

Proyecto Final

SPRINT 4

Arrancar HDFS, NIFI y la plataforma de Confluent.

```
bigdata@bigdatapc: ~/confluent-7.5.1
bigdata@bigdatapc: ~/hadoop-3.3.6 97x8
bigdata@bigdatapc:~$ cd /home/bigdata/hadoop-3.3.6/
bigdata@bigdatapc:~/hadoop-3.3.6$ sbin/start-dfs.sh
starting namenodes on [localhost]
starting datanodes
starting secondary namenodes [bigdatapc]
bigdata@bigdatapc:~/hadoop-3.3.6$

bigdata@bigdatapc: ~/confluent-7.5.1 97x14
bigdata@bigdatapc:~/confluent-7.5.1$ confluent local services kafka start
A minor version update is available for confluent from (current: v3.30.1, latest: v3.59.0).
To view release notes and install the update, please run `confluent update`.

The local commands are intended for a single-node development environment only, NOT for production
usage. See more: https://docs.confluent.io/current/cli/index.html
As of Confluent Platform 8.0, Java 8 will no longer be supported.

Using CONFLUENT_CURRENT: /tmp/confluent.511622
Starting ZooKeeper
ZooKeeper is [UP]
Starting Kafka
Kafka is [UP]
bigdata@bigdatapc:~/confluent-7.5.1$

bigdata@bigdatapc: ~/nifi-1.23.2 97x22
bigdata@bigdatapc:~$ cd /home/bigdata/nifi-1.23.2/
bigdata@bigdatapc:~/nifi-1.23.2$ bin/nifi.sh start

Java home: /usr/lib/jvm/java-8-openjdk-amd64
NiFi home: /home/bigdata/nifi-1.23.2

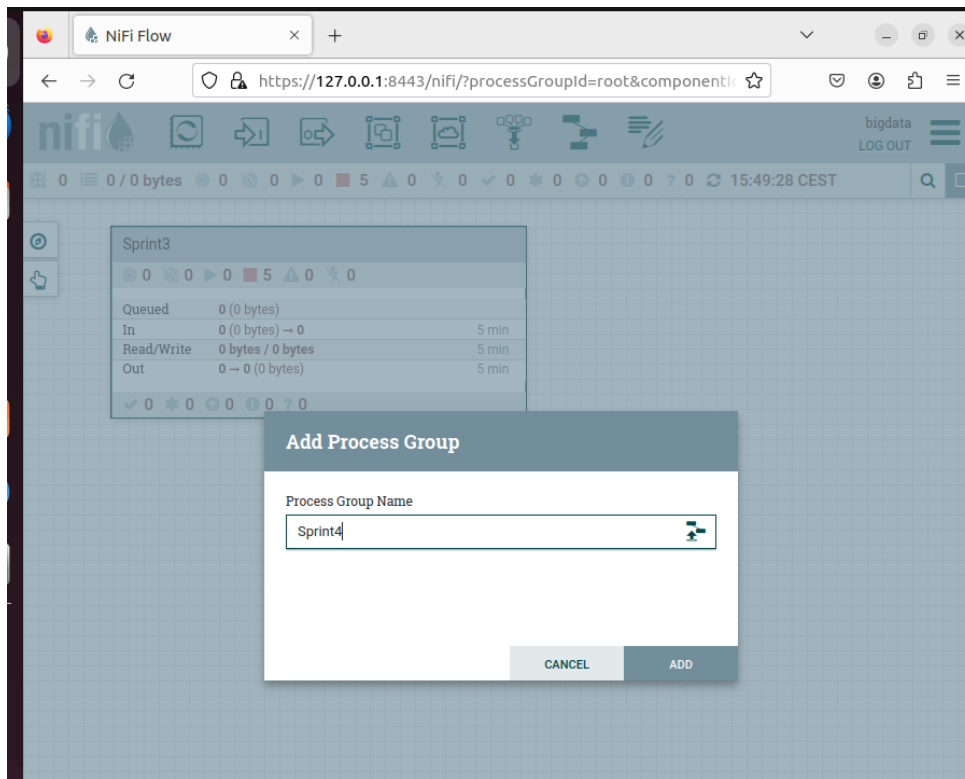
Bootstrap Config File: /home/bigdata/nifi-1.23.2/conf/bootstrap.conf

bigdata@bigdatapc:~/nifi-1.23.2$
```

Se pone un consumidor donde se volcaran las agregaciones que provienen de la ruta HDFS. Este lo volveremos a usar más adelante cuando lo hayamos definido en NiFi.

```
bigdata@bigdatapc:~/confluent-7.5.1$ bin/kafka-console-consumer --bootstrap-server localhost:9092 --topic aggregations
[2024-05-01 17:29:37,252] WARN [Consumer clientId=console-consumer, groupId=console-consumer-9081] Error while fetching metadata
with correlation id 2 : {aggregations=LEADER_NOT_AVAILABLE} (org.apache.kafka.clients.NetworkClient)
```

Accedemos a NiFi y creamos un nuevo Process Group.



El primer procesador que se añadirá es para monitorear datos de una carpeta HDFS, en esta ocasión se utilizara ListHDFS.

Add Processor

Source: all groups

Displaying 12 of 349

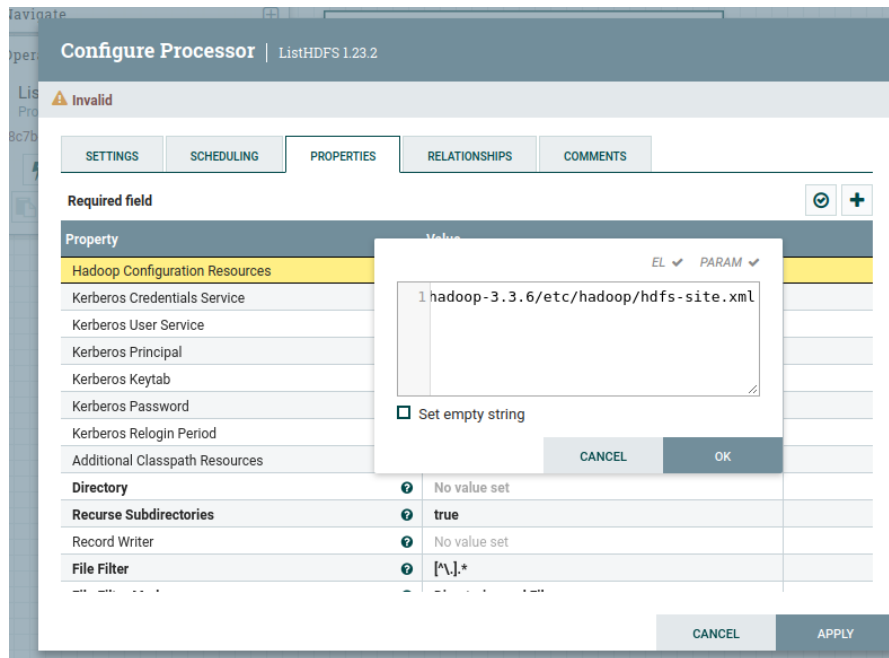
Type	Version	Tags
DeleteHDFS	1.23.2	restricted, HDFS, HCFS, hadoop...
FetchHDFS	1.23.2	restricted, get, fetch, hcfs, hdfs, ...
FetchParquet	1.23.2	restricted, get, fetch, record, HD...
GetHDFS	1.23.2	restricted, get, fetch, HDFS, HC...
GetHDFSEvents	1.23.2	inotify, hadoop, events, notificat...
GetHDFSFileInfo	1.23.2	get, HDFS, HCFS, hadoop, sourc...
GetHDFSSequenceFile	1.23.2	restricted, get, fetch, sequence f...
ListHDFS	1.23.2	get, HDFS, HCFS, hadoop, sourc...
MoveHDFS	1.23.2	move, restricted, HDFS, HCFS, ...
PutHDFS	1.23.2	restricted, HDFS, HCFS, hadoop...
PutKudu	1.23.2	database, NoSQL, record, HDFS...
PutParquet	1.23.2	restricted, record, HDFS, hadoop...

ListHDFS 1.23.2 org.apache.nifi - nifi-hadoop-nar

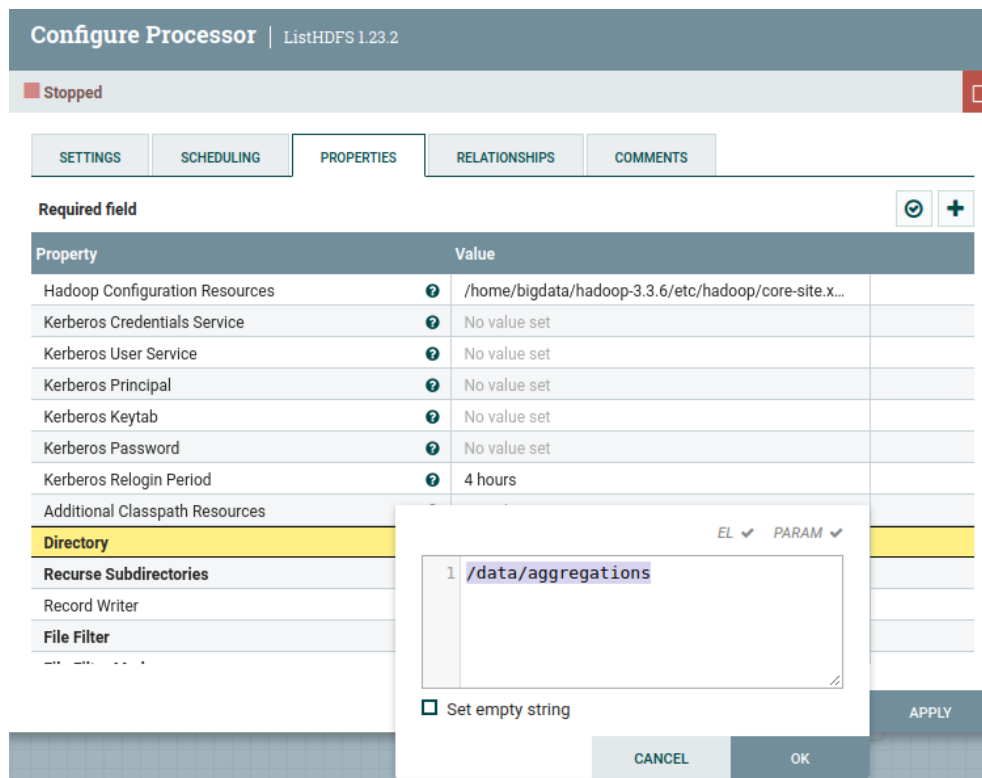
Retrieves a listing of files from HDFS. For each file that is listed in HDFS, this processor creates a FlowFile that represents the HDFS file to be fetched in conjunction with FetchHDFS. This Processor is designed to run on Primary Node only in a cluster. If the primary node changes, the new Primary Node will pick up where the previous node left off without duplicating all of the data. Unlike GetH...

Retrieves a listing of files from HDFS. For each file that is listed in HDFS, this processor creates a FlowFile that represents the HDFS file to be fetched in conjunction with FetchHDFS. This Processor is designed to run on Primary Node only in a cluster. If the primary node changes, the new Primary Node will pick up where the previous node left off without duplicating all of the data. Unlike GetHDFS, this Processor does not delete any data from HDFS.

Se establecen los ficheros de configuración de HDFS.



Se establece la dirección del directorio



El siguiente procesador será fetch parquet para poder leer el contenido de un fichero parquet y escribir el contenido en un flow file.

Add Processor

Source

all groups

Displaying 3 of 349

parque

Type	Version	Tags
ConvertAvroToParquet	1.23.2	convert, avro, parquet
FetchParquet	1.23.2	restricted, get, fetch, record, HD...
PutParquet	1.23.2	restricted, record, HDFS, hadoo...

amazon attributes
avro aws azure
cloud consume
csv fetch get
google ingest
json listen logs
message
microsoft pubsub
put record
restricted source
storage text
update

FetchParquet 1.23.2 org.apache.nifi - nifi-parquet-nar

Reads from a given Parquet file and writes records to the content of the flow file using the selected record writer. The original Parquet file will remain unchanged, and the content of the flow file will be replaced with records of the selected type. This processor can be used with ListHDFS or ListFile to obtain a listing of files to fetch.

CANCEL

ADD

Se conecta la relación de success del anterior procesador, con este.

Create Connection

DETAILS

SETTINGS

From Processor

ListHDFS

ListHDFS

Within Group

Sprint4

For Relationships

☒ success

To Processor

FetchParquet

FetchParquet

Within Group

Sprint4

CANCEL

ADD

Este procesador también necesita la configuración de hadoop.

Configure Processor | FetchParquet 1.23.2

Invalid

SETTINGS SCHEDULING **PROPERTIES** RELATIONSHIPS COMMENTS

Required field

Property	Value
Hadoop Configuration Resources	1 hadoop-3.3.6/etc/hadoop/hdfs-site.xml
Kerberos Credentials Service	
Kerberos User Service	
Kerberos Principal	
Kerberos Keytab	
Kerberos Password	
Kerberos Relogin Period	
Additional Classpath Resources	
Filename	\$(path)/\$(filename)
Record Writer	No value set

Set empty string

CANCEL OK

CANCEL APPLY

Se necesita configurar un record writer, se le da a la opción de create new service.

Configure Processor | FetchParquet 1.23.2

Invalid

SETTINGS SCHEDULING **PROPERTIES** RELATIONSHIPS COMMENTS

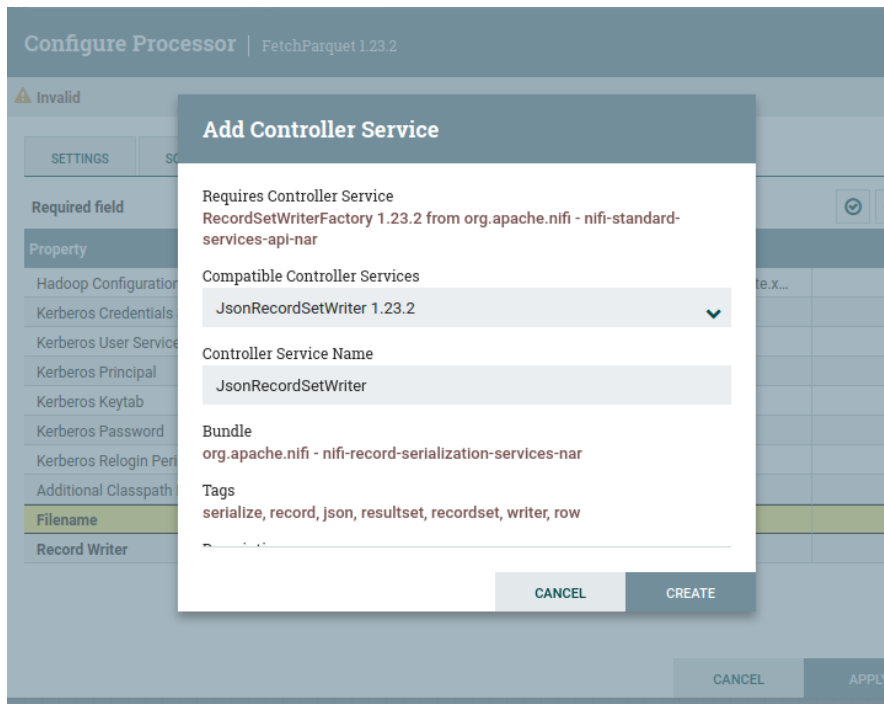
Required field

Property	Value
Hadoop Configuration Resources	/home/bigdata/hadoop-3.3.6/etc/hadoop/core-site.x...
Kerberos Credentials Service	No value set
Kerberos User Service	No value set
Kerberos Principal	No value set
Kerberos Keytab	No value set
Kerberos Password	No value set
Kerberos Relogin Period	4 hours
Additional Classpath Resources	No value set
Filename	
Record Writer	

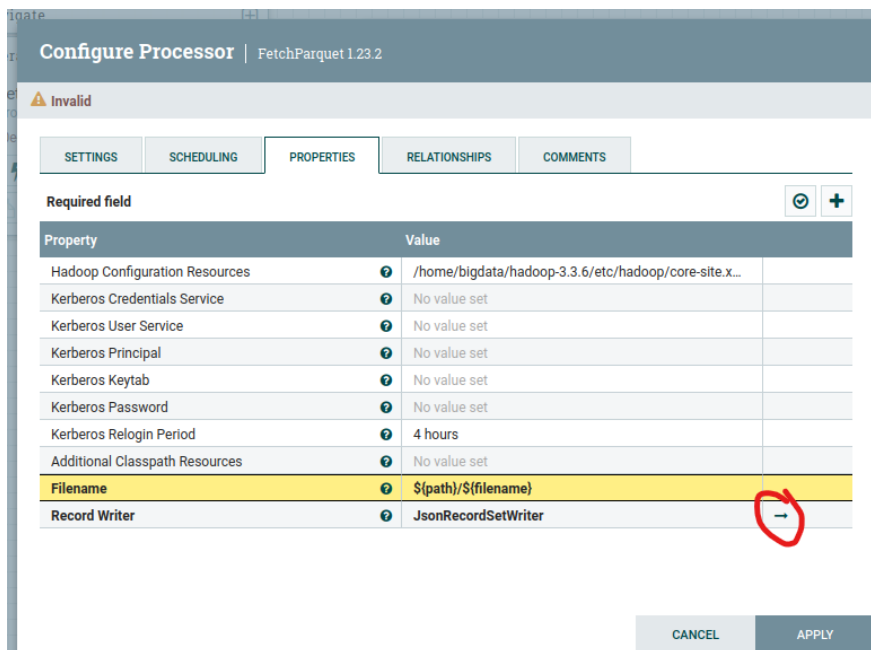
Reference parameter...
Reference parameter...
Create new service...

CANCEL OK APPLY

Se selecciona uno de JSON.



Le damos a la flecha para configurarlo.



Se establece el schema.name para propagar el esquema de los flows files.

Configure Controller Service | JsonRecordSetWriter 1.23.2

SETTINGS

PROPERTIES

COMMENTS

Required field

Property

Schema Write Strategy	Set 'schema.name' Attribute
Schema Cache	
Schema Access Strategy	
Date Format	No value set
Time Format	No value set
Timestamp Format	No value set
Pretty Print JSON	false
Suppress Null Values	Never Suppress
Output Grouping	Array
Compression Format	none

CANCEL

OK

CANCEL

APPLY

Configure Controller Service | JsonRecordSetWriter 1.23.2

SETTINGS

PROPERTIES

COMMENTS

Required field

Property

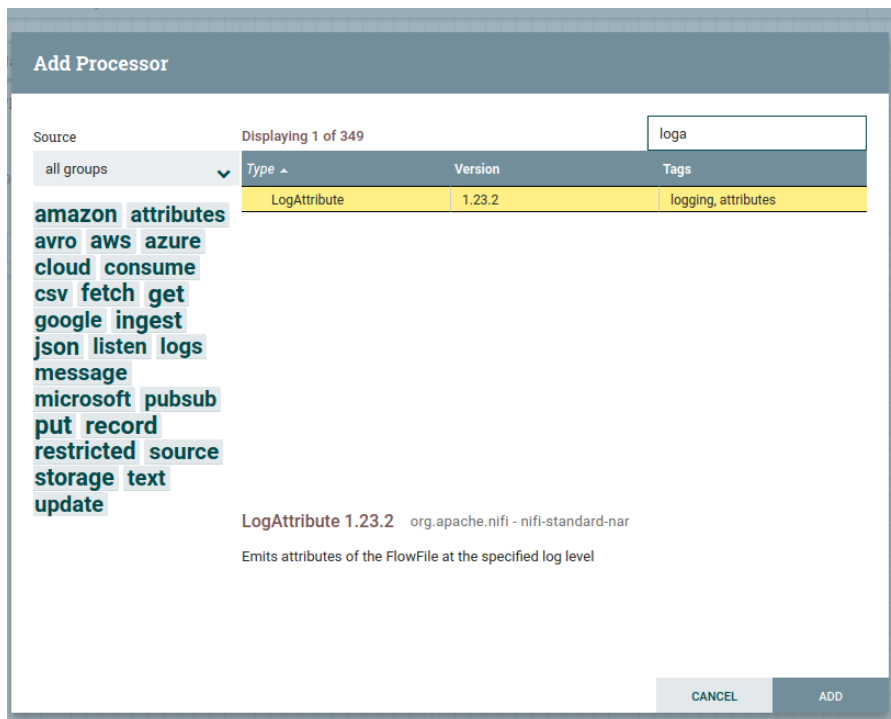
Value

Schema Write Strategy	Set 'schema.name' Attribute
Schema Cache	No value set
Schema Access Strategy	Inherit Record Schema
Date Format	No value set
Time Format	No value set
Timestamp Format	No value set
Pretty Print JSON	false
Suppress Null Values	Never Suppress
Output Grouping	Array
Compression Format	none

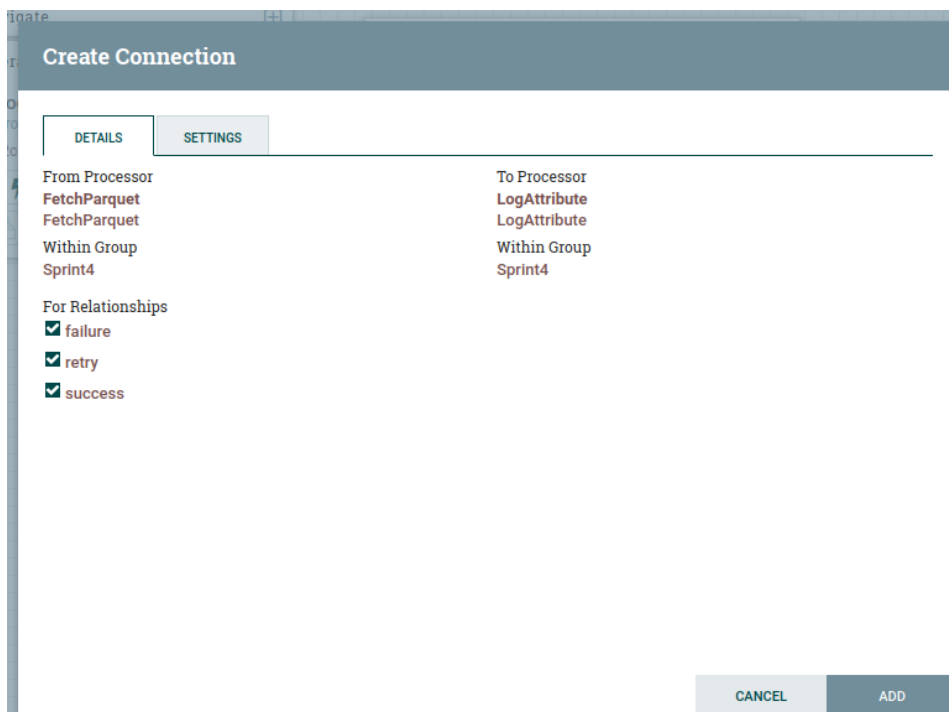
CANCEL

APPLY

Agregamos otro procesador de log attribute, en caso de que algo falle tener el registro.



Se relaciona el fetchparquet con el log attribute de manera que registre todos los eventos de fallo, success y retry.



En el log attribute para remover la señal de warning solo debemos terminar la relación de success.

Configure Processor | LogAttribute 1.23.2

Invalid

SETTINGS

SCHEDULING

PROPERTIES

RELATIONSHIPS

COMMENTS

Automatically Terminate / Retry Relationships

success

☒ terminate

☐ retry

All FlowFiles are routed to this relationship

CANCEL

APPLY

Para volcar el fichero tipo json en un topic de kakfa, usaremos el publish Kafka 2.6 que envía el contenido de un flow file como un mensaje.

Add Processor

Source

Displaying 12 of 349

kafka

all groups

amazon

attributes

avro

aws

azure

cloud

consume

csv

fetch

get

google

ingest

json

listen

logs

message

microsoft

pubsub

put

record

restricted

source

storage

text

update

Type	Version	Tags
ConsumeKafkaRecord_1_0	1.23.2	PubSub, Consume, 1.0, Ingest, ...
ConsumeKafkaRecord_2_0	1.23.2	PubSub, Consume, Ingest, 2.0, ...
ConsumeKafkaRecord_2_6	1.23.2	PubSub, Consume, Ingest, Get, ...
ConsumeKafka_1_0	1.23.2	PubSub, Consume, 1.0, Ingest, ...
ConsumeKafka_2_0	1.23.2	PubSub, Consume, Ingest, 2.0, ...
ConsumeKafka_2_6	1.23.2	PubSub, Consume, Ingest, Get, ...
PublishKafkaRecord_1_0	1.23.2	PubSub, 1.0, Message, csv, Kaf...
PublishKafkaRecord_2_0	1.23.2	PubSub, Message, 2.0, csv, Kaf...
PublishKafkaRecord_2_6	1.23.2	PubSub, Message, csv, Kafka, J...
PublishKafka_1_0	1.23.2	PubSub, 1.0, Message, Kafka, A...
PublishKafka_2_0	1.23.2	PubSub, Message, 2.0, Kafka, A...
PublishKafka_2_6	1.23.2	PubSub, Message, Kafka, 2.6, A...

PublishKafka_2_6 1.23.2

org.apache.nifi - nifi-kafka-2-6-nar

Sends the contents of a FlowFile as a message to Apache Kafka using the Kafka 2.6 Producer API. The messages to send may be individual FlowFiles or may be delimited, using a user-specified delimiter, such as a new-line. The complementary NiFi processor for fetching messages is ConsumeKafka_2_6.

CANCEL

ADD

Configuramos la relación de publish Kafka con log attribute cuando ocurra un error.

Create Connection

DETAILS | SETTINGS

From Processor
PublishKafka_2_6
PublishKafka_2_6

To Processor
LogAttribute
LogAttribute

Within Group
Sprint4
Sprint4

For Relationships
☒ failure
☐ success

CANCEL ADD

Y la relación de success se termina.

Configure Processor | PublishKafka_2_6 1.23.2

Invalid

SETTINGS | SCHEDULING | PROPERTIES | **RELATIONSHIPS** | COMMENTS

Automatically Terminate / Retry Relationships ⓘ

failure

☐ terminate ☐ retry

Any FlowFile that cannot be sent to Kafka will be routed to this Relationship

success

☒ terminate ☐ retry

FlowFiles for which all content was sent to Kafka.

CANCEL APPLY

Como este procesador necesita una entrada le indicamos la relación con el fetch parquet y de esta manera cuando sea success este pase los resultados.

Create Connection

DETAILS

SETTINGS

From Processor

FetchParquet

FetchParquet

Within Group

Sprint4

For Relationships

☐ failure

☐ retry

☒ success

To Processor

PublishKafka_2_6

PublishKafka_2_6

Within Group

Sprint4

CANCEL

ADD

Se configura el topic name con el nombre que se había puesto en el consumidor anteriormente.

Configure Processor | PublishKafka_2_6 1.23.2

Stopped

SETTINGS

SCHEDULING

PROPERTIES

RELATIONSHIPS

COMMENTS

Required field

Property

Value

Kafka Brokers	localhost:9092
Topic Name	aggregations
Use Transactions	true
Transactional Id Prefix	No value set
Message Demarcator	No value set
Failure Strategy	Route to Failure
Delivery Guarantee	Guarantee Replicated Delivery
Attributes to Send as Headers (Regex)	No value set
Message Header Encoding	UTF-8
Security Protocol	PLAINTEXT
SASL Mechanism	GSSAPI
Kerberos Credentials Service	No value set

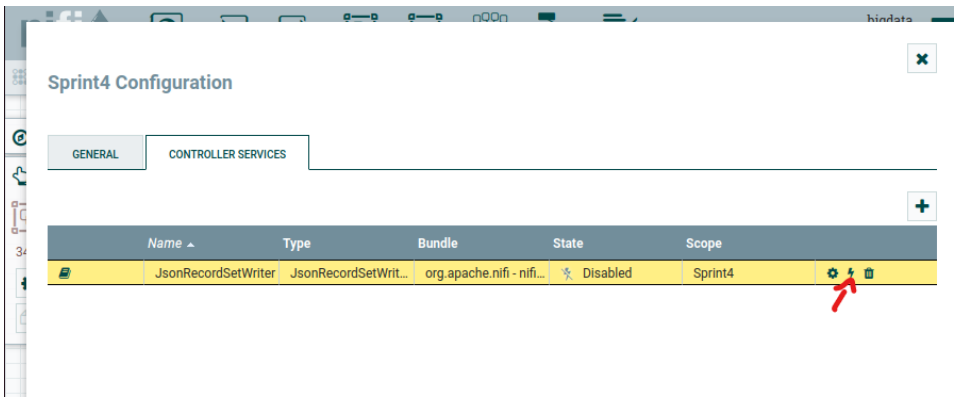
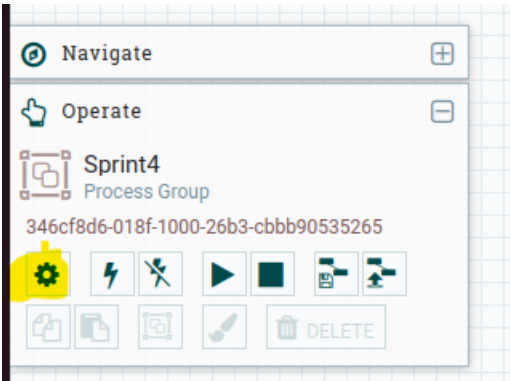
CANCEL

APPLY

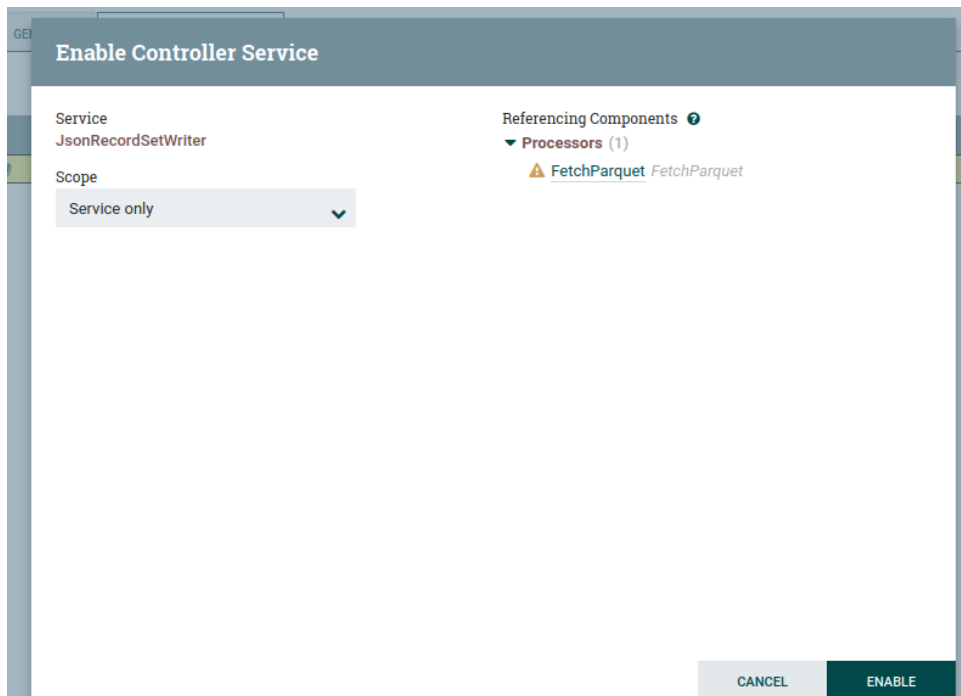
Emitimos los logs de nifi para ver cómo van fluyendo los datos, los atributos y contenidos del flow file.

```
bigdata@bigdatapc: ~/nifi-1.23.2 100x22
2024-05-01 17:18:41,276 INFO [pool-7-thread-1] o.a.n.c.r.WriteAheadFlowFileRepository Initiating the
ckpoint of FlowFile Repository
2024-05-01 17:18:41,276 INFO [pool-7-thread-1] o.a.n.c.r.WriteAheadFlowFileRepository Successfully c
heckpointed FlowFile Repository with 0 records in 0 milliseconds
2024-05-01 17:18:56,885 INFO [Write-Ahead Local State Provider Maintenance] org.wali.MinimalLockingW
riteAheadLog org.wali.MinimalLockingWriteAheadLog@7c840fe3 checkpointed with 78 Records and 0 Swap F
iles in 13 milliseconds (Stop-the-world time = 3 milliseconds, Clear Edit Logs time = 1 millis), max
Transaction ID 376
2024-05-01 17:18:58,162 INFO [Cleanup Archive for default] o.a.n.c.repository.FileSystemRepository S
uccessfully deleted 0 files (0 bytes) from archive
2024-05-01 17:18:58,163 INFO [Cleanup Archive for default] o.a.n.c.repository.FileSystemRepository A
rchive cleanup completed for container default; will now allow writing to this container. Bytes used
= 21,73 GB, bytes free = 2,22 GB, capacity = 23,94 GB
2024-05-01 17:19:01,278 INFO [pool-7-thread-1] o.a.n.c.r.WriteAheadFlowFileRepository Initiating the
ckpoint of FlowFile Repository
2024-05-01 17:19:01,278 INFO [pool-7-thread-1] o.a.n.c.r.WriteAheadFlowFileRepository Successfully c
heckpointed FlowFile Repository with 0 records in 0 milliseconds
2024-05-01 17:19:21,280 INFO [pool-7-thread-1] o.a.n.c.r.WriteAheadFlowFileRepository Initiating the
ckpoint of FlowFile Repository
2024-05-01 17:19:21,280 INFO [pool-7-thread-1] o.a.n.c.r.WriteAheadFlowFileRepository Successfully c
heckpointed FlowFile Repository with 0 records in 0 milliseconds
```

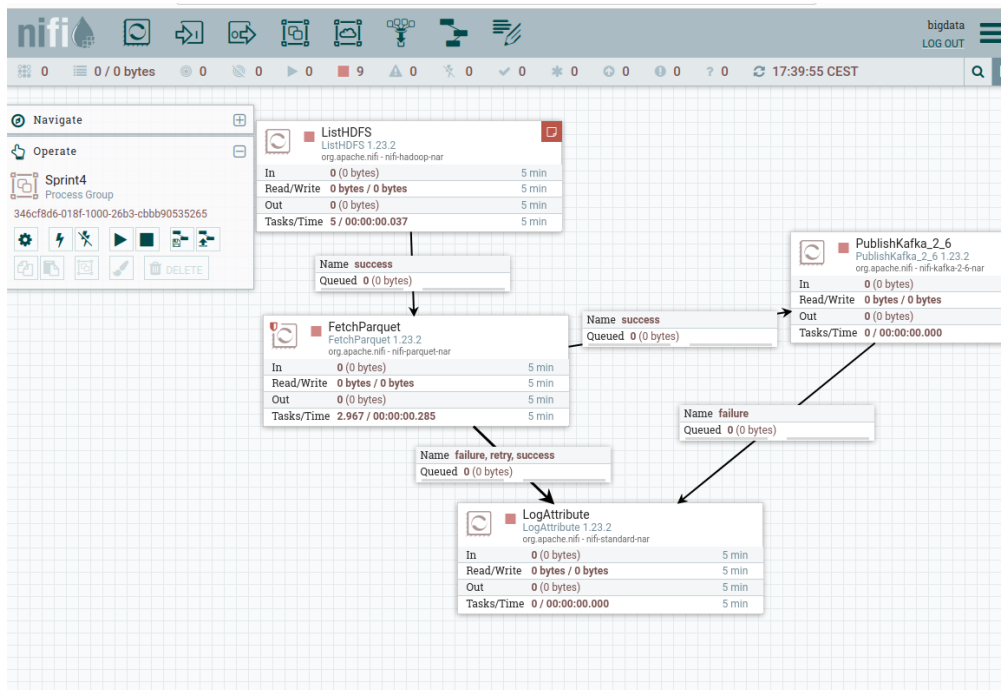
Volvemos a la interfaz y vemos que servicios están sin activar.



Y lo activamos.



Luego de correrlo, vemos que tenemos un error en el ListHDFS debido a que no encuentra el archivo al que nos estamos refiriendo.



Por lo que vamos a Hive a levantar una consola.

```
bigdata@bigdatapc: ~/apache-hive-3.1.3-bin 128x10
24/05/01 17:44:13 [main]: WARN DataNucleus.Metadata: Metadata has jdbc-type of null yet this is not valid. Ignored
24/05/01 17:44:13 [main]: WARN DataNucleus.Metadata: Metadata has jdbc-type of null yet this is not valid. Ignored
24/05/01 17:44:13 [main]: WARN DataNucleus.Metadata: Metadata has jdbc-type of null yet this is not valid. Ignored
24/05/01 17:44:13 [main]: WARN DataNucleus.Metadata: Metadata has jdbc-type of null yet this is not valid. Ignored
Connected to: Apache Hive (version 3.1.3)
Driver: Hive JDBC (version 3.1.3)
Transaction isolation: TRANSACTION_REPEATABLE_READ
Beeline version 3.1.3 by Apache Hive
0: jdbc:hive2://>
```

Se generara una tabla que se llame aggregations y que contiene los datos que está esperando la ruta hdfs.

CREATE TABLE aggregations

STORED AS parquet

LOCATION '/data/aggregations/'

AS

SELECT

a.country,

COUNT(*) AS total_rutas_sin_paradas

FROM

routes r

JOIN

aerolineas a ON r.airlineID = a.airlineID

JOIN

aeropuertos aero ON r.sourceAirportID = aero.airportID

WHERE

r.stops = 0

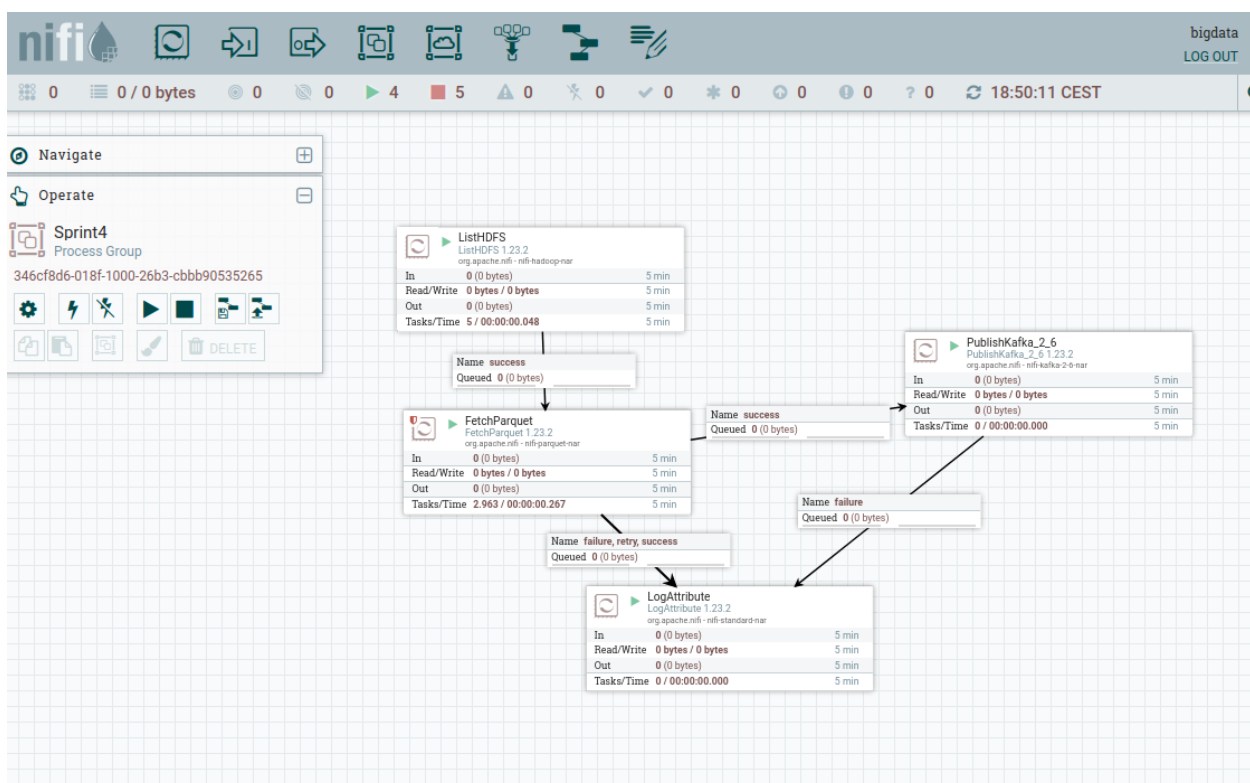
AND a.active = FALSE

AND aero.altitude > 5000


GROUP BY

a.country;

```
bigdata@bigdatapc: ~/apache-hive-3.1.3-bin-128x21
at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:624) [?:1.8.0_382]
at java.lang.Thread.run(Thread.java:750) [?:1.8.0_382]
24/05/01 18:29:27 [pool-21-thread-1]: WARN hadoop.MemoryManager: Total allocation exceeds 50,00% (130.809.856 bytes) of heap memory
Scaling row group sizes to 97,46% for 1 writers
2024-05-01 18:29:28,006 Stage-3 map = 100%, reduce = 0%
2024-05-01 18:29:29,022 Stage-3 map = 100%, reduce = 100%
Ended Job = job_local1234549444_0001
24/05/01 18:29:29 [HiveServer2-Background-Pool: Thread-89]: WARN metastore.ObjectStore: datanucleus.autoStartMechanismMode is set to unsupported value null. Setting it to value: ignored
Moving data to directory /data/aggregations
24/05/01 18:29:29 [HiveServer2-Background-Pool: Thread-89]: WARN metastore.HiveMetaStore: Location: hdfs://localhost:9000/data/aggregations specified for non-external table:aggregations
24/05/01 18:29:29 [HiveServer2-Background-Pool: Thread-89]: WARN metastore.ObjectStore: datanucleus.autoStartMechanismMode is set to unsupported value null. Setting it to value: ignored
MapReduce Jobs Launched:
Stage-Stage-3: HDFS Read: 6711110 HDFS Write: 249 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK
No rows affected (12,28 seconds)
0: jdbc:hive2://>
```




Browse Directory

/data/aggregations Go! 

Show 25 entries Search:

software

<input type="checkbox"/>	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
<input type="checkbox"/>	-rw-r--r--	bigdata	supergroup	203 B	May 01 18:29	1	128 MB	000000_0 

Showing 1 to 1 of 1 entries Previous 1 Next

Hadoop, 2023.

SPRINT 5

Arrancar KSql y todos los servicios requeridos para ello.

```
bigdata@bigdatapc: ~/confluent-7.5.1 128x8
Using CONFLUENT_CURRENT: /tmp/confluent.511622
ZooKeeper is [UP]
Kafka is [UP]
Starting Schema Registry
Schema Registry is [UP]
Starting ksqlDB Server
ksqlDB Server is [UP]
bigdata@bigdatapc:~/confluent-7.5.1$
```

Subimos una consola de ksql

```
bigdata@bigdatapc: ~/confluent-7.5.1 128x8
Copyright 2017-2022 Confluent Inc.

CLI v7.5.1, Server v7.5.1 located at http://localhost:8088
Server Status: RUNNING

Having trouble? Type 'help' (case-insensitive) for a rundown of how things work!

ksql>
```

Usamos este commando para crear el stream

```
CREATE STREAM data(country string, total_rutas_sin_paardas int) WITH (KAFKA_TOPIC='aggregations',
VALUE_FORMAT='JSON');
```



```
bigdata@bigdatapc: ~/confluent-7.5.1 128x8
ksql> CREATE STREAM data(country string, total_rutas_sin_paardas int) WITH (KAFKA_TOPIC='aggregations', VALUE_FORMAT='JSON');
Message
-----
Stream created
-----
ksql>
```

Ejecutamos este query para que sea continua de forma permanente los cambios que lleguen.

```
bigdata@bigdatapc: ~/confluent-7.5.1 128x8
Stream created
-----
ksql> select * from data EMIT CHANGES;
+-----+-----+
|COUNTRY|TOTAL_RUTAS_SIN_PAARDAS|
+-----+-----+
Press CTRL-C to interrupt
bigdata@bigdatapc: ~/nifi-1.23.2 128x12
```

Vamos a NiFi y copiamos y pegamos el process group que teníamos para el 4 y le cambiamos el nombre

Copy of Sprint4 Configuration

GENERAL CONTROLLER SERVICES

Process Group Name
Sprint5

Process Group Parameter Context
No parameter context

Process Group Comments

Process Group FlowFile Concurrency
Unbounded

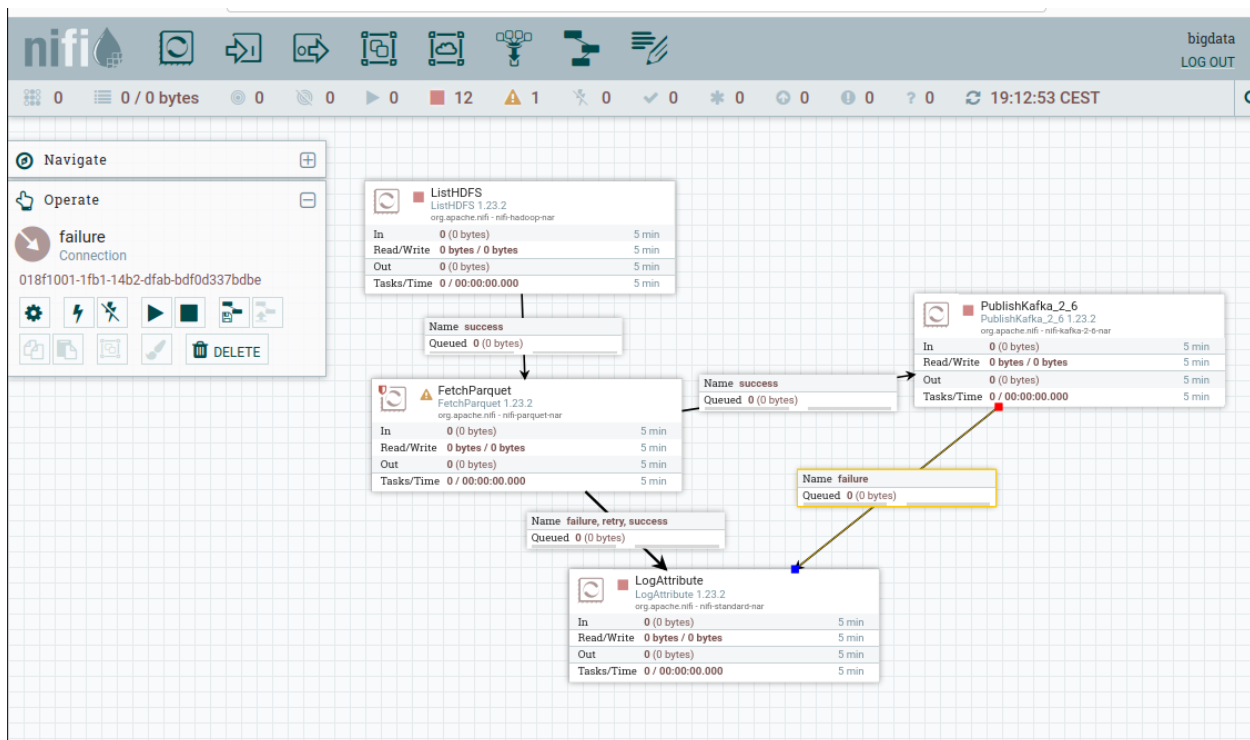
Process Group Outbound Policy
Stream When Available

Default FlowFile Expiration
0 sec

Default Back Pressure Object Threshold
10000

Last updated: 19:10:29 CEST

Se elimina la relación entre log attribute y publish Kafka para poder añadir un nuevo procesador.



Usamos este llamado Split json para separar un flow file en una raíz de elementos.

Add Processor

Source: all groups

Displaying 32 of 349

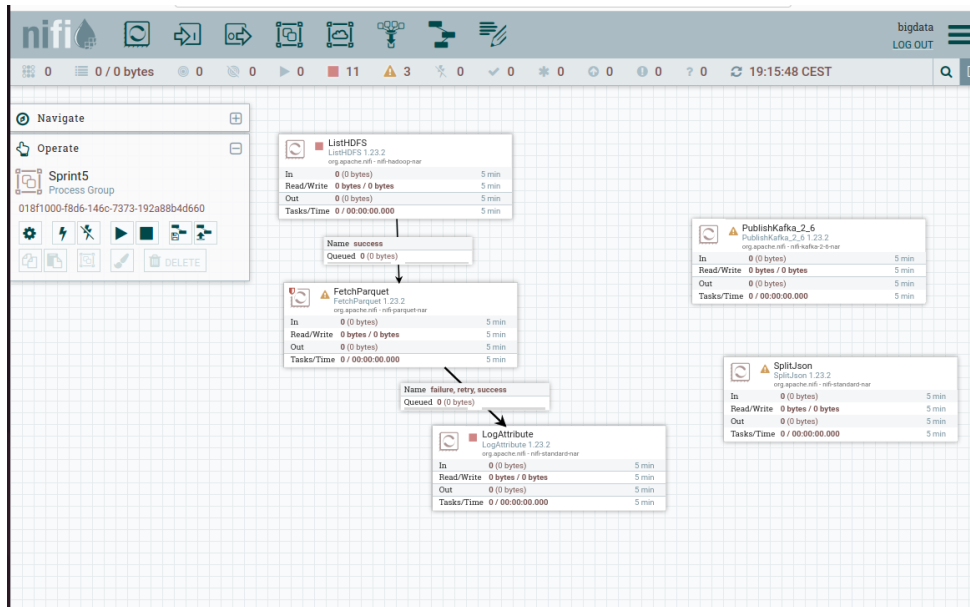
Type	Version	Tags
PublishKafkaRecord_2_0	1.23.2	PubSub, Message, z.U, csv, Kafka, json, Rec...
PublishKafkaRecord_2_6	1.23.2	PubSub, Message, csv, Kafka, json, Record, ...
PutElasticsearchJson	1.23.2	elasticsearch, elasticsearch7, json, elastics...
PutElasticsearchRecord	1.23.2	elasticsearch, record, elasticsearch7, json, ...
PutHBaseJSON	1.23.2	json, hadoop, hbase, put
QueryRecord	1.23.2	select, query, csv, update, calcite, sql, aggre...
RemoveRecordField	1.23.2	schema, record, csv, freeform, update, json, ...
SearchElasticsearch	1.23.2	search, elasticsearch, query, elasticsearch7...
SplitJson	1.23.2	split, jsonpath, json
SplitRecord	1.23.2	schema, split, log, csv, freeform, json, text, L...
UpdateRecord	1.23.2	schema, log, record, csv, freeform, update, j...
ValidateJson	1.23.2	schema, restricted, JSON, validation

SplitJson 1.23.2 org.apache.nifi - nifi-standard-nar

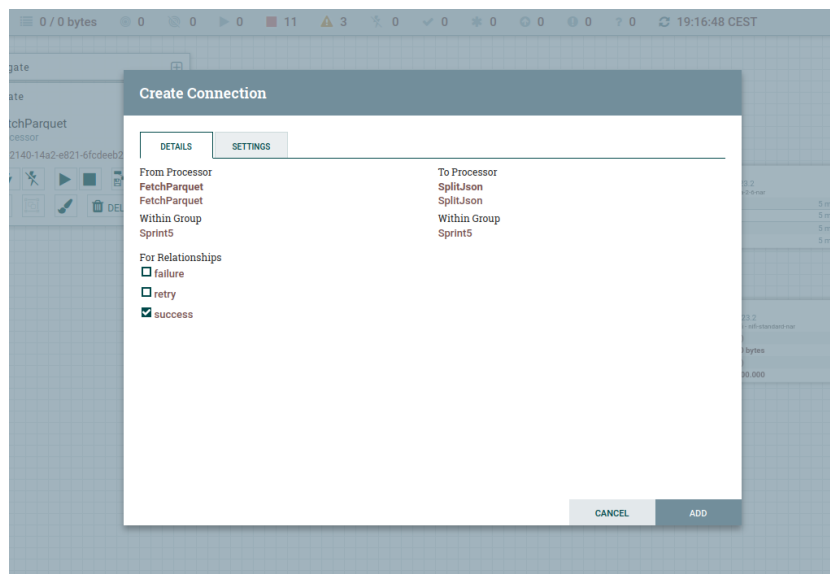
Splits a JSON File into multiple, separate FlowFiles for an array element specified by a JsonPath expression. Each generated FlowFile is comprised of an element of the specified array and transferred to relationship 'split,' with the original file transferred to the 'original' relationship. If the specified JsonPath is not found or does not evaluate to an array element, the original file is routed to 'failure' and no files are generated.

CANCEL ADD

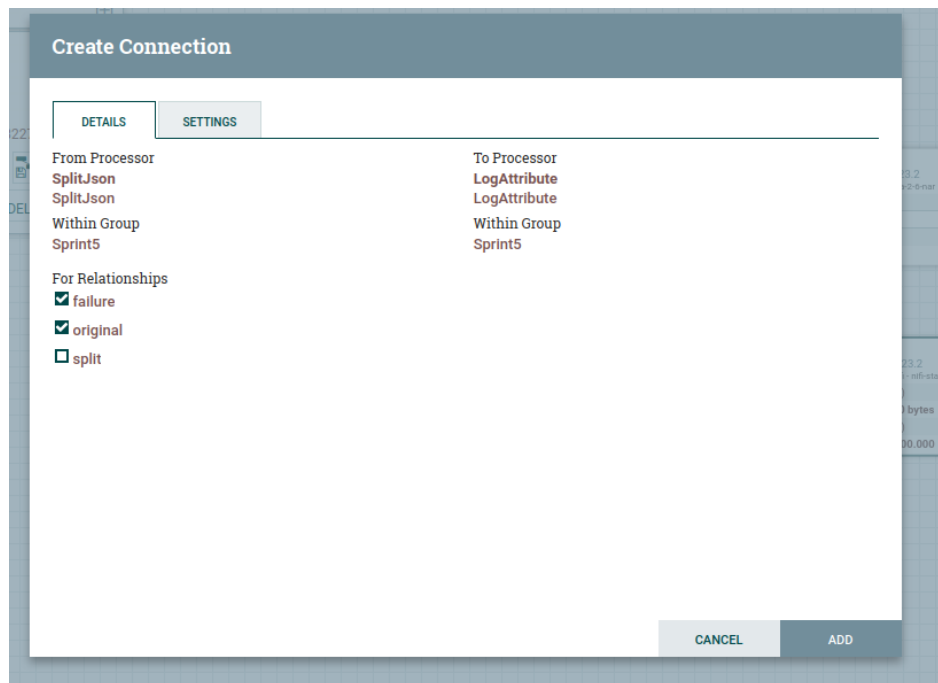
Cambiamos la relación de fetch parquet para que no llegue directamente a publish Kafka si no que pase por Split json y tengamos diferentes elementos de salida.



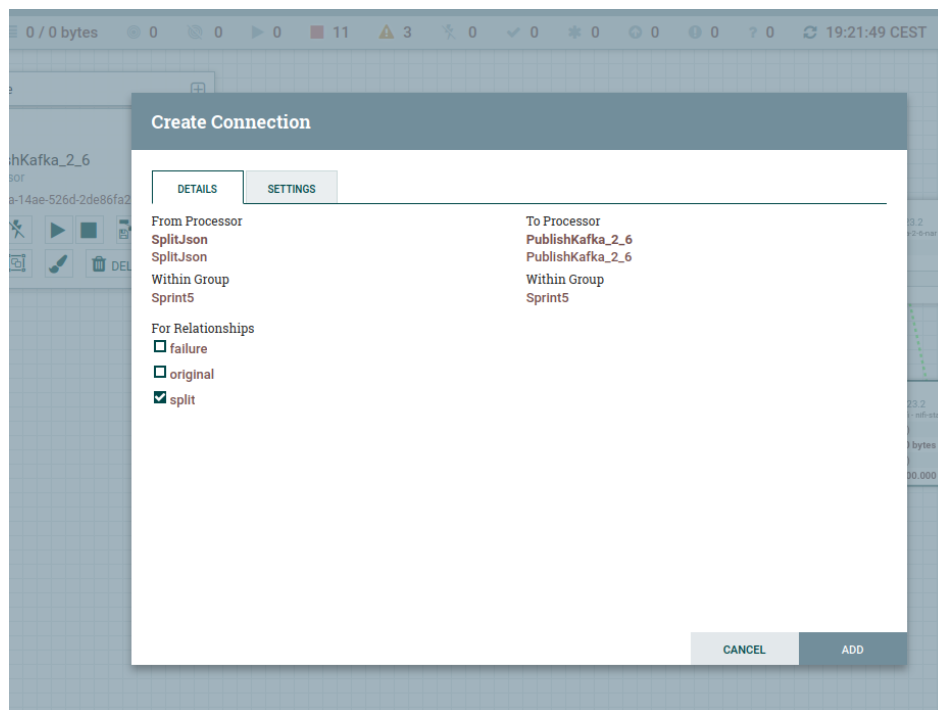
Cuando sea de éxito, lo relacionamos.



Conectamos el Split json con log attribute y de esta manera cuando falle o tenga el original lo registrara.



Y cuando haya hecho bien el Split, entonces lo pasa al procesador de Kafka



Configuramos el json file de manera que cada evento se convertirá en un flow file diferente al usar el signo de \$.

Configure Processor | SplitJson 1.23.2

Invalid

SETTINGS SCHEDULING PROPERTIES RELATIONSHIPS COMMENTS

Required field

Property Value

Property	Value
JsonPath Expression	
Null Value Representation	

EL PARAM ✓

1 \$

☐ Set empty string

CANCEL OK

CANCEL APPLY

Configuramos el procesador de Kafka de manera que cuando haya un fallo este sea terminado.

Configure Processor | PublishKafka_2_6 1.23.2

Invalid

SETTINGS SCHEDULING PROPERTIES RELATIONSHIPS COMMENTS

Automatically Terminate / Retry Relationships ⓘ

failure

☒ terminate ☐ retry

Any FlowFile that cannot be sent to Kafka will be routed to this Relationship

success

☒ terminate ☐ retry

FlowFiles for which all content was sent to Kafka.

CANCEL APPLY

Y activamos el controller para el fetch parquet

Sprint5 Configuration

GENERAL

CONTROLLERS

Enable Controller Service

Service
JsonRecordSetWriter

Scope

Service only

Referencing Components

Processors (1)

FetchParquet FetchParquet

CANCEL

ENABLE

