

Tutorial: Build a New Flow

Date published: 2021-04-06

Date modified: 2024-06-03

CLOUDERA

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

Tutorial: Building a new flow from scratch.....	4
Create a new flow.....	4
Create controller services.....	8
Build your draft flow.....	13
Start a test session.....	25
Publish your flow definition to the Catalog.....	27

Tutorial: Building a new flow from scratch

If you are new to flow design and have never used NiFi before, this tutorial is for you. Learn how to build a draft adding and configuring components, connecting them, creating Controller Services, and testing your flow while creating it.

About this task

This tutorial walks you through the creation of a simple flow design that retrieves the latest changes from Wikipedia through invoking the Wikipedia API. It converts JSON events to Avro, before filtering and routing them to two different processors which merge events together before a file is written to local disk.

You will learn about:

- Creating a draft
- Creating a Controller Service
- Adding processors to your draft
- Configuring processors
- Adding a user-defined property to a processor configuration
- Connecting processors to create relationships between them
- Running a Test Session
- Publishing a draft to the Catalog as a flow definition.

Before you begin

The flow you are about to build can be deployed without any external dependencies and does not require any parameter values during deployment. Still, there are prerequisites you have to meet before you can start building your first draft.

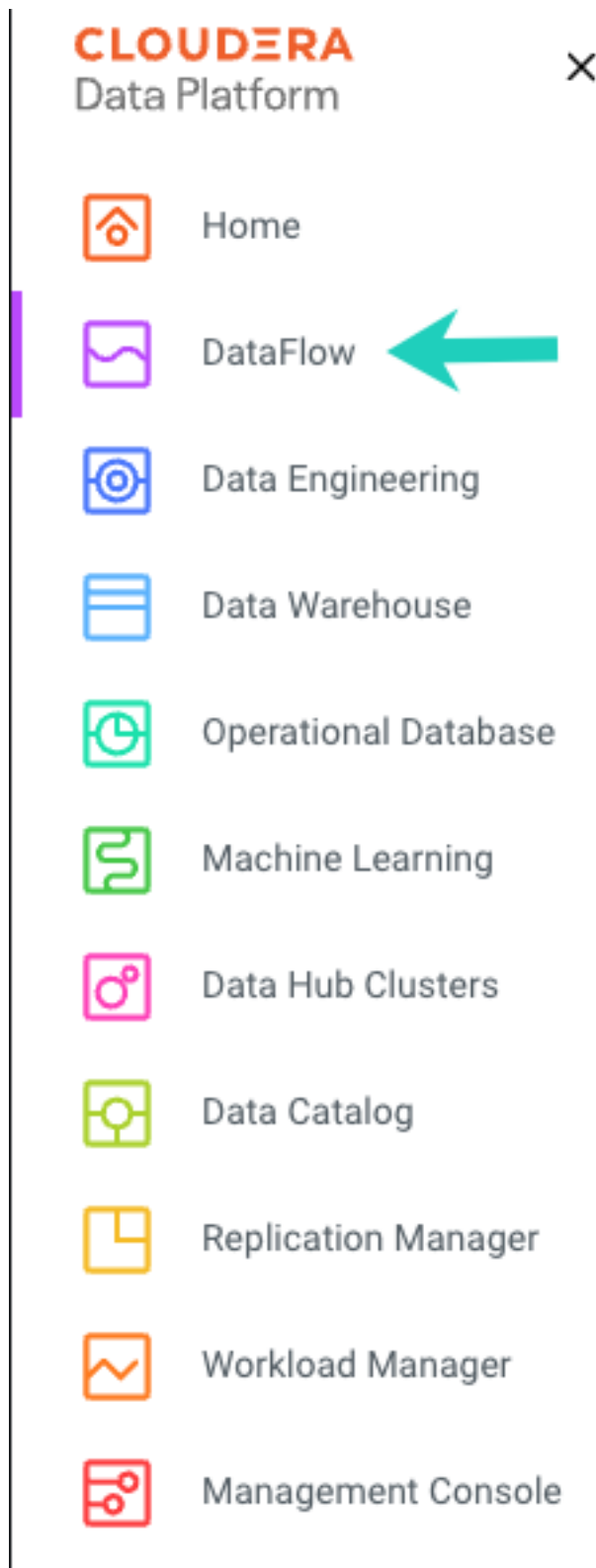
- You have an enabled and healthy Cloudera DataFlow environment.
- You have been assigned the DFDeveloper role granting you access to the Flow Designer.
- You have been assigned the DFCatalogAdmin or DFCatalogViewer role granting you access to the Catalog. You will need this authorization to publish your draft as a flow definition to the Catalog.
- You have been assigned the DFFlowAdmin role for the environment to which you want to deploy the flow definition.

Create a new flow

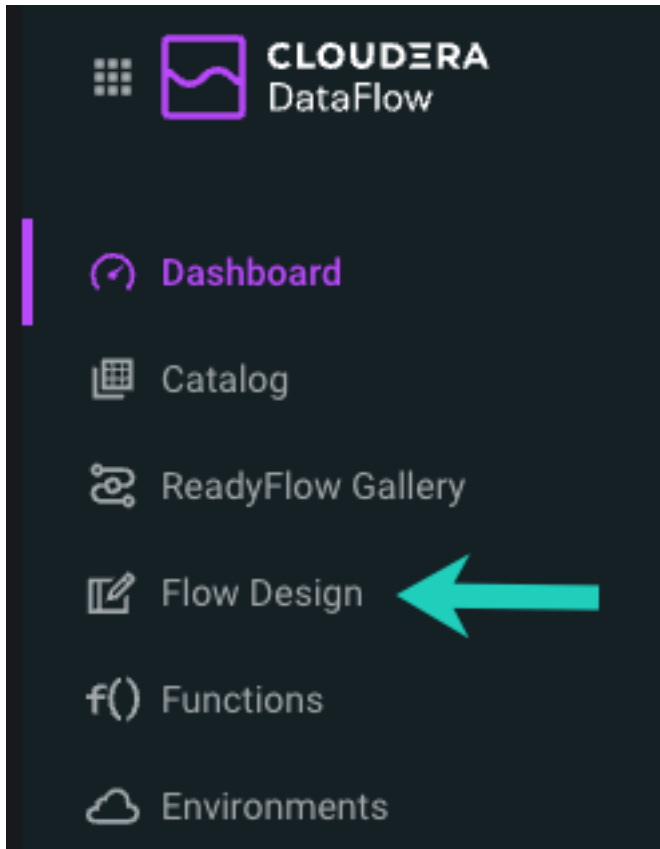
Create a new flow in a Flow Designer Workspace and give it a name.

Procedure

1. Open Cloudera DataFlow by clicking the DataFlow tile in the Cloudera sidebar.

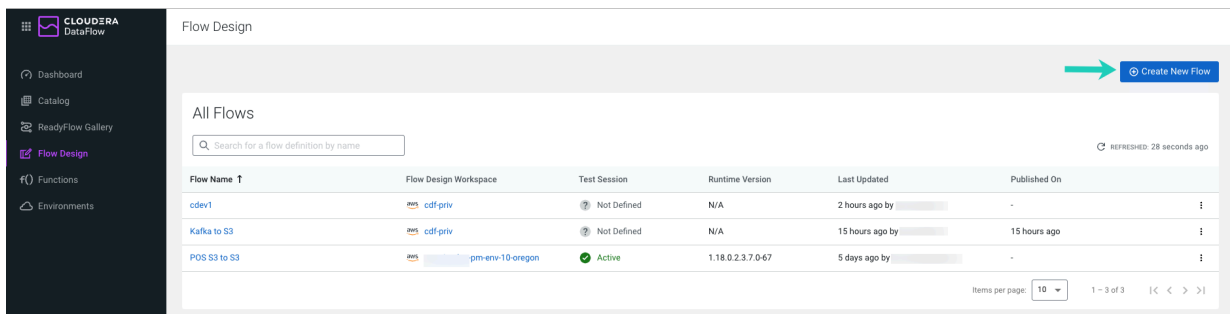


- Click Flow Design in the left navigation pane.



You are redirected to the **Flow Design** page, where previously created draft flows are displayed, one flow per row.

- Click Create New Flow.



4. Select a Target Workspace where you want to create the draft.

Create New Draft ✕

Target Workspace ?

aws cdf-priv ▼

Target Project ?


Select a project ▼

Draft Name

Draft Name

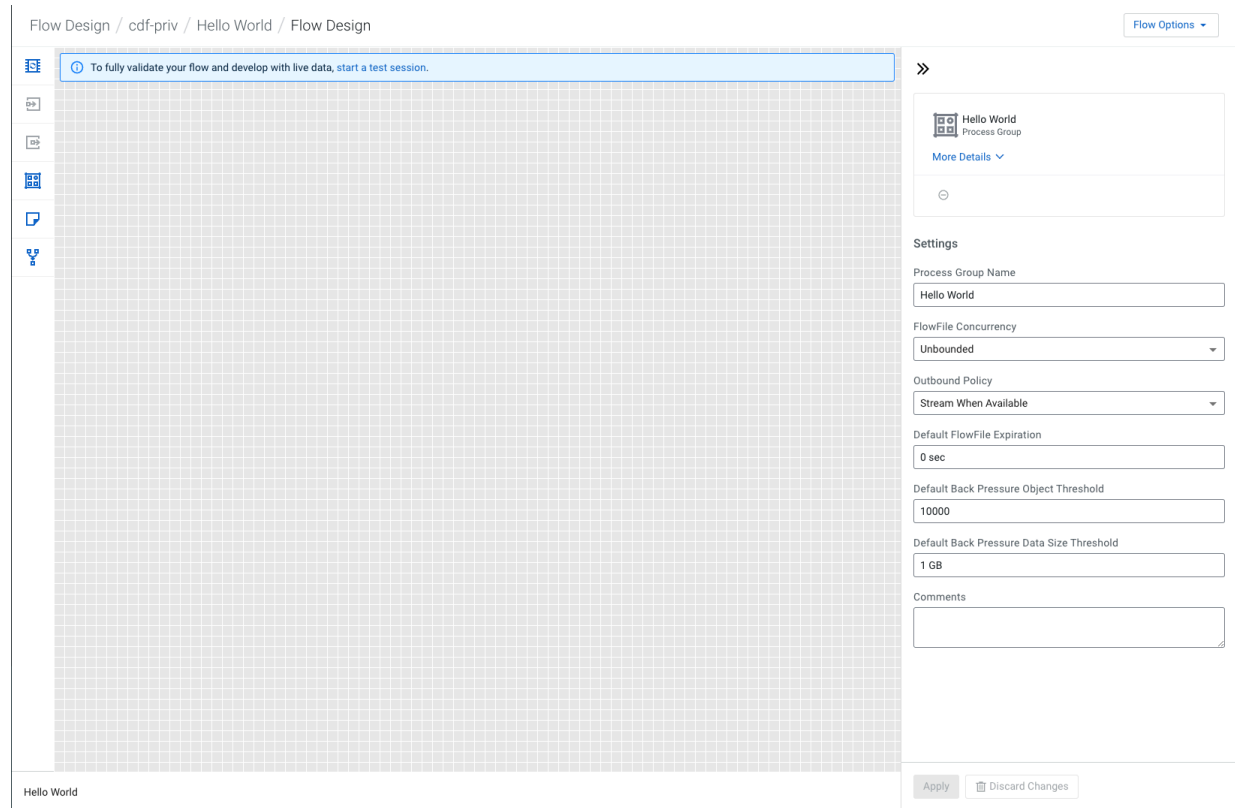
Create

Cancel

5. Under Target Project select  Unassigned.
6. Provide a Draft Name.
Provide Hello World.

7. Click Create.

Flow Designer creates a default Process Group with the Draft Name you provided, 'Hello World' in this case, and you are redirected to the **Flow Design** canvas. The **Configuration** pane on the right displays configuration options for the default Process Group.



What to do next

Proceed to creating controller services.

Related Tasks

[Create controller services](#)

Create controller services

Learn about creating Controller Services in Cloudera DataFlow Flow Designer.

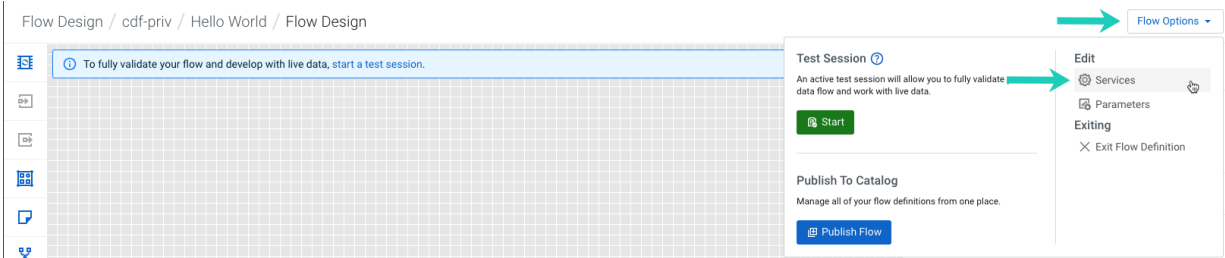
About this task

Controller Services are extension points that provide information for use by other components (such as processors or other controller services). The idea is that, rather than configure this information in every processor that might need it, the controller service provides it for any processor to use as needed.

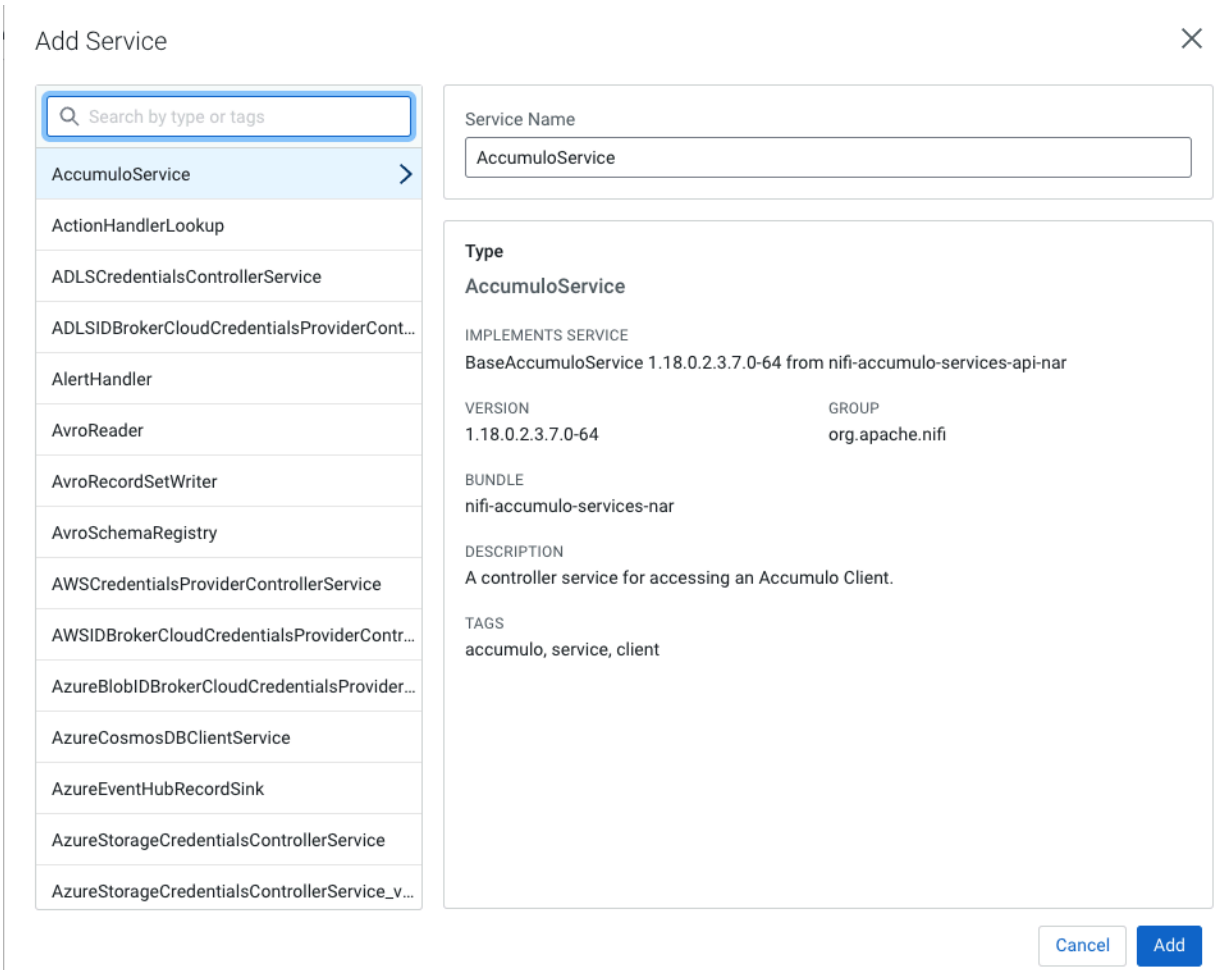
You will use the controller services you create now to configure the behavior of several processors you will add to your flow as you are building it.

Procedure

- 1. Go to Flow Options Services .



- 2. Click Add Service.
The Add Service page opens.



3. In the text box, filter for JsonTreeReader.

Add Service

Q json

X

AvroSchemaRegistry

JsonConfigBasedBoxClientService

JsonPathReader

JsonRecordSetWriter

JsonTreeReader

RestLookupService

Service Name

JsonTreeReader

Type

JsonTreeReader

IMPLEMENTS SERVICE

RecordReaderFactory 1.18.0.2.3.7.0-89 from nifi-standard-services-api-nar

VERSION

1.18.0.2.3.7.0-89

GROUP

org.apache.nifi

BUNDLE

nifi-record-serialization-services-nar

DESCRIPTION

Parses JSON into individual Record objects. While the reader expects each record to be well-formed JSON, the content of a FlowFile may consist of many records, each as a well-formed JSON array or JSON object with optional whitespace between them, such as the common 'JSON-per-line' format. If an array is encountered, each element in that array will be treated as a separate record. If the schema that is configured contains a field that is not present in the JSON, a null value will be used. If the JSON contains a field that is not present in the schema, that field will be skipped. See the Usage of the Controller Service for more information and examples.

TAGS

parser, reader, record, tree, json

Cancel

Add

4. Provide Service Name: JSON_Reader_Recent_Changes.
5. Click Add.

6. Configure the JSON_Reader_Recent_Changes service.

Set the following Properties:

Starting Field Strategy


Nested Field

Starting Field Name


recentchanges


Flow Options ▾


>>


 **JSON_Reader_Recent_Changes**
JsonTreeReader V.1.18.0.2.3.7.0-89

More Details ▾









Settings

*Service Name


JSON_Reader_Recent_Changes


Comments

Properties


Property	Value
Schema Access Strategy ?	Infer Schema ...
Schema Inference Cache ?	No value set ...
Starting Field Strategy ?	Root Node ▾ ...
Date Format ?	Root Node? ...
Time Format ?	Nested Field? ...
Timestamp Format ?	No value set ...

Referencing Components

 No referencing Processors to display.

 No referencing Services to display.

Apply

 Discard Changes

7. Click Apply.

8. Click Add Service to create another service.

9. In the text box, filter for AvroRecordSetWriter.

10. Provide Service Name: AvroWriter_Recent_Changes.

11. Click Add.

You do not need to configure the AvroWriter_Recent_Changes service. You can leave all properties with their default values.

12. Click Add Service to create a third service.

13. In the text box, filter for AvroReader.

14. Provide Service Name: AvroReader_Recent_Changes.

15. Click Add.

You do not need to configure the AvroReader_Recent_Changes service. You can leave all properties with their default values.

16. Click Back To Flow Designer to return to the flow design Canvas.

What to do next

After creating the necessary Controller Services, you can start building and configuring your flow.

Related Tasks

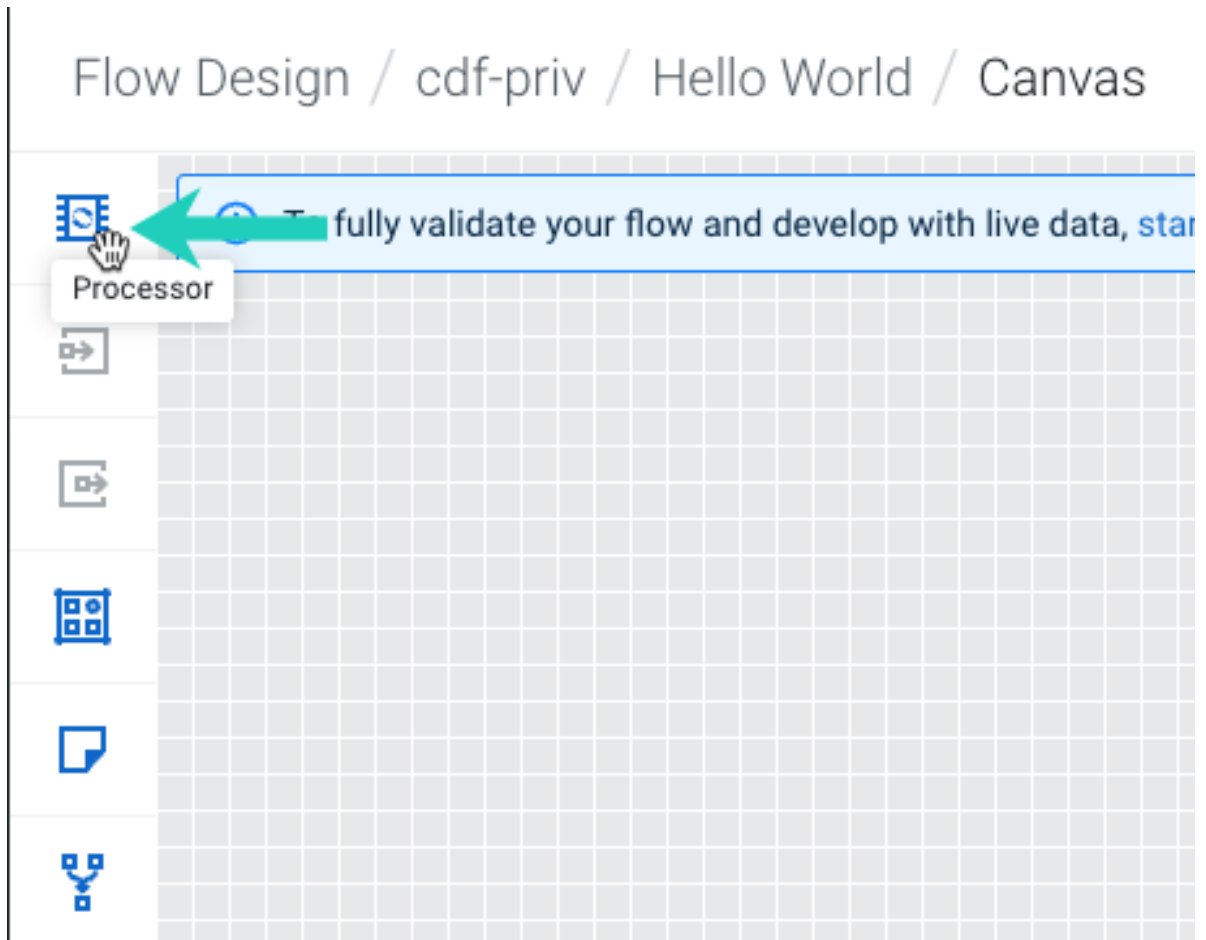
[Build your draft flow](#)

Build your draft flow

Start building your draft flow by adding components to the Canvas and configuring them.

Procedure

1. Add an InvokeHTTP processor to the canvas.
 - a) Drag a Processor from the Components sidebar to the canvas.



- b) In the text box, filter for InvokeHTTP.

Add Processor

🔍 invokeH

✕

InvokeHTTP

Processor Name

InvokeHTTP

Type

InvokeHTTP

IMPLEMENTS SERVICE

VERSION

1.18.0.2.3.7.0-89

GROUP

org.apache.nifi

BUNDLE

nifi-standard-nar

DESCRIPTION

An HTTP client processor which can interact with a configurable HTTP Endpoint. The destination URL and HTTP Method are configurable. FlowFile attributes are converted to HTTP headers and the FlowFile contents are included as the body of the request (if the HTTP Method is PUT, POST or PATCH).

TAGS

rest, http, client, https

Cancel

Add

c) Change the Processor Name to Get Recent Wikipedia Changes.

d) Click Add.

After configuration, this processor calls the Wikipedia API to retrieve the latest changes.

2. Configure the Get Recent Wikipedia Changes processor.

Properties

HTTP URL

provide `https://en.wikipedia.org/w/api.php?action=query&list=recentchanges&format=json&rcprop=user%7Ccomment%7Cparsedcomment%7Ctimestamp%7Ctitle%7Csize%7Ctags`

Flow Design / cdf-priv / Hello World / Canvas Flow Options ▾

To fully validate your flow and develop with live data, start a test session.

Get Recent Wikipedia Changes
Invoke HTTP 1.18.0.2.3.7.0-89

IN	N/A
READ/WRITE	N/A
OUT	N/A
TASKS	N/A

5 min

Scheduling

*Scheduling Strategy ⓘ Timer Driven ▾ *Concurrent Tasks ⓘ 1

*Run Duration ⓘ 0ms ▾ *Run Schedule ⓘ 0 sec

*Execution ⓘ All Nodes ▾

Properties ⓘ Add Property

Property	Value	
HTTP Method ⓘ		⋮
HTTP URL ⓘ	<code>https://en.wikipedia.org/w/api.php?</code>	⋮
HTTP/2 Disabled ⓘ		⋮
SSL Context Service ⓘ		⋮
Socket Connect Timeout ⓘ		⋮
Socket Read Timeout ⓘ		⋮
Socket Idle Timeout ⓘ	5 mins	⋮
Socket Idle Connections ⓘ	5	⋮
Proxy Configuration Service ⓘ	No value set	⋮
Proxy Host ⓘ	No value set	⋮

Apply Discard Changes

Relationships

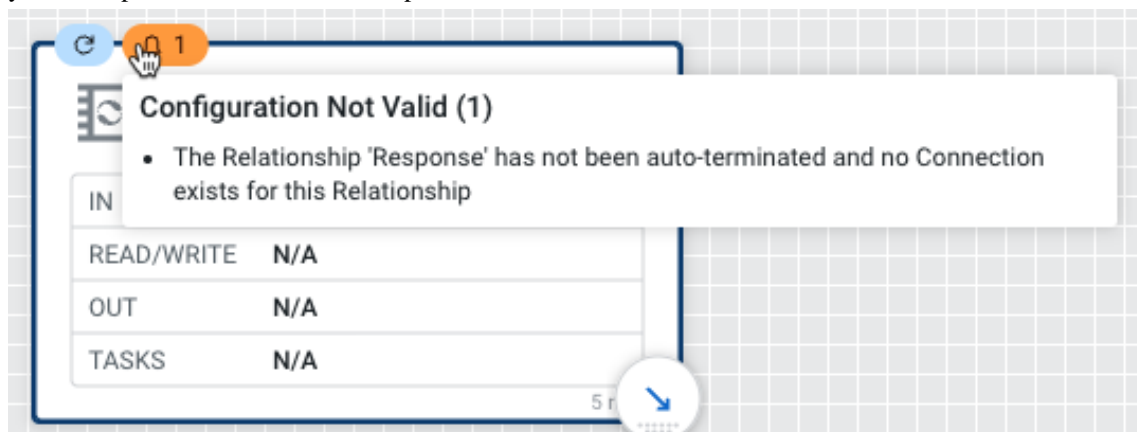
Select the following relationships:

- Original - Terminate
- Failure - Terminate, Retry
- Retry - Terminate
- No Retry - Terminate

3. Click Apply.



Notice: Note the orange Notification Pill in the upper left corner of your processor. It is there to warn you about possible issues with a component.



In this particular case, it warns you about one of the Relationships not being addressed when configuring your processor. Do not worry, it will disappear when you create a connection to another processor for the 'Response' relationship.

4. Add a ConvertRecord processor to the canvas.

- Drag a Processor from the Components sidebar to the canvas.
- In the text box, filter for ConvertRecord.
- Change the Processor Name to Convert JSON to AVRO.
- Click Add.

This processor converts the JSON response to AVRO format. It uses RecordReaders and RecordWriters to accomplish this. It infers the JSON schema starting from the recent changes field.

5. Configure the Convert JSON to AVRO processor.

Properties

Record Reader

Select the JSON_Reader_Recent_Changes controller service you have created from the drop-down list.

Record Writer

Select the AvroWriter_Recent_Changes controller service you have created from the drop-down list.

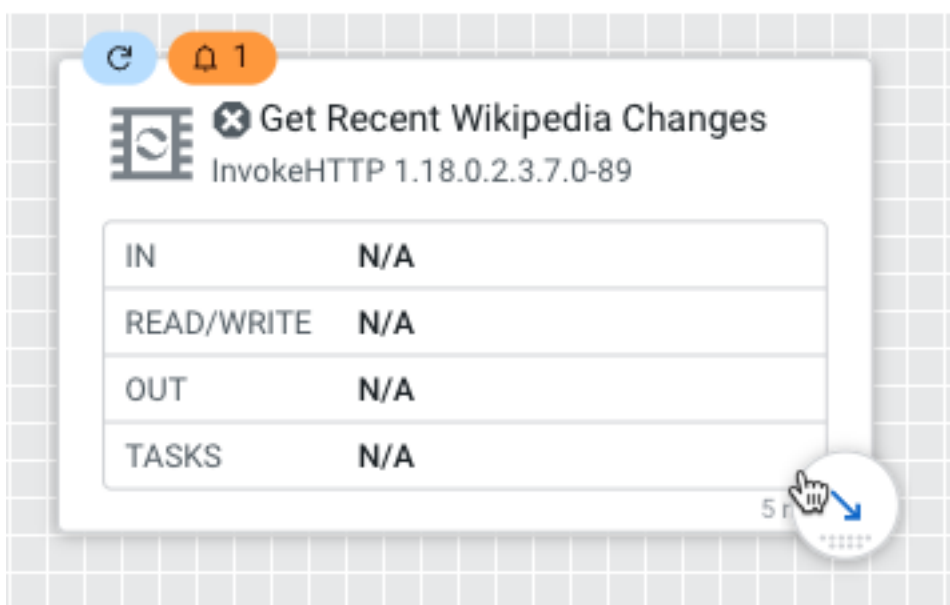
Relationships

Select the following relationships:

failure - Terminate, Retry

6. Click Apply.

7. Connect the Get Recent Wikipedia Changes and Convert JSON to AVRO processors by hovering over the lower-right corner of the Get Recent Wikipedia Changes processor, clicking the arrow that appears and dragging it to the Convert JSON to AVRO processor.



8. In the configuration popup, select the Response relationship and click Add.

Create Connection ✕

From Processor

Get Recent Wikipedia Changes
InvokeHTTP

Within Group

Hello World

Relationships

☐ Original

☐ Failure

☐ Retry

☐ No Retry

☒ Response

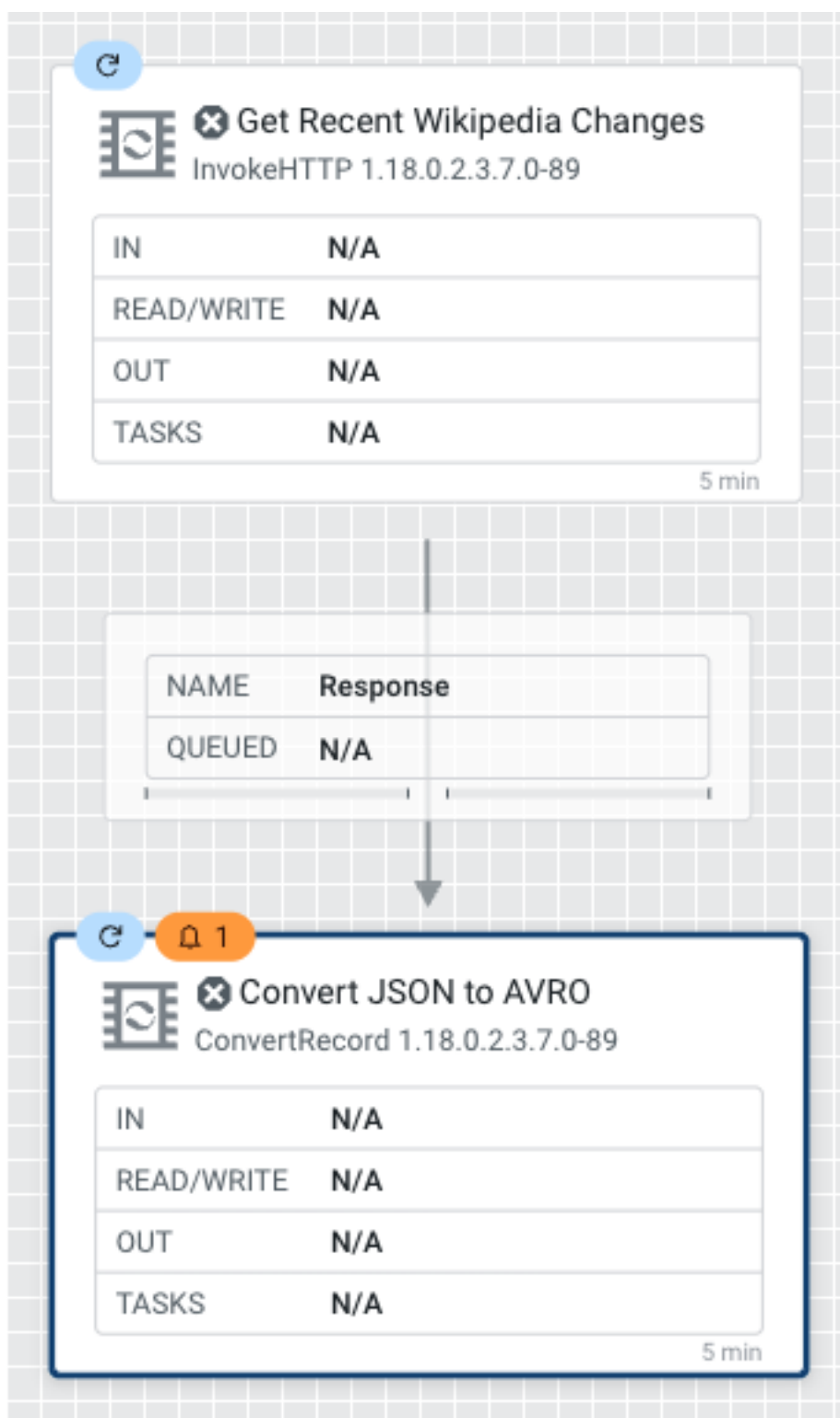
To Processor


Convert JSON to AVRO
ConvertRecord

Within Group

Hello World

Cancel Add



Notice: Note, that the  Notification Pill warning about the unconfigured 'Response' relationship disappeared from your Get Recent Wikipedia Changes processor.

9. Add a QueryRecord processor.

- Drag a Processor from the Components sidebar to the canvas.
- In the text box, filter for QueryRecord.
- Name it Filter Edits.
- Click Add.

This processor filters out anything but actual page edits. To achieve this, it's running a query that selects all FlowFiles (events) of the type edit.

10. Configure the Filter Edits processor.

Properties

Record Reader

Select the AvroReader_Recent_Changes controller service you have created from the drop-down list.

Record Writer

Select the AvroWriter_Recent_Changes controller service you have created from the drop-down list.

Relationships

Select the following relationship:

- failure - Terminate
- original - Terminate

11. For the Filter Edits processor you also need to add a user-defined property. Click Add Property.

- Provide Filtered edits as Name
- Provide Select * from FLOWFILE where type='edit' as Value.
- Click Apply.

12. Connect the Convert JSON to AVRO and Filter Edits processors by hovering over the lower-right corner of the Convert JSON to AVRO processor, clicking the arrow that appears and dragging it to the Filter Edits processor.**13. In the configuration pane, select the success and failure relationships and click Add.****14. Add a second QueryRecord processor.**

- Drag a Processor from the Components sidebar to the canvas.
- In the text box, filter for QueryRecord.
- Name it Route on Content Size.
- Click Add.

This processor uses two SQL statements to separate edit events that resulted in a longer article from edit events that resulted in a shorter article.

15. Configure the Route on Content Size processor.

Properties

Record Reader

Select the AvroReader_Recent_Changes controller service you have created from the drop-down list.

Record Writer

Select the AvroWriter_Recent_Changes controller service you have created from the drop-down list.

Relationships

Select the following relationships:

- failure - Terminate, Retry
- original - Terminate

16. For the Route on Content Size processor you also need to add two user-defined properties.

- a) Click Add Property.
- b) Provide Added content as Name
- c) Provide Select * from FLOWFILE where newlen>=oldlen as Value.
- d) Click Add.
- e) Click Add Property, to create the second property.
- f) Provide Removed content as Name.
- g) Provide Select * from FLOWFILE where newlen<oldlen as Value.
- h) Click Add.
- i) Click Apply.

17. Connect the Filter Edits and Route on Content Size processors by hovering over the lower-right corner of the Filter Edits processor, clicking the arrow that appears and drawing it to Route on Content Size.

In the Create Connection pop up select the Filtered edits relation and click Add.

18. Add two MergeRecord processors.

- a) Drag a Processor from the Components sidebar to the canvas.
- b) In the text box, filter for MergeRecord.
- c) Name it Merge Edit Events.
- d) Click Add.
- e) Repeat the above steps to add another identical processor.

These processors are configured to merge at least 100 records into one flowfile to avoid writing lots of small files. The MaxBinAge property is set to 2 minutes which makes the processors merge records after two minutes even if less than 100 records have arrived.

19. Configure the two Merge Edit Events processors.

Properties

Record Reader

Select the AvroReader_Recent_Changes controller service you have created from the drop-down list.

Record Writer

Select the AvroWriter_Recent_Changes controller service you have created from the drop-down list.

Max Bin Age

Set to two minutes by providing a value of 2 min.

Relationships

Select the following relationships:

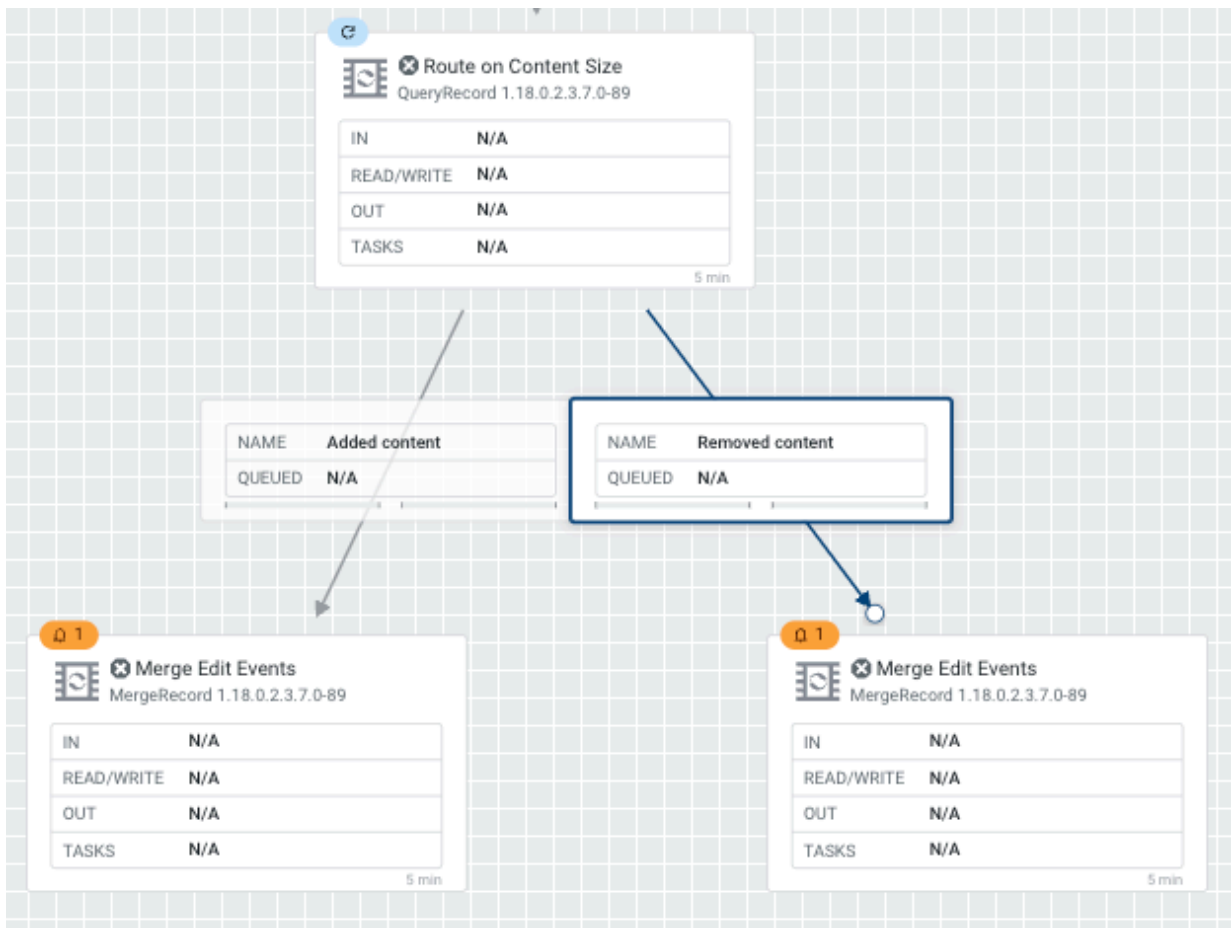
- failure - Terminate
- original - Terminate



Note: Do not forget to perform configuration for both Merge Edit Events processors.

20. Connect the Route on Content Size processor to both of the Merge Edit Events processors.

- a) For the first Merge Edit Events processor, select Added content from Relationships and click Add.
- b) For the second Merge Edit Events processor, select Removed content from Relationships and click Add.



21. Add two PutFile processors to the canvas.

- a) Drag a Processor from the Components sidebar to the canvas.
- b) In the text box, filter for PutFile.
- c) Name it Write "Added Content" Events To File.
- d) Click Add.
- e) Repeat the above steps to add another identical processor, naming this second PutFile processor Write "Removed Content" Events To File.

These processors write the filtered, routed edit events to two different locations on the local disk. In Cloudera DataFlow, you would typically not write to local disk but replace these processors with processors that resemble your destination system (Kafka, Database, Object Store etc.)

22. Configure the Write "Added Content" Events To File processor.

Properties:

Directory

Provide /tmp/larger_edits

Maximum File Count

Provide 500

Relationships

Select the following relationships:

- SUCCESS - Terminate
- failure - Terminate

23. Click Apply.**24. Configure the Write "Removed Content" Events To File processor.**

Properties:

Directory

Provide /tmp/smaller_edits

Maximum File Count

Provide 500

Relationships

Select the following relationships:

- SUCCESS - Terminate
- failure - Terminate

25. Click Apply.**26. Connect the Merge Edit Events processor with the Added content connection to the Write "Added Content" Events To File processor.**

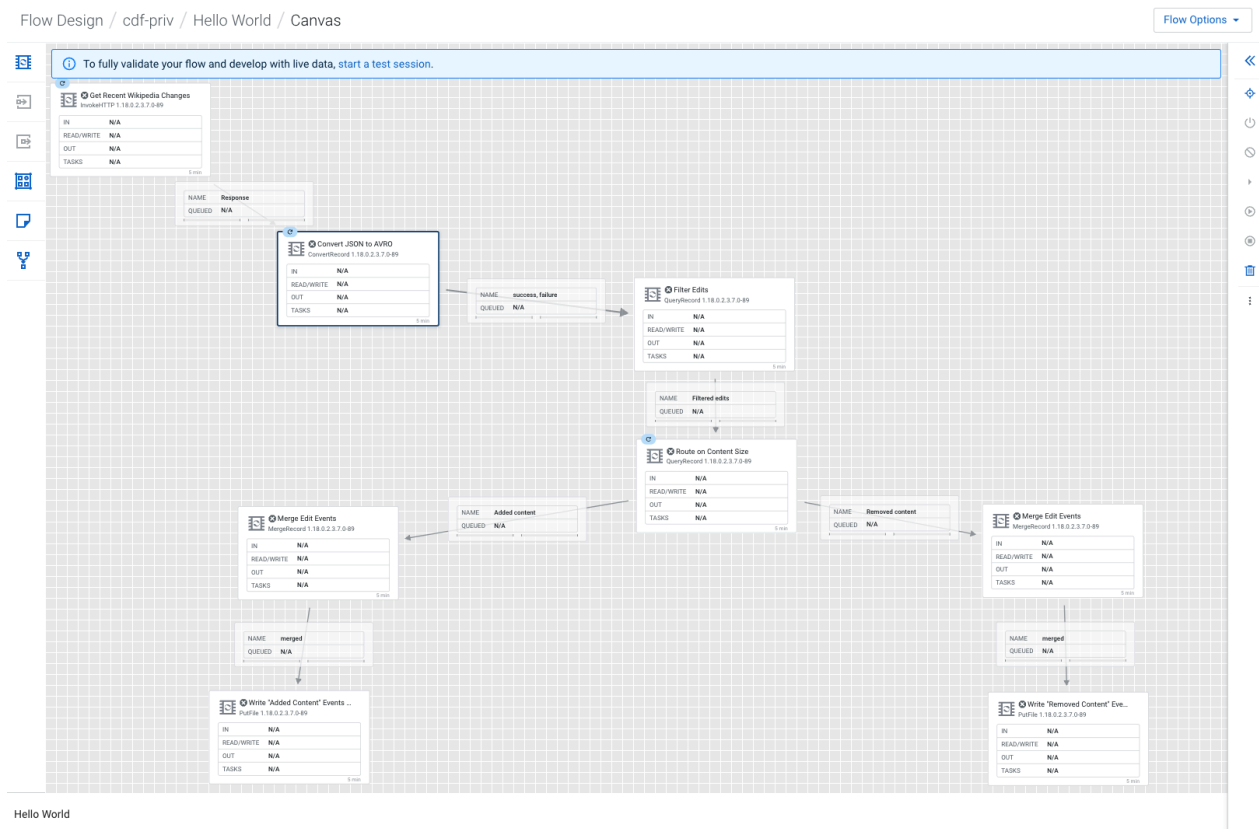
In the Create Connection pop up select merged and click Add.

27. Connect the Merge Edit Events processor with the Removed content connection to the Write "Removed Content" Events To File processor.

In the Create Connection pop up select merged and click Add.

Results

Congratulations, you have created your first draft flow. Now proceed to testing it by launching a Test Session.



Start a test session

To validate your draft, start a test session. This provisions an Apache NiFi cluster where you can test your draft.

About this task

Starting a Test Session provisions NiFi resources, acting like a development sandbox for a particular draft. It allows you to work with live data to validate your data flow logic while updating your draft. You can suspend a test session any time and change the configuration of the NiFi cluster then resume testing with the updated configuration.

Procedure

1. Click start a test session in the banner on top of the Canvas.



2. Click Start Test Session.

Test Session status **Initializing Test Session...** Initializing Test Session... appears on top of the page.


3. Wait for the status to change to

Active Test Session **Active Test Session**

This may take several minutes.

4. Click Flow Options Services to enable Controller Services.


5.

Select a service you want to enable, then click  Enable Service and Referencing Components.

This option does not only enable the controller service, but also any component that references it. This way, you do not need to enable the component separately to run the test session. In the context of this tutorial, enabling the 'AvroReader_Recent_Changes' controller service will also enable 'Filter Edits', 'Route on Content Size', and 'Merge Edit Events' processors as well.

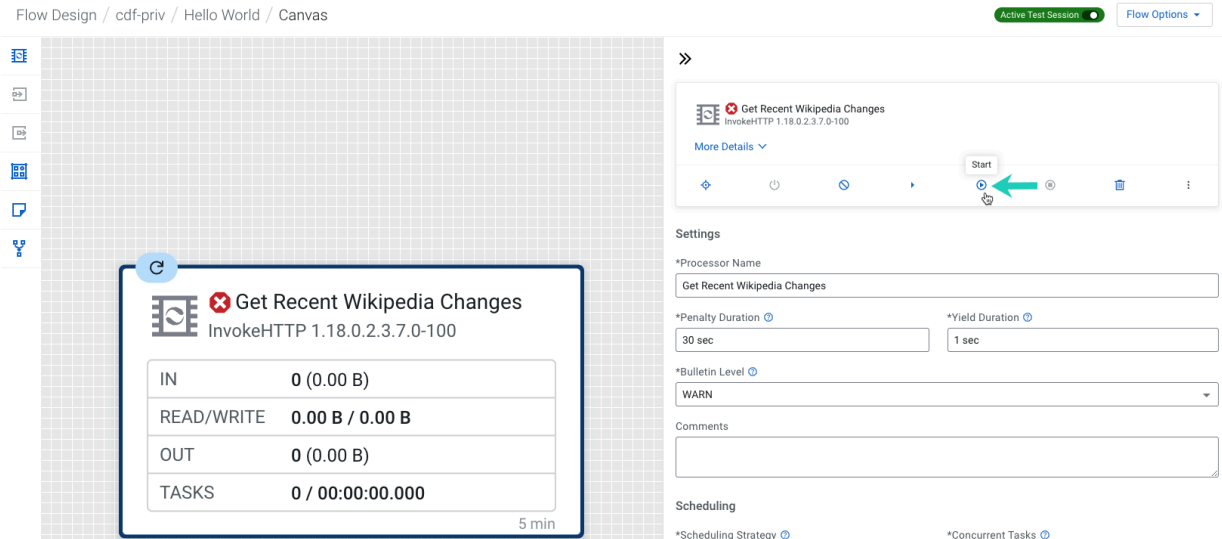
Repeat this step for all Controller Services.

6. Click Back To Flow Designer to return to the flow design Canvas.

7. Start the Get Recent Wikipedia Changes, Write "Added Content" Events To File, and Write "Removed Content" Events To File components by selecting them on the Canvas then clicking  Start.

Flow Design / cdf-priv / Hello World / Canvas

Active Test Session • Flow Options ▾



The screenshot shows the Cloudera DataFlow interface. On the left is a sidebar with icons for components. The main canvas displays a component named 'Get Recent Wikipedia Changes' (InvokeHTTP 1.18.0.2.3.7.0-100) with a blue border and a 'Start' button in its toolbar. A green arrow points to the 'Start' button. On the right, the configuration panel for this component is visible, showing settings for Processor Name, Penalty Duration, Yield Duration, Bulletin Level, and Comments.

IN	0 (0.00 B)
READ/WRITE	0.00 B / 0.00 B
OUT	0 (0.00 B)
TASKS	0 / 00:00:00.000

5 min

Settings

*Processor Name
Get Recent Wikipedia Changes

*Penalty Duration 30 sec *Yield Duration 1 sec

*Bulletin Level WARN

Comments

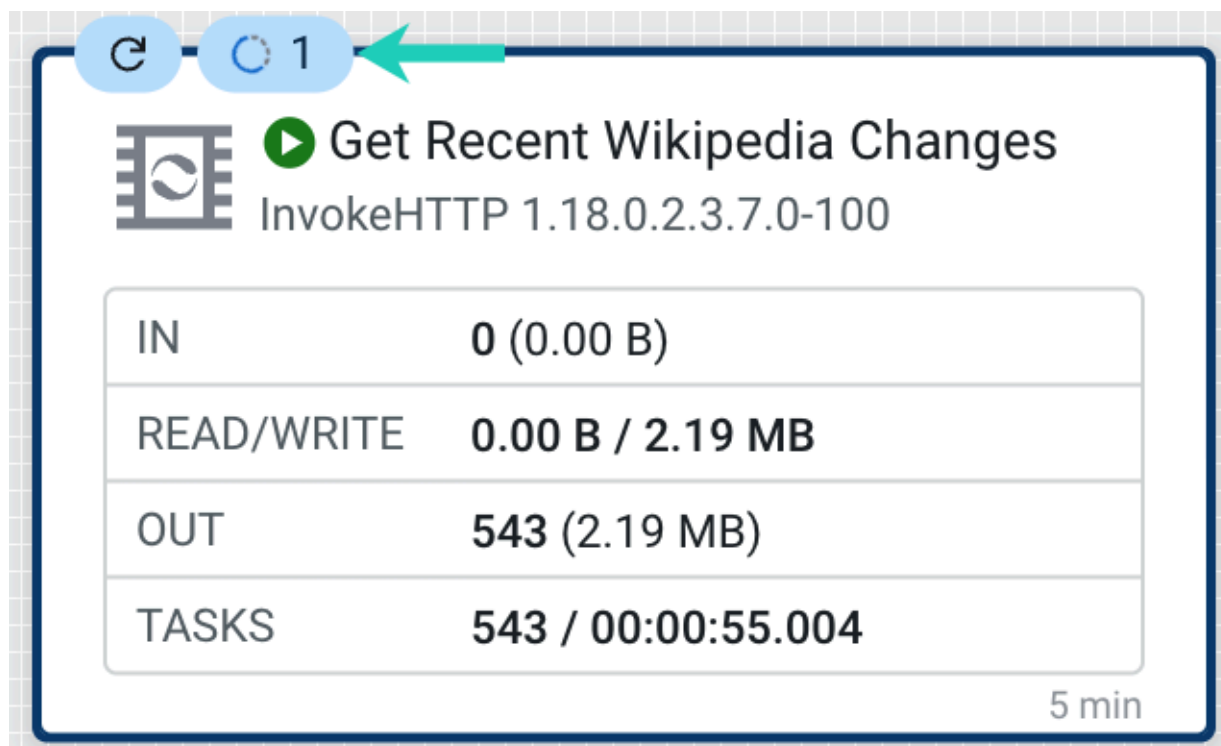
Scheduling

*Scheduling Strategy *Concurrent Tasks

All other components were auto-started when you selected the Enable Service and Referencing Components option.

8. Observe your first draft flow processing data.

On the Flow Design Canvas you can observe statistics on your processors change as they consume and process data from Wikipedia. You can also observe one or more blue Notification Pills, providing information about the current task.



Publish your flow definition to the Catalog

Now that you have tested your draft and it works fine, you can go on and publish it to the Catalog as a flow definition so that you can create a Cloudera DataFlow deployment.

Procedure

1. On the **Flow Designer** canvas, click **Flow Options Publish To Catalog Publish**.
2. Fill in the fields in the **Publish A New Flow** box.
 - Provide a Flow Name for your flow definition.
You can only provide a name when you publish your flow for the first time.
 - Optionally provide a Flow Description.
You can only provide a description when you publish your flow for the first time.
 - Optionally provide Custom Tags.
You can filter flow definition versions by tags in the Catalog.
 - Optionally provide Version Comments.
3. Click **Publish**.

Results

Your draft is published to the Catalog as a flow definition.

Related Information

[Deploying a flow definition](#)