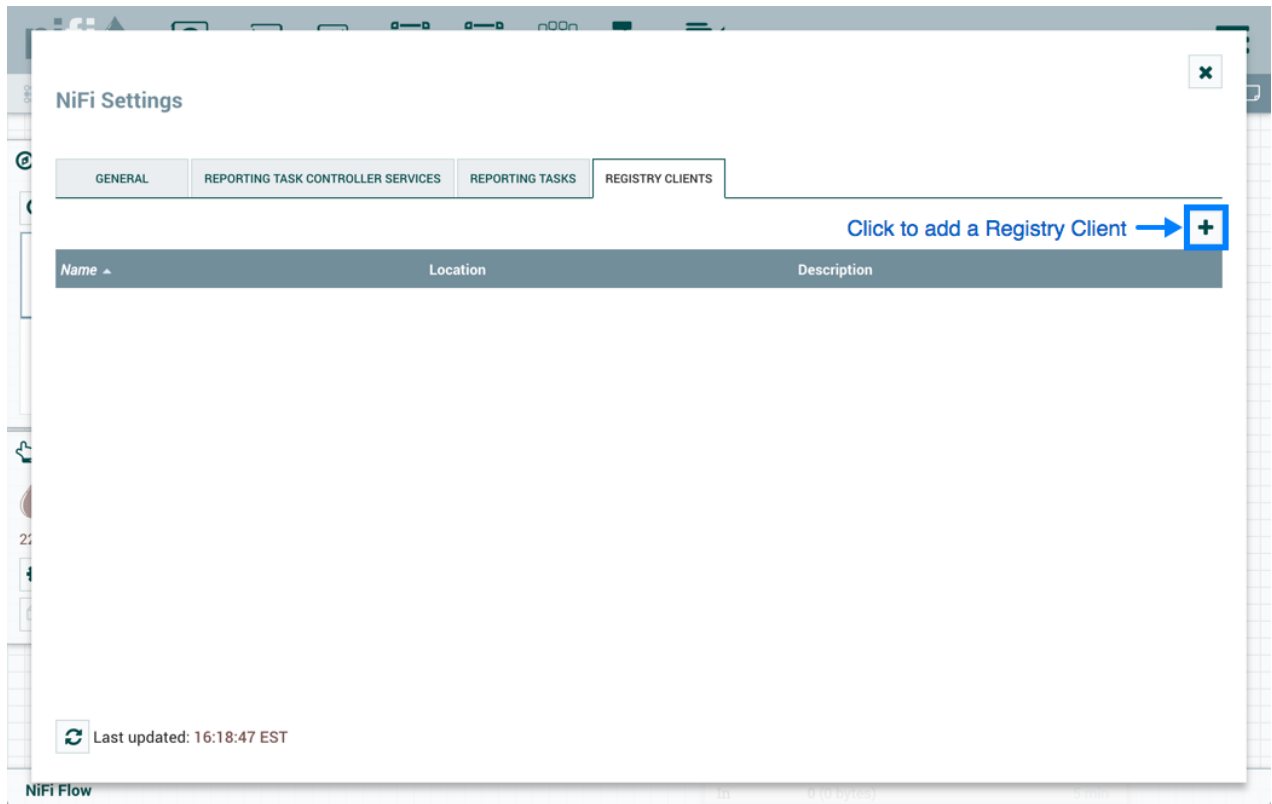
When NiFi is connected to a NiFi Registry, dataflows can be version controlled on the process group level. For more information about NiFi Registry usage and configuration, see the *NiFi Registry documentation*.

Connecting to a NiFi Registry

To connect NiFi to a Registry, select Controller Settings from the Global Menu.



This displays the NiFi Settings window. Select the Registry Clients tab and click the + button in the upper-right corner to register a new Registry client.



In the Add Registry Client window, provide a name and URL.

Add Registry Client

Name

Local Registry

URL

http://localhost:18080

Description

CANCEL **ADD**

Click "Add" to complete the registration.

NiFi Settings

GENERAL | REPORTING TASK CONTROLLER SERVICES | REPORTING TASKS | **REGISTRY CLIENTS**

Name	Location	Description
Local Registry	http://localhost:18080	


Last updated: 16:18:47 EST





Note: Versioned flows are stored and organized in registry buckets. Bucket Policies and Special Privileges configured by the registry administrator determine which buckets a user can import versioned flows from and which buckets a user can save versioned flows to. Information on Bucket Policies and Special Privileges can be found in the *NiFi Registry User Guide*.


Version States


Versioned process groups exist in the following states:

- 

Up to date: The flow version is the latest.
- 

Locally modified: Local changes have been made.
- 

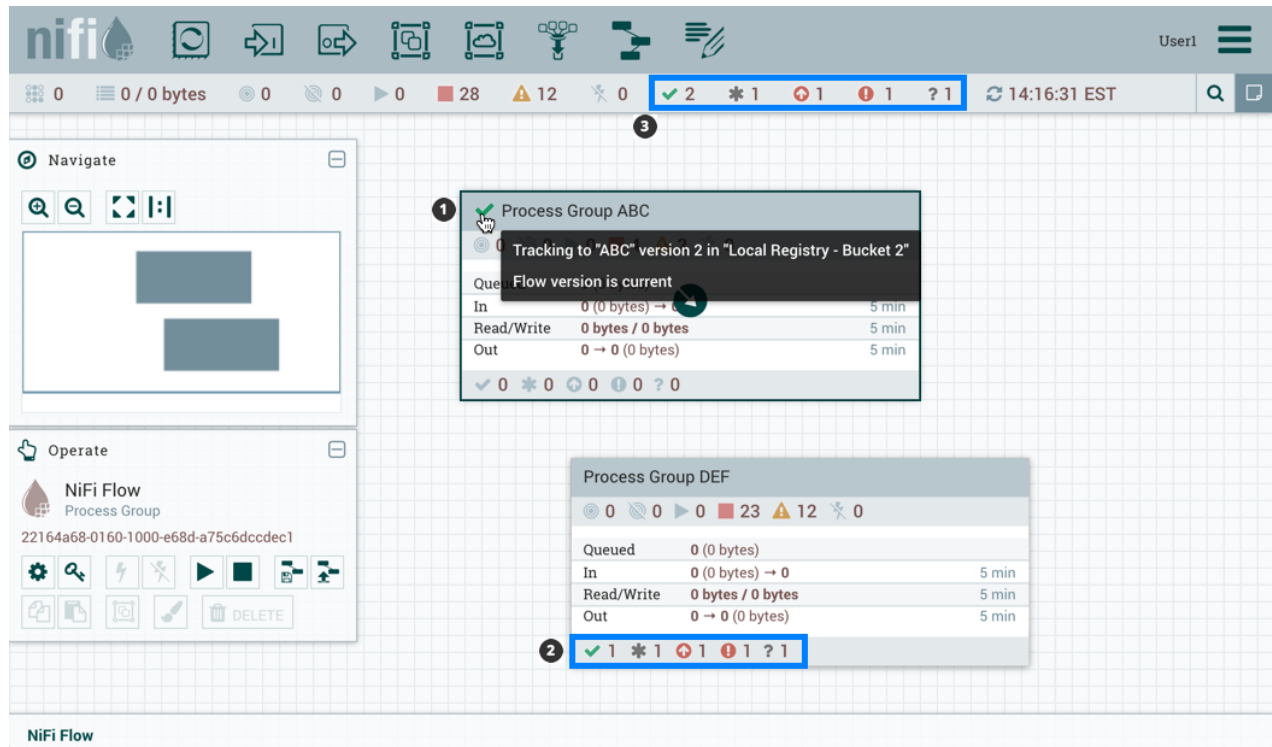
Stale: A newer version of the flow is available.
- 

Locally modified and stale: Local changes have been made and a newer version of the flow is available.
- 

Sync failure: Unable to synchronize the flow with the registry.

Version state information is displayed:

1. Next to the process group name, for the versioned process group itself. Hovering over the state icon displays additional information about the versioned flow.
2. At the bottom of a process group, for the versioned flows contained in the process group.
3. In the Status Bar at the top of the UI, for the versioned flows contained in the root process group.



Version state information is also shown in the "Process Groups" tab of the Summary Page.

NiFi Summary

PROCESSORS | INPUT PORTS | OUTPUT PORTS | REMOTE PROCESS GROUPS | CONNECTIONS | PROCESS GROUPS

Displaying 7 of 7

Filter by name

Name	Version State	Transferr...	In / Size 5...	Read / Write 5 mi...	Out / Size ...	Sent / Size...	Received / ...
NiFi Flow		0 (0 bytes)	0 (0 bytes)	0 bytes / 0 bytes	0 (0 bytes)	0 (0 bytes)	0 (0 bytes)
Process Group DEF		0 (0 bytes)	0 (0 bytes)	0 bytes / 0 bytes	0 (0 bytes)	0 (0 bytes)	0 (0 bytes)
Flow 3	* Locally modified	0 (0 bytes)	0 (0 bytes)	0 bytes / 0 bytes	0 (0 bytes)	0 (0 bytes)	0 (0 bytes)
test	⚠ Locally modified and stale	0 (0 bytes)	0 (0 bytes)	0 bytes / 0 bytes	0 (0 bytes)	0 (0 bytes)	0 (0 bytes)
MySQL CDC	⚠ Stale	0 (0 bytes)	0 (0 bytes)	0 bytes / 0 bytes	0 (0 bytes)	0 (0 bytes)	0 (0 bytes)
PG 13	? Sync failure	0 (0 bytes)	0 (0 bytes)	0 bytes / 0 bytes	0 (0 bytes)	0 (0 bytes)	0 (0 bytes)
Process Group ABC	✓ Up to date	0 (0 bytes)	0 (0 bytes)	0 bytes / 0 bytes	0 (0 bytes)	0 (0 bytes)	0 (0 bytes)

Last updated: 15:15:50 EST system diagnostics



Note: To see the most recent version states, it may be necessary to right-click on the NiFi canvas and select 'Refresh' from the context menu.

Import a Versioned Flow

When a NiFi instance is connected to a registry, an "Import" link will appear in the Add Process Group dialog.

Add Process Group

Process Group Name

[Import...](#)

Import a flow from a registry

CANCEL ADD

Selecting the link will open the Import Version dialog.

Import Version

Registry
Local Registry

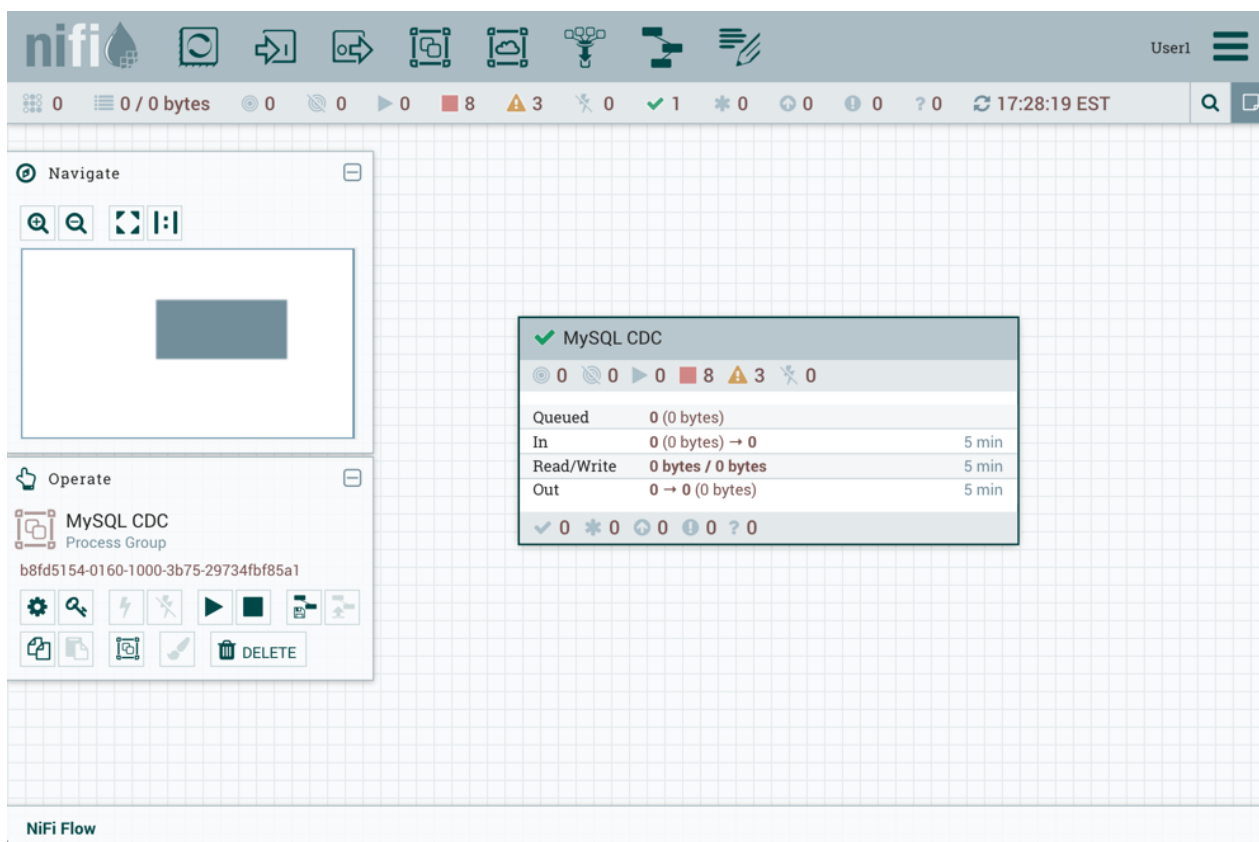
Bucket
Bucket 1

Name
MySQL CDC

Version	Created	Comments
3	01/02/2018 17:18:44.686	Changed Run Schedule of CDC processor to 300 ms.
2	01/02/2018 17:17:38.128	Corrected LogAttribute Processor by auto-terminating.
1	01/02/2018 17:12:17.339	Start from template

CANCEL IMPORT

Connected registries will appear as options in the Registry drop-down menu. For the chosen Registry, buckets the user has access to will appear as options in the Bucket drop-down menu. The names of the flows in the chosen bucket will appear as options in the Name drop-down menu. Select the desired version of the flow to import and select "Import" for the dataflow to be placed on the canvas.



Since the version imported in this example is the latest version (MySQL CDC, Version 3), the state of the versioned process group is "Up to date"



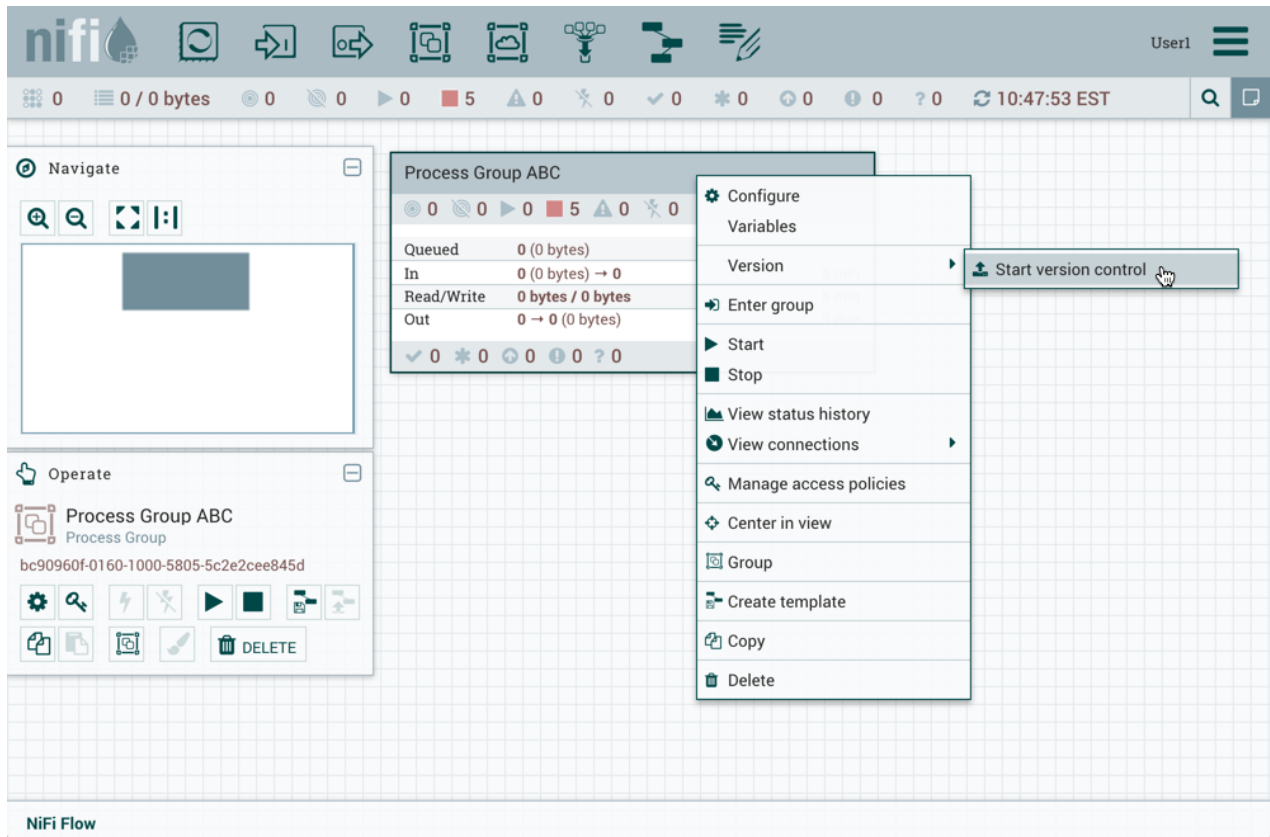
If the version imported had been an older version, the state would be



"Stale"

Start Version Control

To place a process group under version control, right-click on the process group and in the context menu, select "Version#Start version control".



In the Save Flow Version window, select a Registry and Bucket and enter a Name for the Flow. If desired, add content for the Description and Comment fields.

Save Flow Version

Registry
Local Registry

Bucket
Bucket 2

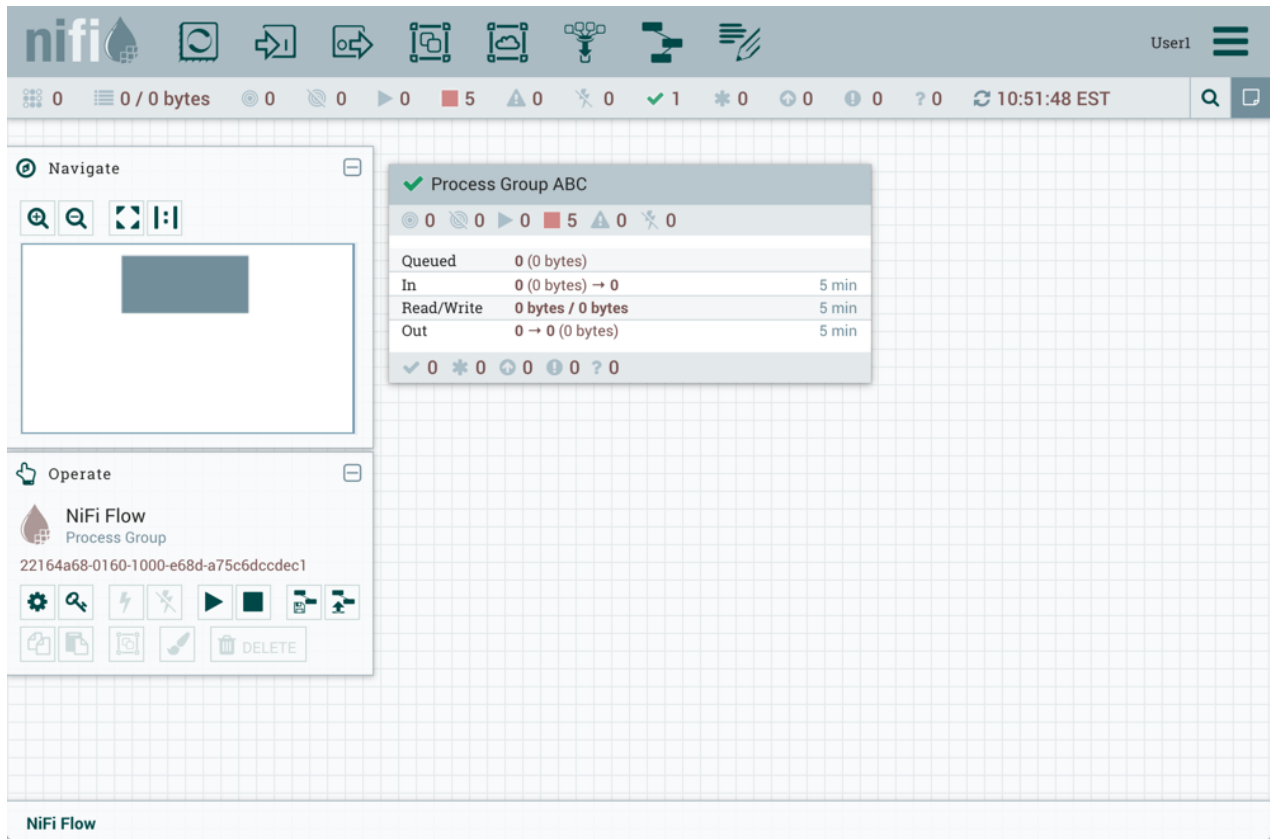
Name
ABC


Description
Get movie CSV data, convert to JSON and publish to Kafka topic "Movies"

Comments
Initial Save

CANCEL SAVE

Select Save and Version 1 of the flow is saved.




As the first and latest version of the flow, the state of the versioned process group is "Up to date" ().

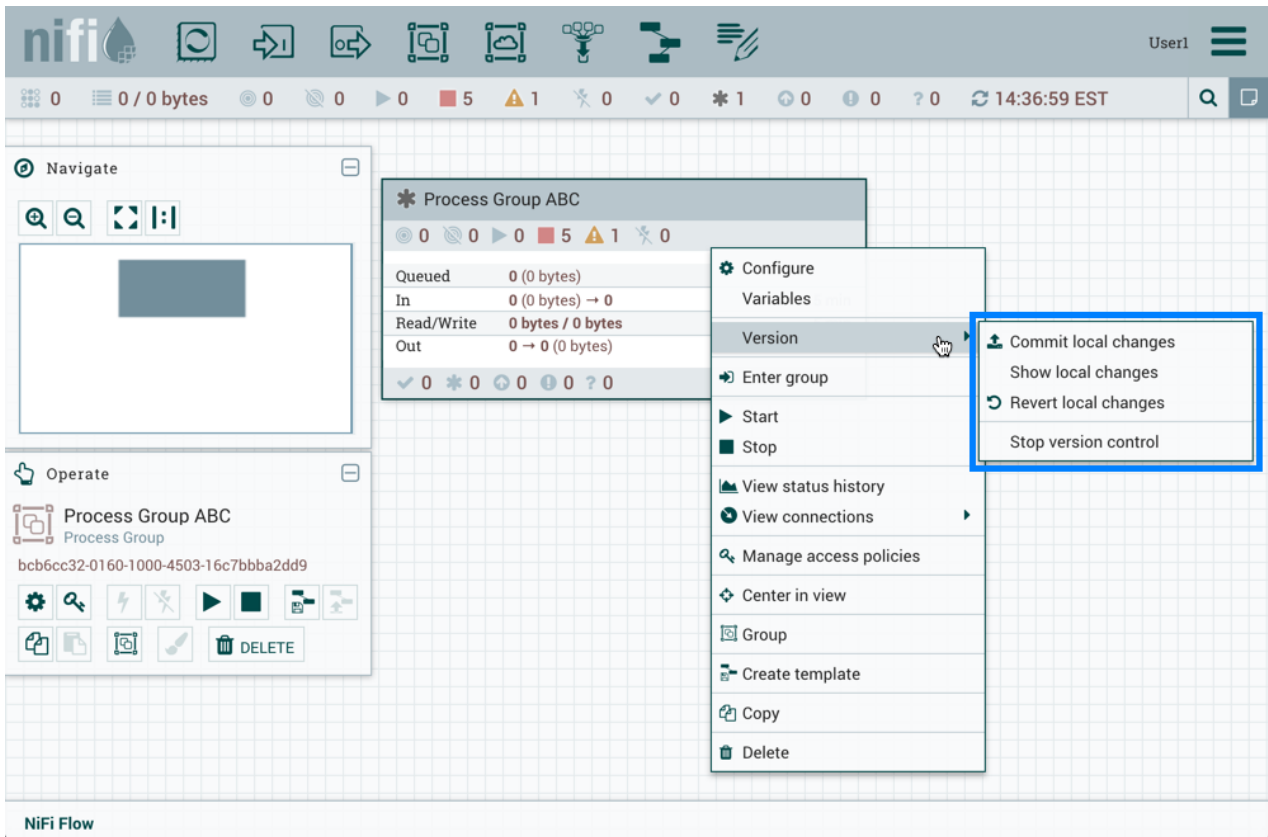


Note: The root process group can not be placed under version control.

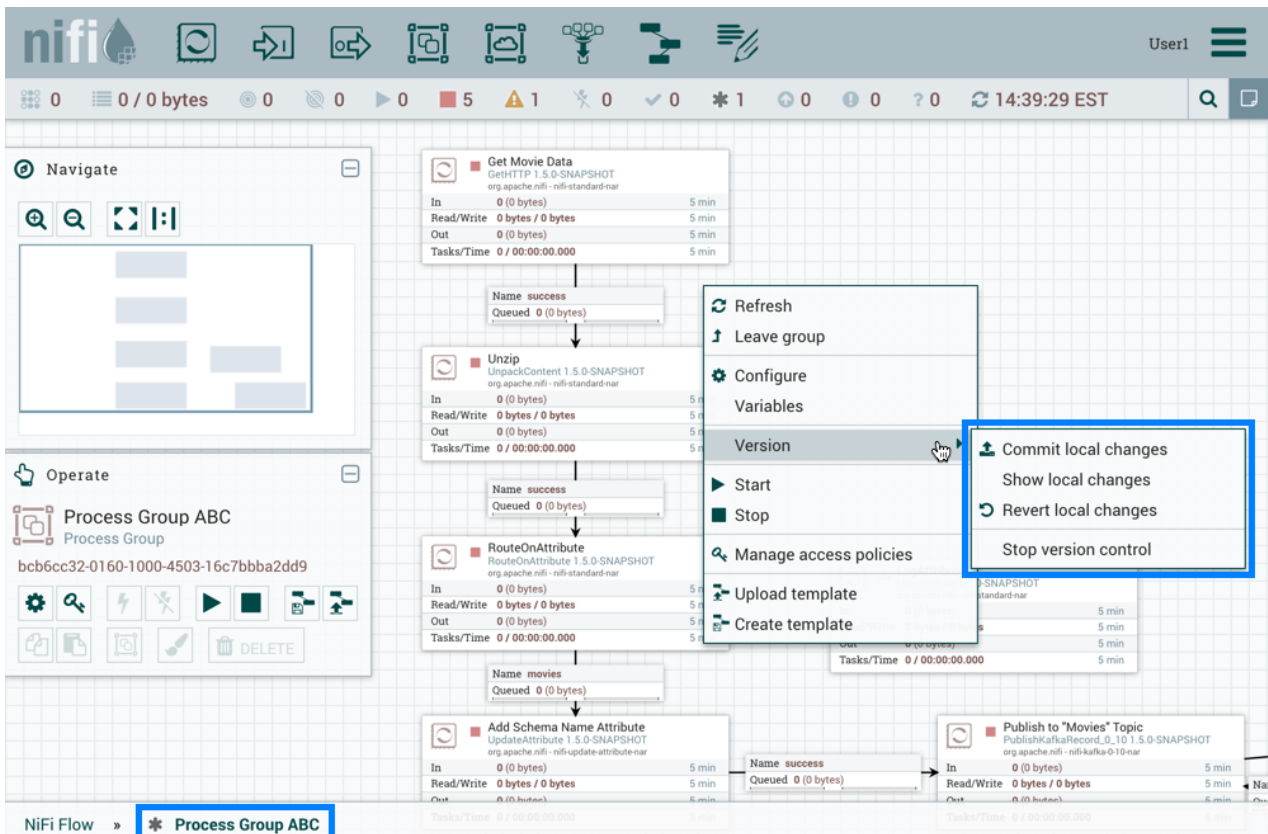
Managing Local Changes

When changes are made to a versioned process group, the state of the component updates to "Locally modified" ().

The DFM can show, revert or commit the local changes. These options are available for selection in the context menu when right-clicking on the process group:



or when right-clicking on the canvas inside the process group:



The following actions are not considered local changes:

- disabling/enabling processors and controller services
- stopping/starting processors
- modifying sensitive property values
- modifying remote process group URLs
- updating a processor that was referencing a non-existent controller service to reference an externally available controller service
- assigning, creating, modifying or deleting parameter contexts
- creating, modifying or deleting variables



Note: Assigning or creating a parameter context does not trigger a local change because assigning or creating a parameter context on its own has not changed anything about what the flow processes. A component will have to be created or modified that uses a parameter in the parameter context, which will trigger a local change. Modifying a parameter context does not trigger a local change because parameters are intended to be different in each environment. When a versioned flow is imported, it is assumed there is a one-time operation required to set those parameters specific for the given environment. Deleting a parameter context does not trigger a local change because any components that reference parameters in that parameter context will need need to be modified, which will trigger a local change.



Note: Creating a variable does not trigger a local change because creating a variable on its own has not changed anything about what the flow processes. A component will have to be created or modified that uses the new variable, which will trigger a local change. Modifying a variable does not trigger a local change because variable values are intended to be different in each environment. When a versioned flow is imported, it is assumed there is a one-time operation required to set those variables specific for the given environment. Deleting a variable does not trigger a local change because the component that references that variable will need need to be modified, which will trigger a local change.



Note: Variables do not support sensitive values and will be included when versioning a Process Group. Variables are still supported for compatibility purposes but do not have the same power as Parameters such as support for sensitive properties and more granular control over who can create, modify or use them. Variables will be removed in a future release. As a result, it is highly recommended to switch to Parameters.

Show Local Changes

The local changes made to a versioned process group can be viewed in the Show Local Changes dialog by selecting "Version#Show local changes" from the context menu.

Show Local Changes

The following changes have been made to ABC (Version 1).

Displaying 4 of 4

Component Name ▲	Change Type	Difference	
Get Movie Data	Run Schedule Changed	From '10 mins' to '5 mins'	→
LogAttribute	Component Added	Processor was added	→
Publish to "Movies" Topic	Property Value Changed	From 'Compression Type=none' to 'Compression Type=gzip'	→
RouteOnAttribute	Bulletin Level Changed	From 'WARN' to 'ERROR'	→

CLOSE

You can navigate to a component by selecting the "Go To" icon

()
 in its row.



Note: As described in the Managing Local Changes section, there are exceptions to which actions are reviewable local changes. Additionally, multiple changes to the same property will only appear as one change in the list as the changes are determined by diffing the current state of the process group and the saved version of the process group noted in the Show Local Changes dialog.

Revert Local Changes

Revert the local changes made to a versioned process group by selecting "Version#Revert local changes" from the context menu. The Revert Local Changes dialog displays a list of the local changes for the DFM to review and consider prior to initiating the revert. Select "Revert" to remove all changes.

Revert Local Changes

The following changes have been made to ABC (Version 1). **Revert will remove all changes.**

Displaying 4 of 4

Filter

Component Name ▲	Change Type	Difference	
Get Movie Data	Run Schedule Changed	From '10 mins' to '5 mins'	→
LogAttribute	Component Added	Processor was added	→
Publish to "Movies" Topic	Property Value Changed	From 'Compression Type=none' to 'Compression Type=gzip'	→
RouteOnAttribute	Bulletin Level Changed	From 'WARN' to 'ERROR'	→

CANCEL
REVERT

You can navigate to a component by selecting the "Go To" icon

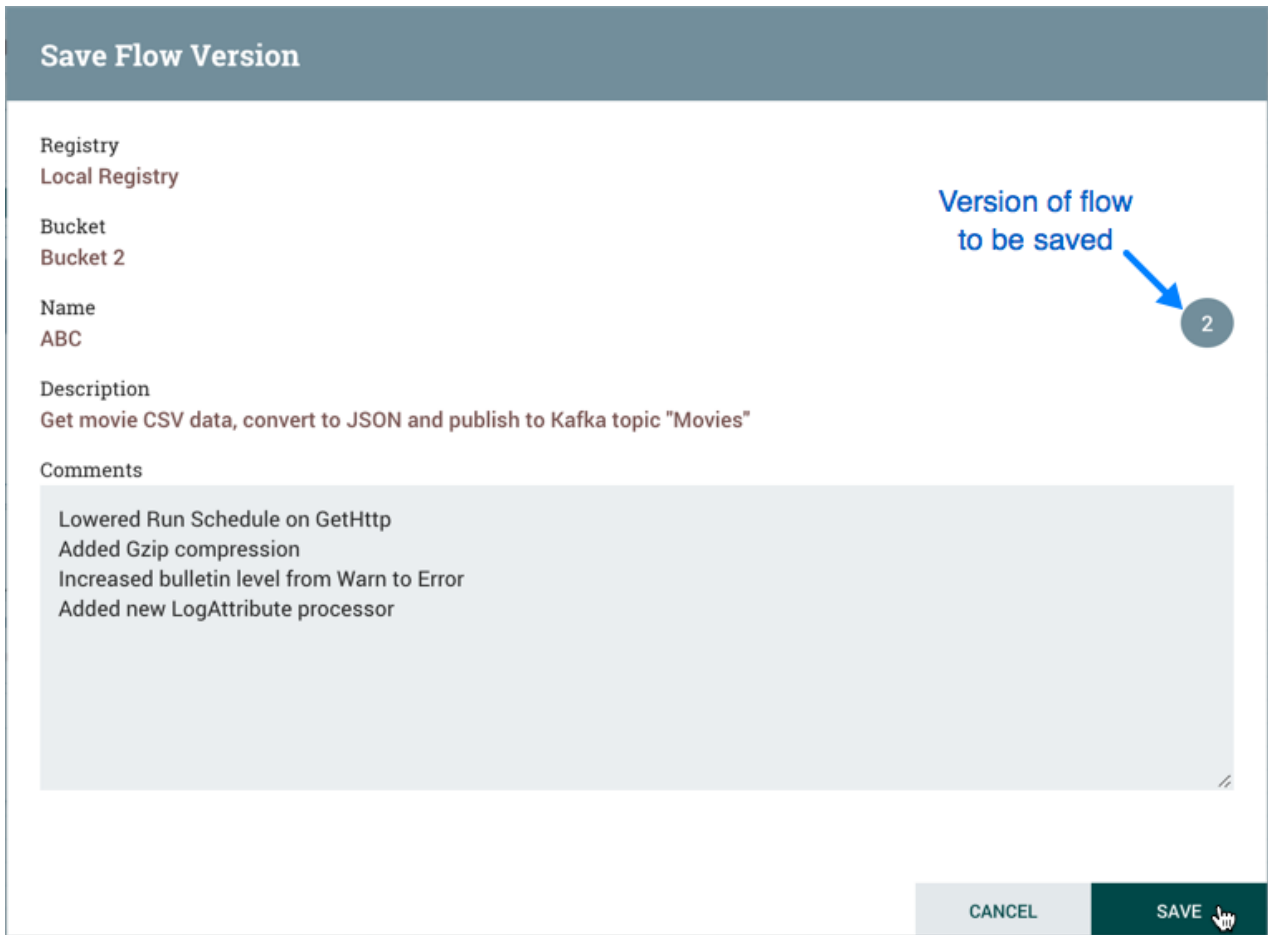
()
 in its row.



Note: As described in the Managing Local Changes section, there are exceptions to which actions are revertible local changes. Additionally, multiple changes to the same property will only appear as one change in the list as the changes are determined by diffing the current state of the process group and the saved version of the process group noted in the Revert Local Changes dialog.

Commit Local Changes

To commit and save a flow version, select "Version#Commit local changes" from the context menu. In the Save Flow Version dialog, add comments if desired and select "Save".

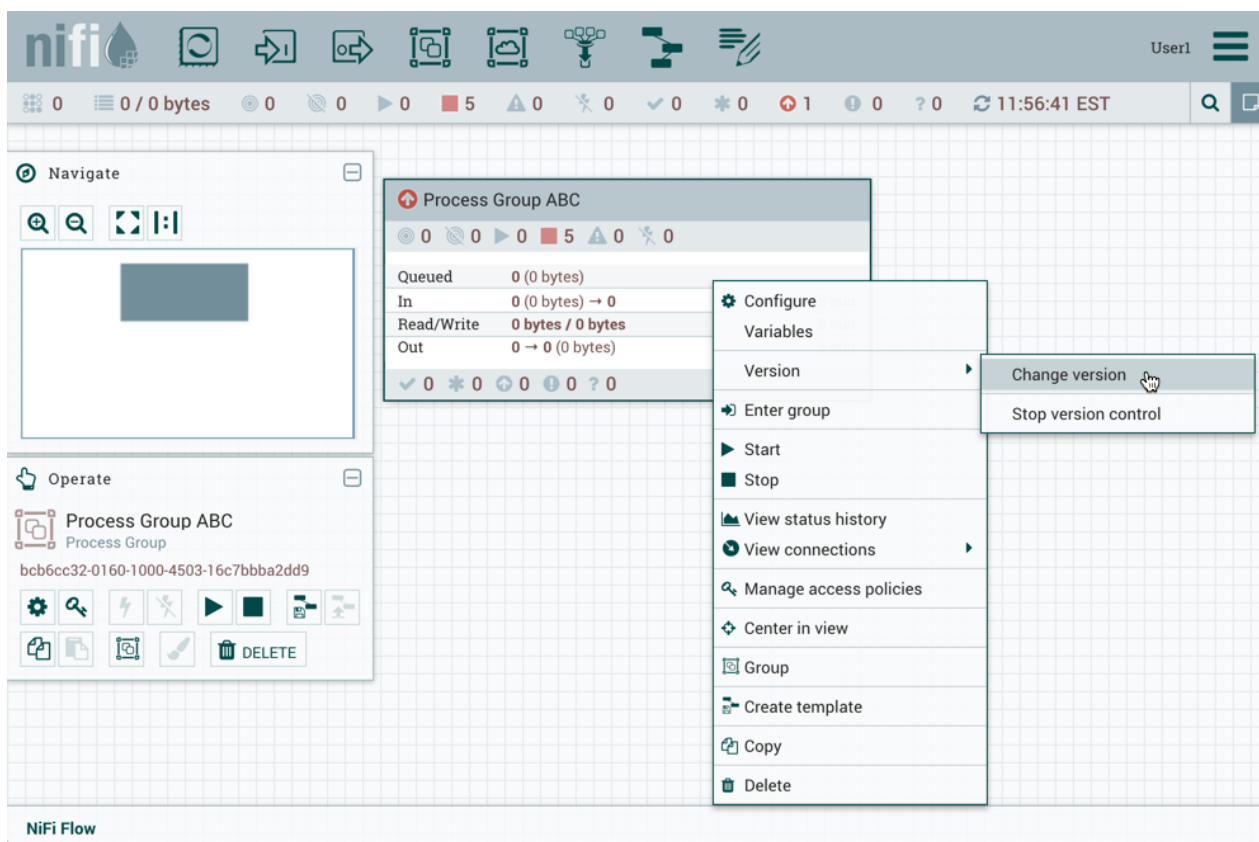


Local changes can not be committed if the version that has been modified is not the latest version. In this scenario, the version state is "Locally modified and

stale" ().

Change Version

To change the version of a flow, right-click on the versioned process group and select "Version#Change version".



In the Change Version dialog, select the desired version and select "Change":

Change Version

Registry
Local Registry

Bucket
Bucket 2

Name
ABC

Current Version
1

Version	Created	Comments
2	01/04/2018 11:23:10.564	Lowered Run Schedule on GetHttp Added Gzip compression Increased bulletin leve...
1	01/03/2018 10:39:36.520	Initial Save

CANCEL CHANGE

The version of the flow is changed:

The screenshot shows the Apache NiFi web interface. At the top, there is a navigation bar with the NiFi logo and various icons. Below the navigation bar is a status bar showing metrics like '0 / 0 bytes', '0', '0', '0', '5', '1', '0', '1', '0', '0', '0', '0', '0', '0', and the time '14:36:41 EST'. The main workspace is a grid with a 'Process Group ABC' tooltip. The tooltip contains the following information:

- Process Group ABC
- Tracking to "ABC" version 2 in "Local Registry - Bucket 2"
- Queue: Flow version is current
- In: 0 (0 bytes) → 5 min
- Read/Write: 0 bytes / 0 bytes 5 min
- Out: 0 → 0 (0 bytes) 5 min

At the bottom of the tooltip, there are status icons: a checkmark, a star, a refresh icon, a warning icon, and a question mark.

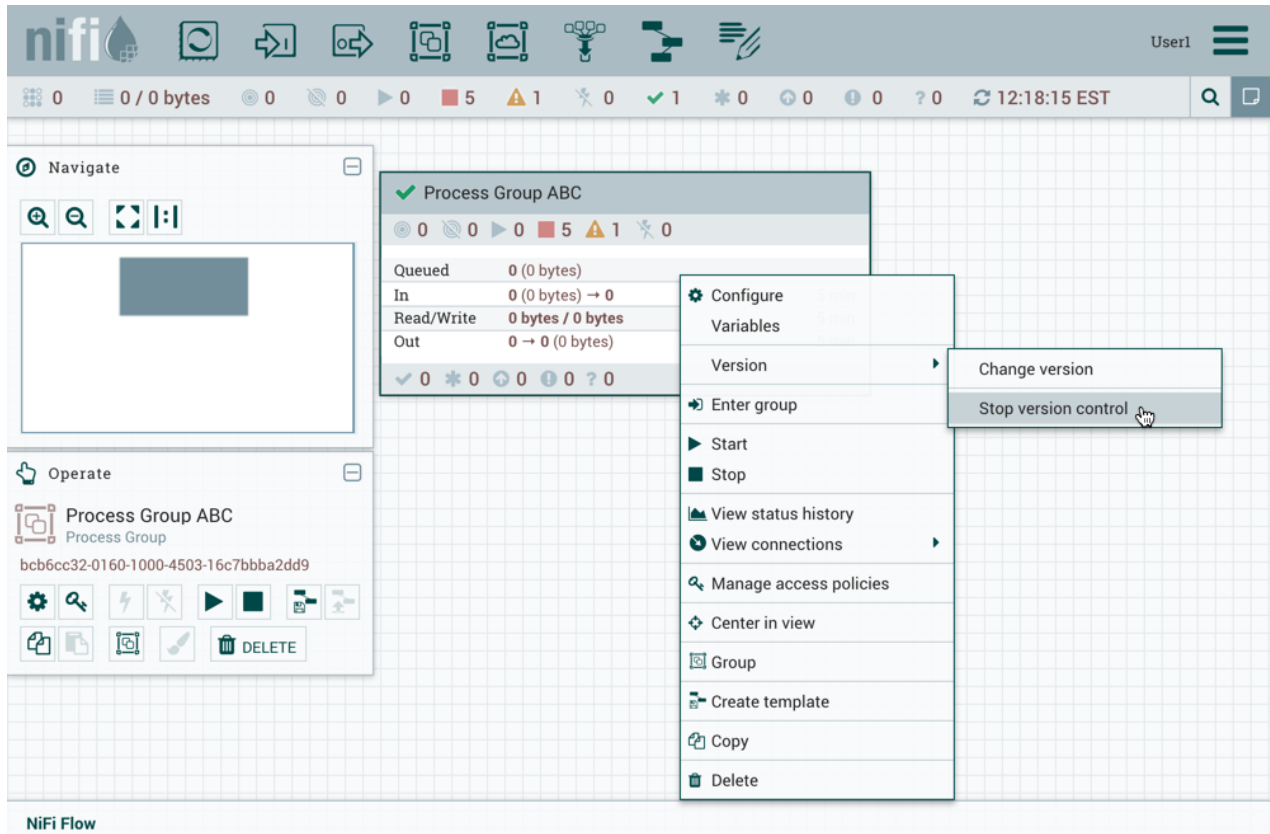
In the example shown, the versioned flow is upgraded from an older to the newer latest version. However, a versioned flow can also be rolled back to an older version.



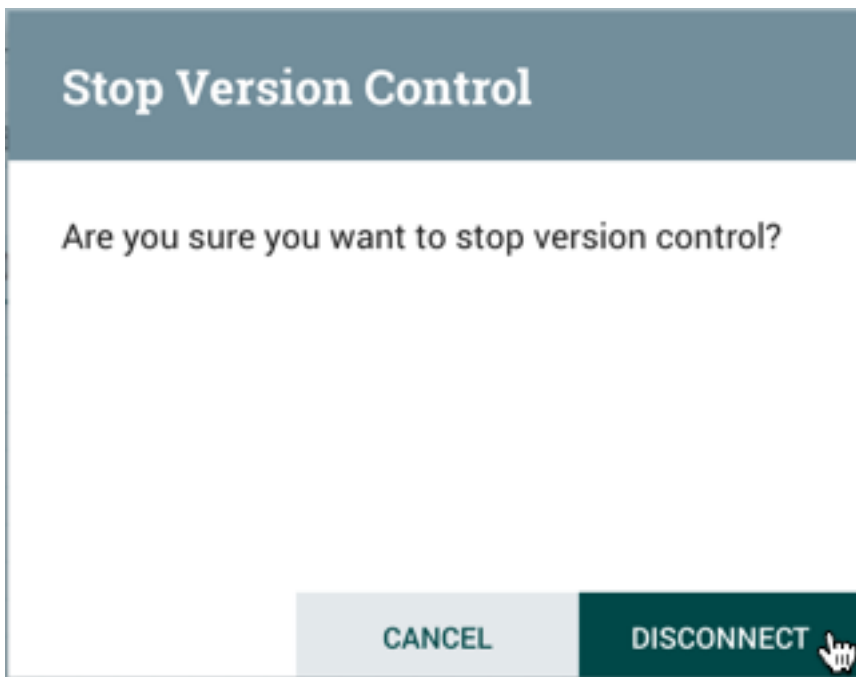
Note: For "Change version" to be an available selection, local changes to the process group need to be reverted.

Stop Version Control

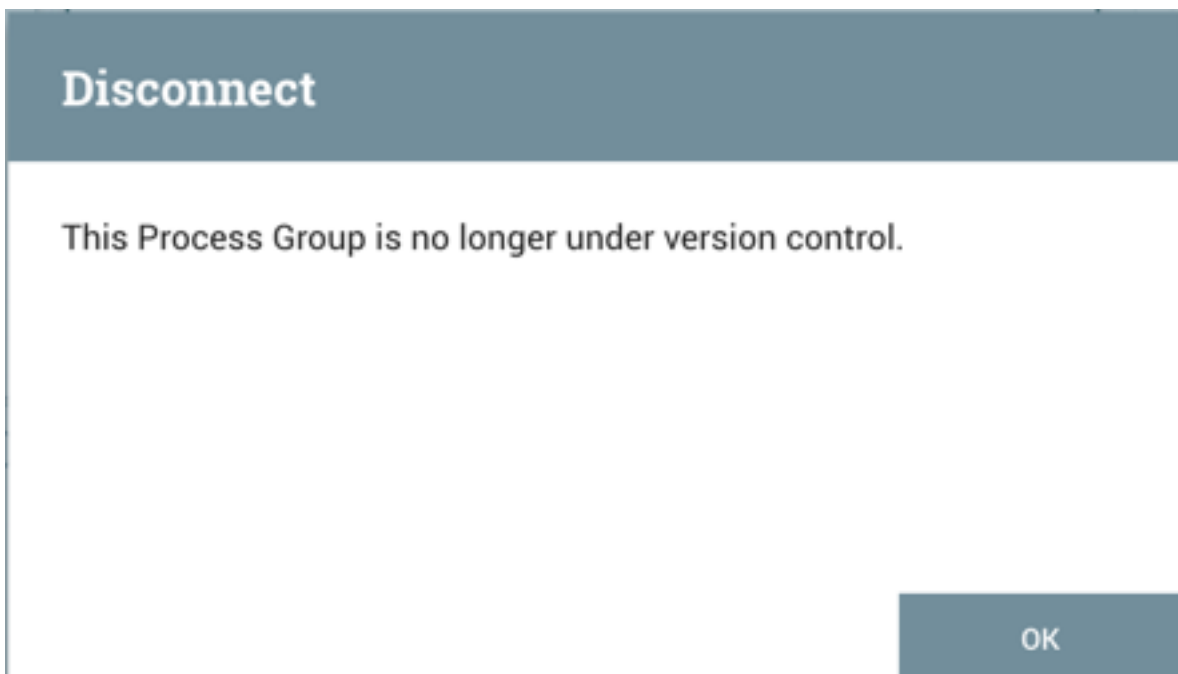
To stop version control on a flow, right-click on the versioned process group and select "Version#Stop version control":

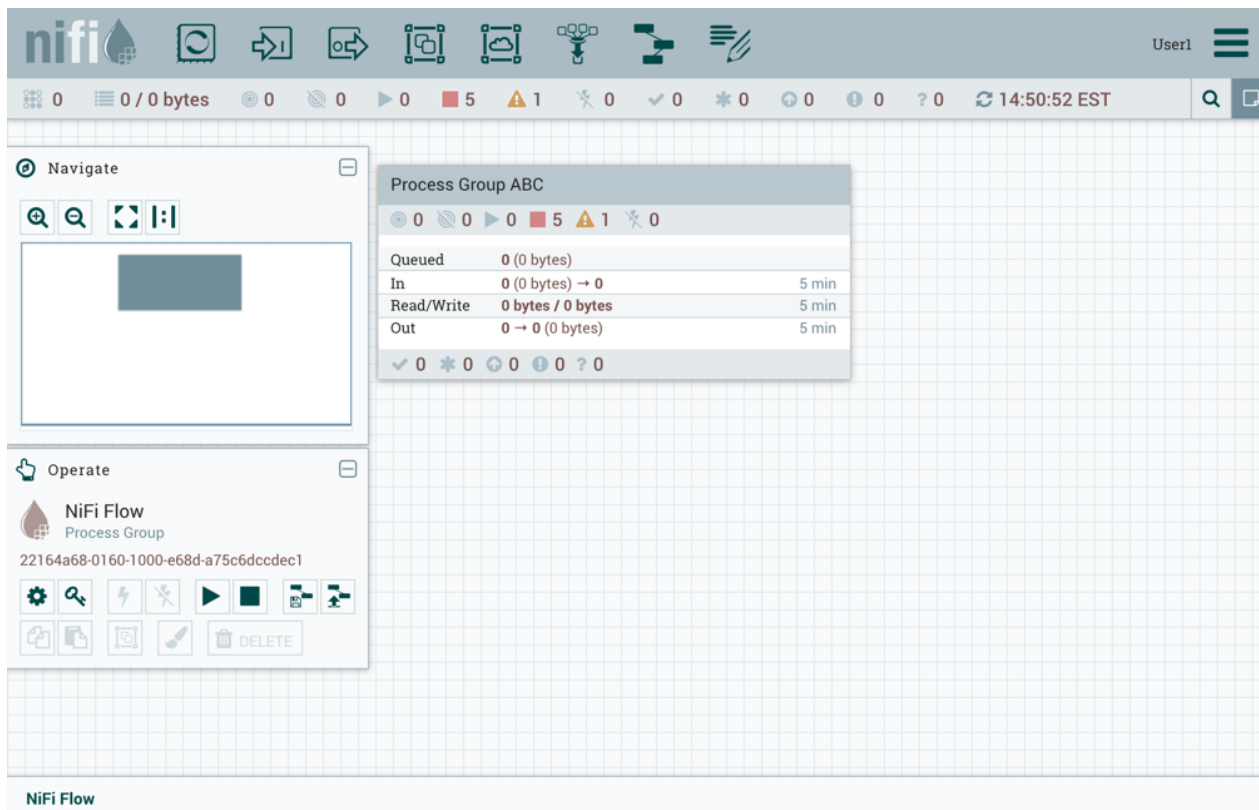


In the Stop Version Control dialog, select "Disconnect".



The removal of the process group from version control is confirmed.





Nested Versioned Flows

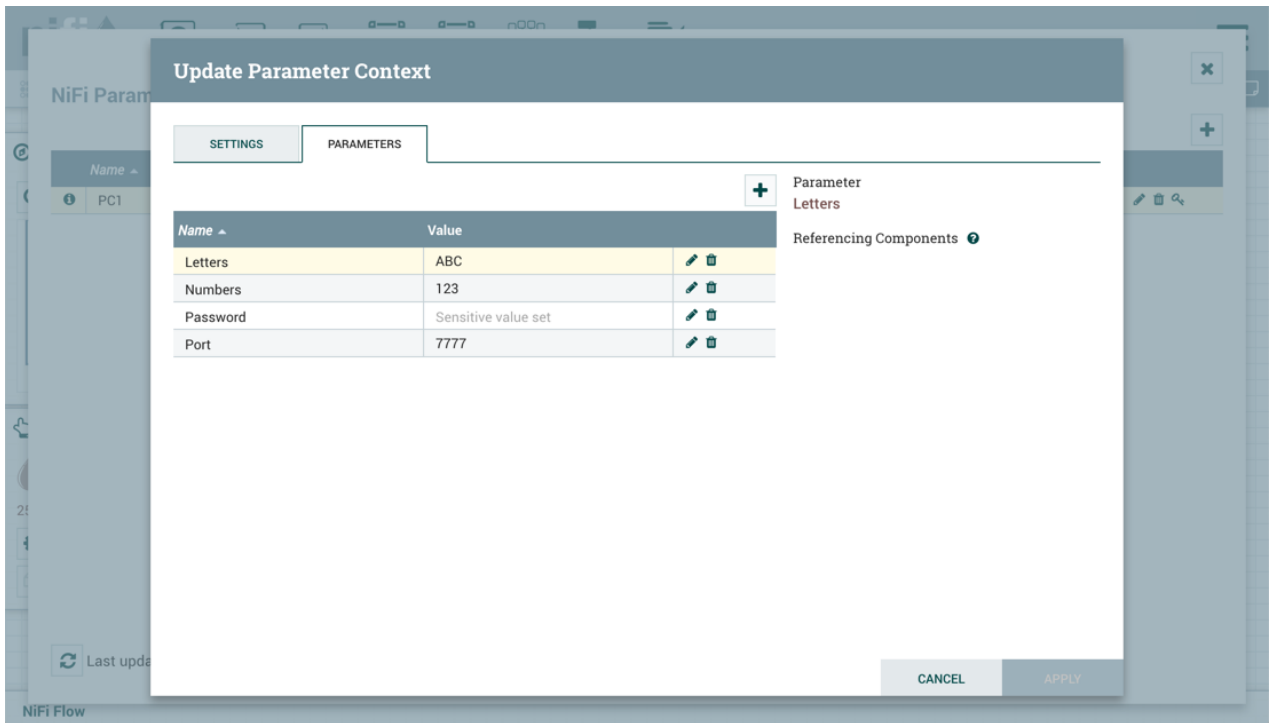
A versioned process group can contain other versioned process groups. However, local changes to a parent process group cannot be reverted or saved if it contains a child process group that also has local changes. The child process group must first be reverted or have its changes committed for those actions to be performed on the parent process group.

Parameters in Versioned Flows

When exporting a versioned flow to a Flow Registry, the name of the Parameter Context is sent for each process group that is stored. The Parameters (names, descriptions, values, whether or not sensitive) are also stored with the flow. However, Sensitive Parameter values are not stored.

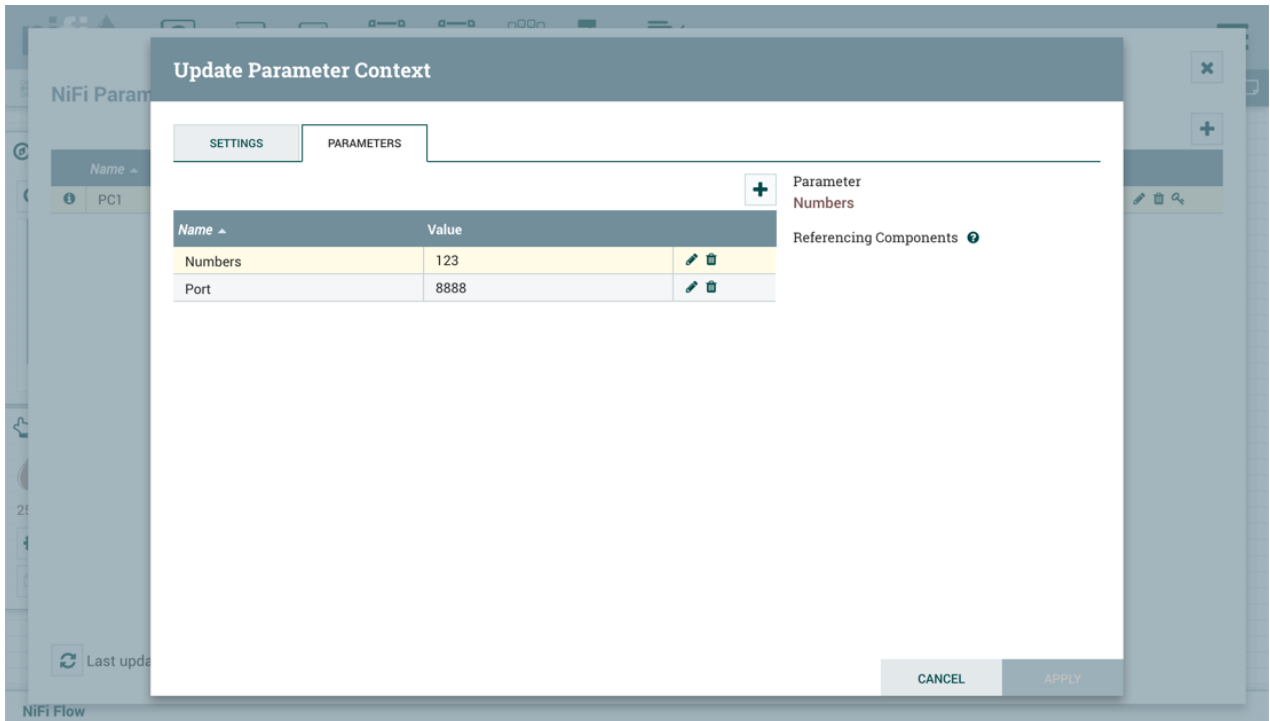
When a versioned flow is imported, a Parameter Context will be created for each one that doesn't already exist in the NiFi instance. When importing a versioned flow from Flow Registry, if NiFi has a Parameter Context with the same name, the values are merged, as described in the following example:

A flow has a Parameter Context "PC1" with the following parameters:

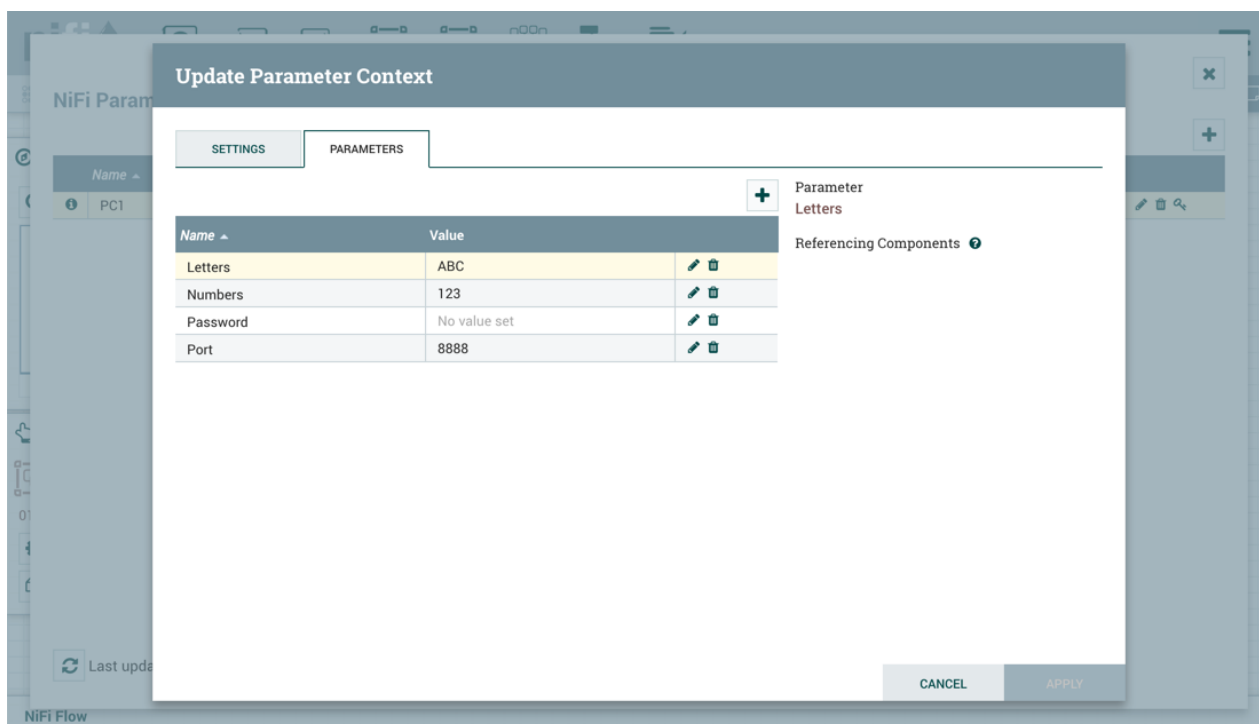


The flow is exported and saved to the Flow Registry.

A NiFi instance has a Parameter Context also named "PC1" with the following parameters:



The versioned flow is imported into the NiFi instance. The Parameter Context "PC1" now has the following parameters:



The "Letters" parameter did not exist in the NiFi instance and was added. The "Numbers" parameter existed in both the versioned flow and NiFi instance with identical values, so no changes were made. "Password" is a sensitive Parameter missing from the NiFi instance, so it was added but with no value. "Port" existed in the NiFi instance with a different value than the versioned flow, so its value remained unchanged.

Parameter Contexts are handled similarly when a flow version is changed. Consider the following two examples:

If the versioned flow referenced earlier is changed to another version (Version 2) and Version 2's Parameter Context "PC1" has a "Colors" Parameter, "Colors" will be added to "PC1" in the NiFi instance.

Version 1 of a flow does not have a Parameter Context associated with it. A new version (Version 2) does. When the flow is changed from Version 1 to Version 2, one of the following occurs:

- A new Parameter Context is created if it does not already exist
- An existing Parameter Context is assigned (by name) to the Process Group and the values of the Parameter Contexts are merged

Variables in Versioned Flows

Variables are included when a process group is placed under version control. If a versioned flow is imported that references a variable not defined in the versioned process group, the reference is maintained if the variable exists. If the referenced variable does not exist, a copy of the variable will be defined in the process group. To illustrate, assume the variable "RPG_Var" is defined in the root process group:

Variables

Process Group
NiFi Flow

Scope	Name	Value	
NiFi Flow	RPG_Var	/Users/tmp	

+ Variables
RPG_Var

Referencing Processors ?
None

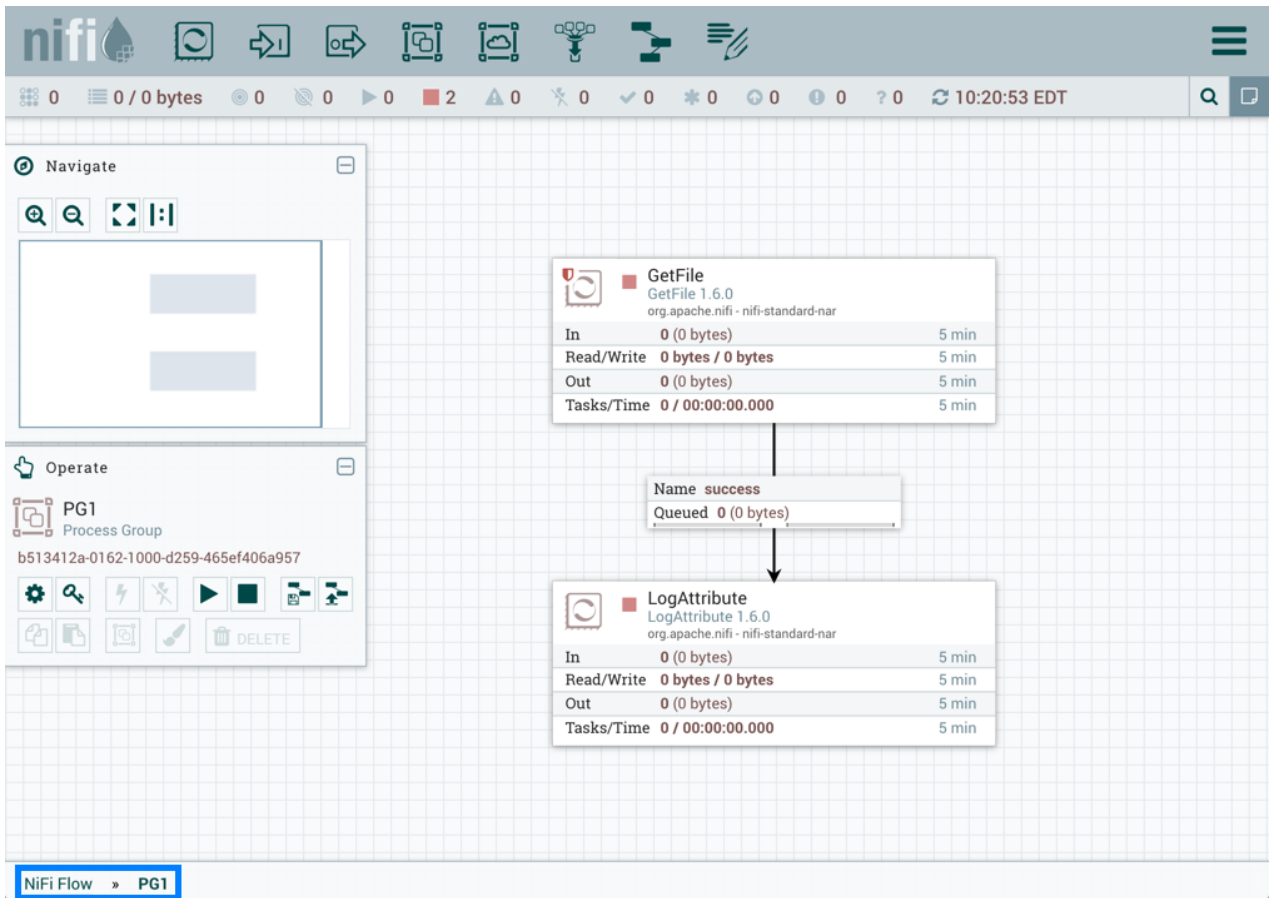
Referencing Controller Services ?
None

Unauthorized Referencing Components ?
None

Variables do not support sensitive values and will be included when versioning a Process Group.

CANCEL APPLY

A process group PG1 is created:



The GetFile processor in PG1 references the variable "RPG_Var":

Variables

Process Group
PG1

Scope	Name	Value	
NiFi Flow	RPG_Var	/Users/tmp	→

Variables
RPG_Var

Referencing Processors
GetFile

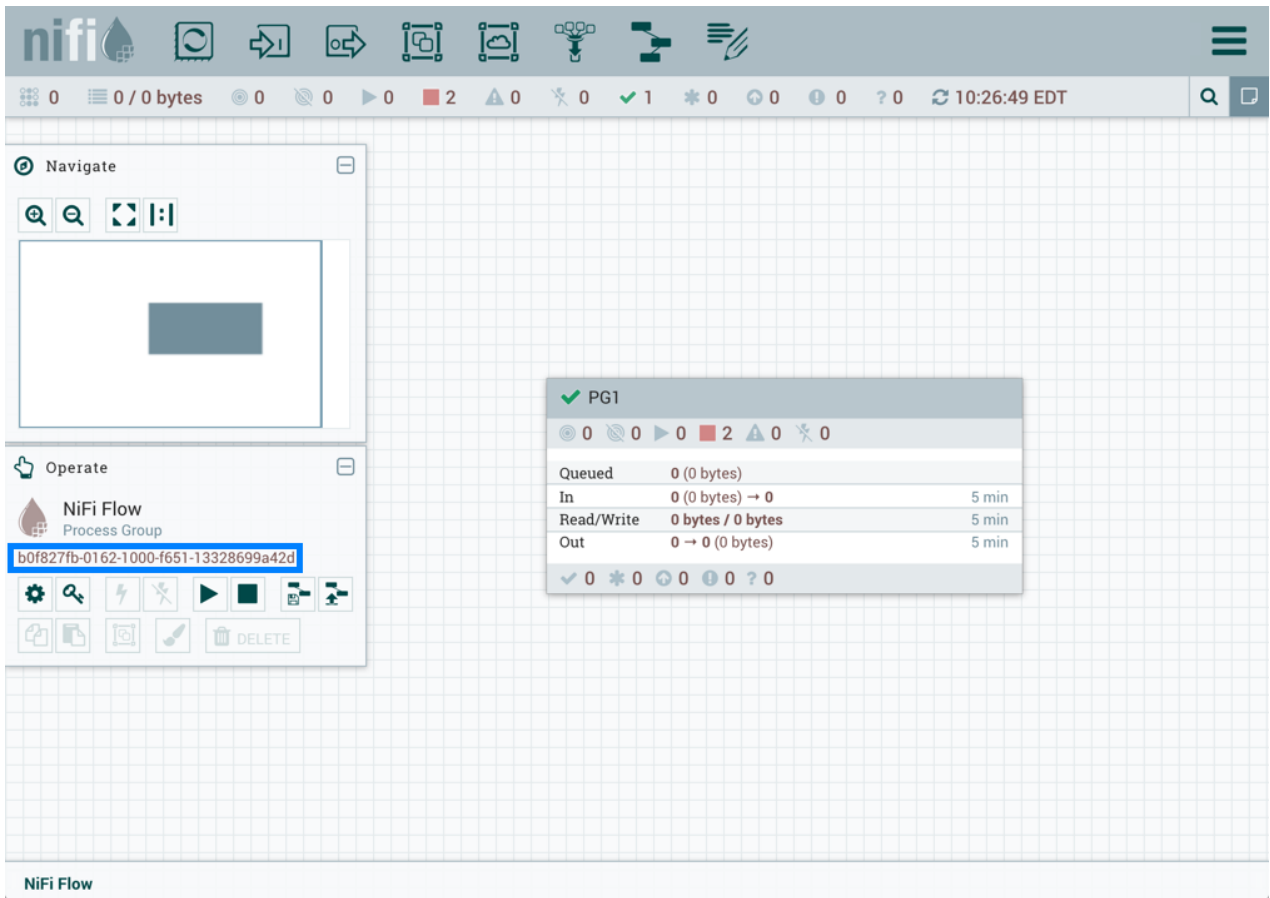
Referencing Controller Services
None

Unauthorized Referencing Components
None

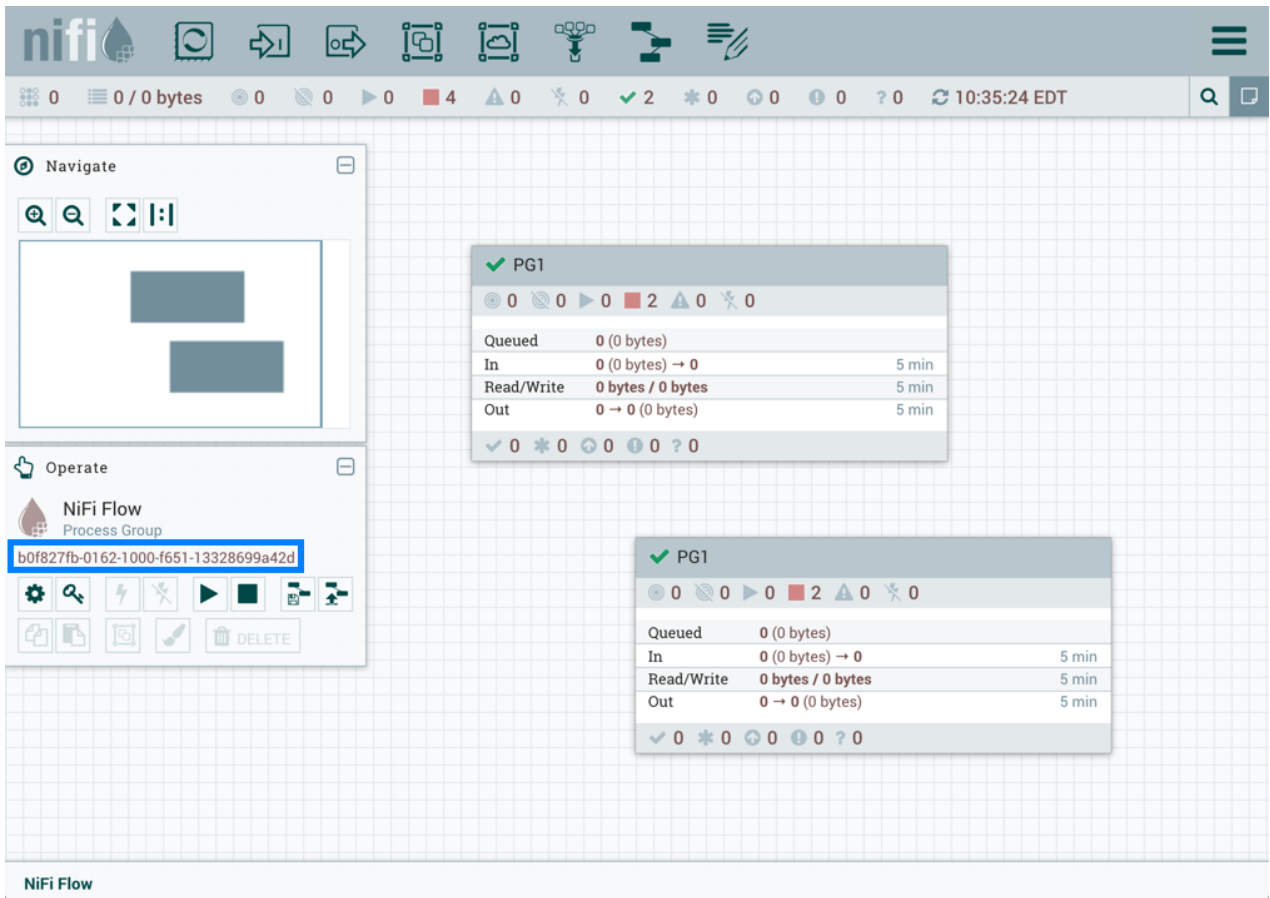
Variables do not support sensitive values and will be included when versioning a Process Group.

CANCEL APPLY

PG1 is saved as a versioned flow:



If PG1 versioned flow is imported into this same NiFi instance:



the added GetFile processor will also reference the "RPG_Var" variable that exists in the root process group:

Variables

Process Group
PG1

Scope	Name	Value
NiFi Flow	RPG_Var	/Users/tmp

Variables
RPG_Var

Referencing Processors

- GetFile
- GetFile

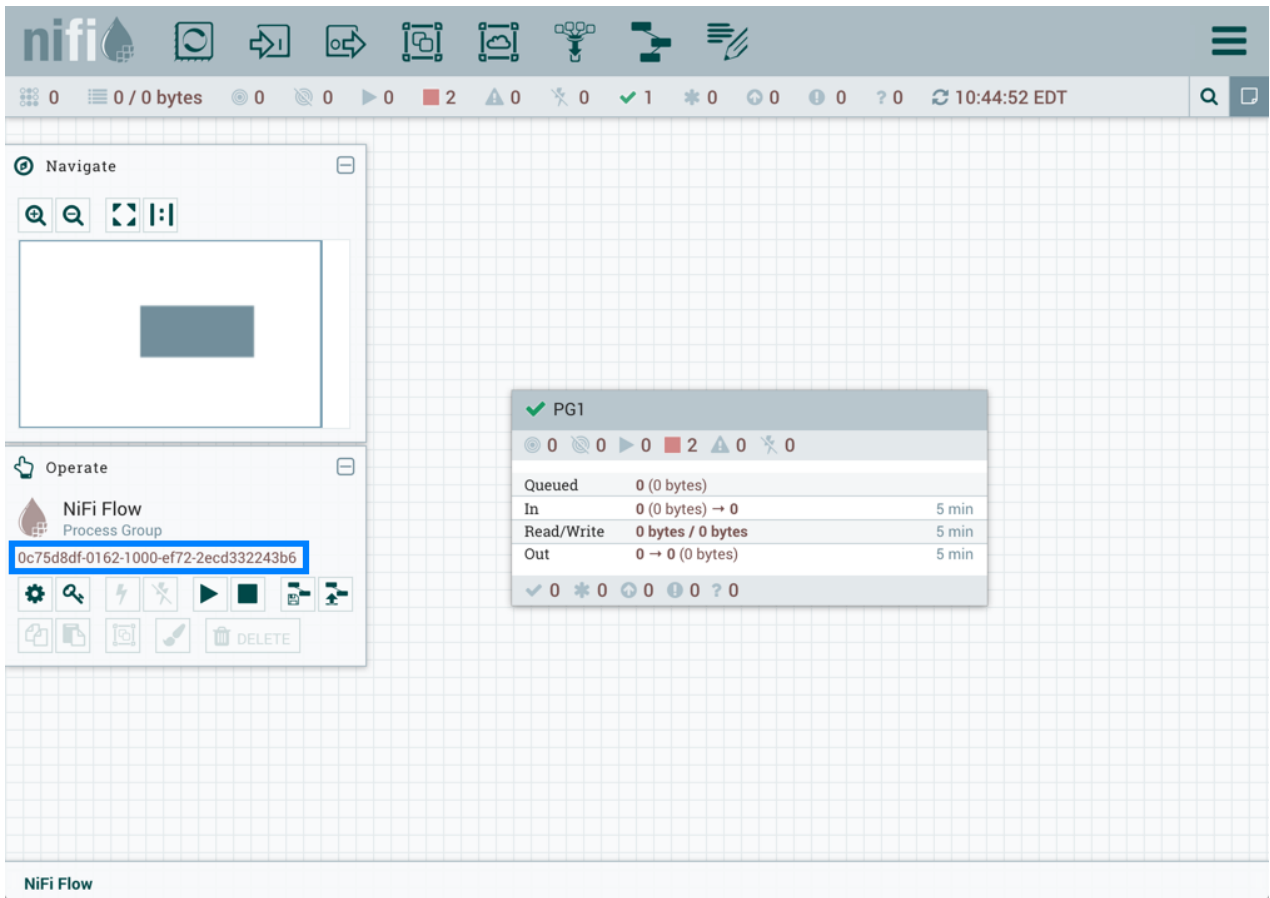
Referencing Controller Services
None

Unauthorized Referencing Components
None

Variables do not support sensitive values and will be included when versioning a Process Group.

CANCEL APPLY

If PG1 versioned flow is imported into a different NiFi instance where "RPG_Var" does not exist:



A "RPG_Var" variable is created in the PG1 process group:

Variables

Process Group
PG1

Scope	Name	Value
PG1	RPG_Var	/Users/tmp

Variables
RPG_Var

Referencing Processors
GetFile

Referencing Controller Services
None

Unauthorized Referencing Components
None

Variables do not support sensitive values and will be included when versioning a Process Group.

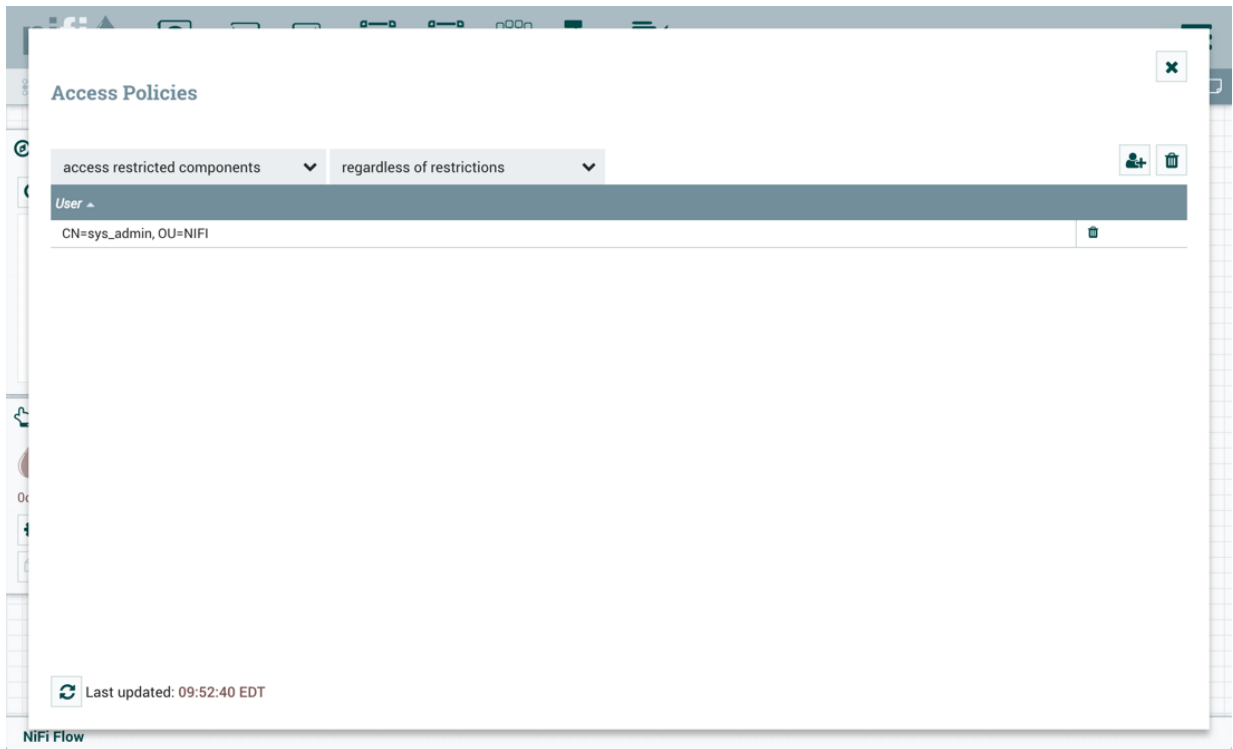
CANCEL APPLY

Restricted Components in Versioned Flows

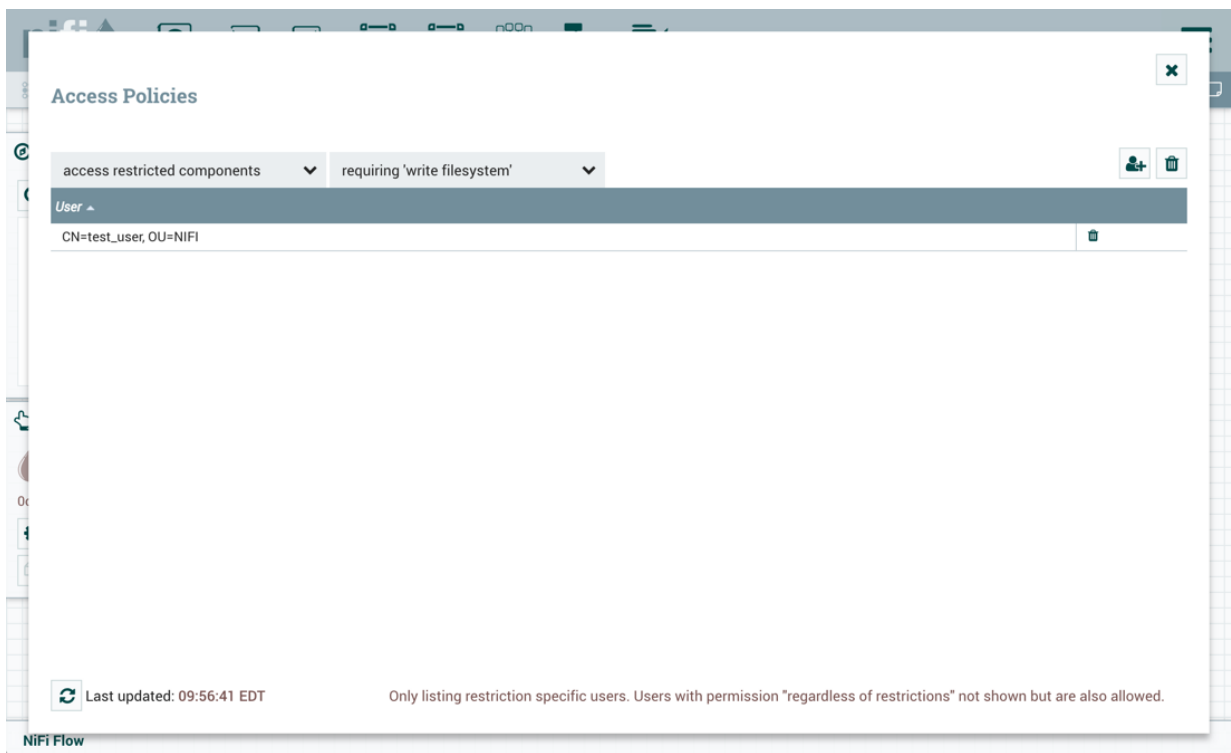
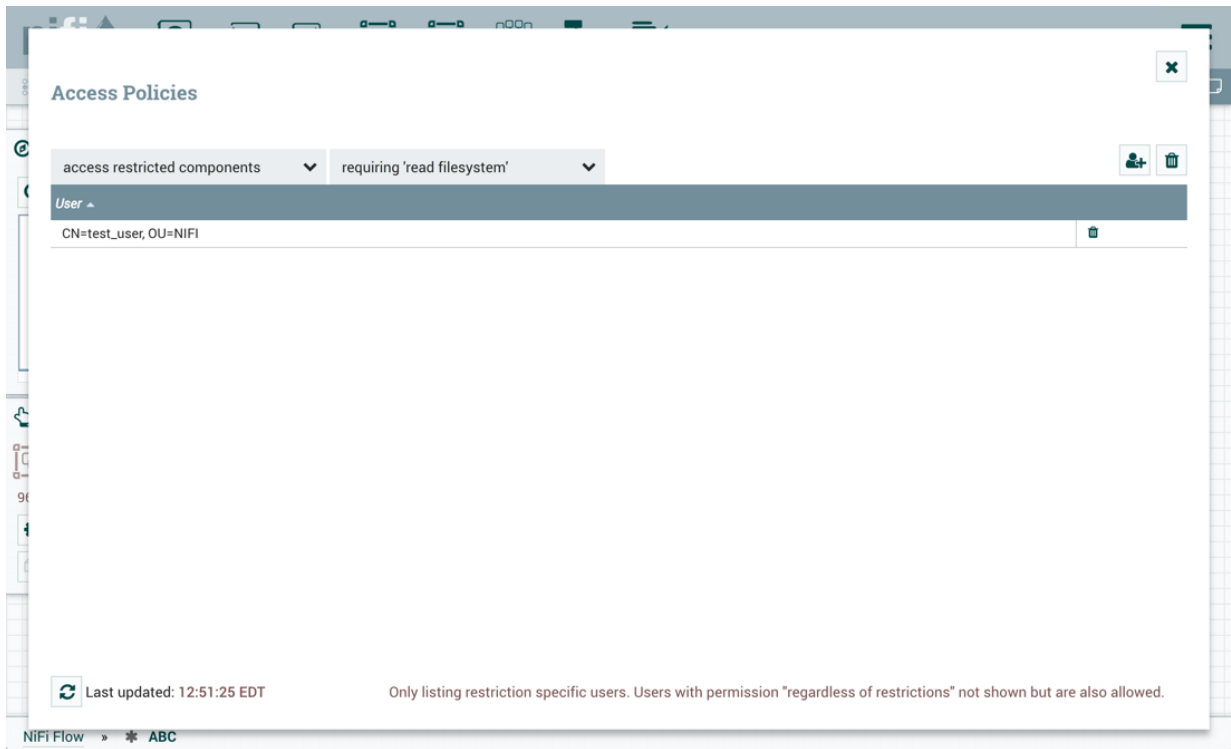
To import a versioned flow or revert local changes in a versioned flow, a user must have access to all the components in the versioned flow. As such, it is recommended that restricted components are created at the root process group level if they are to be utilized in versioned flows. Let's walk through some examples to illustrate the benefits of this configuration. Assume the following:

- There are two users, "sys_admin" and "test_user" who have access to both view and modify the root process group.

- "sys_admin" has access to all restricted components.

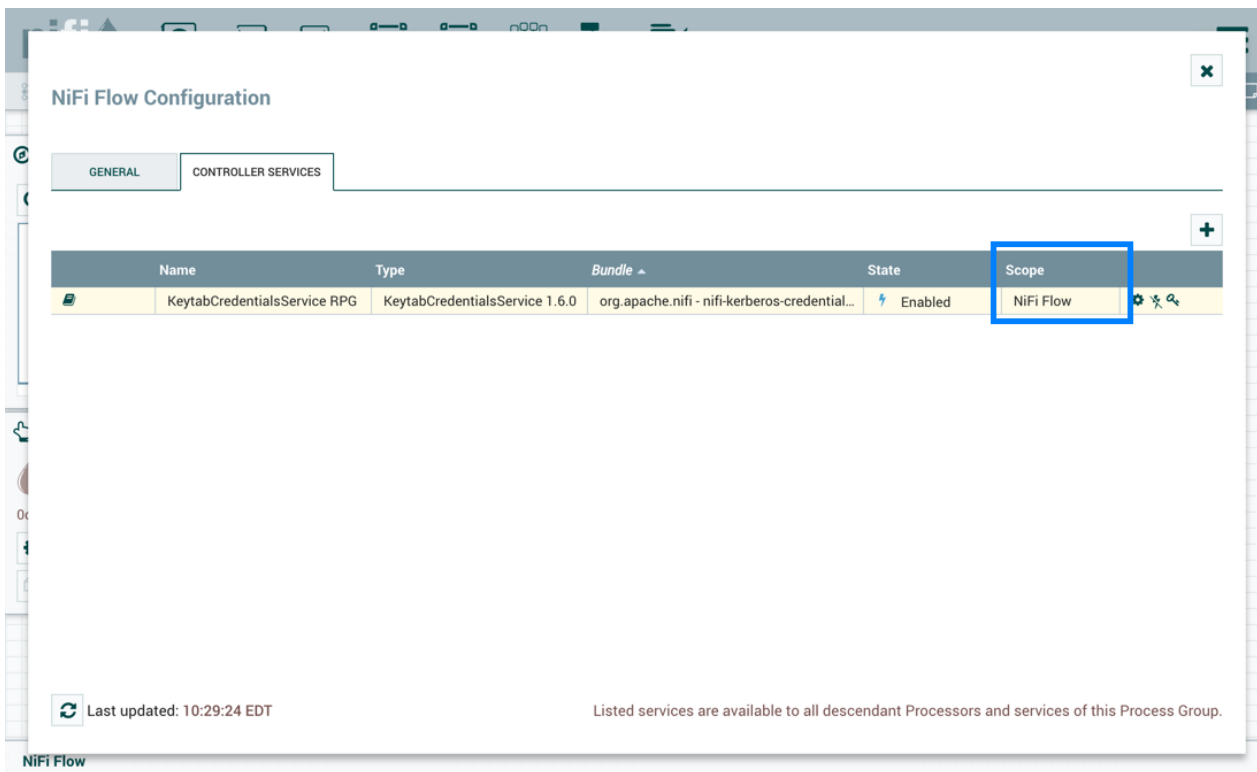


- "test_user" has access to restricted components requiring 'read filesystem' and 'write filesystem'.

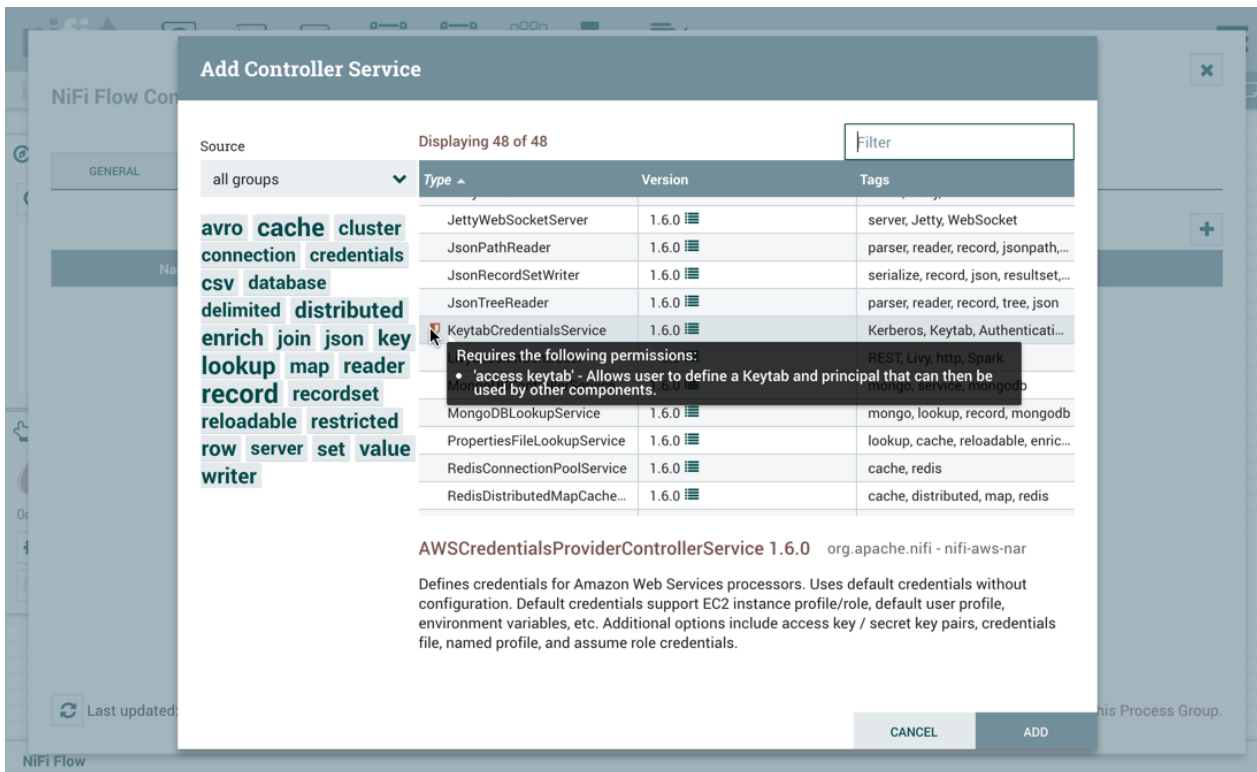


Restricted Controller Service Created in Root Process Group

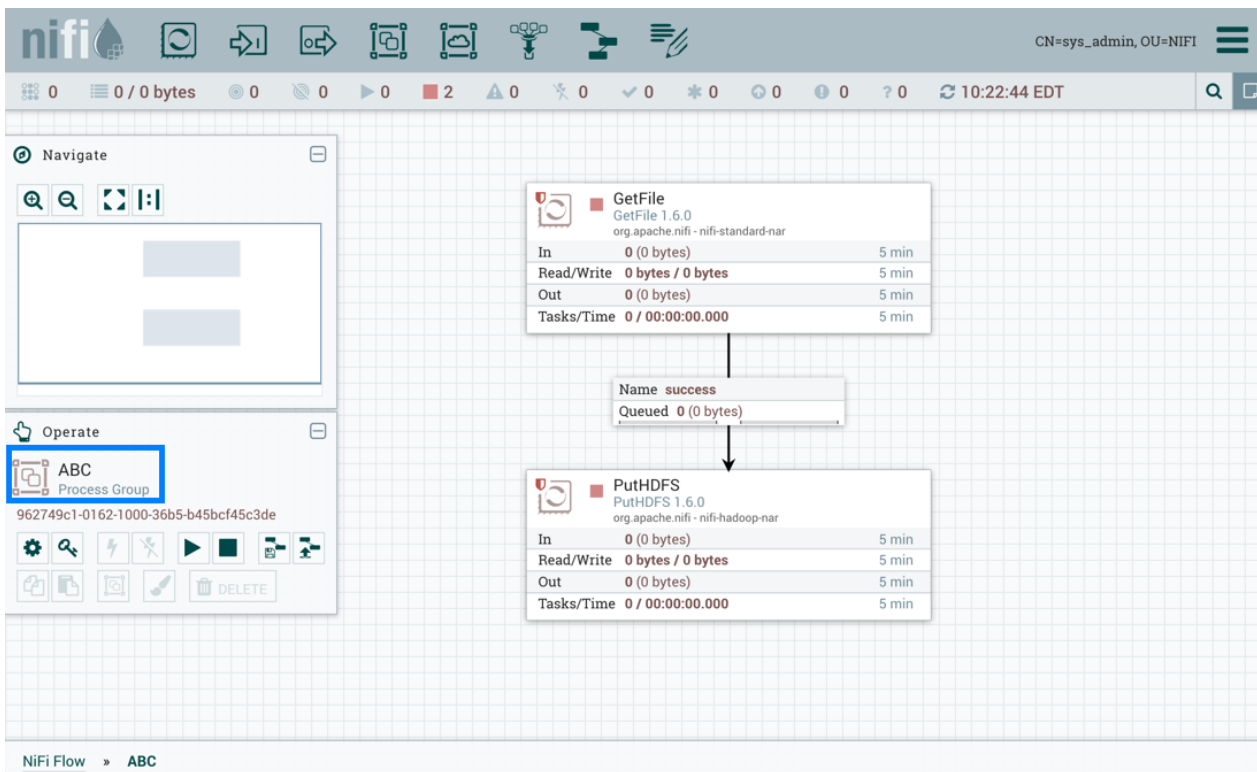
In this first example, `sys_admin` creates a `KeytabCredentialsService` controller service at the root process group level.



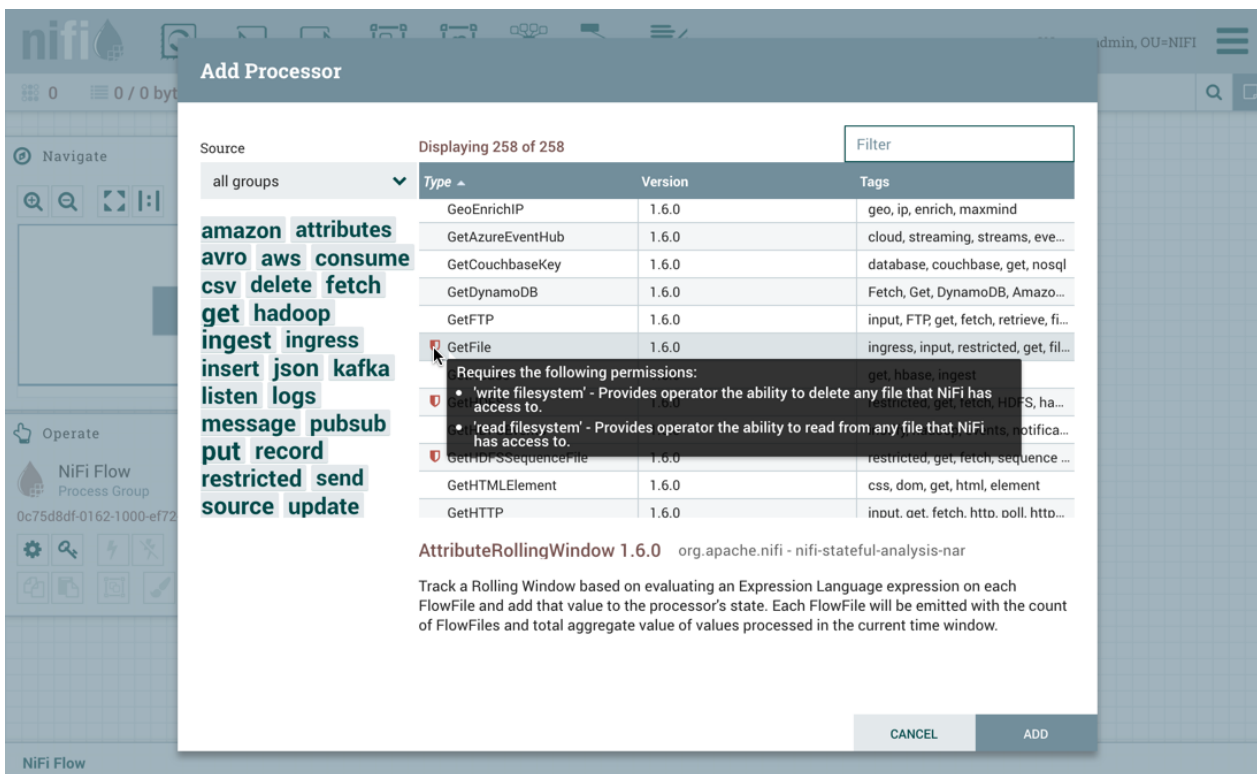
KeytabCredentialService controller service is a restricted component that requires 'access keytab' permissions:



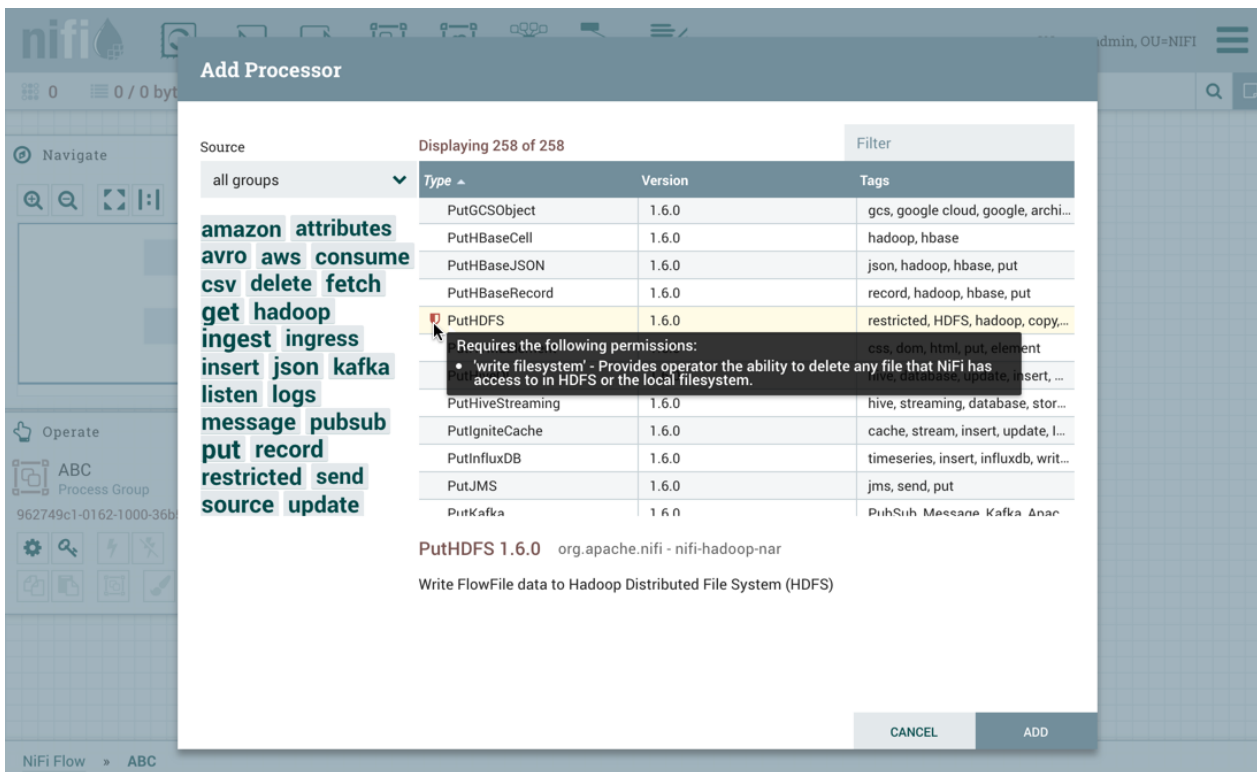
Sys_admin creates a process group ABC containing a flow with GetFile and PutHDFS processors:



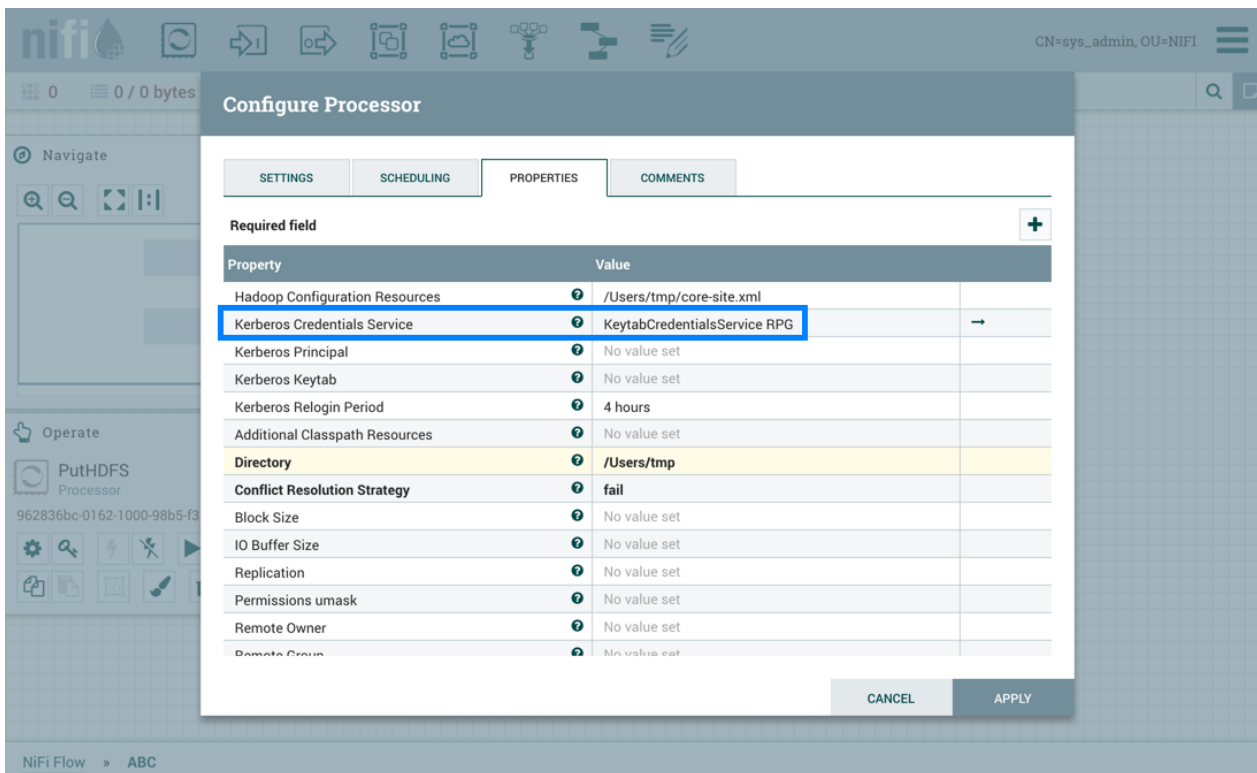
GetFile processor is a restricted component that requires 'write filesystem' and 'read filesystem' permissions:



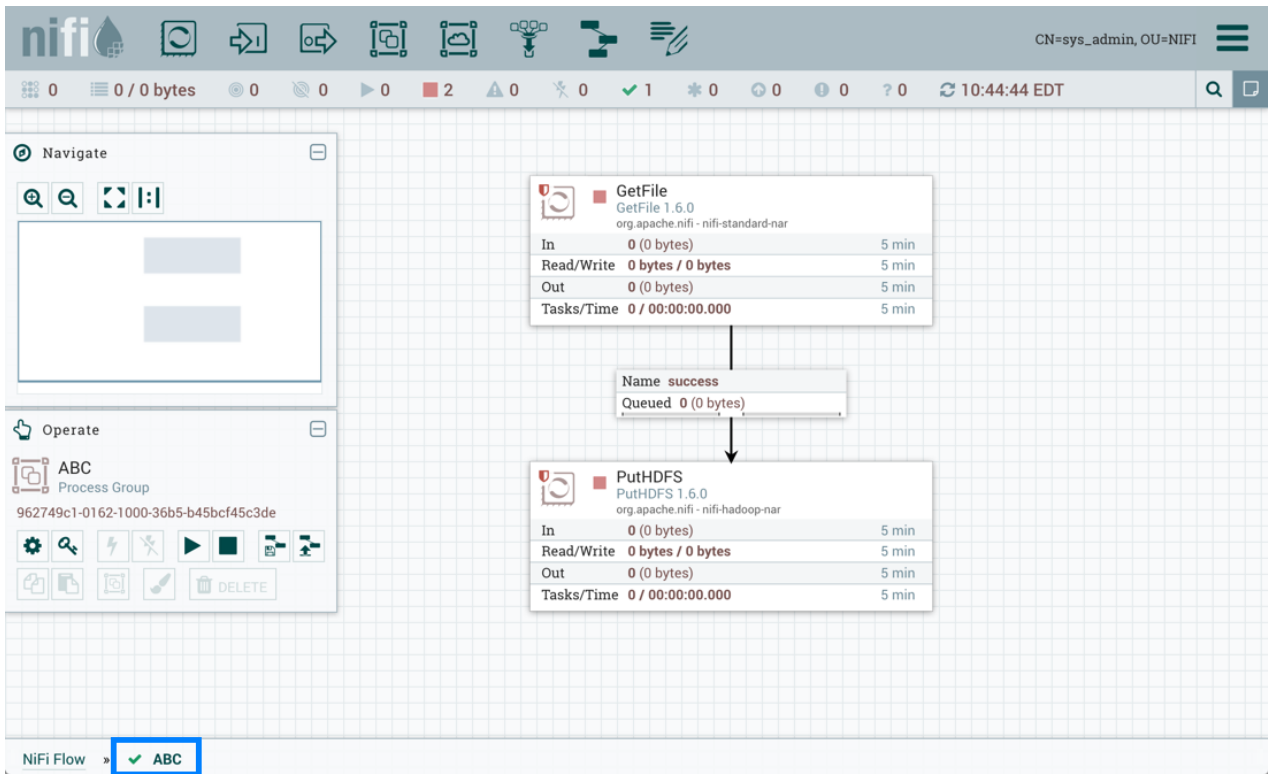
PutHDFS is a restricted component that requires 'write filesystem' permissions:



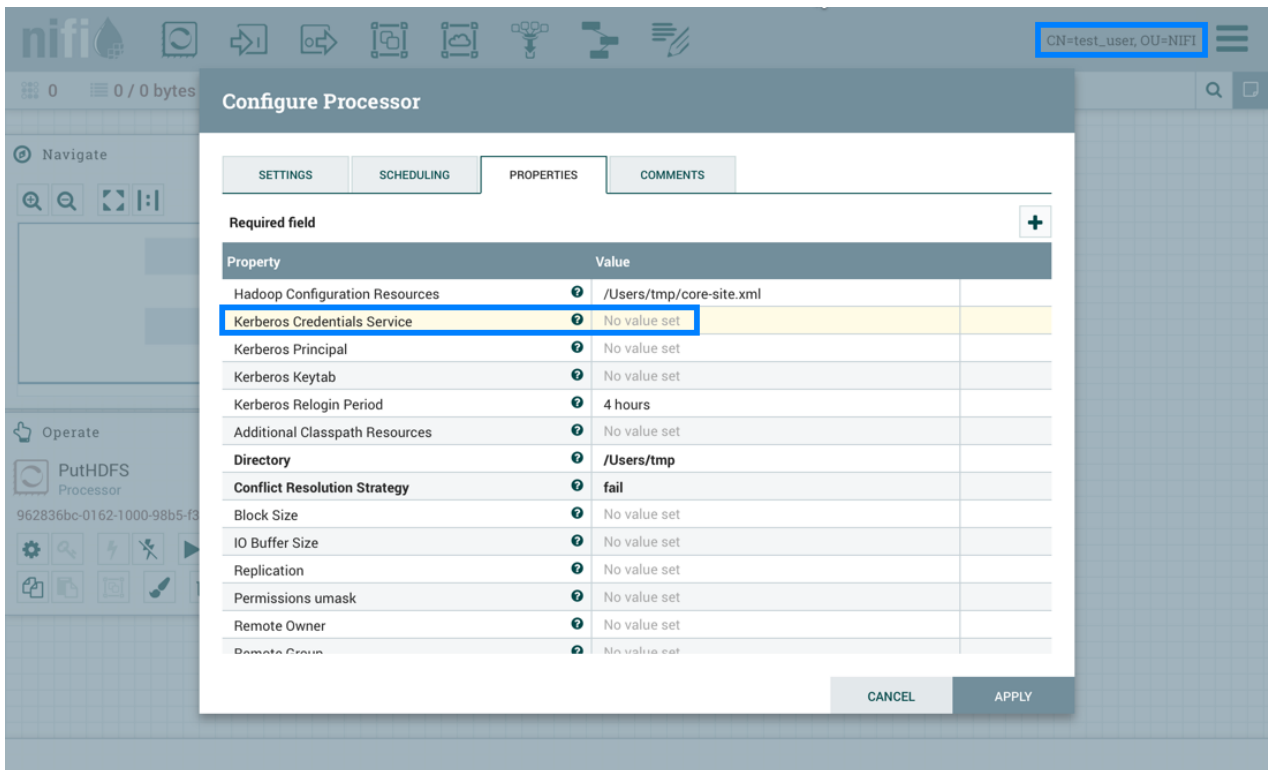
The PutHDFS processor is configured to use the root process group level KeytabCredentialsService controller service:



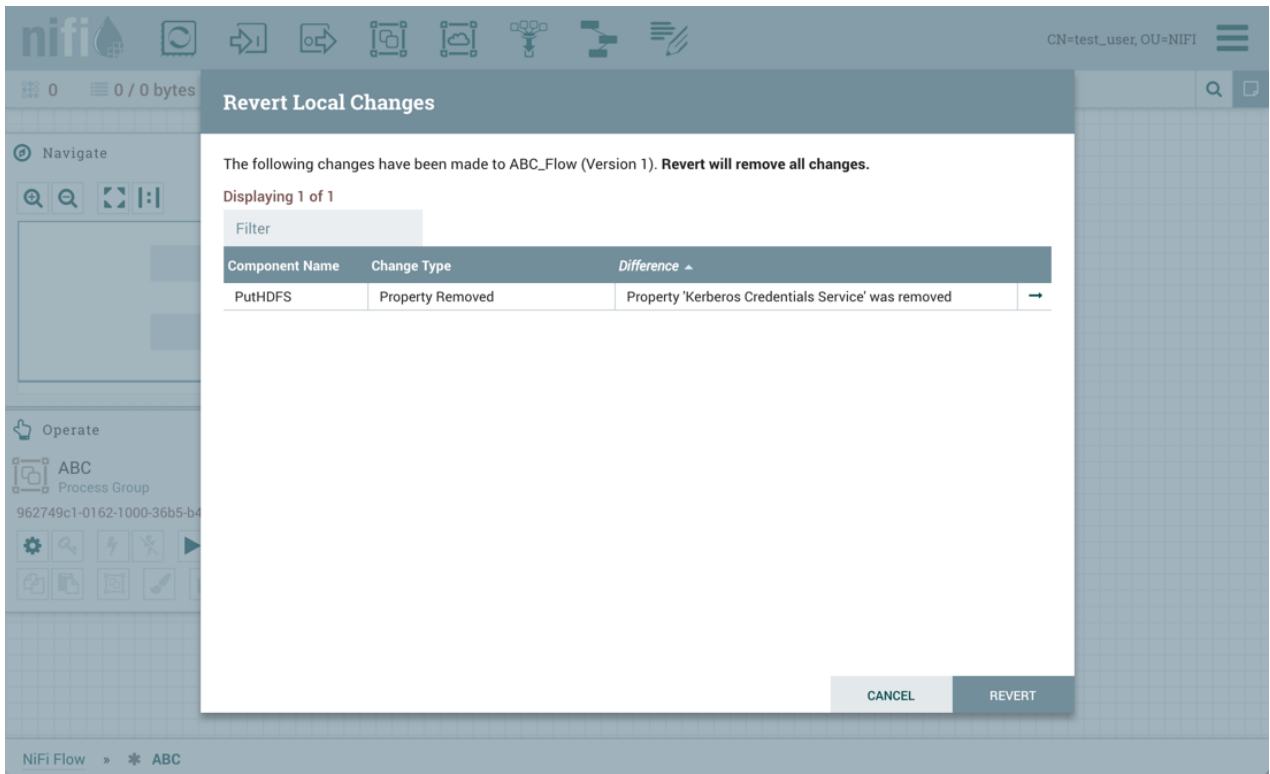
Sys_admin saves the process group as a versioned flow:



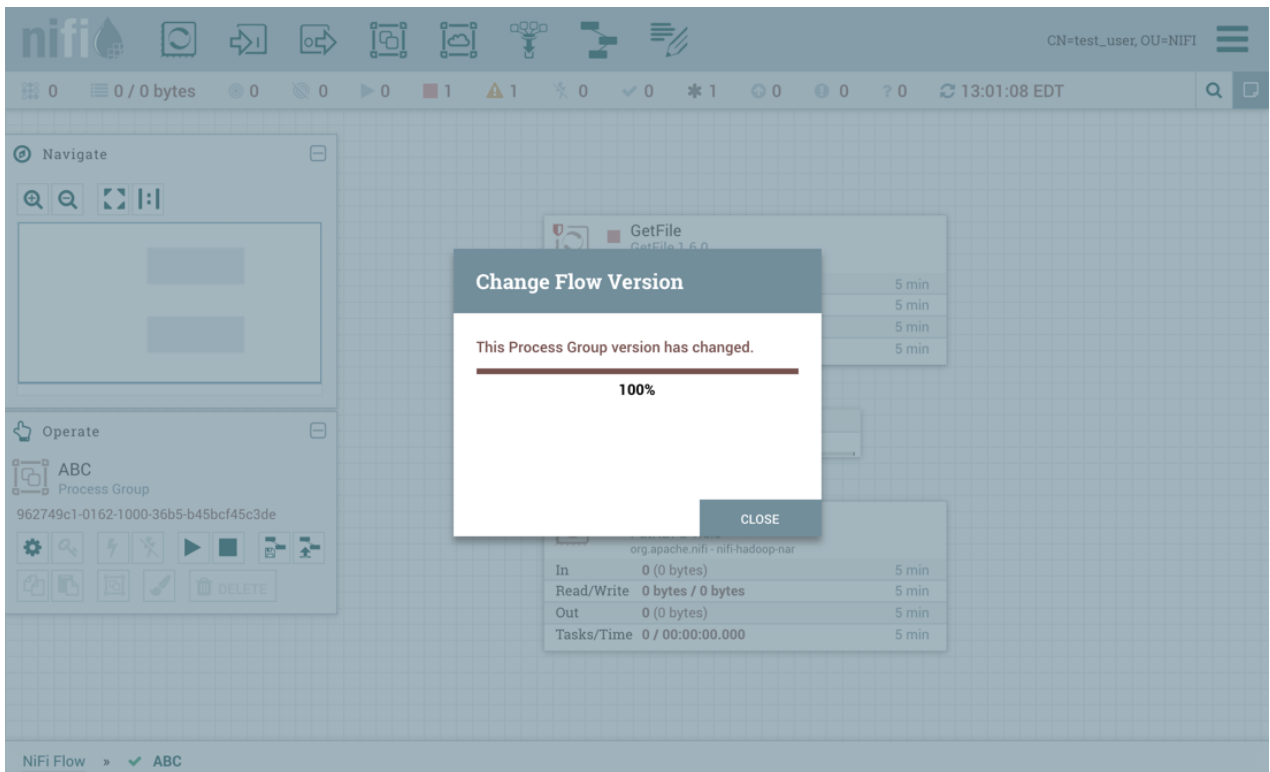
Test_user changes the flow by removing the KeytabCredentialsService controller service:



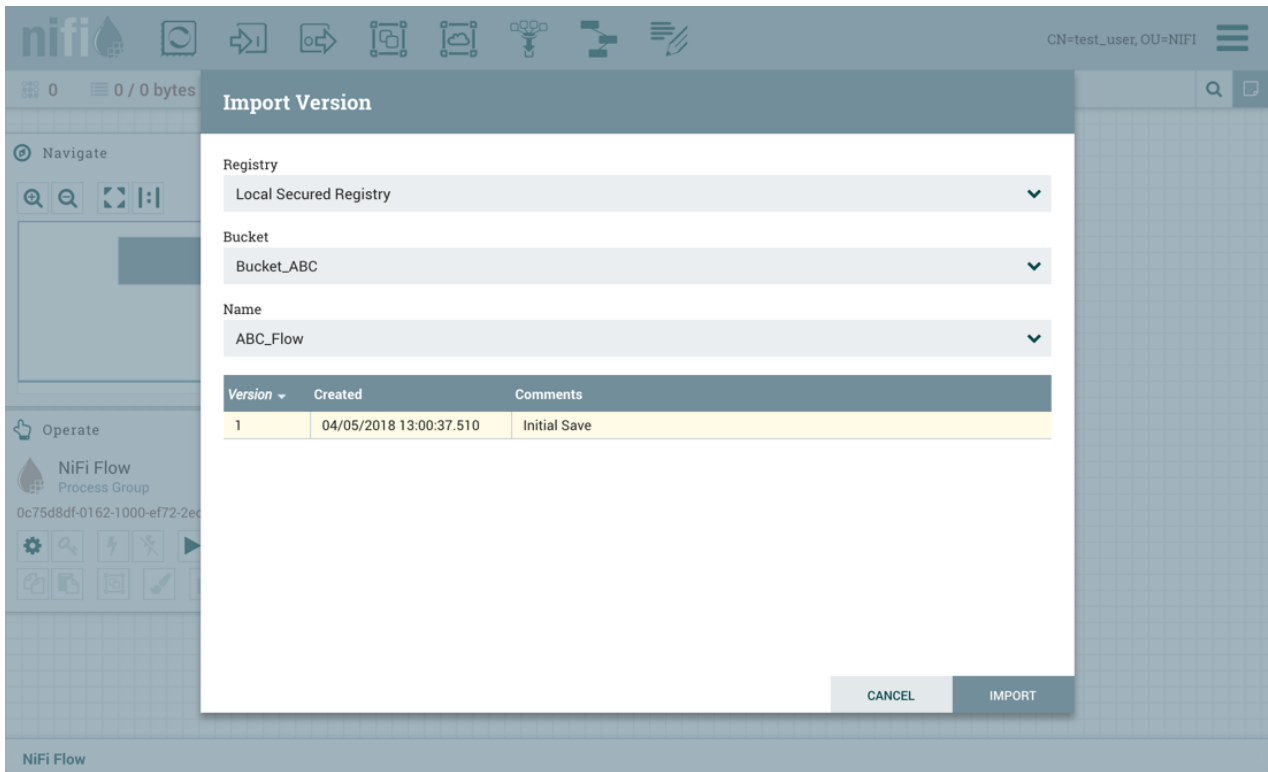
If test_user chooses to revert this change:



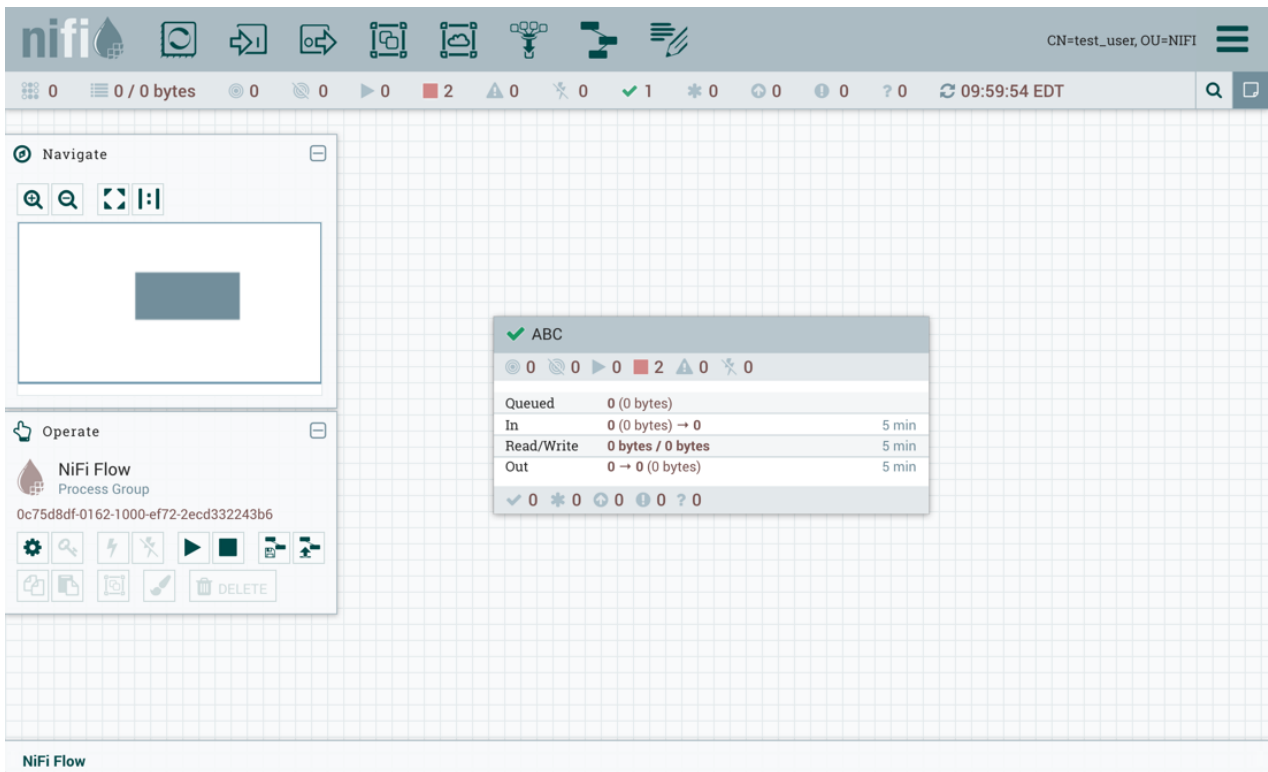
the revert is successful:



Additionally, if test_user chooses to import the ABC versioned flow:



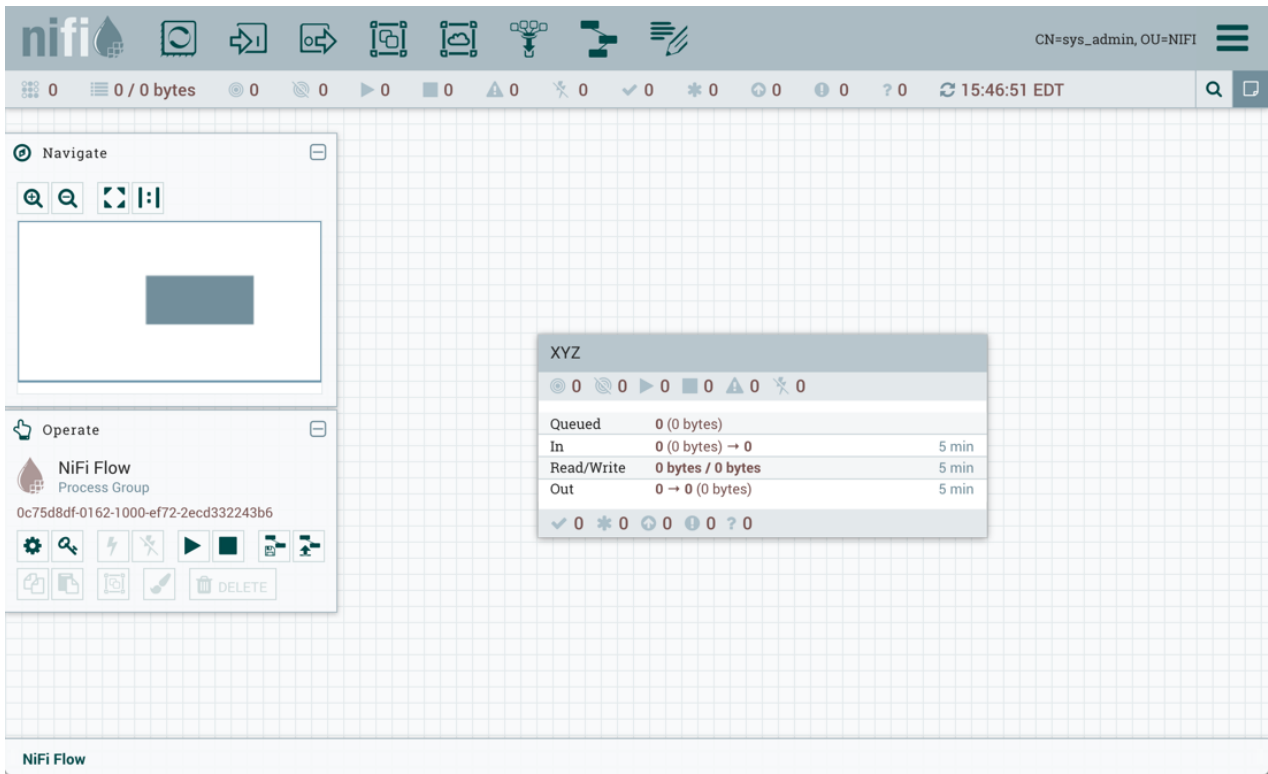
The import is successful:



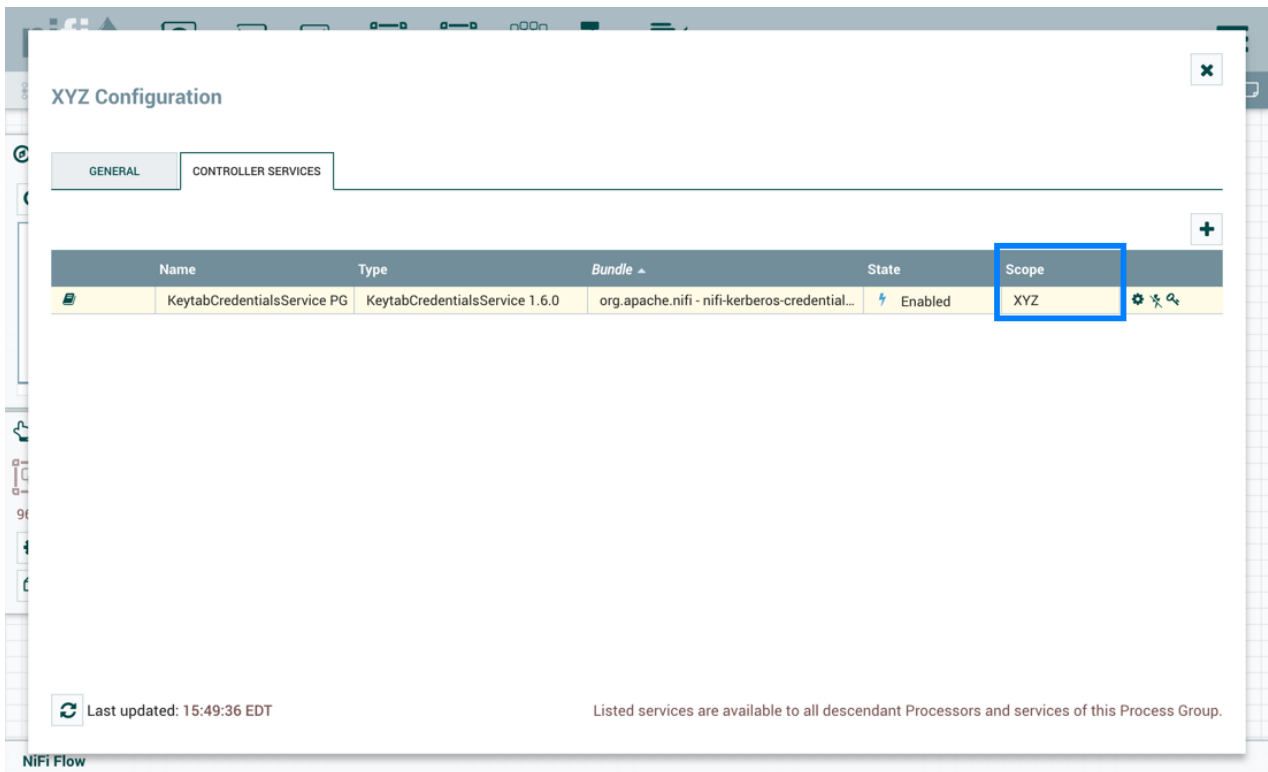
Restricted Controller Service Created in Process Group

Now, consider a second scenario where the controller service is created on the process group level.

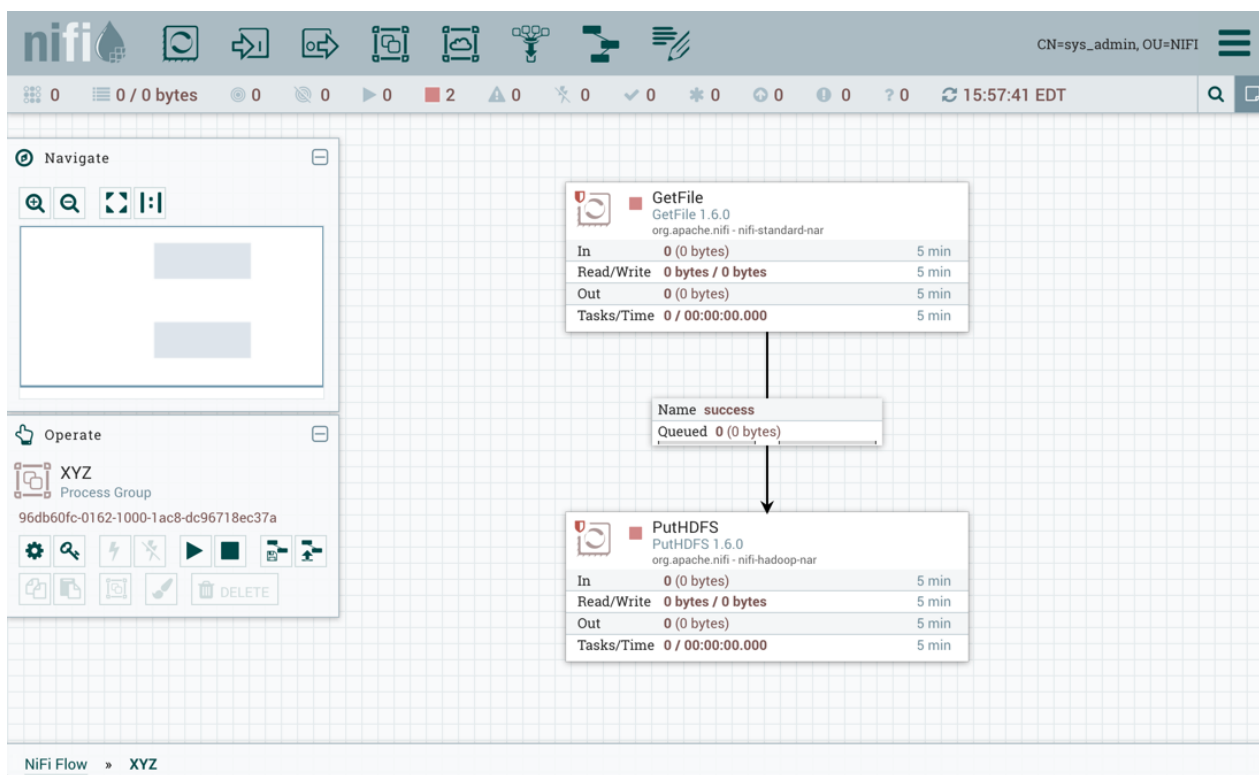
Sys_admin creates a process group XYZ:



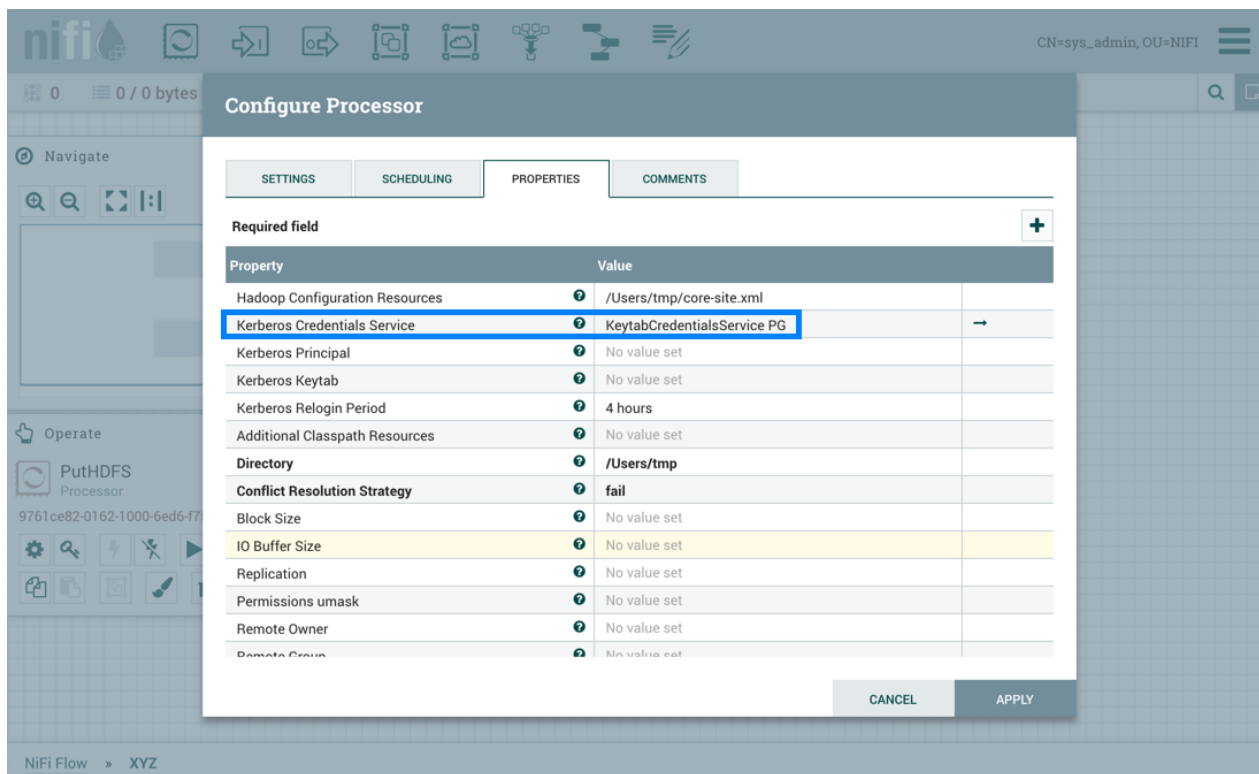
Sys_admin creates a KeytabCredentialsService controller service at the process group level:



The same GetFile and PutHDFS flow is created in the process group:

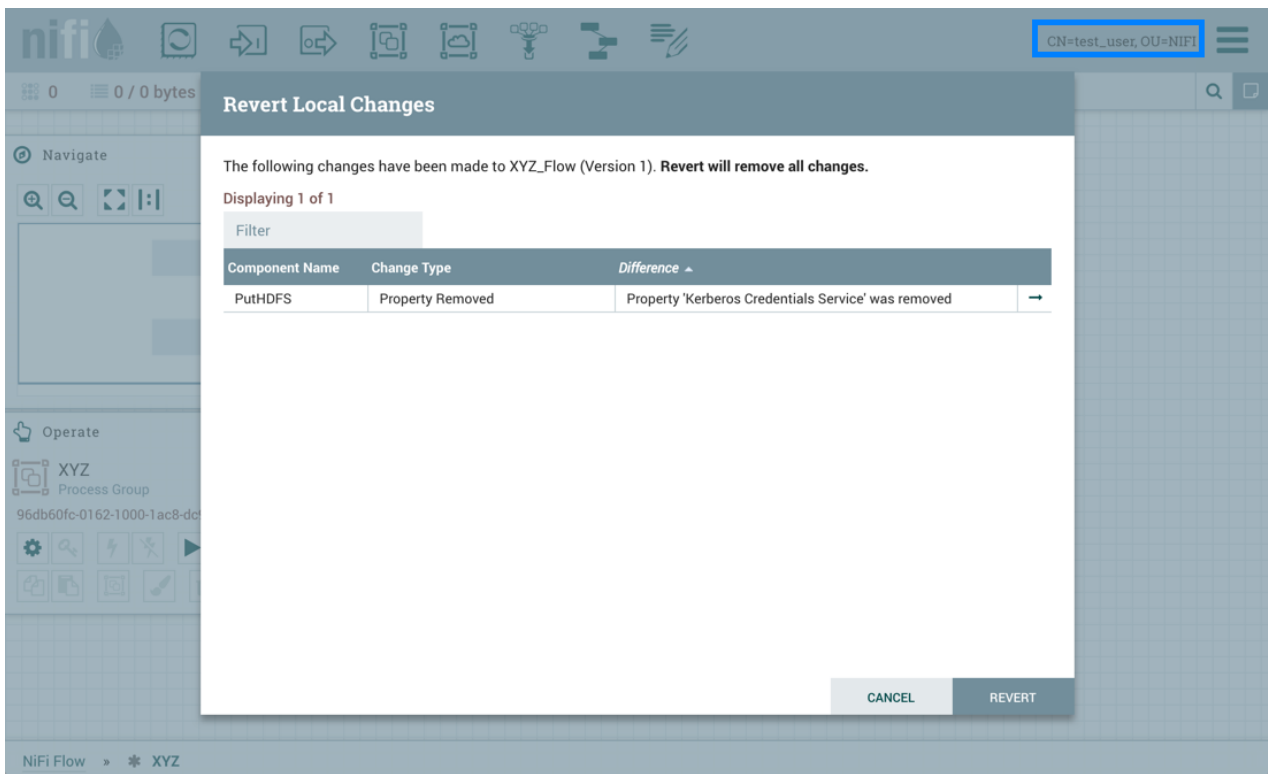


However, PutHDFS now references the process group level controller service:

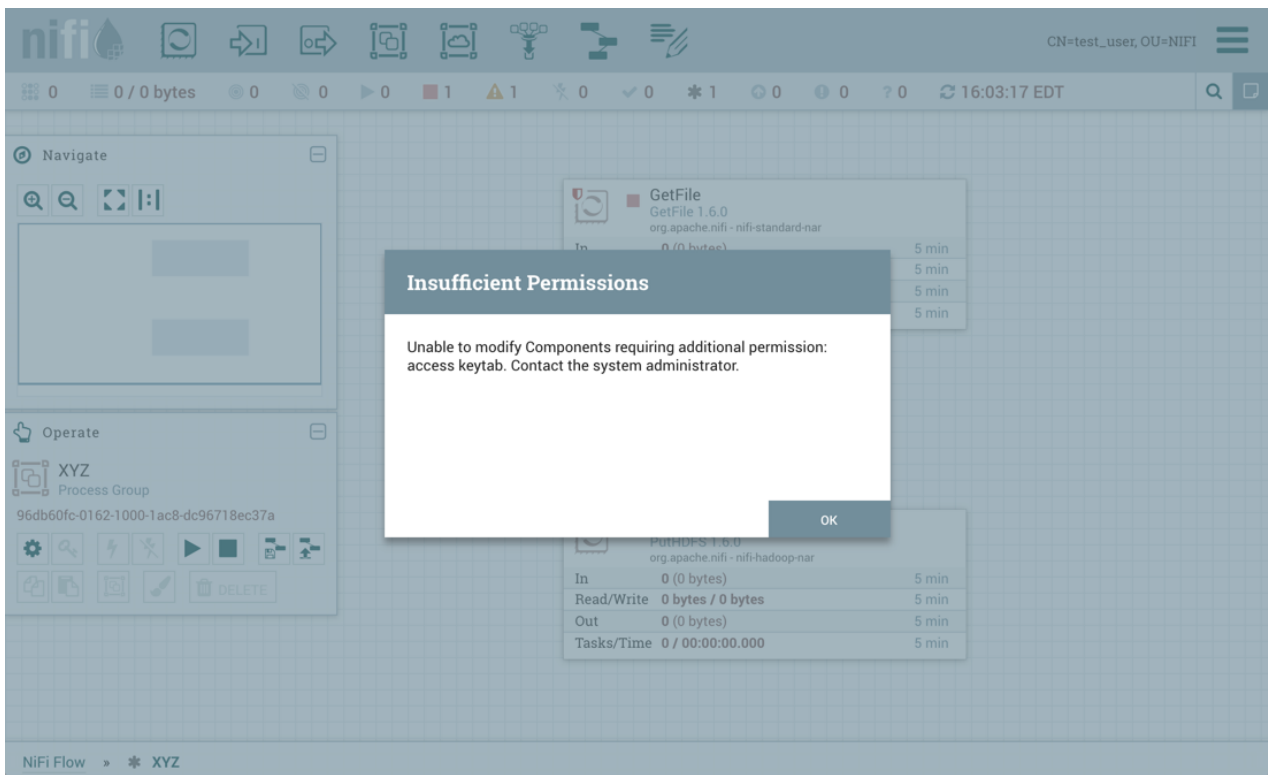


Sys_admin saves the process group as a versioned flow.

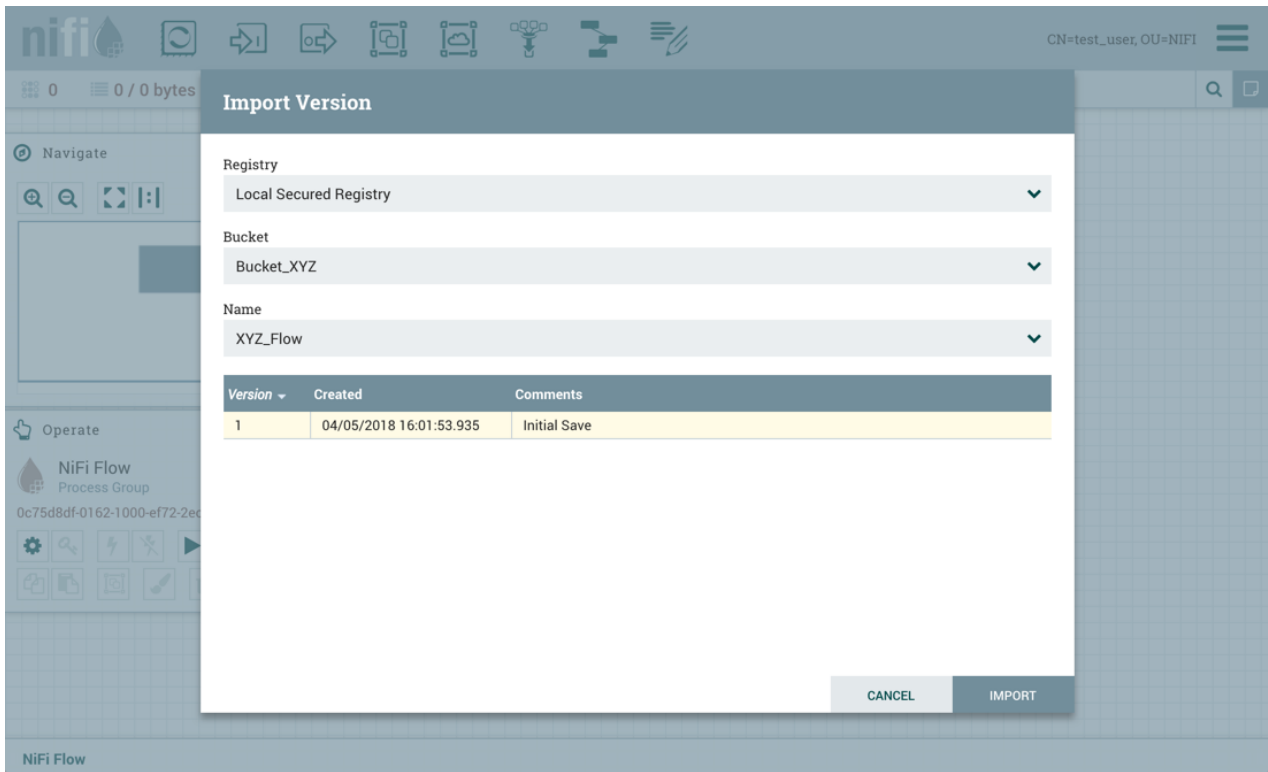
Test_user changes the flow by removing the KeytabCredentialsService controller service. However, with this configuration, if test_user attempts to revert this change:



the revert is unsuccessful because test_user does not have the 'access keytab' permissions required by the KeytabCredentialService controller service:



Similarly, if test_user tries to import the XYZ versioned flow:



The import fails:

