

## Using Cloudera Edge Management

Date published: 2019-10-25

Date modified: 2019-10-25



# 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

<b>Cloudera Edge Management User Interface.....</b>	<b>4</b>
Deployment Monitor.....	4
Events Monitor.....	4
Flow Designer.....	5
<b>Monitoring Deployments.....</b>	<b>7</b>
<b>Monitoring Events.....</b>	<b>9</b>
<b>Expression Language.....</b>	<b>12</b>
<b>Parameters.....</b>	<b>13</b>
Adding Parameters.....	14
Referencing Parameters.....	17
Using Parameters.....	17
<b>Building a DataFlow.....</b>	<b>26</b>
Adding Components to the Canvas.....	26
Configuring a Processor.....	28
Configuring a Remote Process Group.....	33
Adding Services.....	35
Connecting Components.....	36
Configuring a Connection.....	37
Bending Connections.....	38
Refreshing a DataFlow.....	39
<b>Publishing a DataFlow.....</b>	<b>39</b>
<b>Reverting a DataFlow.....</b>	<b>40</b>
<b>Example DataFlow.....</b>	<b>41</b>
<b>Monitoring Metrics with Grafana.....</b>	<b>43</b>

# Cloudera Edge Management User Interface

This section describes the Cloudera Edge Management (CEM) UI and highlights the different segments of the UI.

The CEM UI provides mechanisms for building, editing and publishing dataflows, as well as monitoring C2 server and agent events.

When you start the application, you can navigate to the UI by going to the default address of `http://<hostname>:10080/efm/ui` in a web browser.

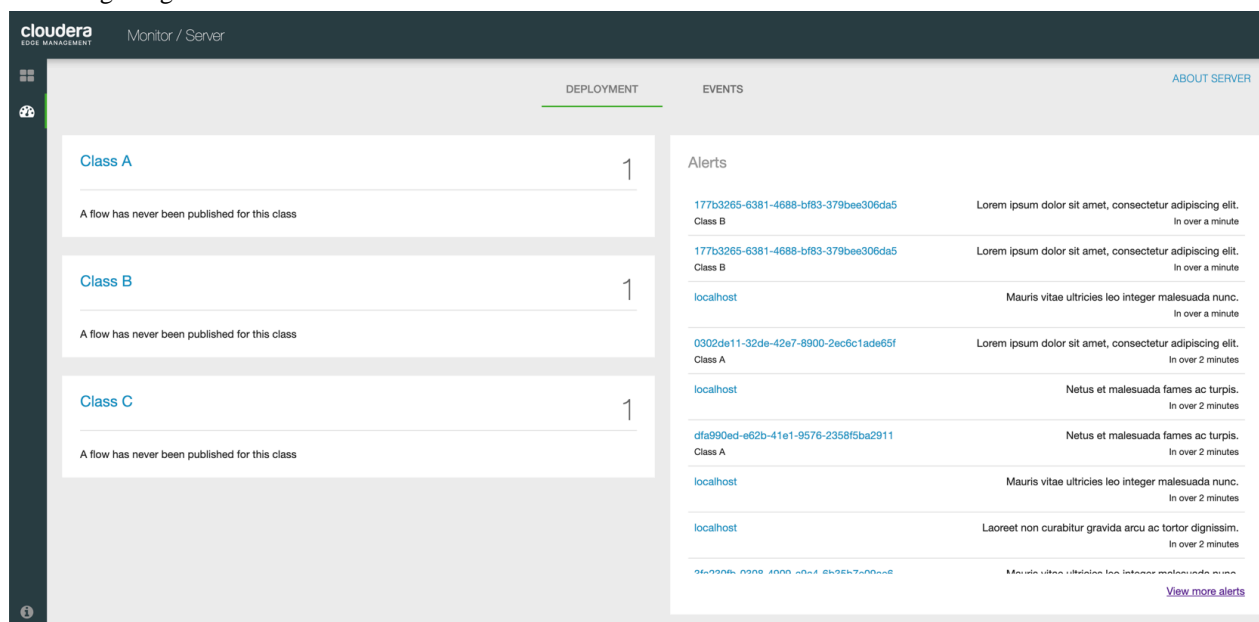


**Note:** Anyone who has access to the CEM UI can view events, and view or modify dataflows.

## Deployment Monitor

The Deployment Monitor screen enables you to monitor C2 server and agent deployments.

When you navigate to the CEM UI for the first time, the Deployment Monitor screen appears as shown in the following image:



For more information on how to monitor deployments, see *Monitoring Deployments*.

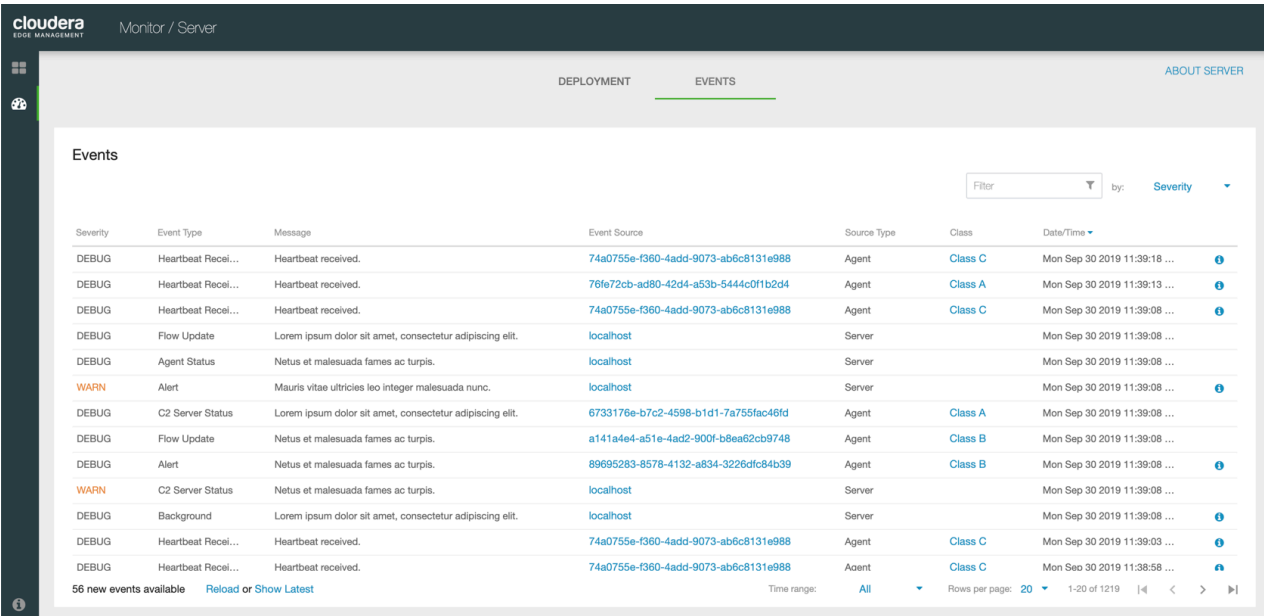
### Related Information

[Monitoring Deployments](#)

## Events Monitor

The Events Monitor screen enables you to monitor C2 server and agent events.

When you navigate to the Events page, the Events Monitor screen appears as shown in the following image:



For more information on how to monitor events, see *Monitoring Events*.

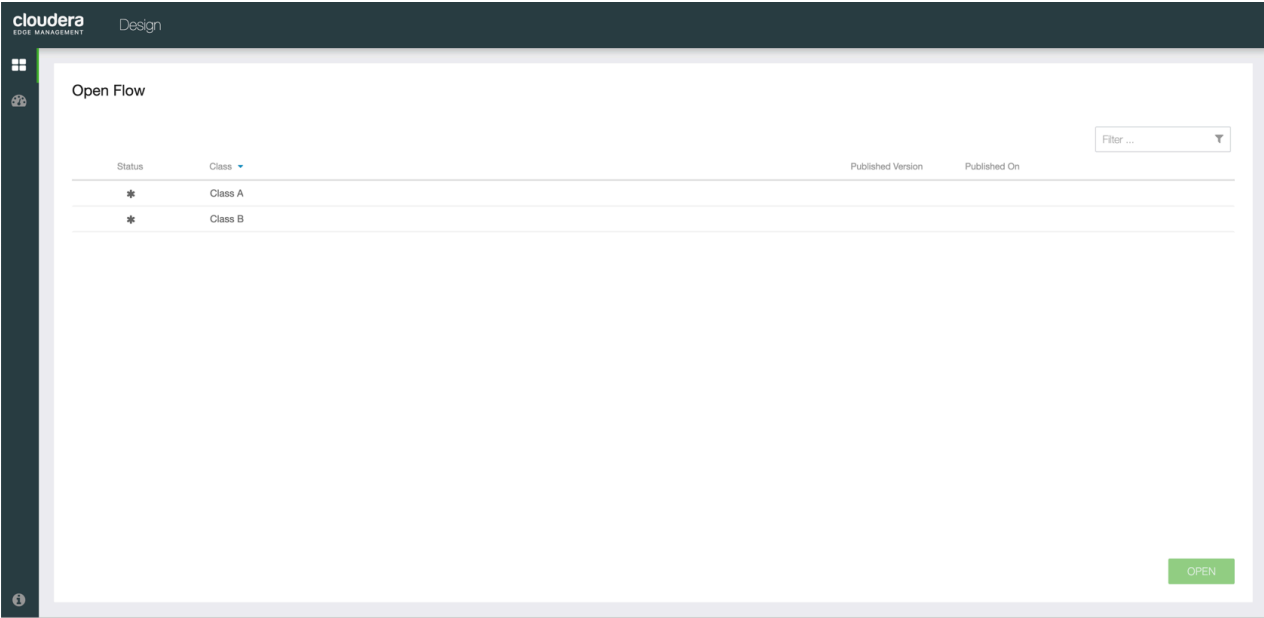
Related Information

Monitoring Events

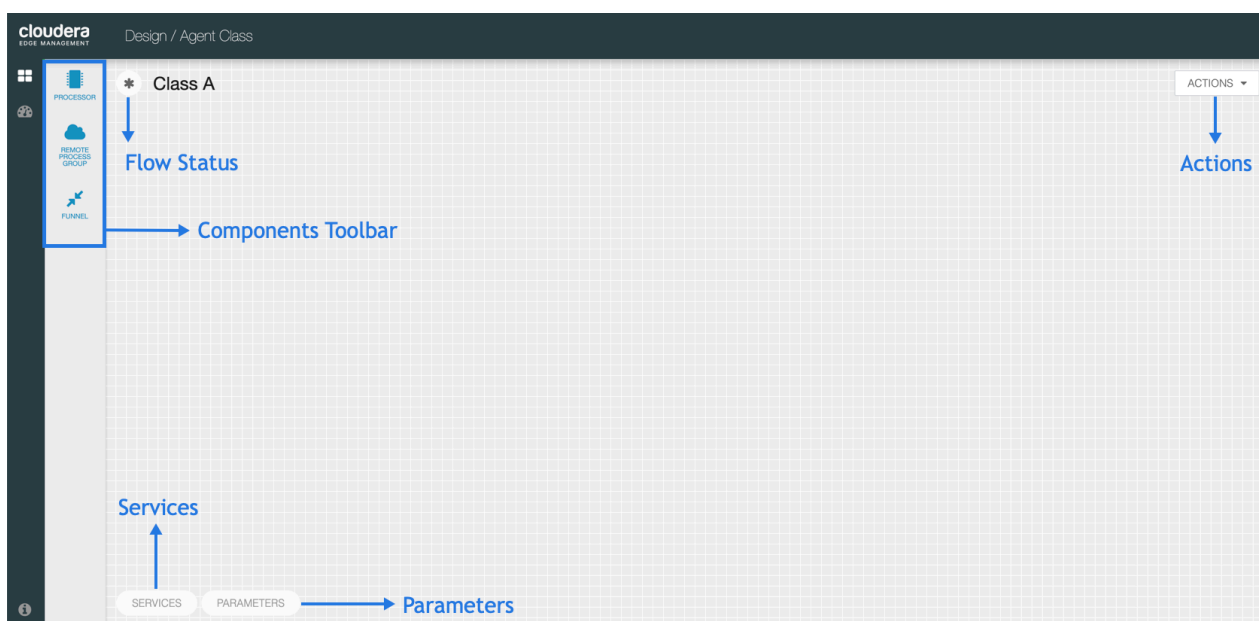
Flow Designer

You can build and publish dataflows by using flow designer.

To navigate to the Flow Designer, click the Design icon (🔧). The Open Flow screen initially appears, as shown in the following image, and displays information about the classes in the system:



To build a dataflow, select the desired class from the table and click OPEN. Alternatively, you can double-click on the desired class. The Design screen appears as shown in the following image:



For more information on how to build a dataflow, see *Building a DataFlow*. For more information on publishing a dataflow, see *Publishing a DataFlow*.

### Open Flow

The Open Flow screen appears initially when you navigate to Flow Designer. It displays the following information about the dataflows and classes in the system:

- Status
- Class
- Published Version
- Published On

You can sort data in each column in ascending or descending order by clicking the column name. For example, you can sort the dataflows based on which class they are associated with by clicking the Class column name.



You can also filter the data by class.

### Design

The Design screen contains the following segments, each responsible for different functionalities of the application:

- Components Toolbar. The components toolbar consists of the components that you can drag onto the canvas to build your dataflow.
- Services. SERVICES are shared services that can be used by processors and other services to utilize for configuration or task execution.
- Parameters. Provides the ability to parameterize the values of processor and service properties in the flow.
- Actions. Use the ACTIONS drop-down to open a different dataflow, publish a dataflow, or revert to the last published version of a dataflow.

- Flow Status. The Flow Status displays information about the flow including when the flow was published, what was the last published version of the flow, and the status of the flow.

-  Modified ( ). The flow has been modified since the flow has been published.
-  Current ( ). The current published version of the flow.



**Note:** Once you open a dataflow, it is the default flow that displays in the Flow Designer. To build or edit another dataflow, select Open from the ACTIONS menu.

Related Information

[Building a DataFlow](#)

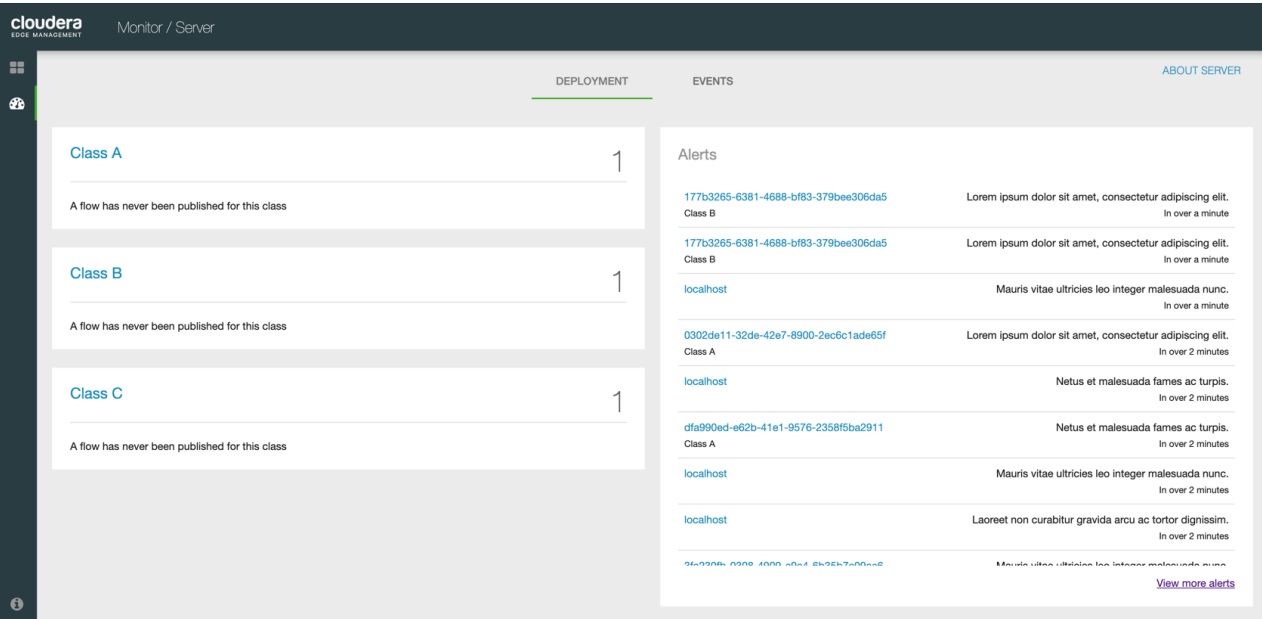
[Publishing a DataFlow](#)

# Monitoring Deployments

This section highlights options, available in the Deployments Monitor screen, that enables you to monitor C2 server and agent deployments.



Click the Monitor icon to navigate to the Deployments Monitor screen. The Deployments Monitor screen appears as shown in the following image:



CEM provides the following details for deployments:

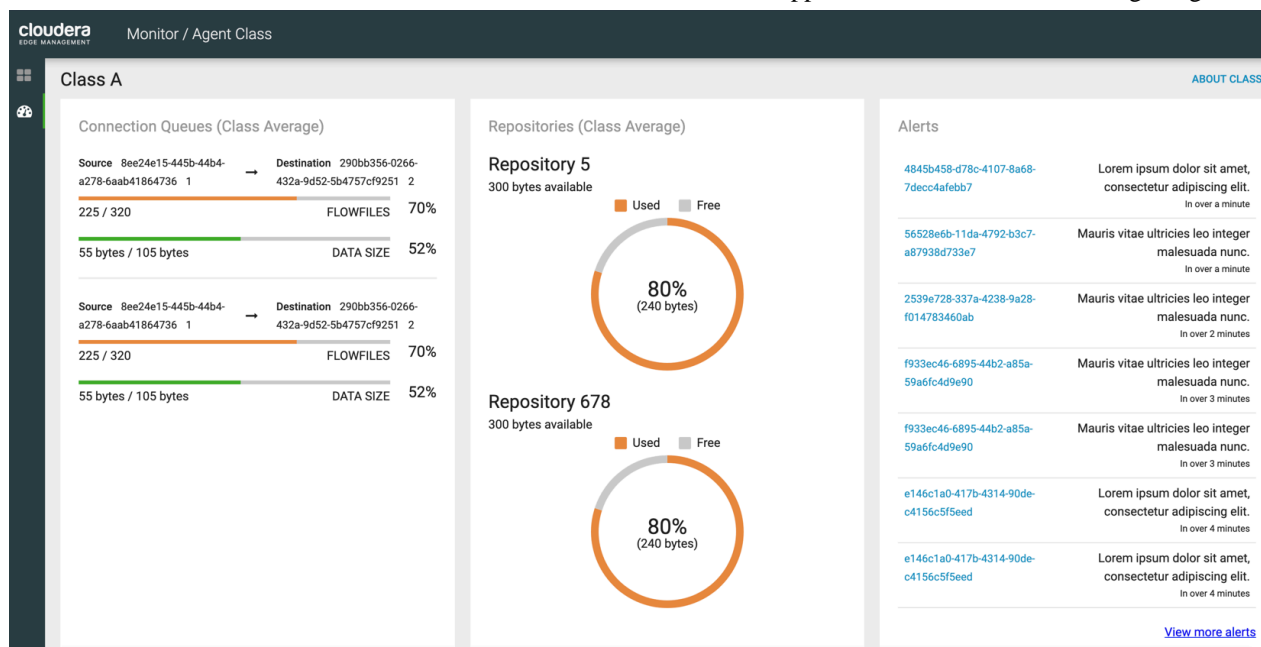
- Classes. The first column, from the left, displays the classes available in the system.
- Alerts. The Alerts column displays the alerts available in the system.
- About Server. Click the About Server link to fetch details about the C2 server. The About This Server dialog displays the server URL, registry URL, registry bucket ID, and registry bucket name.

Click OK to return to the Deployments Monitor screen.

## Monitoring Class Details

The class column displays the class names that also act as links to navigate to more details about the classes. The column also displays the number of agents running in the class beside each class name. You can also view the number of agents that are currently not responding in a class.

Click a class name link to fetch details about the class. The class screen appears as shown in the following image:



The class screen displays the following details:

- Connection Queues (Class Average). Aggregate views of all the connections throughout the instances in the selected class.
- Repositories (Class Average). Usage details of the repositories the selected class is connected to.
- Alerts. Alerts available for the selected class.
- About Class. When you click the About Class link, the About This Class page appears which displays details about the class including flow name, latest version, registry ID, registry bucket, and Grafana dashboard link if you enabled it. For more information, see *Monitoring Metrics with Grafana*.

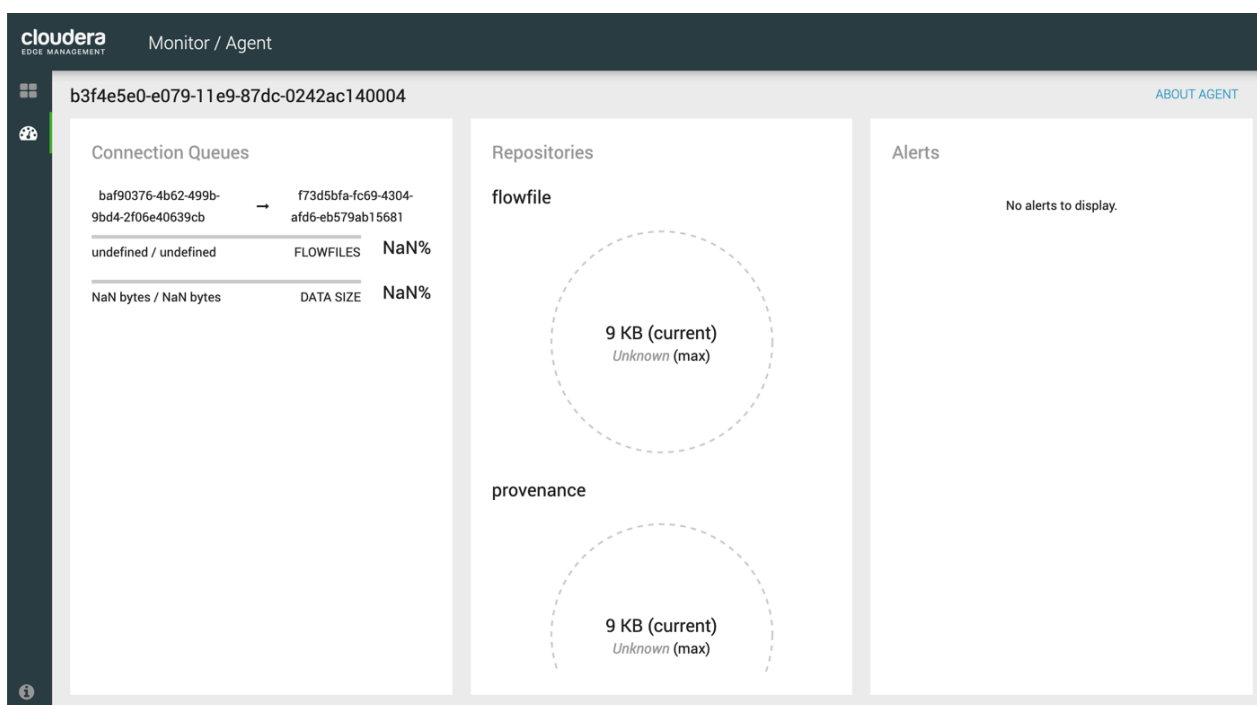
## Monitoring Alert Details

The alert column displays the alert messages.

You can scroll through the Alert column to see more alerts in the system. Click the View more alerts link to navigate to the Events page to monitor C2 server and agent events.

In the Alerts column, click an alert link to fetch details about the alert. The alert screen appears as shown in the following image:





The alert screen displays the following details.

- Connection Queues. Aggregate views of all the connections throughout the instances in the agent.
- Repositories. Usage details of the repositories the agent is connected to.
- Alerts. Alerts available in the agent.
- About Agent. When you click the About Agent link, the About This Agent page appears which displays the following details about the agent of the alert:
  - Last Seen
  - Location
  - Agent Class
  - Flow Name
  - Flow Version
  - Registry ID
  - Bucket Name
  - Device OS
  - Device Architecture
  - Device Memory
  - Device Cores
  - Network Device ID
  - Network Hostname
  - Network IP\0000


### Related Information

[Monitoring Metrics with Grafana](#)

## Monitoring Events

This section highlights options, available in the Events Monitor screen, that enables you to monitor C2 server and agent events.



Click the Monitor icon (  ) and then select Events to navigate to the Events Monitor screen. CEM provides the following details for events:

- Severity
- Event Type
- Message
- Event Source
- Source Type
- Class
- Date/Time

You can display events based on time range (All, Last Hour, Last 4 Hours, Last 24 Hours, Last 7 Days) by selecting the desired Time range value in the drop-down. The number of rows displayed (20, 50, 100) can be configured by selecting the desired value from the Rows per page drop-down. The following image shows the Time range and Rows per page options in the UI:

The screenshot displays the Cloudera Edge Management Events Monitor interface. The top navigation bar includes the Cloudera logo and the text 'Monitor / Server'. Below this, there are tabs for 'DEPLOYMENT' and 'EVENTS', with 'EVENTS' being the active tab. A link for 'ABOUT SERVER' is visible in the top right corner. The main content area is titled 'Events' and contains a table of event data. The table has the following columns: Severity, Event Type, Message, Event Source, Source Type, Class, and Date/Time. The events listed include various types such as 'Heartbeat received', 'Flow Update', 'Agent Status', 'Alert', and 'C2 Server Status'. At the bottom of the table, it indicates '56 new events available' and provides links to 'Reload' or 'Show Latest'. In the bottom right corner, there are controls for 'Time range' (set to 'All') and 'Rows per page' (set to '20').

## Sorting and Filtering

You can sort data in each column in ascending or descending order by clicking the column name. For example, you can sort the events based on class by clicking the Class column name.

You can also filter the events by Severity, Event Type, Message, Event Source, Source Type, and Class. Select the column name in the drop-down box at the top-right corner of the UI, enter the filter value, and select RETURN on the keyboard to apply the filter.

You can use multiple column names and filter values to filter your data. Here is an example of filtering by Event Source and Source Type:

cloudera  
EDGE MANAGEMENT

Monitor / Server

DEPLOYMENT EVENTS ABOUT SERVER

Events

Event Source... Source Type... Filter by: Source Type

Severity	Event Type	Message	Event Source	Source Type	Class	Date/Time
WARN	Background	Laoreet non curabitur gravida arcu ac tortor dignissim.	localhost	Server		Mon Sep 30 2019 13:32:54 ...
DEBUG	Heartbeat	Laoreet non curabitur gravida arcu ac tortor dignissim.	localhost	Server		Mon Sep 30 2019 13:32:54 ...
INFO	Heartbeat	Mauris vitae ultricies leo integer malesuada nunc.	localhost	Server		Mon Sep 30 2019 13:32:54 ...
WARN	Agent Status	Lorem ipsum dolor sit amet, consectetur adipiscing elit.	localhost	Server		Mon Sep 30 2019 13:32:44 ...
DEBUG	Alert	Netus et malesuada fames ac turpis.	localhost	Server		Mon Sep 30 2019 13:32:44 ...
INFO	Agent Status	Lorem ipsum dolor sit amet, consectetur adipiscing elit.	localhost	Server		Mon Sep 30 2019 13:32:04 ...
WARN	Alert	Mauris vitae ultricies leo integer malesuada nunc.	localhost	Server		Mon Sep 30 2019 13:32:44 ...
DEBUG	Background	Mauris vitae ultricies leo integer malesuada nunc.	localhost	Server		Mon Sep 30 2019 13:32:34 ...
WARN	Background	Laoreet non curabitur gravida arcu ac tortor dignissim.	localhost	Server		Mon Sep 30 2019 13:32:14 ...
DEBUG	Agent Status	Netus et malesuada fames ac turpis.	localhost	Server		Mon Sep 30 2019 13:32:14 ...
WARN	Alert	Mauris vitae ultricies leo integer malesuada nunc.	localhost	Server		Mon Sep 30 2019 13:32:04 ...
INFO	Alert	Lorem ipsum dolor sit amet, consectetur adipiscing elit.	localhost	Server		Mon Sep 30 2019 13:32:04 ...
INFO	Alert	Netus et malesuada fames ac turpis.	localhost	Server		Mon Sep 30 2019 13:31:54 ...

Time range: All Rows per page: 20 1-20 of 741

After you filter the event details as per your requirement, you can share the URL with other users who can then view your filtered event list.

## Reload and Show Latest

You can view new events in the system by selecting either the Reload or Show Latest links. Reload refreshes events using the existing search and sort criteria. Show Latest reloads events but sorts by the latest events using the existing search criteria.

The following image shows the Reload and Show Latest links in the Events screen:

Events

Filter by: Severity

Severity	Event Type	Message	Event Source	Source Type	Class	Date/Time
INFO	Operation Updated	C2 operation state changed from QUEUED to DEPLOYED: UPD...	0.0.0.0	Server		Thu Apr 04 2019 11:41:14 G...
DEBUG	Heartbeat Receiv...	Heartbeat received.	syslog-agent-1	Agent		Thu Apr 04 2019 11:41:15 G...
DEBUG	Heartbeat Receiv...	Heartbeat received.	gizmo-agent	Agent		Thu Apr 04 2019 11:41:14 G...
DEBUG	Heartbeat Receiv...	Heartbeat received.	minifi-jpi-1	Agent		Thu Apr 04 2019 11:41:14 G...
DEBUG	Heartbeat Receiv...	Heartbeat received.	appserver-agent	Agent		Thu Apr 04 2019 11:41:14 G...
DEBUG	Heartbeat Receiv...	Heartbeat received.	webserver-agent-1	Agent		Thu Apr 04 2019 11:41:12 G...
INFO	Operation Updated	C2 operation state changed from DEPLOYED to DONE: UPDAT...	0.0.0.0	Server		Thu Apr 04 2019 11:41:12 G...
INFO	Operation Created	C2 operation created: UPDATE configuration location=flows/a...	server-id	Server		Thu Apr 04 2019 11:41:12 G...
DEBUG	Heartbeat Receiv...	Heartbeat received.	appserver-agent	Agent		Thu Apr 04 2019 11:41:09 G...
INFO	Operation Updated	C2 operation state changed from QUEUED to DEPLOYED: UPD...	0.0.0.0	Server		Thu Apr 04 2019 11:41:09 G...
DEBUG	Heartbeat Receiv...	Heartbeat received.	syslog-agent-1	Agent		Thu Apr 04 2019 11:41:09 G...
DEBUG	Heartbeat Receiv...	Heartbeat received.	minifi-jpi-1	Agent		Thu Apr 04 2019 11:41:09 G...
DEBUG	Heartbeat Receiv...	Heartbeat received.	gizmo-agent	Agent		Thu Apr 04 2019 11:41:09 G...
INFO	Operation Created	C2 operation created: UPDATE configuration location=flows/a...	server-id	Server		Thu Apr 04 2019 11:41:07 G...
DEBUG	Heartbeat Receiv...	Heartbeat received.	webserver-agent-1	Agent		Thu Apr 04 2019 11:41:07 G...
INFO	Operation Updated	C2 operation state changed from DEPLOYED to DONE: UPDAT...	0.0.0.0	Server		Thu Apr 04 2019 11:41:07 G...

1968 new events available Reload or Show Latest Time range: All Rows per page: 20 1-20 of 1996

## Event Details

You can find the details of events by clicking the Event Details icon (ⓘ) beside each event. The following image shows event details of the event selected in the Events section:

The screenshot shows the Cloudera Edge Management interface for monitoring a server. The 'EVENTS' tab is active, displaying a table of events. The table has columns for Severity, Event Type, Message, Event Source, Source Type, Class, and Date/Time. Three events are listed: a WARN event for Background, a DEBUG event for Heartbeat, and an INFO event for Heartbeat. The INFO event is selected, and its details are shown in a modal window below. The details include a JSON object with fields like id, level, eventType, message, created, and eventSource.

Severity	Event Type	Message	Event Source	Source Type	Class	Date/Time
WARN	Background	Laoreet non curabitur gravida arcu ac tortor dignissim.	localhost	Server		Mon Sep 30 2019 13:32:54 ...
DEBUG	Heartbeat	Laoreet non curabitur gravida arcu ac tortor dignissim.	localhost	Server		Mon Sep 30 2019 13:32:54 ...
INFO	Heartbeat	Mauris vitae ultricies leo integer malesuada nunc.	localhost	Server		Mon Sep 30 2019 13:32:54 ...



Event Details

```
{
  "id": "175b5b5b-130b-41a7-942a-e5d49c9d3fe7",
  "level": "INFO",
  "eventType": "Heartbeat",
  "message": "Mauris vitae ultricies leo integer malesuada nunc.",
  "created": "2019-09-30T08:02:54.712Z",
  "eventSource": {
    "type": "Server",
    "id": "localhost"
  }
}
```

## Expression Language

The NiFi Expression Language provides the ability to reference the attributes of FlowFiles, compare them to other values, and manipulate their values.

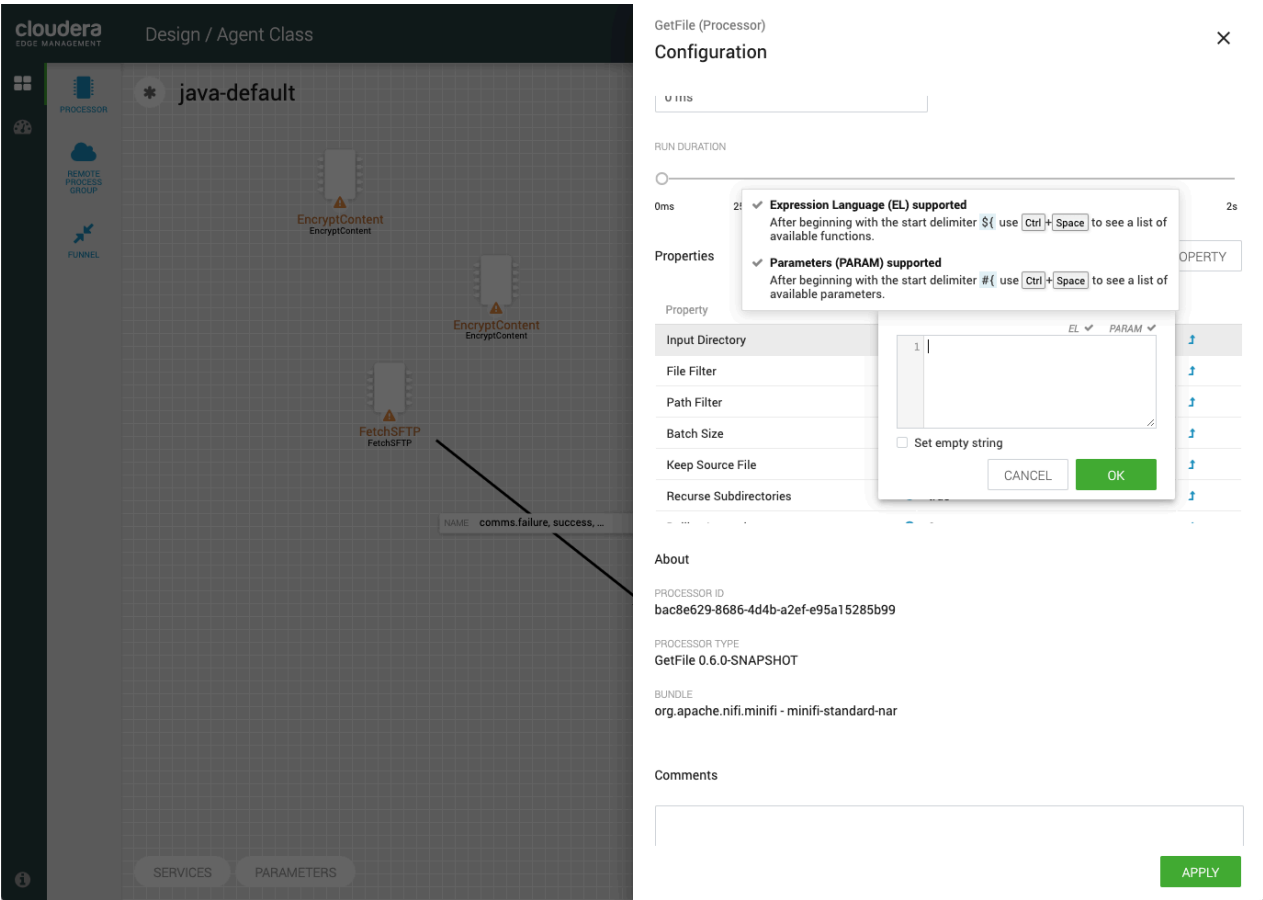
As you extract attributes from content of the FlowFiles and add user-defined attributes, they do not help much as an operator unless you have some mechanism by which you can use them. The NiFi Expression Language allows you to access and manipulate FlowFile attribute values as you configure your flows. Not all processor properties allow the Expression Language to be used, but many do. In order to determine whether or not a processor or service property supports the Expression Language, click in the property value field. The eligibility indicators show if Expression

Language is supported (  ) or unsupported (  ). Additionally, you can hover over the Help icon next to the property name. The Help icon provides a tooltip that displays the Expression Language scope.

To configure an eligible property to utilize Expression Language, use the \$ symbol as the start, with the expression enclosed in curly braces:

```
${Expression}
```

If you start your new entry with the start delimiter \$, selecting the keystroke control+space shows a list of available functions. Help text describing this process appears when you hover over the Expression Language eligibility indicator.



An expression can be as simple as an attribute name. For example, to reference the uuid attribute, you can simply use the value `${uuid}`. If the attribute name begins with any character other than a letter, or if it contains a character other than a number, a letter, a period (`.`), or an underscore (`_`), you need to quote the attribute name. For example, `${My Attribute Name}` is not valid, but `${'My Attribute Name'}` refers to the My Attribute Name attribute.

## Parameters

Parameters provide the ability to parameterize the values of processors and service properties in the flow including sensitive properties. Parameters are created and configured within the CEM UI.

To determine whether a parameter can be used for a property, click in the property value field. The eligibility

indicators show if parameters are supported ( ✓ ) or unsupported ( ✗ ).

cloudera  
EDGE MANAGEMENT

Design / Agent Class

PROCESSOR

REMOTE PROCESS GROUP

FUNNEL

java-default

EncryptContent  
EncryptContent

FetchSFTP  
FetchSFTP

NAME: comms.failure, success, ...

SERVICESPARAMETERS

GetFile (Processor)

Configuration

U 1115s

RUN DURATION

0ms2s

Expression Language (EL) supported

After beginning with the start delimiter \$ use Ctrl+Space to see a list of available functions.

Parameters (PARAM) supported

After beginning with the start delimiter # use Ctrl+Space to see a list of available parameters.

Properties

Property

Input Directory

File Filter

Path Filter

Batch Size

Keep Source File

Recurse Subdirectories

About

PROCESSOR ID

bac8e629-8686-4d4b-a2ef-e95a15285b99

PROCESSOR TYPE


GetFile 0.6.0-SNAPSHOT

BUNDLE

org.apache.nifi.minifi - minifi-standard-nar

Comments

APPLY

 **Note:** Properties that reference services can not use parameters.

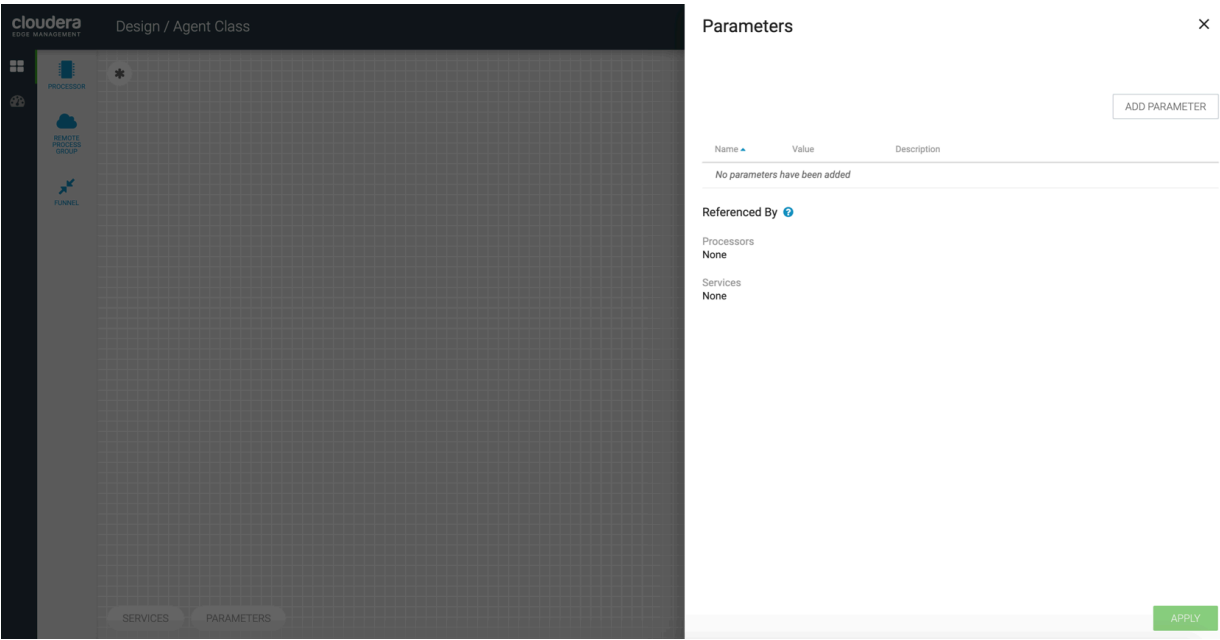
### Adding Parameters

This section describes how to add parameters.

Procedure

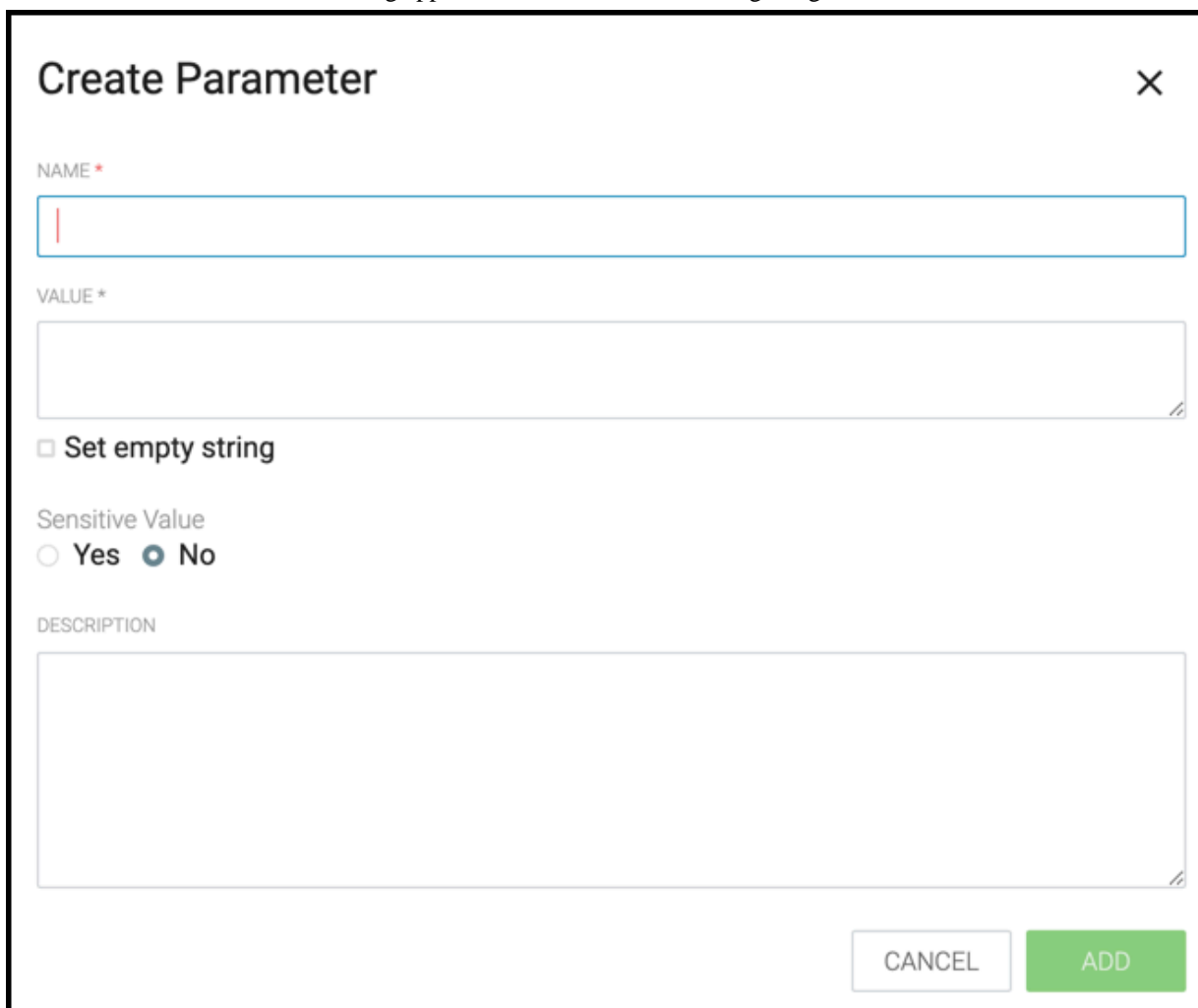
- 1. To add a parameter, click the Parameters button at the bottom-left corner of the canvas, or simply right-click on the canvas and select Parameters.

The Parameters window appers as shown in the following image:



- Click the ADD PARAMETER button.

The Create Parameter window dialog appears as shown in the following image:



The image shows a 'Create Parameter' dialog box with a title bar and a close button (X). It contains the following fields and controls:

- NAME \***: A text input field with a red vertical line indicating the start of the input.
- VALUE \***: A text input field.
- ☐ **Set empty string**: A checkbox.
- Sensitive Value**: A label above two radio buttons.
  - ☐ **Yes**
  - ☒ **No**
- DESCRIPTION**: A large text area.
- CANCEL** and **ADD** buttons at the bottom right.

- Configure the following properties:

Property	Description
Name	Enter a name that is used to denote a parameter. Only alpha-numeric characters (a-z, A-Z, 0-9), hyphens ( - ), underscores ( _ ), periods ( . ), and spaces are allowed.
Value	Enter a value that is used when the parameter is referenced. Parameter values do not support Expression Language or embedded parameter references.
Set empty string	Select the checkbox to set parameter value to an empty string.
Sensitive Value	Set whether the parameter value should be considered sensitive. If so, the value of the parameter is not shown in the UI once applied. Once you create a parameter, its sensitivity flag cannot be changed.
Description	Enter a description that explains what the parameter is for. This field is optional.

- Click ADD button to add the parameter.
- Click APPLY button to save the changes.



## Referencing Parameters

This section describes how to reference parameters.

To configure an eligible property to reference a parameter, use the # symbol as the start, with the name of the parameter enclosed in curly braces:

```
# {Parameter.Name}
```

This can be escaped using an additional # character at the beginning. To illustrate this, assume that the parameter abc has a value of xxx and parameter def has a value of yyy. Then, the following user-defined property values will evaluate to these effective values:

User-Entered Literal Property Value	Effective Property Value	Explanation
# {abc}	xxx	Simple substitution
# {abc}/data	xxx/data	Simple substitution with additional literal data
# {abc}/# {def}	xxx/yyy	Multiple substitution with additional literal data
# {abc	# {abc	No { } for parameter replacement
#abc	#abc	No { } for parameter replacement
## {abc}	# {abc}	Escaped # for literal interpretation
### {abc}	#xxx	Escaped # for literal interpretation, followed by simple substitution
#### {abc}	## {abc}	Escaped # for literal interpretation, twice
##### {abc}	##xxx	Escaped # for literal interpretation, twice, followed by simple substitution
# {abc/data}	Exception thrown on property set operation	/ not a valid parameter name character

When referencing a parameter from within expression language, the parameter reference is evaluated first. As an example, to replace xxx with zzz for the abc parameter:

```
${ # {abc} : replace ( 'xxx' , 'zzz' ) }
```

## Using Parameters

This section describes how to use parameters.

Parameters can be easily referenced or created as you configure the components in your flow.

### Referencing existing parameters

Existing parameters can be referenced for a processor or service property value.

1. To reference an existing parameter, select the property value field and clear the default value if one exists.

2. Enter the start delimiter `#{`.

### FetchFile (Processor)

## Configuration

Timer Driven

10

RUN SCHEDULE \*

0 ms

RUN DURATION

0ms25ms50ms100ms250ms500ms1s2s

Properties

ADD PROPERTY

Property	Value
File to Fetch	<div><div>?</div><div></div><div>→</div></div>
Completion Strategy	
Move Destination Directory	
Move Conflict Strategy	
Log level when file not found	
Log level when permission denied	

1

`#{`

EL ✓PARAM ✓

☐ Set empty string

CANCEL

OK

About

PROCESSOR ID

f7f4d523-8a0b-48ab-a577-bc6cfc3003e9

PROCESSOR TYPE

FetchFile 0.6.0-SNAPSHOT

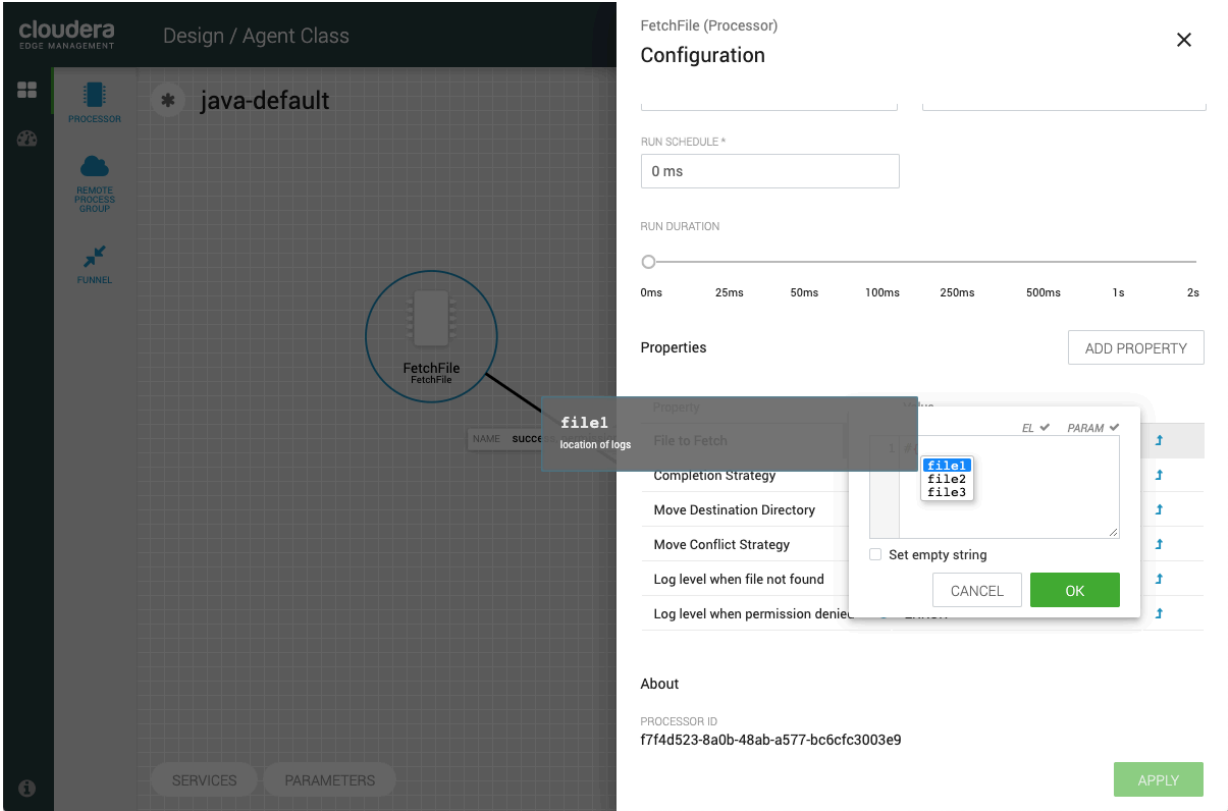
BUNDLE

org.apache.nifi.minifi - minifi-standard-nar

Comments

APPLY

3. If you know the parameter you want to reference, enter its name. Alternatively, select the keystroke control+space to show the list of available parameters:



4. Complete the entry with a closing curly brace `}` and select OK.

### FetchFile (Processor) ✕

## Configuration

RUN SCHEDULE \*

0 ms

RUN DURATION

0ms25ms50ms100ms250ms500ms1s2s

### Properties ADD PROPERTY

Property	Value	EL ✓	PARAM ✓
File to Fetch	1 <code>#file1</code>		
Completion Strategy			
Move Destination Directory			
Move Conflict Strategy			
Log level when file not found			
Log level when permission denied			

### About

PROCESSOR ID

**f7f4d523-8a0b-48ab-a577-bc6cfc3003e9**

APPLY

1 `#file1`

☐ Set empty string

CANCEL

OK

5. Click APPLY to save the changes.

Help text describing this process appears when you hover over the Expression Language and Parameters eligibility indicators.

The screenshot shows the 'FetchFile (Processor) Configuration' dialog. It has a sidebar with configuration options: 'File to Fetch', 'Completion Strategy', 'Move Destination Directory', 'Move Conflict Strategy', 'Log level when file not found', and 'Log level when permission denied'. The 'File to Fetch' option is selected. The main area shows 'RUN SCHEDULE \*' set to '0 ms' and 'RUN DURATION' set to '0ms'. A help tooltip is displayed over the 'PARAM' eligibility indicator, showing the following text:

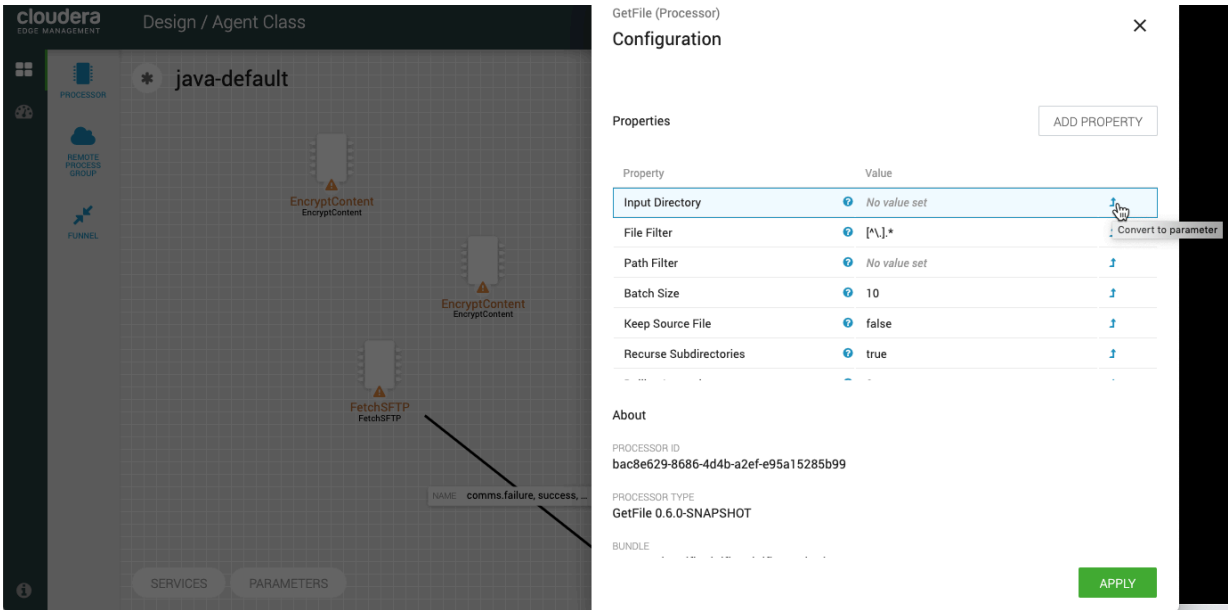
- ✓ **Expression Language (EL) supported**  
After beginning with the start delimiter `${` use `Ctrl + Space` to see a list of available functions.
- ✓ **Parameters (PARAM) supported**  
After beginning with the start delimiter `#{` use `Ctrl + Space` to see a list of available parameters.

The tooltip also shows a code editor with the text `1 #{file1}` and a checkbox labeled 'Set empty string'. The 'OK' button is highlighted in green. At the bottom right of the dialog is an 'APPLY' button.

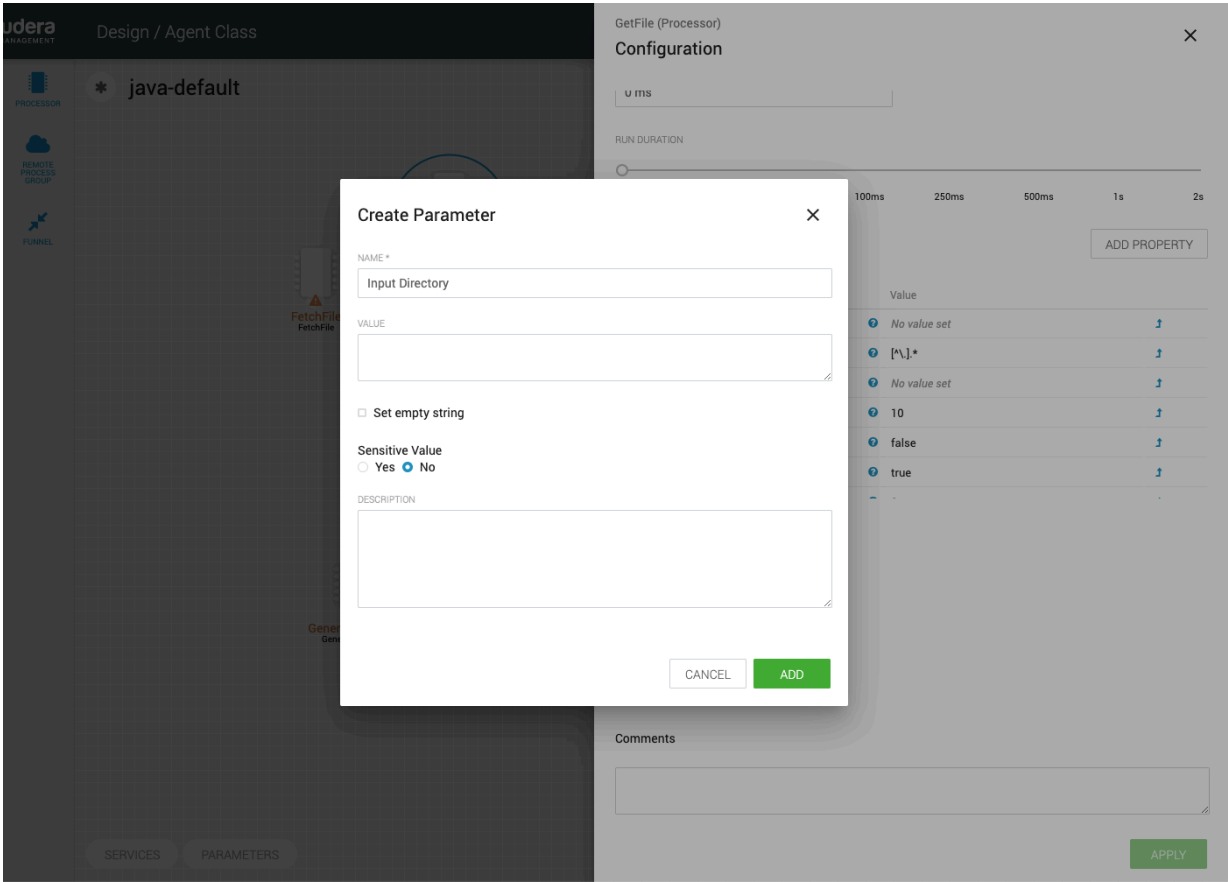
Converting property values to parameters

Parameters can be created during process or service configuration. Instead of entering a property value, you can convert the property value to a parameter.

- 1. To create a parameter for a property value, select the Convert to parameter icon (  ) in the row of that property.



The Create Parameter dialog appears as shown in the following image:



- 2. Configure the parameter properties.

- Click ADD to add the parameter. The property references the new parameter with the proper syntax applied automatically.

The screenshot displays the Cloudera Edge Management interface. On the left, a workflow design is visible with a 'java-default' agent class. A 'FetchFile' processor is connected to a 'GenerateFlowFile' processor. A 'GetFile' processor is also shown, with its configuration panel open on the right. The configuration panel for 'GetFile (Processor)' includes a 'Configuration' section with a 'Run Duration' slider set to 10ms. The 'Properties' section contains a table with the following data:

Property	Value
Input Directory	#(Input Directory)
File Filter	['\']*
Path Filter	No value set
Batch Size	10
Keep Source File	false
Recurse Subdirectories	true

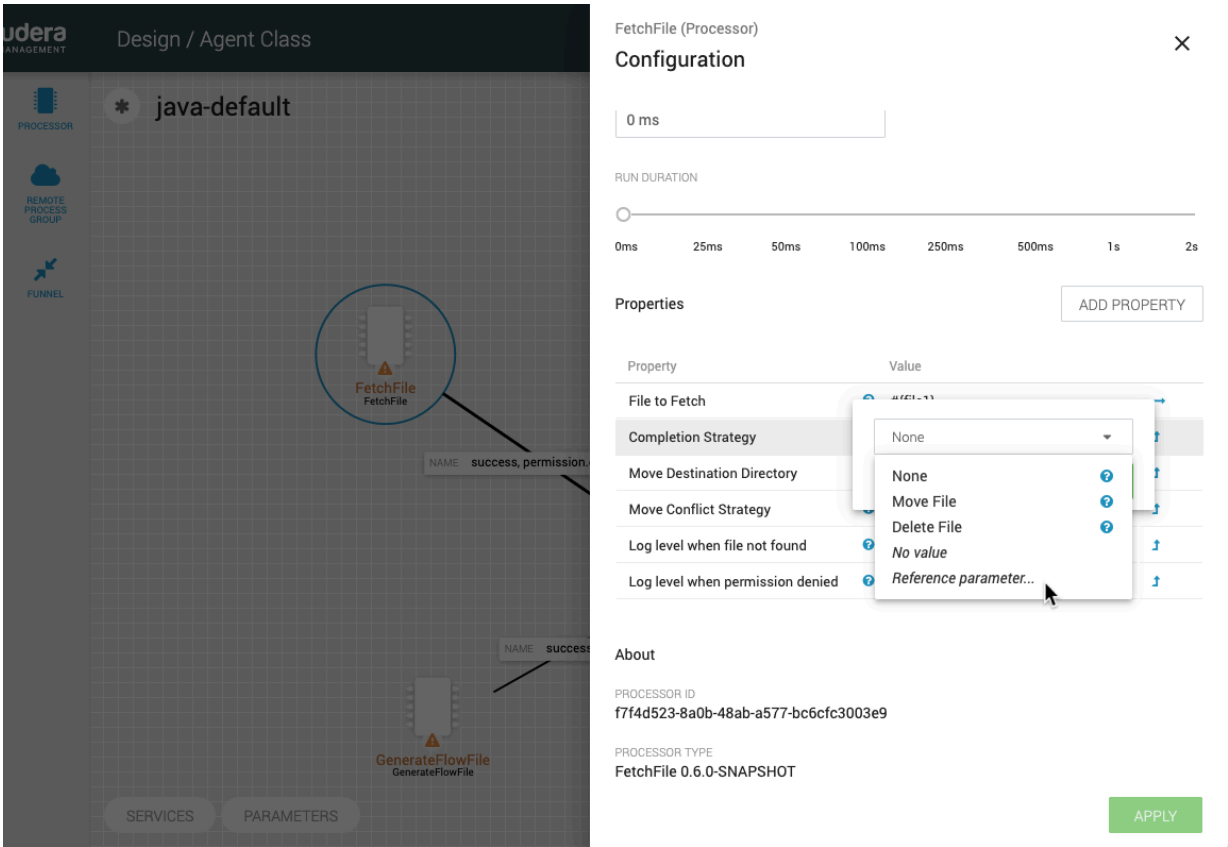
The 'About' section provides details about the processor: Processor ID is bb54c8eb-fafd-44ac-a6a0-cb6163f89cc0, Processor Type is GetFile 0.6.0-SNAPSHOT, and Bundle is org.apache.nifi.minifi - minifi-standard-nar. A green 'APPLY' button is located at the bottom right of the configuration panel.

- Click APPLY to save the changes.

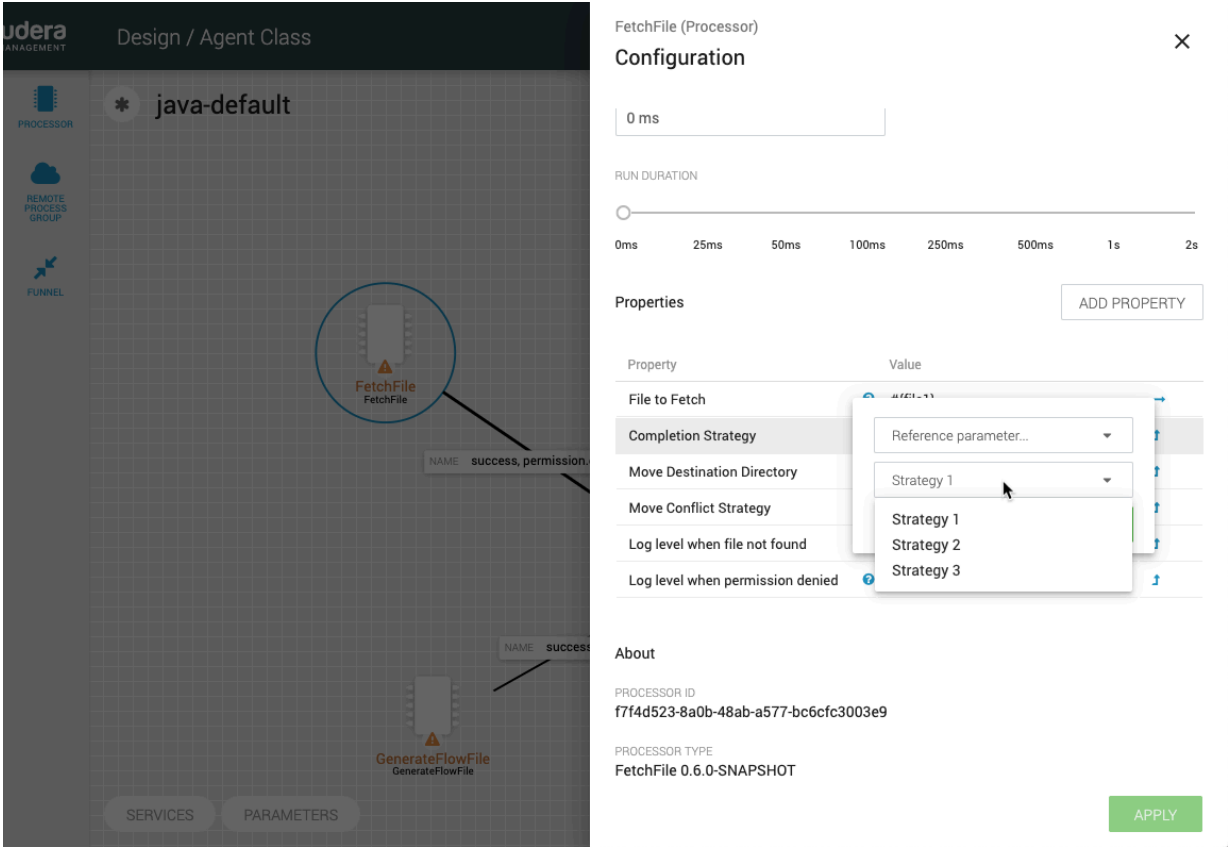
### Converting selectable property values to parameters

Property values that are selectable can also reference parameters.

- 1. Select the property value drop-down. The option Reference parameter... is available for eligible properties.



- 2. Select Reference parameter... option. A list of parameters to choose appears.

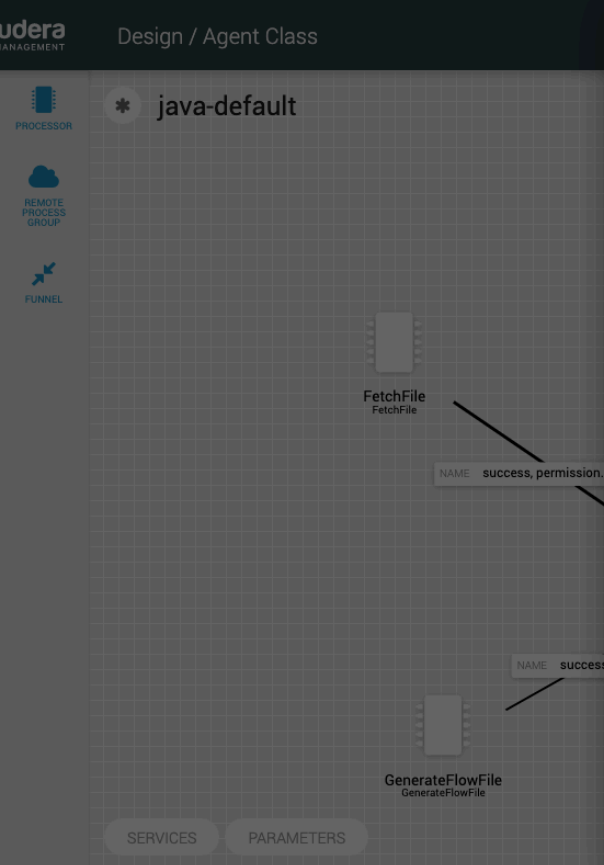




3. Select a parameter and click OK.
4. Click APPLY to save the changes.

### Referenced parameters

When parameters have been referenced in the flow, the Parameters window lists all the components that use them in the Referenced By section.



The screenshot shows the Cloudera Edge Management interface. On the left, there's a sidebar with icons for PROCESSOR, REMOTE PROCESS GROUP, and FUNNEL. The main area is titled 'Design / Agent Class' and shows a flow design with components like 'FetchFile' and 'GenerateFlowFile'. The 'Parameters' window is open on the right, displaying a table of parameters and their values. The 'Referenced By' section lists the processors and services that reference the selected parameter.

Name	Value	Description
Batch Size	1	
Password1	*****	
Password2	23	
file1	/something/file...	location of logs

**Referenced By**

Processors  
[FetchSFTP](#); [FetchFile](#); [GenerateFlowFile](#)

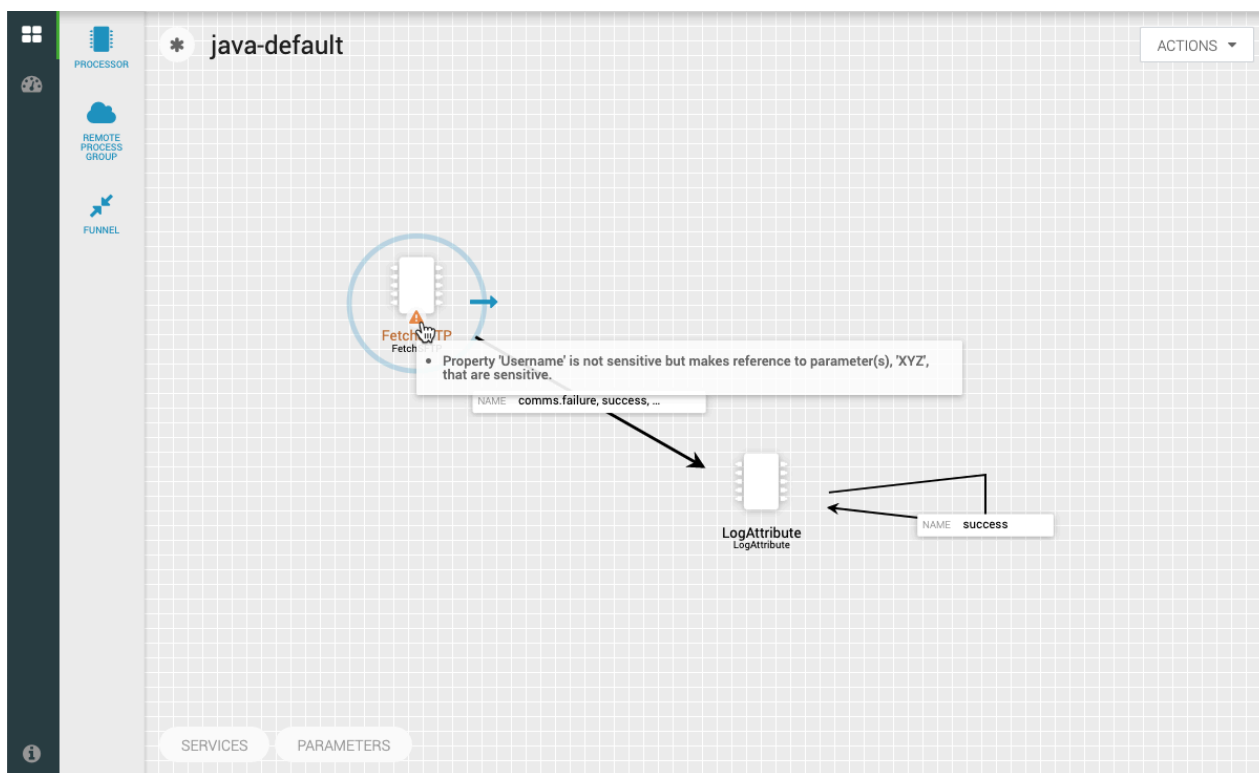
Services  
[StandardRestrictedSSLContextService](#)

Select a specific parameter to see the processors and services that reference just that parameter.

### Using parameters with sensitive properties

Non-sensitive properties should only reference non-sensitive parameters. Sensitive properties should only reference sensitive parameters. This is important when publishing versioned flows. The value of the sensitive parameter itself is not sent to the flow registry, but only the fact that the property references the sensitive parameter.

If a non-sensitive property references a sensitive parameter, or vice-versa, the UI marks the component as invalid:



## Building a DataFlow

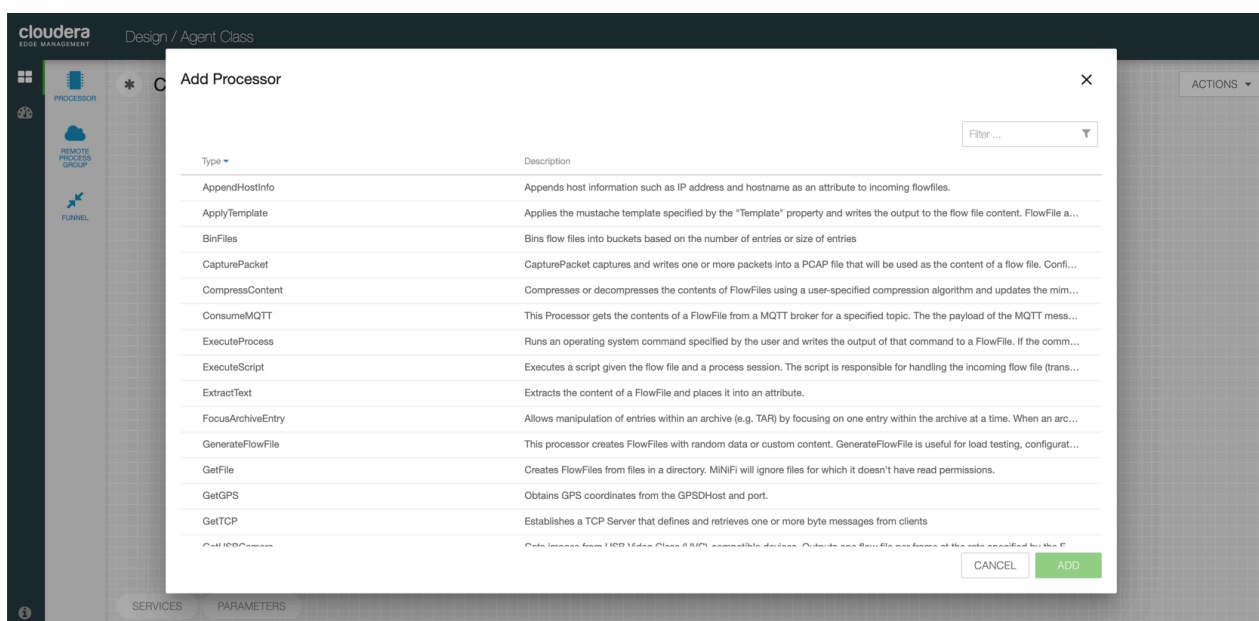
You can build an automated dataflow using the CEM UI. Simply drag components from the toolbar to the canvas, configure the components to meet specific needs, and connect the components together.

### Adding Components to the Canvas

This section describes each of the components available in the Components Toolbar.

#### Processor

The processor is the most commonly used component, as it is responsible for data ingress, egress, routing, and manipulating. There are many different types of processors. When you drag a processor onto the canvas, the Add Processor dialog appears, as shown in the following image, which allows you to choose which type of processor to use:

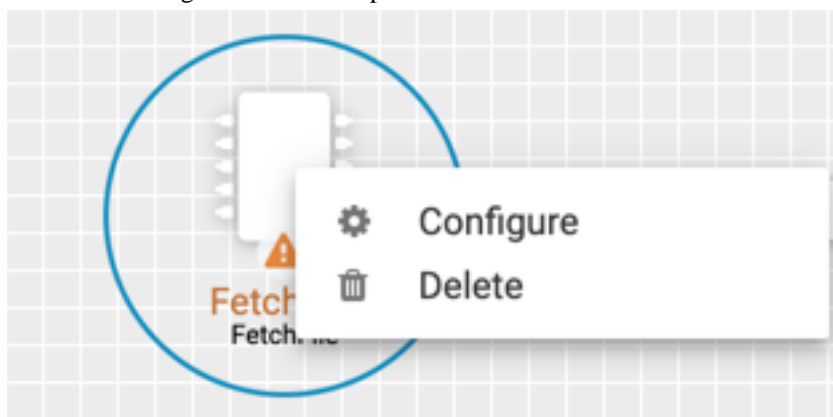


You can filter the list based on the processor type by using the Filter field at top-right corner of the Add Processor dialog. After selecting a processor, you can click the ADD button to add the selected processor to the canvas at the location that it was dropped. Alternatively, you can double-click on a processor type.



**Note:** For any component added to the canvas, it is possible to select it with the mouse and move it anywhere on the canvas. Also, it is possible to move all items at once by clicking and dragging the mouse on the canvas.

After you drag a processor onto the canvas, you can configure properties of the processor, parameterize processor property values, or delete the processor. To configure properties, double-click on the processor, or right-click on the processor and select Configure from the context menu. To delete a processor, right-click on the processor and select Delete from the context menu, or highlight the processor and select DELETE on your keyboard. The following image shows the Configure and Delete options in the context menu:

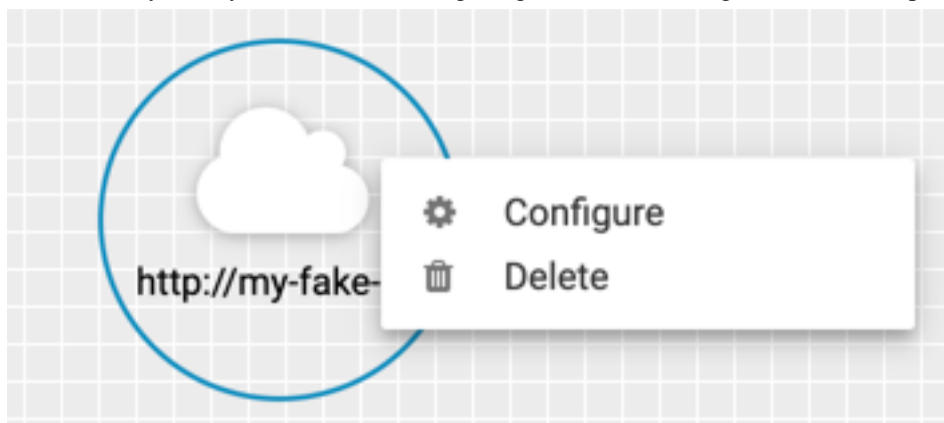


## Remote Process Group

A Remote Process Group (RPG) references a remote instance of NiFi. When you drag an RPG onto the canvas, rather than being prompted for a name, you are prompted for the URL of the remote NiFi instance. If the remote NiFi is clustered, you need to provide at least one URL of any NiFi instance in that cluster. When data is transferred from an RPG running in MiNiFi, the RPG first connects to the remote instance whose URL is configured to determine which nodes are in the cluster and how busy each node is. This information is then used to load balance the data that is pushed to each node. The remote instances are then interrogated periodically to determine information about any nodes that are dropped from or added to the cluster and to recalculate the load balancing based on the load of each node. If the cluster node specified in the URL is down, the RPG cannot establish a connection with the cluster. To

mitigate this scenario, you can enter multiple URLs, allowing the RPG to establish a connection with more than one node.

After you drag an RPG onto the canvas, you can configure settings of the RPG or delete the RPG. To configure settings, double-click on the RPG, or right-click on the RPG and select Configure from the context menu. To delete an RPG, right-click on the processor and select Delete from the context menu, or highlight the RPG and select DELETE on your keyboard. The following image shows the Configure and Delete options in the context menu:



### Funnel

Funnels are used to combine data from many connections into a single connection. If many connections are created with the same destination, the canvas can become cluttered if those connections have to span a large space. By funneling these connections into a single connection, that single connection can then be drawn to span that large space instead.

To delete a funnel, right-click on the funnel and select Delete from the context menu, or highlight the funnel and select DELETE on your keyboard.

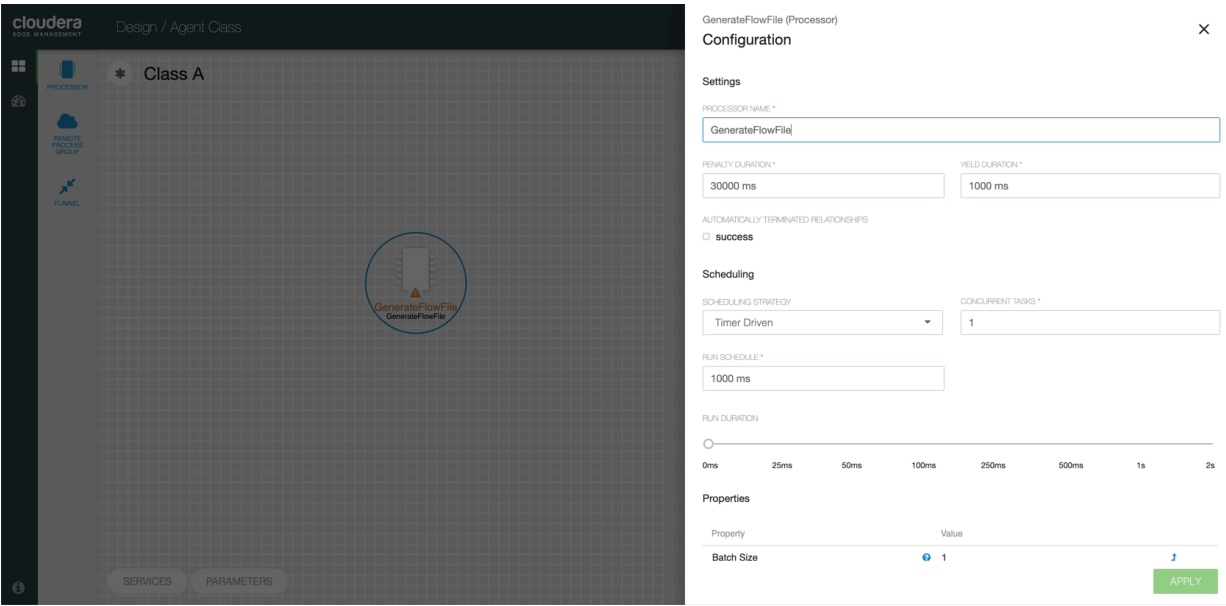
## Configuring a Processor

This section describes how to configure a processor.

Procedure

- 1. To configure a processor, right-click on the processor and select the Configure option.  
Alternatively, just double-click on the processor.

The Configuration dialog opens as shown in the following image:



The Configuration dialog contains the following sections:

- Settings. The Settings section contains the following configuration items:

Properties	Description
Processor Name	Allows you to change the name of the processor. The name of a processor by default is the same as the processor type.
Penalty Duration	The amount of time used when a processor penalizes a FlowFile. During the normal course of processing a piece of data (a FlowFile), an event might occur that indicates that the data cannot be processed at this time but the data might be processable at a later time. When this occurs, the processor might choose to penalize the FlowFile. This prevents the FlowFile from being processed for some period of time. For example, if the processor needs to push the data to a remote service, but the remote service already has a file with the same name as the filename that the processor is specifying, the processor might penalize the FlowFile. The penalty duration allows you to specify how long the FlowFile must be penalized. The default value is 30,000 milliseconds.
Yield Duration	When a processor yields, the amount of time that elapses before the processor is re-scheduled is the yield duration. A processor might determine that some situation exists such that the processor can no longer make any progress, regardless of the data that it is processing. For example, if a processor needs to push data to a remote service and that service is not responding, the processor cannot make any progress. As a result, the processor must yield, which prevents the processor from being scheduled to run for some period of time. The default value is 1,000 milliseconds.

Properties	Description
Automatically Terminated Relationships	Each of the relationships that is defined by the processor is listed here. In order for a processor to be considered valid, each relationship defined by the processor must be either connected to a downstream component or auto-terminated. If a relationship is auto-terminated, any FlowFile that is routed to that relationship is removed from the flow and its processing is considered as complete.

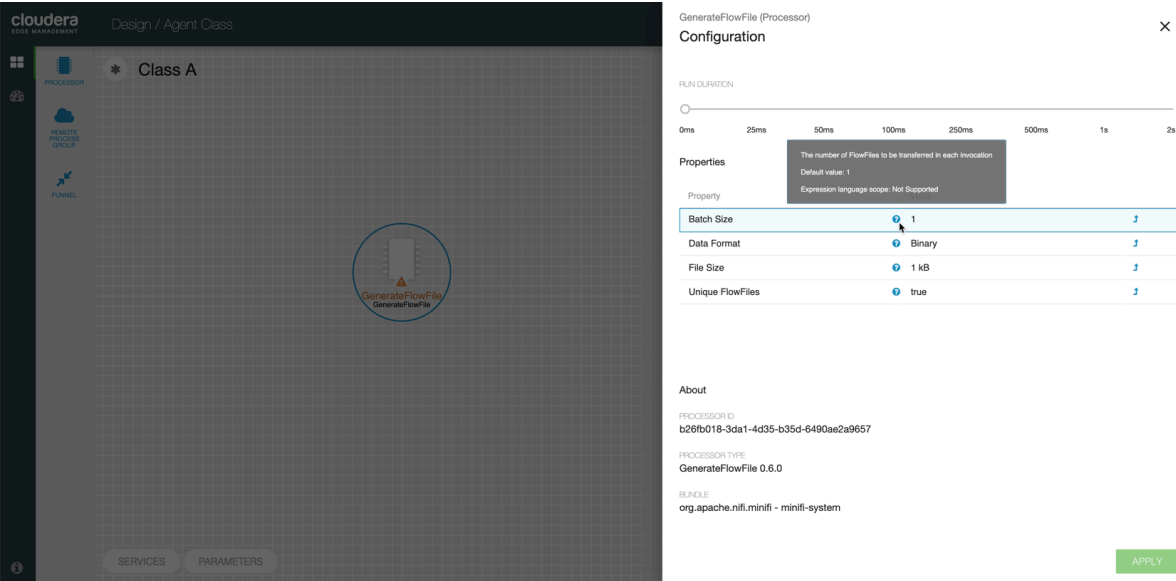
- **Scheduling.** The Scheduling section contains the following configuration items:

Properties	Description
Scheduling Strategy	There are two options for scheduling components: <ul style="list-style-type: none"> <li>• <b>Timer Driven.</b> This is the default mode. The processor is scheduled to run on a regular interval. The interval at which the processor runs is defined by the Run Schedule option (see below).</li> <li>• <b>Event Driven.</b> When this mode is selected, the processor is triggered to run by an event, and that event occurs when FlowFiles enter connections feeding this processor. This mode is currently considered experimental and is not supported by all processors. When this mode is selected, the Run Schedule option is not configurable, as the processor is not triggered to run periodically but as the result of an event.</li> </ul>
Concurrent Tasks	This controls how many threads the processor uses or how many FlowFiles must be processed by this processor at the same time. Increasing this value allows the processor to handle more data in the same amount of time. However, it does this by using system resources that then are not usable by other processors. This essentially provides a relative weighing of processors. For example, it controls how much resources of the system must be allocated to this processor instead of other processors. This field is available for most processors. There are, however, some types of processors that can only be scheduled with a single concurrent task.
Run Schedule	This dictates how often the processor must be scheduled to run. The valid values for this field depend on the selected scheduling strategy (see above). When you select the Event Driven scheduling strategy, this field is not available. When you select the Timer Driven scheduling strategy, this value is a time duration specified by a number followed by a time unit, for example, 1 second or 5 mins. A value of 0 second means that the processor must run as often as possible as long as it has data to process. This is true for any time duration of 0, regardless of the time unit (for example, 0 sec, 0 mins, 0 days).
Run Duration	This slider controls how long the processor must be scheduled to run each time it is triggered. When a processor finishes running, it must update the repository in order to transfer the FlowFiles to the next connection. Updating the repository is expensive, so the more work that can be done at once before updating the repository, the more work the processor can handle (higher throughput). However, this means that the next processor cannot start processing those FlowFiles until the previous process updates this repository. As a result, the latency (the time required to process the FlowFile from beginning to end) becomes longer. As a result, the slider provides a spectrum from which you can choose to favor Lower Latency or Higher Throughput.

- **Properties.** The Properties section provides a mechanism to configure processor-specific behavior. There are no default properties. Each type of processor must define which properties make sense for its use case.

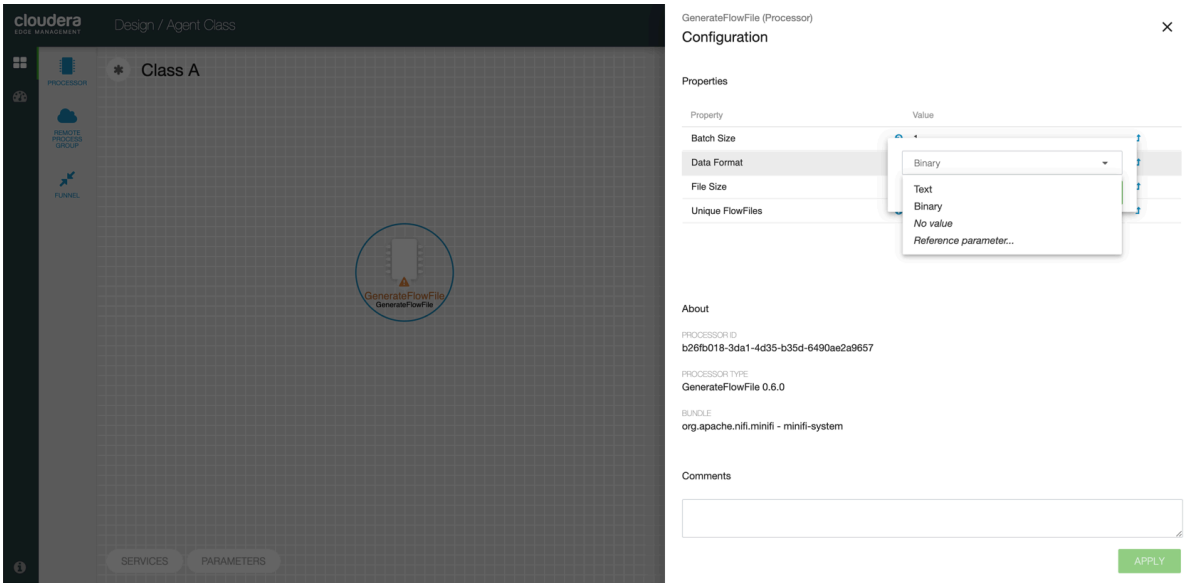
A GenerateFlowFile processor, by default, has four properties including Batch Size, Data Format, File Size, and Unique FlowFiles. Next to the name of each property, there appears a small question-mark symbol (•) indicating that additional information is available. Hovering over this symbol with the mouse provides

additional details about the property, the default value and whether Expression Language is supported. Here is an example of GenerateFlowFile processor with additional information for the Batch Size property:

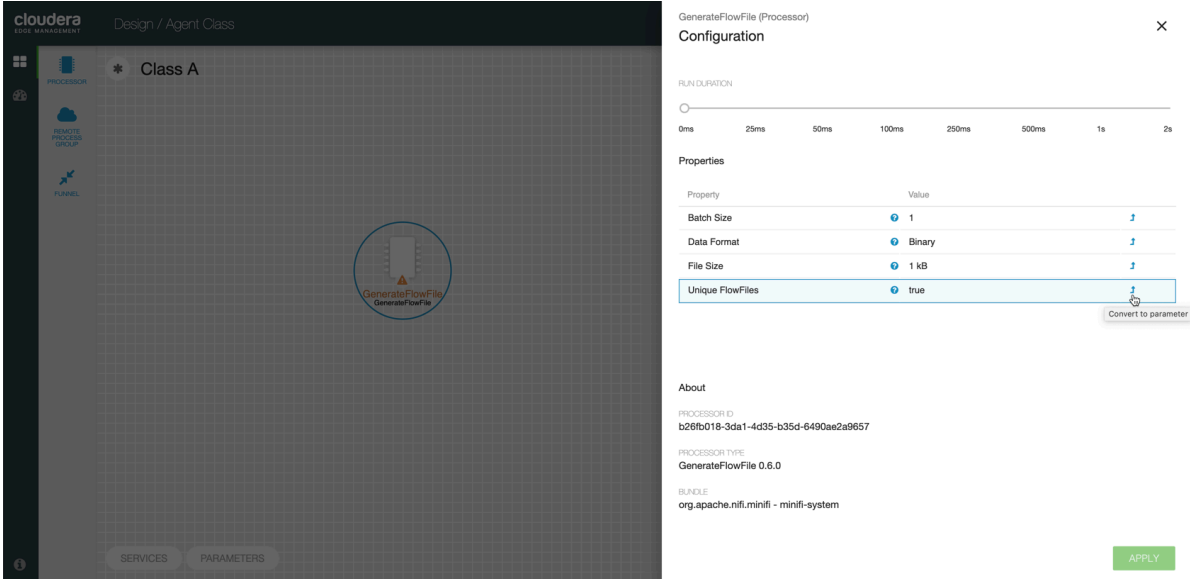


Clicking on the value for the property allows you to change the value. Depending on the values that are allowed for the property, you are either provided a drop-down from which to choose a value, or a text area

to type a value. Here is an example of GenerateFlowFile processor with the drop-down for the Data Format property:



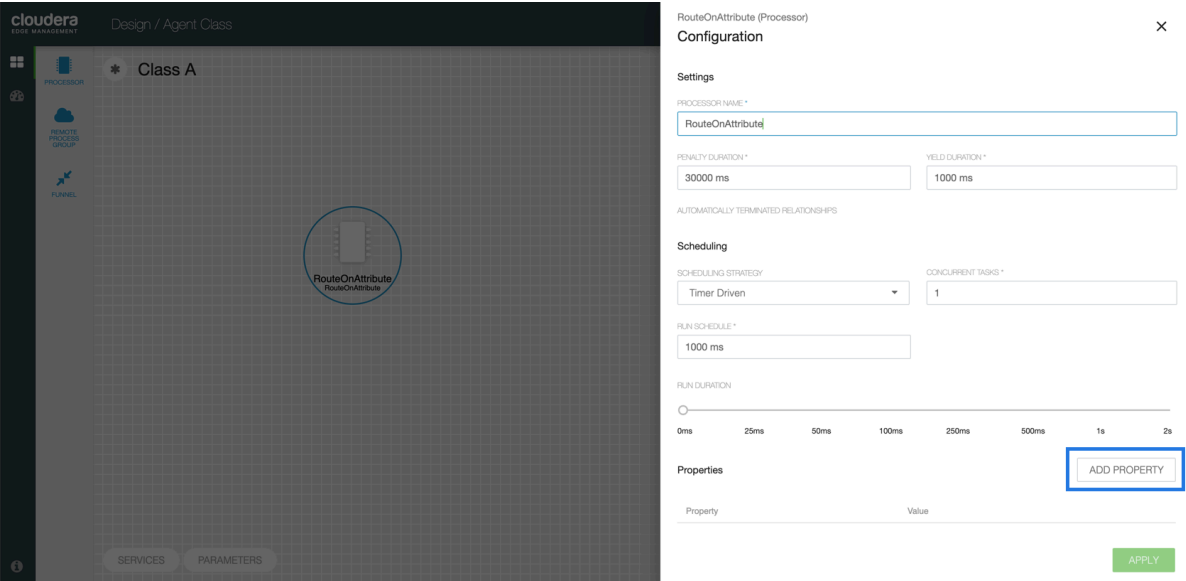
Each of the properties has an arrow in the row showing that they can be converted to parameters. The following image shows the Convert to parameter option for the Unique FlowFiles property:





For some processors, there appears an ADD PROPERTY button, beside the Properties option, for adding a user-defined property. When you click this button, a dialog opens, which allows you to enter the name and value of a new property. Not all processors allow user-defined properties. The RouteOnAttribute processor, however, allows user-defined properties. In fact, this Processor will not be valid until you add a property.

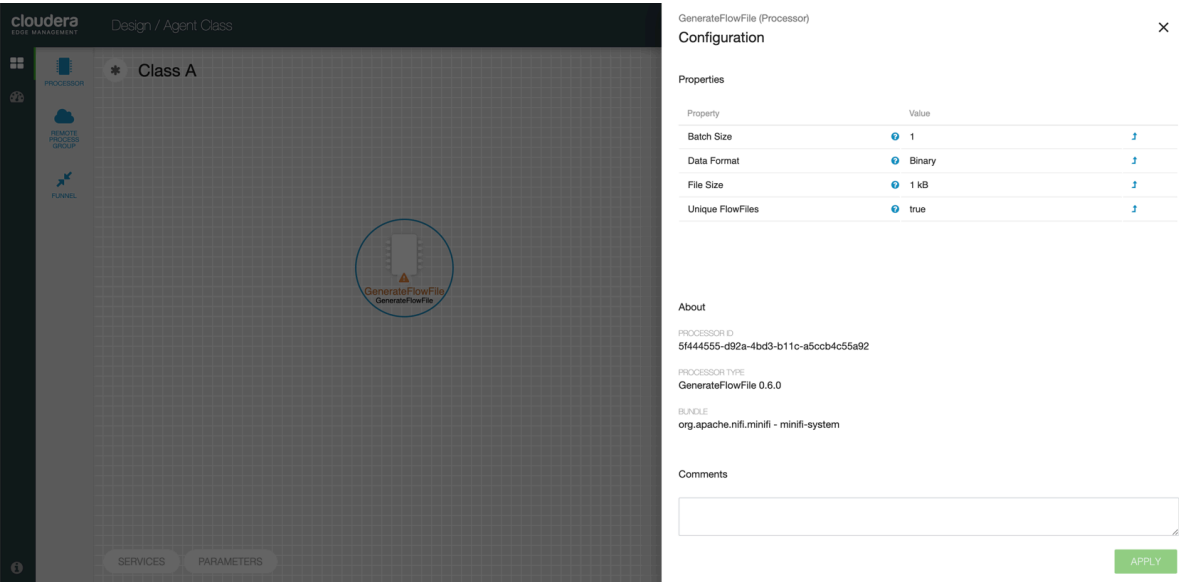


The following image shows the Add Property button in the Configuration dialog of the RouteOnAttribute processor:



 **Note:** After a user-defined property has been added, a trash icon () appears on the right-hand side of that row. You can remove the user-defined property from the processor by clicking the trash icon.

- About. The About section provides the Processor ID, Processor Type, and Bundle details of the processor, as shown in the following image:



- Comments. This tab simply provides an area for you to include whatever comments are appropriate for this component.

2. After you configure a processor, click the APPLY button to apply the changes.

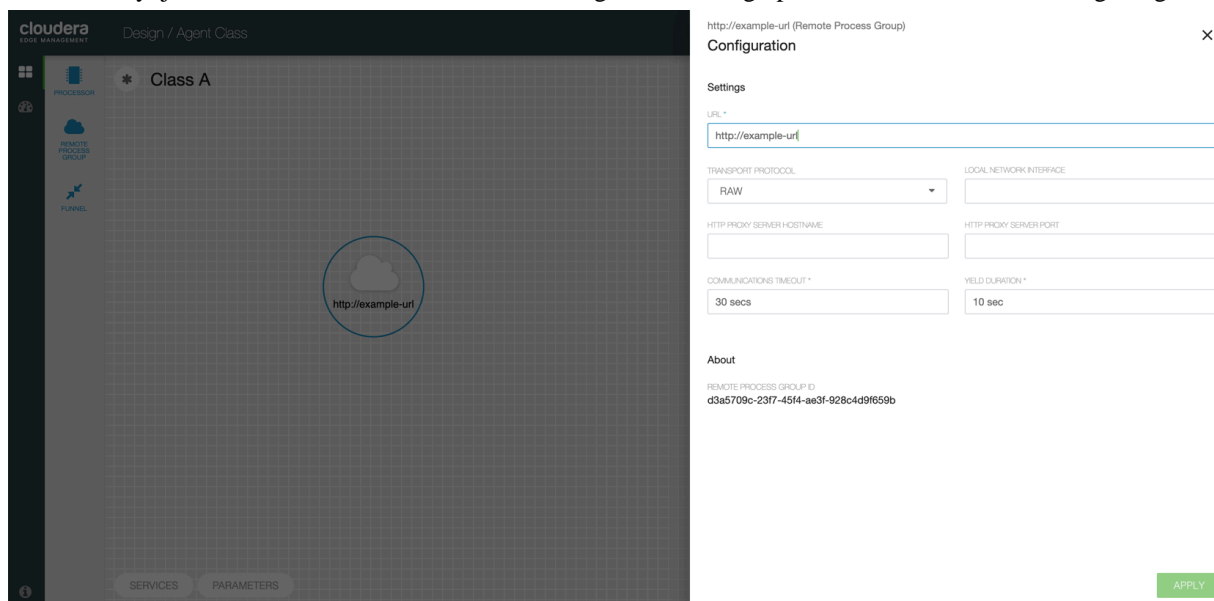
## Configuring a Remote Process Group

This section describes how to configure a remote process group.

## Procedure

1. To configure an RPG, right-click on the RPG and select the Configure option.

Alternatively, just double-click on the RPG. The Configuration dialog opens as shown in the following image:



The Configuration dialog contains the following two sections:

- Settings
  - About. The About section provides the Remote Process Group ID.
2. Configure the following properties in Settings section:

Properties	Description
URL	Allows you to change the URL of the RPG.
Transport Protocol	There are two options for transport protocol: <ul style="list-style-type: none"> <li>• RAW. This is the default protocol which uses raw socket communication by using a dedicated port.</li> <li>• HTTP. The HTTP transport protocol is useful if the remote NiFi instance is in a restricted network that only allows access through HTTP(S) protocol or only accessible from a specific HTTP Proxy server.</li> </ul>
Local Network Interface	In some cases, it might be desirable to prefer one network interface over another. For example, if a wired interface and a wireless interface exist, the wired interface might be preferred. This can be configured by specifying the name of the network interface to use in this box. If the value entered is not valid, the Remote Process Group will not be valid and will not communicate with other NiFi instances until this is resolved.
HTTP Proxy Server Hostname	Specify the host name of the proxy server, if you select HTTP transport protocol.
HTTP Proxy Server Port	Specify the port number of the proxy server, if you select HTTP transport protocol.
Communications Timeout	When communication with the RPG takes longer than this amount of time, it will timeout. The default value is 30 seconds.
Yield Duration	When communication with the RPG fails, it will not be scheduled again until this amount of time elapses. The default value is 10 seconds.

3. After you configure an RPG, apply the changes by clicking the APPLY button.

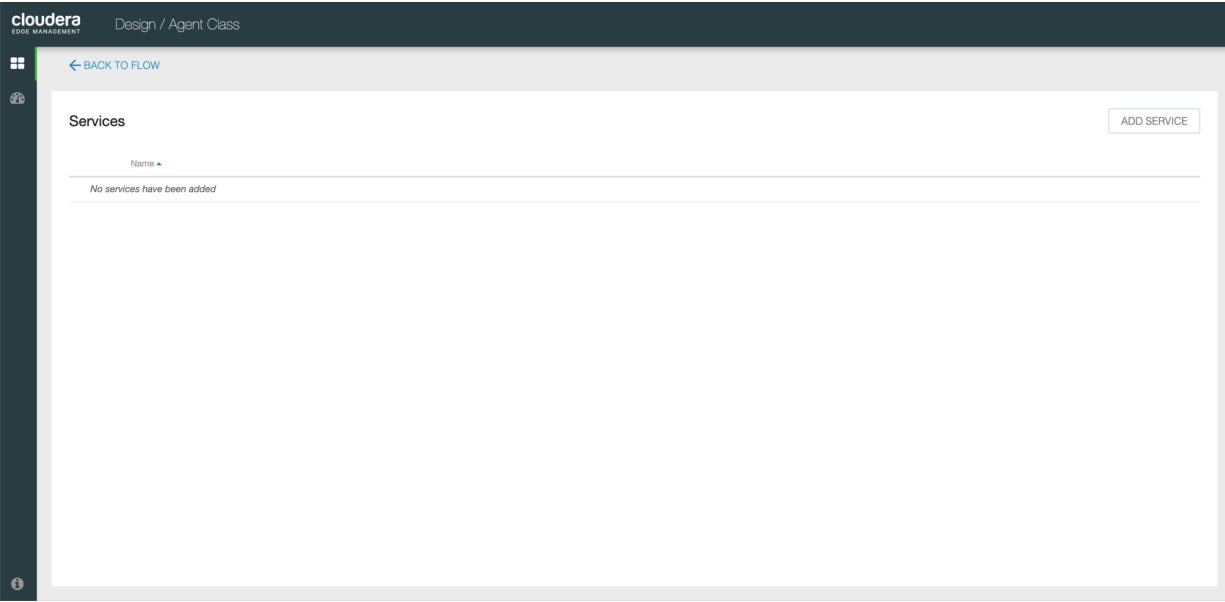
## Adding Services

Services are shared services that can be used by processors and other services to utilize for configuration or task execution.

### Procedure

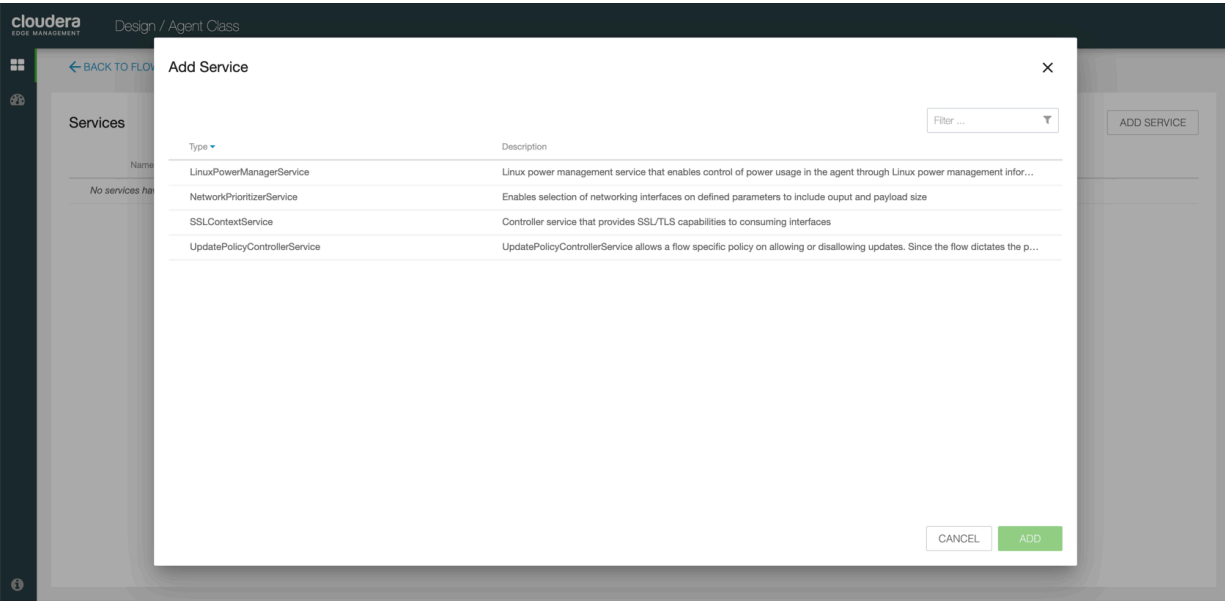
- 1. To add a service, click the SERVICES button at the bottom-left corner of the canvas, or simply right-click on the canvas and select Services.

The Services window opens as shown in the following image:




- 2. Click the ADD SERVICE button.

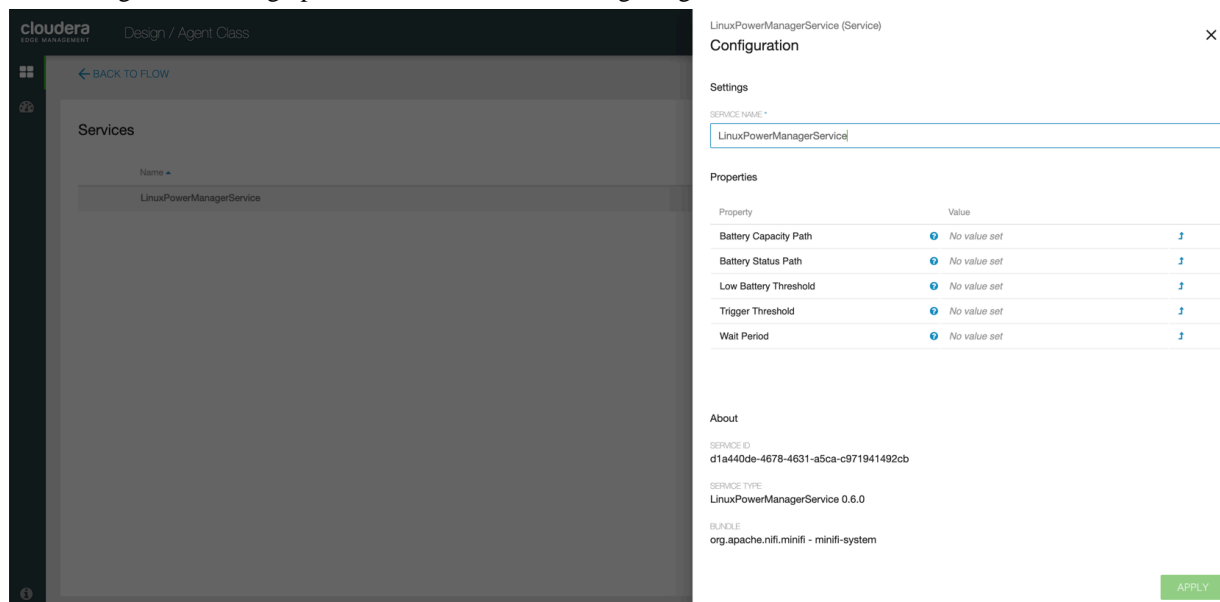
The Add Service dialog opens. It provides a list of the available services as shown in the following image:



- 3. Select the service you want to add and click ADD, or simply double-click on the name of the service to add it. You can also use the Filter field at the top-right corner of the window to search for the desired service by name.

4.


After you add a service, configure it by clicking the Configure icon (  ) in the far-right column. The Configuration dialog opens as shown in the following image:



The Configuration dialog contains the following sections:

- **Settings.** The Settings section provides a place for you to give the service a unique name. The name of a service by default is the same as the service type.
- **Properties.** The Properties section lists the various properties that apply to the particular service. You can hover over the question mark icons with the mouse to see more information about each property.
- **About.** The About section provides the Service ID, Service Type, and Bundle details of the service.
- **Comments.** The Comments section is just an open-text field, where you can include comments about the service.

5. After you configure a service, click the APPLY button to apply the configuration

If you want to delete a service, click the trash icon (  ) in the far-right column. To return to the canvas, click the BACK TO FLOW link.

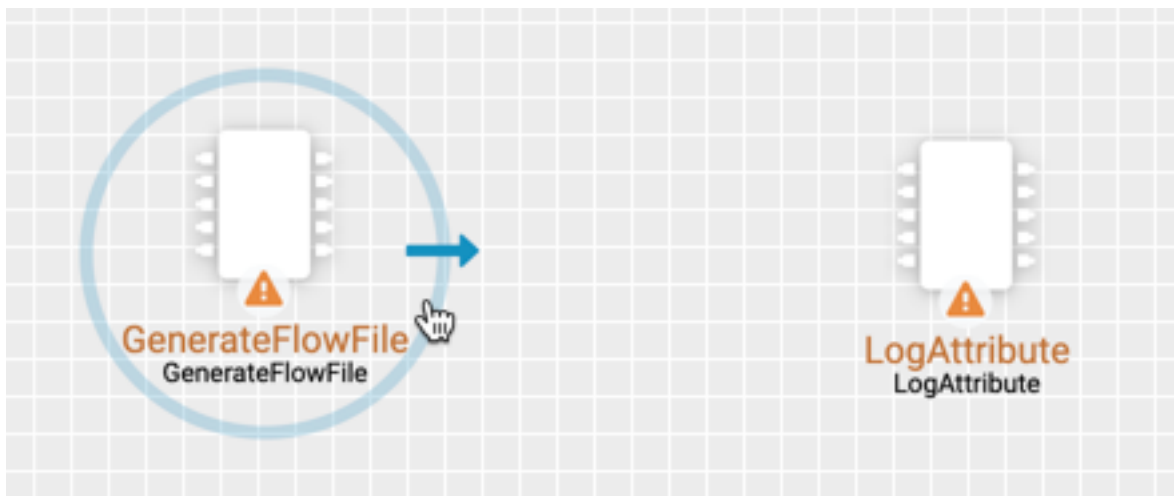
## Connecting Components

After you add processors and other components to the canvas and configure them, the next step is to connect them to one another. This is accomplished by creating a connection between each component.

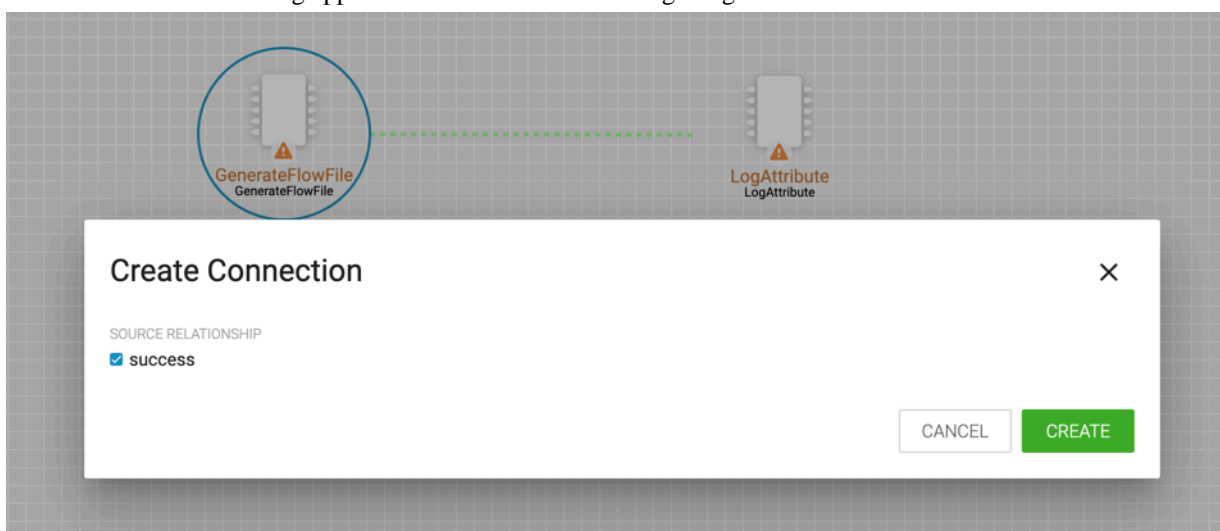
### Procedure

1. Hover the mouse over a component.

An arrow appears as shown in the following image:



2. Drag the arrow from one component to another until the second component is highlighted, then release the mouse.  
A Create Connection dialog appears as shown in the following image:



The dialog allows you to choose the Source Relationships that must be included in the connection. At least one relationship must be selected. If only one relationship is available, it is automatically selected.

3. Select CREATE to create the connection.



**Note:** It is possible to draw a connection so that it loops back on the same processor. This can be useful if you want the processor to try to re-process FlowFiles if the FlowFiles go down a failure relationship. To create this type of looping connection, simply drag the connection arrow away and then back to the same processor until it is highlighted. Then release the mouse and the same Create Connection dialog, referenced earlier, appears.

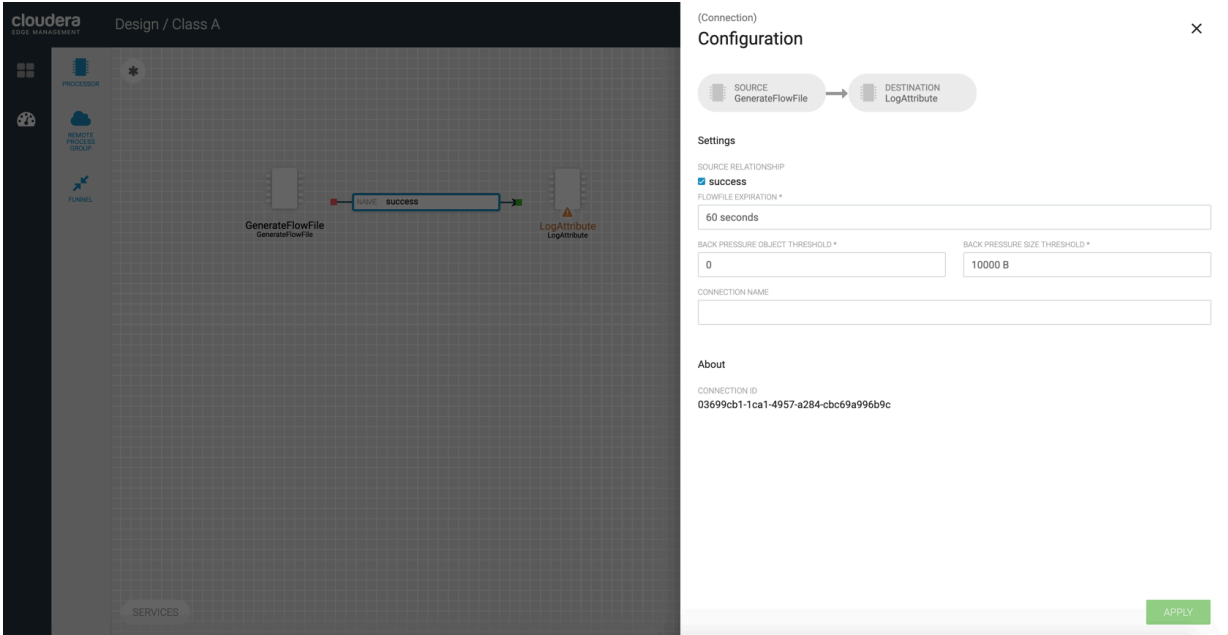
## Configuring a Connection

After you create a connection, you can change the configuration properties of the connection or move the connection.

Procedure

- 1. To change the configuration of a connection, right-click on the connection and select the Configure option, or double-click on the connection.

The Configuration dialog opens as shown in the following image:



The Configuration dialog contains the following two sections:

- Settings
  - About. The About section provides the Connection ID.
- 2. Configure the following properties in the Settings section:

Property	Description
Source Relationship	Allows you to change the Source Relationships of the connection.
FlowFile Expiration	FlowFile expiration is a concept by which data that cannot be processed in a timely fashion can be automatically removed from the flow. This is useful, for example, when the volume of data is expected to exceed the volume that can be sent to a remote site. The expiration period is based on the time that the data entered the MiNiFi instance. In other words, if the file expiration on a given connection is set to 1 hour, and a file that has been in the MiNiFi instance for one hour reaches that connection, it will expire. The default value is 60 seconds. A value of 0 seconds indicates that the data will never expire.
Back Pressure Object Threshold	This is the number of FlowFiles that can be in the queue before back pressure is applied. The default value is 0.
Back Pressure Size Threshold	This specifies the maximum amount of data (in size) that must be queued up before applying back pressure. The default value is 10,000 Bytes.
Connection Name	This field allows you to change the name of the connection. It is blank by default.

- 3. After you configure a connection, click the APPLY button to apply the changes.

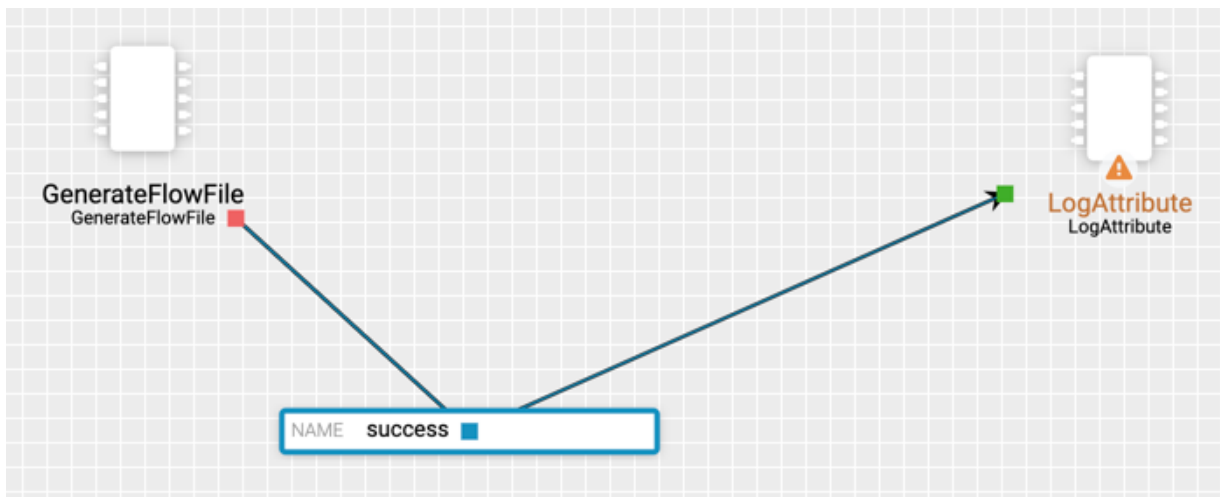
Bending Connections

This section describes how to bend a connection.

### Procedure

1. To add a bend point (or elbow) to an existing connection, simply double-click on the connection in the spot where you want the bend point to be.
2. Use the mouse to grab the bend point and drag it so that the connection is bent in the desired way.

The following image shows a bend point in the connection between GenerateFlowFile and LogAttribute processors:



You can add as many bend points as you want. You can also use the mouse to drag and move the label on the connection to any existing bend point. To remove a bend point, simply double-click on it again.

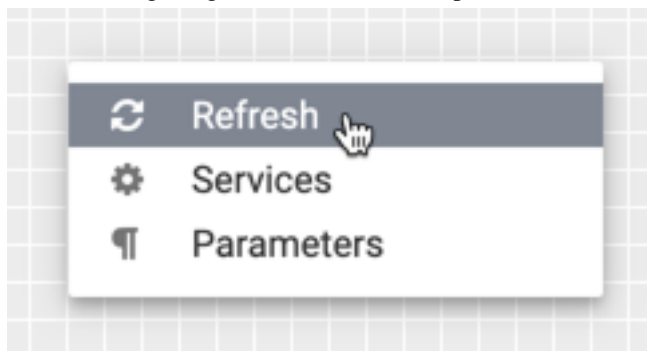
## Refreshing a DataFlow

This section describes how to refresh a dataflow.

### Procedure

To refresh a flow that you monitor, right-click on the canvas and select Refresh.

The following image shows the Refresh option:



## Publishing a DataFlow

This section describes how to publish a dataflow.

### About this task





**Note:** Publishing is an asynchronous process and agents update their flow as they periodically heartbeat to the CEM server.

### Procedure

1. To publish a dataflow and make it available to all agents associated with its class, select Publish from the ACTIONS drop-down menu on the canvas.

The Publish Flow dialog appears as shown in the following image:

2. Enter comments if desired, and click PUBLISH.

The flow status changes from modified (  ) to current (  ).

## Reverting a DataFlow

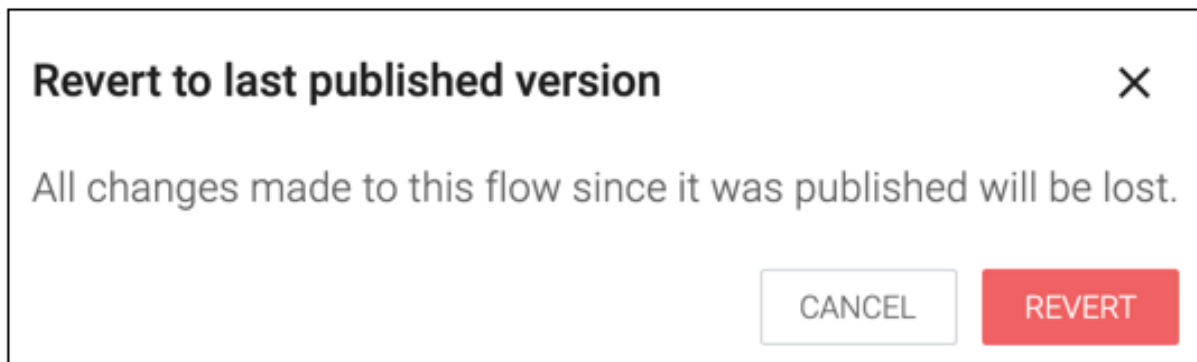
This section describes how to revert a dataflow.





### Procedure

1. To remove all changes that you made since a flow was published, select Revert to last published from the ACTIONS drop-down menu on the canvas.

The Revert to last published version dialog appears as shown in the following image:



2. Select REVERT to complete the process.

The flow status changes from modified (  ) to current (  ).

## Example DataFlow

The example dataflow consists of just two processors: GenerateFlowFile and LogAttribute. These processors are normally used for testing, but they can also be used to build a quick flow and see CEM in action.

### Procedure

1. Drag the GenerateFlowFile and LogAttribute processors to the canvas and connect them (using the guidelines provided above).

The dataflow appears on the canvas as shown in the following image:



## 2. Configure the GenerateFlowFile processor and click APPLY to apply the changes.

- In the Scheduling section, set Run Schedule to: 5 sec. Note that the GenerateFlowFile processor can create many FlowFiles very quickly. Therefore, setting the Run Schedule is important, so that the flow does not overwhelm the system CEM is running on.
- In the Properties section, set File Size to 10 KB.

The following image shows the configuration properties of the GenerateFlowFile processor:

**GenerateFlowFile (Processor) Configuration**

**Settings**

PROCESSOR NAME \*  
GenerateFlowFile

PENALTY DURATION \* 30000 ms YIELD DURATION \* 1000 ms

AUTOMATICALLY TERMINATED RELATIONSHIPS  
☐ success

**Scheduling**

SCHEDULING STRATEGY Timer Driven CONCURRENT TASKS \* 1

RUN SCHEDULE \* 5 s

RUN DURATION  
0ms 25ms 50ms 100ms 250ms 500ms 1s 2s

**Properties**

Property	Value
Batch Size	1

APPLY

## 3. Configure the LogAttribute processor and click APPLY to apply the changes.

- In the Settings section, under Automatically Terminated Relationships, select the checkbox next to success. This terminates FlowFiles after the processor successfully processes them.
- In the Properties section, set the Log Payload property to true. This way, the payload of the FlowFile is logged, in addition to its attributes. Otherwise, if set to false, just the attributes are logged.

The following image shows the configuration properties of the LogAttribute processor:

**LogAttribute (Processor) Configuration**

**Settings**

PROCESSOR NAME \*  
LogAttribute

PENALTY DURATION \* 30000 ms YIELD DURATION \* 1000 ms

AUTOMATICALLY TERMINATED RELATIONSHIPS  
☒ success

**Scheduling**

SCHEDULING STRATEGY Timer Driven CONCURRENT TASKS \* 1

RUN SCHEDULE \* 1000 ms

RUN DURATION  
0ms 25ms 50ms 100ms 250ms 500ms 1s 2s

**Properties**

Property	Value
Attributes to Ignore	No value set

APPLY

## 4. Publish the flow by selecting Publish from the ACTIONS drop-down.

## Monitoring Metrics with Grafana

CEM can export time series metrics to several metric storage providers. The recommended metrics store service is Prometheus. Prometheus integrates with Grafana for time series metric visualization. With Prometheus and Grafana, you can store and visualize metrics for CEM.

You need to perform the following tasks before you start visualizing CEM metrics with Prometheus and Grafana.

### Enabling Prometheus metrics in CEM

Ensure that the following metrics exporting property is enabled in CEM in the `efm.properties` file:

```
management.metrics.export.prometheus.enabled=true
```

You need to customize the following `efm.dashboard.*` properties:

```
efm.dashboard.base-url=http://grafana.example.com:3000
efm.dashboard.url.agentclass=/d/efm-agent-class/?var-agentClass={agentClass}
efm.dashboard.url.agent=/d/efm-agent/?var-agentId={agentId}
```

The `base-url` must reflect the location where you host Grafana. For details, see the *Setting up Grafana* section.

The dashboard URLs must point to the locations where you have set up agent and agent class specific URLs (see below).

### Setting up Prometheus

1. Install Prometheus on a host that has network connectivity to CEM. For instructions to install Prometheus, see <https://prometheus.io>.
2. Configure your `prometheus.yml` file to scrape the CEM instance. For example:

# Global config

```
global:
  scrape_interval: 1m
  evaluation_interval: 1m
```

# The following is a scrape configuration for CEM. Add this to any other scrape configurations you desire. In this example, it is Prometheus.

```
scrape_configs:
- job_name: 'cem-efm'
  metrics_path: '/efm/actuator/prometheus'
  scrape_interval: 15s
```

```
static_configs:
- targets: ['efm.example.com:10080']
```

For additional scrape configuration properties, such as TLS settings, see the Prometheus configuration guide: [https://prometheus.io/docs/prometheus/latest/configuration/configuration/#scrape\\_config](https://prometheus.io/docs/prometheus/latest/configuration/configuration/#scrape_config).

3. Verify Prometheus configuration.

### Setting up Grafana

1. Install Grafana on a host that has network connectivity to Prometheus. For instructions to install Grafana, see <https://grafana.com>.

2. Configure Grafana to use Prometheus as a datasource. This can be done through the Grafana UI or through a data sources.yml provisioning file. For example:

```
# config file version
apiVersion: 1
```

# List of datasources that must be deleted from the database:

```
deleteDatasources:
- name: CEM EFM Prometheus orgId: 1
```

# List of datasources to insert or update depending what is available in the database:

```
datasources:
- name: CEM EFM Prometheus
  type: prometheus
  access: proxy
  orgId: 1
  url: http://prometheus:9090
  password:
  user:
  database:
  basicAuth: false
  basicAuthUser:
  basicAuthPassword:
  withCredentials: false
  isDefault: true
  jsonData:
    graphiteVersion: "1.1"
    tlsAuth: false
    tlsAuthWithCACert: false
  secureJsonData:
    tlsCACert: "..."
    tlsClientCert: "..."
    tlsClientKey: "..."
  version: 1
  editable: true
```

For more information for configuring a Prometheus datasource in Grafana, see <https://grafana.com/docs/features/datasources/prometheus/>.

3. Download the CEM Grafana dashboard templates.

You can download the Grafana dashboard templates included in source files from <https://www.cloudera.com/downloads/cdf.html>.

4. Import each JSON dashboard. Follow the instructions provided in the following Grafana document: [https://grafana.com/docs/reference/export\\_import/#importing-a-dashboard](https://grafana.com/docs/reference/export_import/#importing-a-dashboard).
5. CEM dashboards should now be available in the Grafana UI.

### Navigating to Grafana dashboard in CEM

To navigate to class specific Grafana dashboards, select a class in the Deployment screen. In the class page, click the About Class link at the top-right corner. Select the View Grafana Dashboard link in the About This Class window.

To navigate to agent specific Grafana dashboards, select an event source in the Events screen. In the event page, click the About Agent link at the top-right corner. Select the View Grafana Dashboard link in the About This Agent window.