

オープンソースのデータフローエンジン <<Apache NiFi>>を利用して 簡単なデータフローを動かしてみよう！

～超初心者向け、お気軽に！～

データフロー活用研究会
in OSC 2019 Hiroshima

今日は何を話す？

- ▶ Apache NiFiというオープンソースのソフトウェアについて紹介します！
 - ▶ とっても初心者向けです。
ごゆるりと聞いてください

自己紹介

▶ 松山千尋

- ▶ 出身は沖縄県
- ▶ 都内にてSEをしています。
- ▶ 広島に来るのは人生二度目です。
 - ▶ ↓これを飲みにこの後宮島に行きます



Apache NiFi って？？

- ▶ まずは読み方

あぱっち ないふあい

- ▶ ライセンスは？

Apache License Version 2.0

- ▶ なにで書かれている？

Java言語

Apache NiFiは、Apache Software Foundationの米国およびその他の国における登録商標または商標です。

Apache NiFi で何が出来る？

Apache NiFiは、

データフローオーケストレーションツール！

データの流れ(データフロー)をモデリングする環境と、そのモデリング内容に則った動作をする実行環境を提供します。

- ▶ ブラウザ（WebベースUI）でデータフローをお絵かき
- ▶ お絵かきしたデータフローがボタンひとつで動作

Apache NiFi の歴史

- ▶ 元は米国家安全保障局（NSA）により「Niagara Files」として開発されていたもの
- ▶ 「**Ni**agara **F**iles」 → 「NiFi」の語源
- ▶ 2014年にオープンソースソフトウェアとして公開
- ▶ 2015年にApacheのトップレベルプロジェクト（TLP）に昇格

Apache NiFi の歴史

- ▶ リリース時期と頻度は以下の通りです

	2015	2016	2017	2018	2019
0系		▲7月 (0.2.1) ▲3月 (0.6.0) ▲9月 (0.3.0)▲4月 (0.6.1) ▲12月 (0.4.0) ▲7月 (0.7.0) ▲12月 (0.4.1) ▲10月 (0.7.1) ▲2月 (0.5.0) ▲2月 (0.5.1)	▲2月 (0.7.2) ▲5月 (0.7.3) ▲6月 (0.7.4)		
1系			▲8月 (1.0.0) ▲5月 (1.2.0) ▲11月 (1.1.0) ▲6月 (1.3.0) ▲12月 (1.1.1) ▲2月 (1.1.2)	▲1月 (1.5.0) ▲4月 (1.6.0) ▲6月 (1.7.0) ▲7月 (1.7.1) ▲10月 (1.8.0)	▲2月 (1.9.0) ▲3月 (1.9.1) ▲4月 (1.9.2)

Apache NiFi を動かしてみよう！

▶ ここで一度おさらい

- ▶ Apache NiFiって何ができるんだっけ？
- ▶ 実現したい**データの流れ(データフロー)**をApache NiFiの部品(プロセッサ)を使ってお絵かきして動かせる

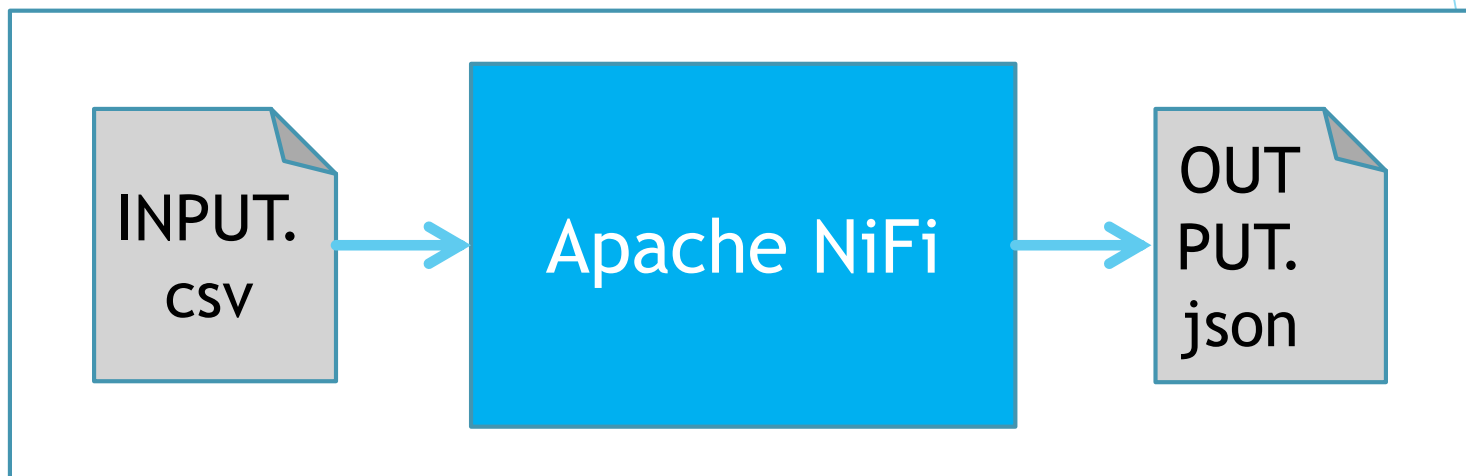
▶ データの流れとは？

- ▶ 具体的なデータの入出力やそのデータの取り回しを定義すること
 - ▶ 例えば...**Twitterのタイムライン**をデータの入力元、**DB**をデータの出力先、とするなど！
 - ▶ Apache NiFiならGUIでデータの流れを書けます◎

データフローをお絵描き！

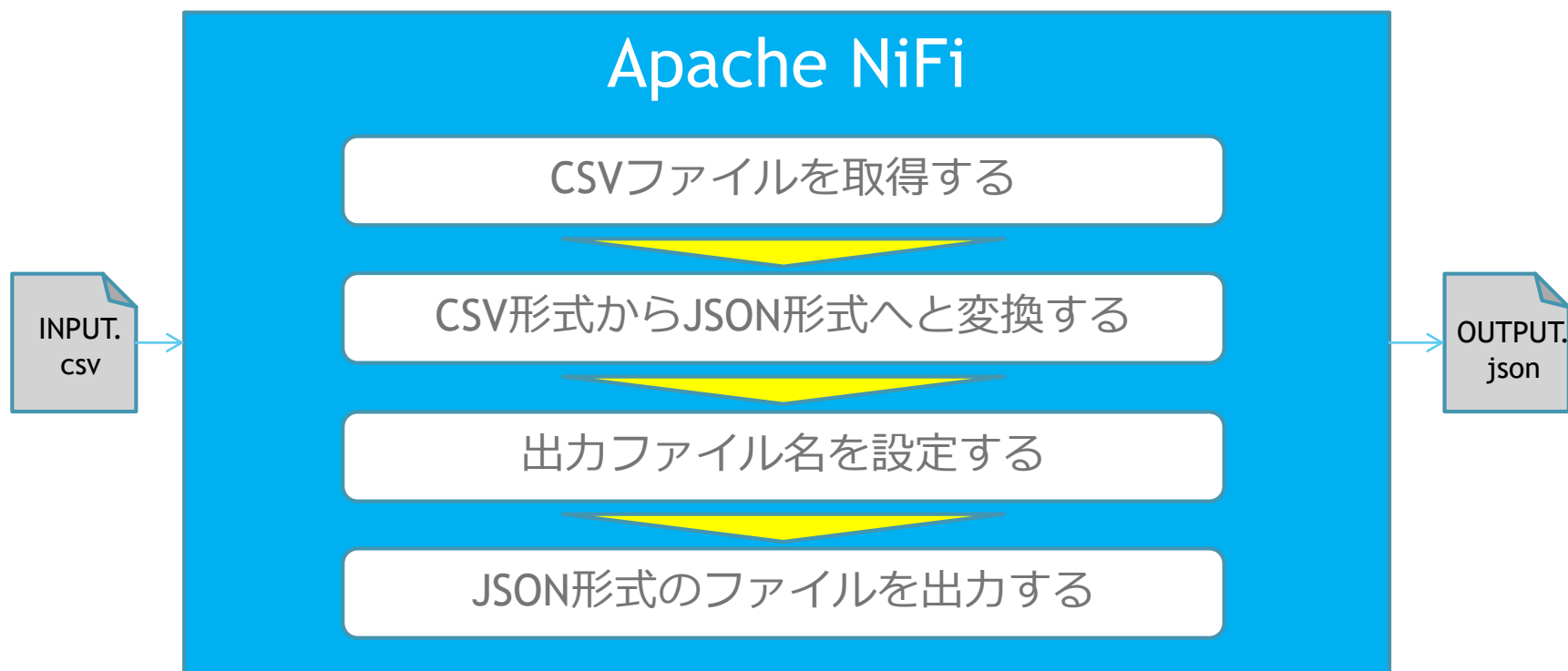
- ▶ 例)ファイル(CSV形式)を読み込み、読み込んだデータをJSON形式に変換し別ファイル名として書き込む。

データの流れと処理のイメージを簡単にお絵かきすると、
こんな感じ



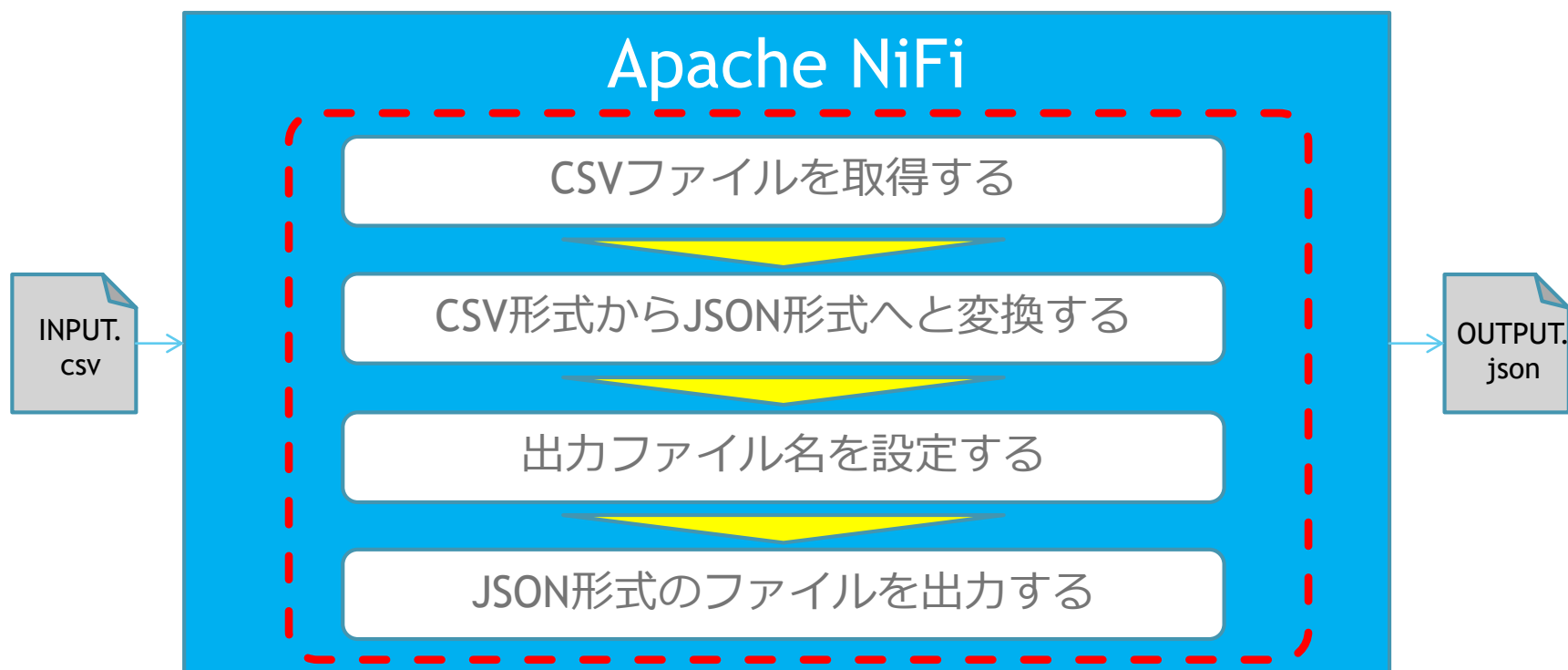
データフローをお絵描き！

- ▶ 例) ファイル(CSV形式)を読み込み、読み込んだデータをJSON形式に変換し別ファイル名として書き込む。



データフローをお絵描き！

- ▶ 例) ファイル(CSV形式)を読み込み、読み込んだデータをJSON形式に変換し別ファイル名として書き込む。



この部分がデータフロー

(かなり拡大解釈してます。本来は取り扱うデータにもっと着目します)

データフローをお絵描き！

- ▶ 例) ファイル(CSV形式)を読み込み、読み込んだデータをJSON形式に変換し別ファイル名として書き込む。



この個々の単位がプロセッサ（NiFi部品）

まとめると

- ▶ プロセッサを利用して、データフローを構築する

やりたいこと

CSVファイルを取得する

CSV形式からJSON形式へと変換する

出力ファイル名を設定する

JSON形式のファイルを出力する

NiFi部品

ファイル入力用
プロセッサ

データ型変換用
プロセッサ

ファイル名設定用
プロセッサ

ファイル出力用
プロセッサ

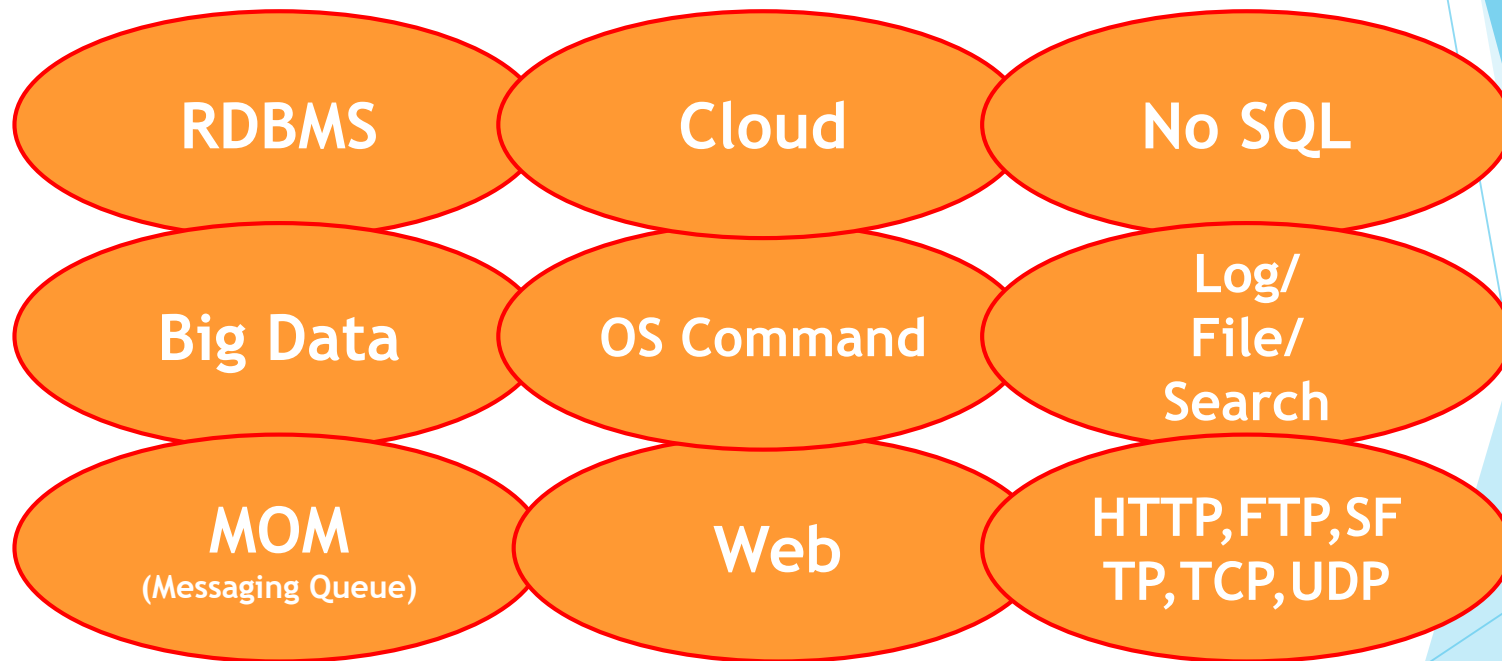
実際のNiFiの開発画面

The screenshot displays the Apache NiFi web interface with a data flow consisting of four processors connected in sequence. Each processor has a status bar at the bottom showing 'Name success' and 'Queued 0 (0 bytes)'. The processors are:

- GetFile** (1.9.2): org.apache.nifi - nifi-standard-nar. In: 0 (0 bytes), Read/Write: 0 bytes / 0 bytes, Out: 0 (0 bytes), Tasks/Time: 0 / 00:00:00.000.
- ConvertRecord** (1.9.2): org.apache.nifi - nifi-standard-nar. In: 0 (0 bytes), Read/Write: 0 bytes / 0 bytes, Out: 0 (0 bytes), Tasks/Time: 0 / 00:00:00.000.
- UpdateAttribute** (1.9.2): org.apache.nifi - nifi-update-attribute-nar. In: 0 (0 bytes), Read/Write: 0 bytes / 0 bytes, Out: 0 (0 bytes), Tasks/Time: 0 / 00:00:00.000.
- PutFile** (1.9.2): org.apache.nifi - nifi-standard-nar. In: 0 (0 bytes), Read/Write: 0 bytes / 0 bytes, Out: 0 (0 bytes), Tasks/Time: 0 / 00:00:00.000.

プロセッサにはどんなものがある？

- ▶ NiFiにはデータ入出力用プロセッサが豊富！！



プロセッサにはどんなものがある？

- ▶ プロセッサは本当にたくさんの種類があります。
ぜひ、ドキュメントを見てみてください。

<<https://nifi.apache.org/docs.html>>

The screenshot shows the Apache NiFi Documentation website. The browser address bar displays <https://nifi.apache.org/docs.html>. The navigation menu includes Project, Documentation, Downloads, Community, Development, ASF Links, and Subprojects. The main content area is titled 'Apache NiFi Overview' and includes a 'Table of Contents' with links to 'What is Apache NiFi?', 'The core concepts of NiFi', 'NiFi Architecture', 'Performance Expectations and Characteristics of NiFi', 'High Level Overview of Key NiFi Features', and 'References'. The 'Processors' section is highlighted with a red box, listing various processors such as 'AttributeRollingWindow 1.9.2', 'AttributesToCSV 1.9.2', 'AttributesToJSON 1.9.2', 'Base64EncodeContent 1.9.2', 'CalculateRecordStats 1.9.2', 'CaptureChangeMySQL 1.9.2', 'CompareFuzzyHash 1.9.2', 'CompressContent 1.9.2', 'ConnectWebSocket 1.9.2', and 'ConsumeAMQP 1.9.2'. A red arrow points to the 'Processors' section with the Japanese character '続' (Continued).

Apache NiFiをインストールしてみよう！

▶ 動作環境

- ▶ Mac OS

- ▶ Linux

- ▶ Windows

- ▶ お手持ちのPCで簡単に動かします！

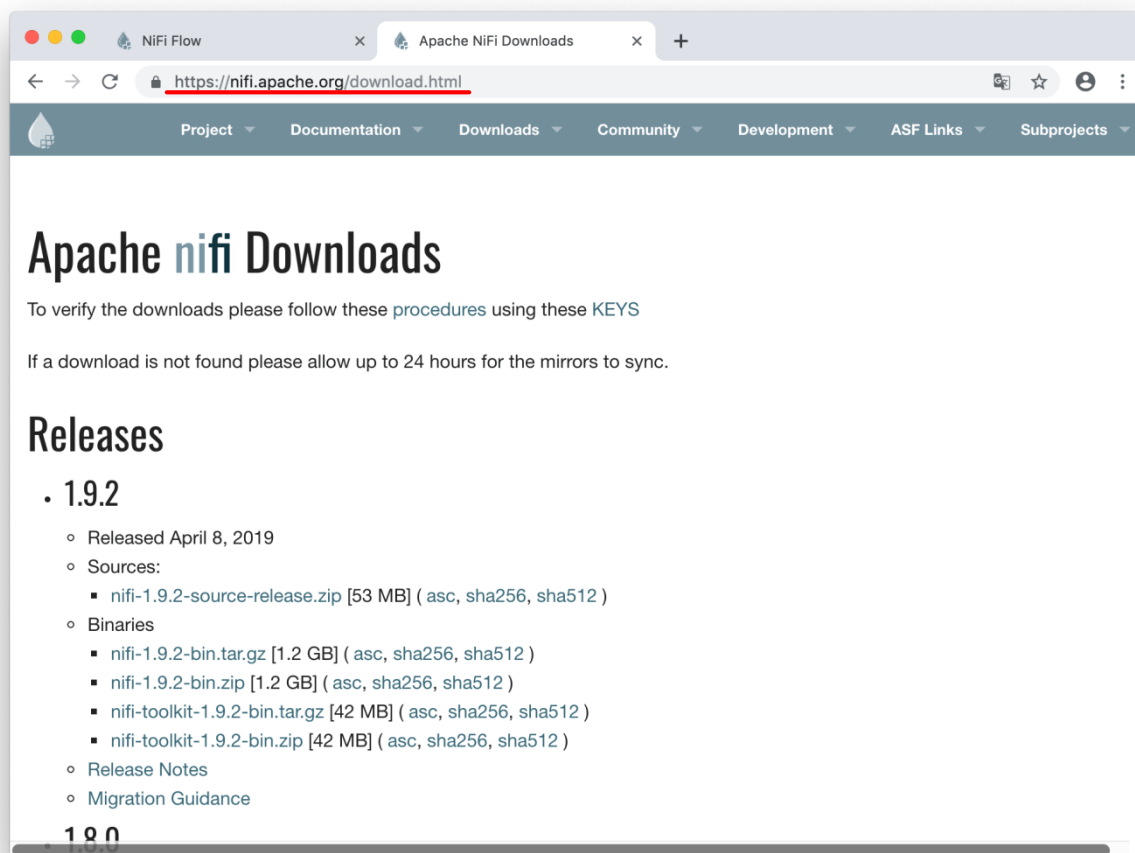
- ▶ Raspberry Pi3上では...重くて動作不安定でした。

- ▶ Java 1.8.0以上が必要

Apache NiFiをインストールしてみよう！

1. Apache NiFiのダウンロード

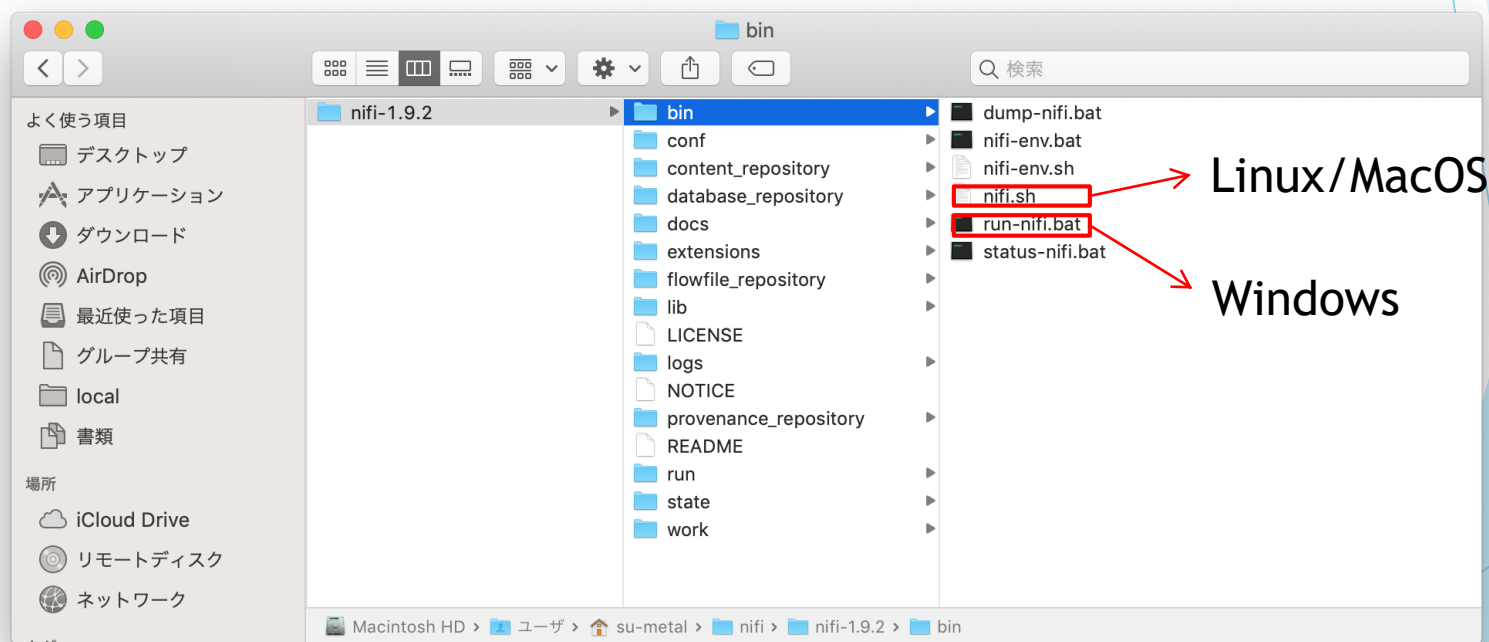
<<https://nifi.apache.org/download.html>>



Apache NiFiをインストールしてみよう！

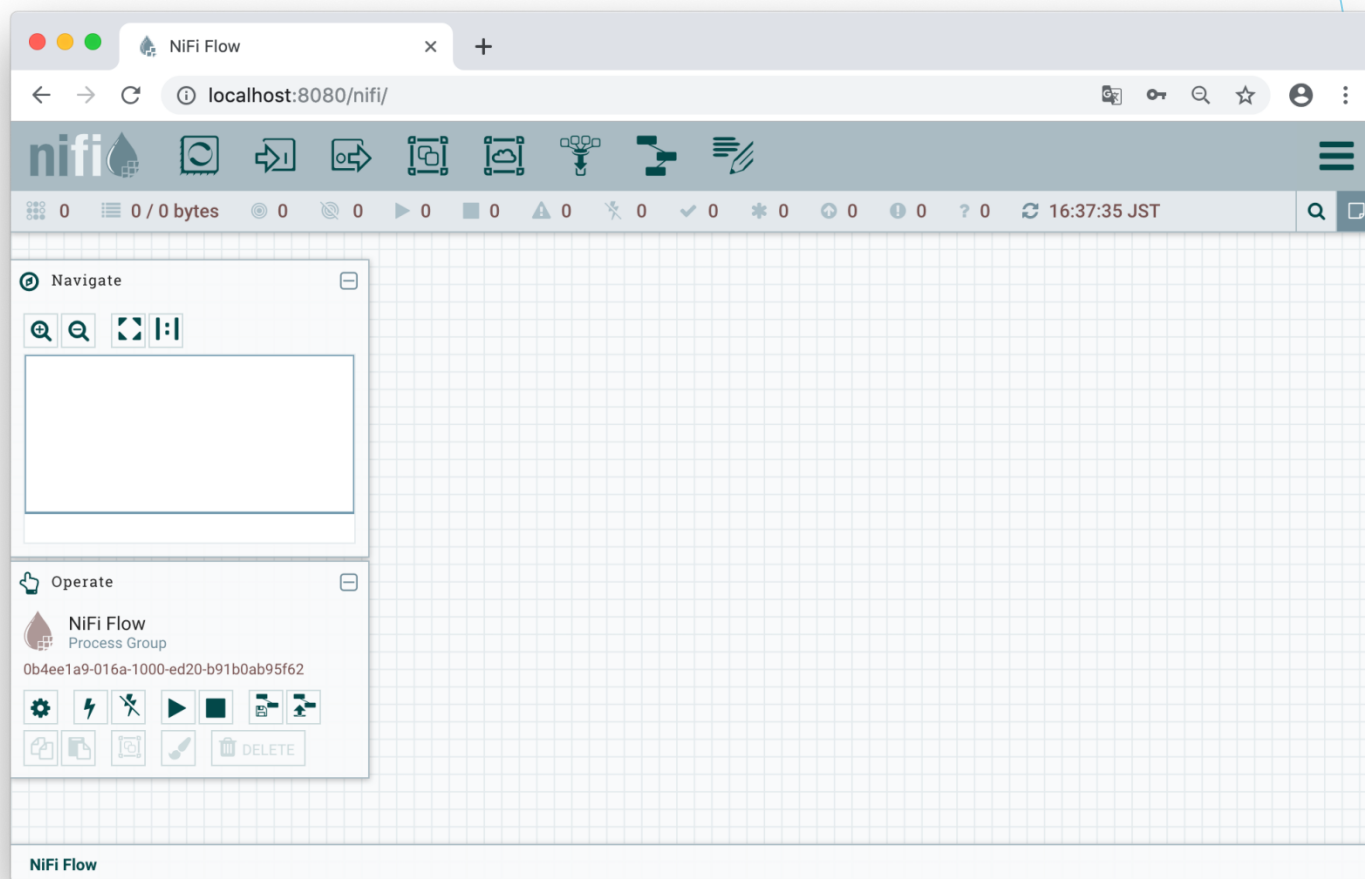
2. ダウンロード資料の（zip or tar.gz）解凍

3. NiFiの起動



Apache NiFiをインストールしてみよう！

4. ブラウザからNiFiにアクセス
<<http://localhost:8080/nifi/>>



Apache NiFiでデータフローを作ろう！

さきほど例に上げたデータフローを作ってみましょう！

Apache NiFi

CSVファイルを取得する

CSV形式からJSON形式へと変換する

出力ファイル名を設定する

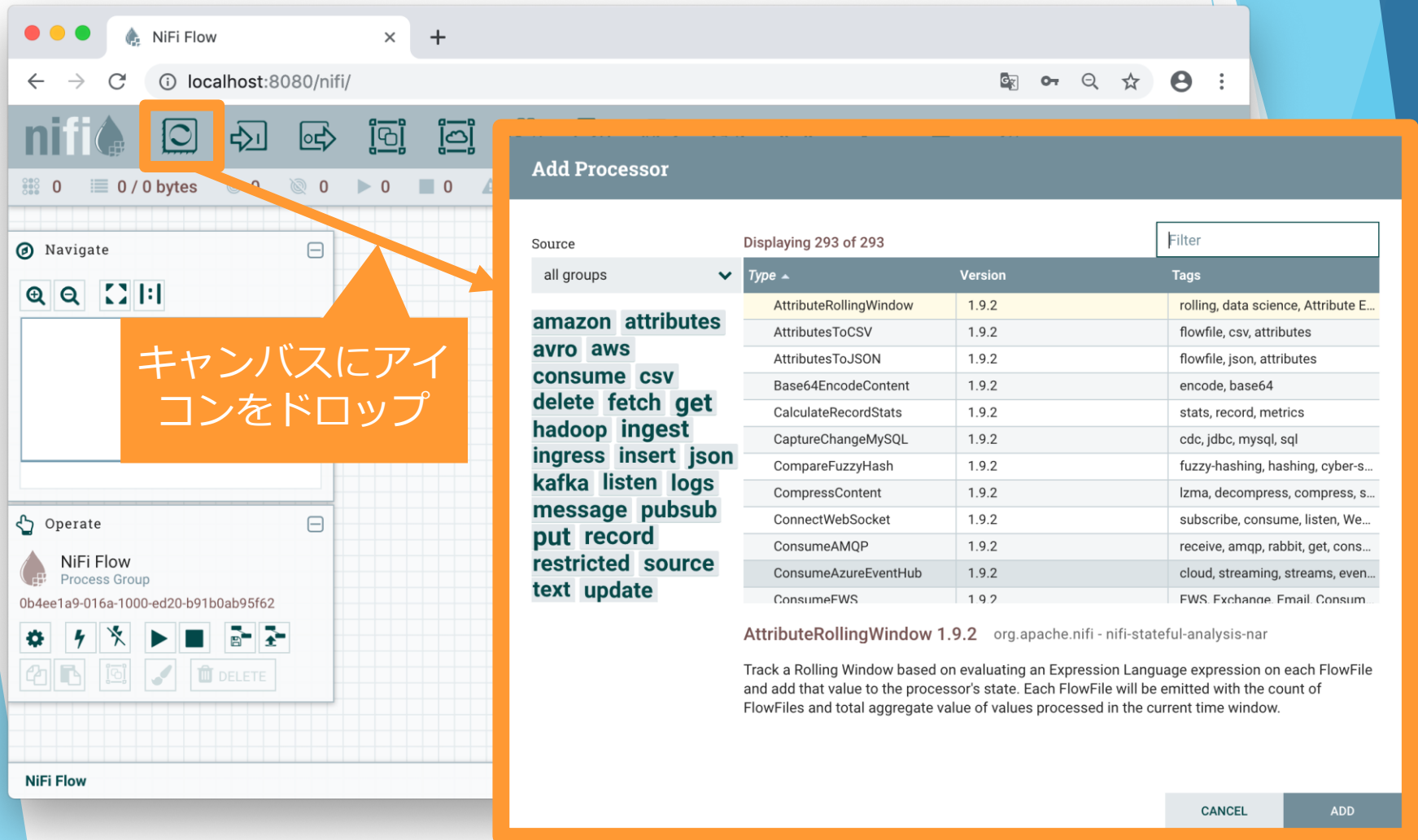
JSON形式のファイルを出力する

INPUT.
CSV

OUTPUT.
json

Apache NiFiでデータフローを作ろう！

1. プロセッサを選択する



キャンバスにアイコンをドロップ

Add Processor

Source: all groups

Displaying 293 of 293

Type	Version	Tags
AttributeRollingWindow	1.9.2	rolling, data science, Attribute E...
AttributesToCSV	1.9.2	flowfile, csv, attributes
AttributesToJSON	1.9.2	flowfile, json, attributes
Base64EncodeContent	1.9.2	encode, base64
CalculateRecordStats	1.9.2	stats, record, metrics
CaptureChangeMySQL	1.9.2	cdc, jdbc, mysql, sql
CompareFuzzyHash	1.9.2	fuzzy-hashing, hashing, cyber-s...
CompressContent	1.9.2	lzma, decompress, compress, s...
ConnectWebSocket	1.9.2	subscribe, consume, listen, We...
ConsumeAMQP	1.9.2	receive, amqp, rabbit, get, cons...
ConsumeAzureEventHub	1.9.2	cloud, streaming, streams, even...
ConsumeFWS	1.9.2	FWS, Exchange, Email, Consum...

AttributeRollingWindow 1.9.2 org.apache.nifi - nifi-stateful-analysis-nar

Track a Rolling Window based on evaluating an Expression Language expression on each FlowFile and add that value to the processor's state. Each FlowFile will be emitted with the count of FlowFiles and total aggregate value of values processed in the current time window.

CANCEL ADD

Apache NiFiでデータフローを作ろう！

1. プロセッサを選択する

Add Processor

Source: all groups ▼

Displaying 1 of 293

Search: getfile

Type ▲	Version	Tags
GetFile	1.9.2	ingress, input, restricted, get, fil...

amazon attributes
avro aws
consume csv
delete fetch get
hadoop ingest
ingress insert json
kafka listen logs
message pubsub
put record
restricted source
text update

GetFile 1.9.2 org.apache.nifi - nifi-standard-nar

Creates FlowFiles from files in a directory. NiFi will ignore files it doesn't have at least read permissions for.

CANCEL ADD

検索・絞り込みができる

プロセッサを選択したら、ADDする

Apache NiFiでデータフローを作ろう！

1. プロセッサを選択する

The screenshot displays the Apache NiFi web interface. The top navigation bar includes the NiFi logo and various icons for navigation and actions. Below the navigation bar, a status bar shows metrics such as '0 / 0 bytes', '0', '0', '0', '4', '3', '0', '0', '0', '0', '0', '0', '0', '0', and the time '15:17:46 JST'. The main workspace is a grid where a 'GetFile' processor is being added. The 'GetFile' processor details are shown in a tooltip:

GetFile	
GetFile 1.9.2	
org.apache.nifi - nifi-standard-nar	
In	0 (0 bytes) 5 min
Read/Write	0 bytes / 0 bytes 5 min
Out	0 (0 bytes) 5 min
Tasks/Time	0 / 00:00:00.000 5 min

An orange callout box points to the 'GetFile' processor with the text: 'GetFileのプロセッサが用意された' (GetFile processor is ready).

The left sidebar contains two panels: 'Navigate' and 'Operate'. The 'Operate' panel shows the 'NiFi Flow' process group with the ID '299f5721-016a-1000-d5d9-399aa47b97d6' and various action icons including settings, refresh, delete, and execute. The bottom status bar indicates 'NiFi Flow'.

Apache NiFiでデータフローを作ろう！

1. プロセッサを選択する

The screenshot shows the Apache NiFi Flow Designer interface. The top bar includes the NiFi logo and various status icons. The left sidebar contains two panels: 'Navigate' and 'Operate'. The main canvas displays a flow diagram with four processors: GetFile, ConvertRecord, UpdateAttribute, and PutFile. An orange callout box on the right contains the text '全部で4つのプロセッサを準備します' (Prepare all 4 processors).

GetFile Processor Details:

Property	Value	Unit
In	0	(0 bytes)
Read/Write	0 bytes / 0 bytes	
Out	0	(0 bytes)
Tasks/Time	0 / 00:00:00.000	5 min

ConvertRecord Processor Details:

Property	Value	Unit
In	0	(0 bytes)
Read/Write	0 bytes / 0 bytes	
Out	0	(0 bytes)
Tasks/Time	0 / 00:00:00.000	5 min

UpdateAttribute Processor Details:

Property	Value	Unit
In	0	(0 bytes)
Read/Write	0 bytes / 0 bytes	
Out	0	(0 bytes)
Tasks/Time	0 / 00:00:00.000	5 min

PutFile Processor Details:

Property	Value	Unit
In	0	(0 bytes)
Read/Write	0 bytes / 0 bytes	
Out	0	(0 bytes)
Tasks/Time	0 / 00:00:00.000	5 min

全部で4つのプロセスを準備します

Apache NiFiでデータフローを作ろう！

2. データの流れ(コネクション)を設定する

	GetFile GetFile 1.9.2 org.apache.nifi - nifi-standard-nar	
In	0 (0 bytes)	5 min
Read/Write	0 bytes / 0 bytes	5 min
Out	0 (0 bytes)	5 min
Tasks/Time	0 / 00:00:00.000	5 min

①プロセッサの上にカーソルを置くと現れる→を、接続先にドロップする

	 ConvertRecord ConvertRecord 1.9.2 org.apache.nifi - nifi-standard-nar	
In	0 (0 bytes)	5 min
Read/Write	0 bytes / 0 bytes	5 min
Out	0 (0 bytes)	5 min
Tasks/Time	0 / 00:00:00.000	5 min

	 UpdateAttribute UpdateAttribute 1.9.2 org.apache.nifi - nifi-update-attribute-nar	
In	0 (0 bytes)	5 min
Read/Write	0 bytes / 0 bytes	5 min
Out	0 (0 bytes)	5 min
Tasks/Time	0 / 00:00:00.000	5 min

	 PutFile PutFile 1.9.2 org.apache.nifi - nifi-standard-nar	
In	0 (0 bytes)	5 min
Read/Write	0 bytes / 0 bytes	5 min
Out	0 (0 bytes)	5 min
Tasks/Time	0 / 00:00:00.000	5 min

Create Connection

DETAILS

SETTINGS

From Processor

GetFile
GetFile

Within Group
NiFi Flow

For Relationships

☒ success

To Processor

ConvertRecord
ConvertRecord

Within Group
NiFi Flow

②コネクションの設定画面が開くので、設定してADDする

CANCEL

ADD

Apache NiFiでデータフローを作ろう！

2. データの流れ(コネクション)を設定する

The screenshot displays the Apache NiFi web interface with a data flow canvas. Four processors are visible, each with its own status panel. An orange callout box highlights the 'Name success' and 'Queued 0 (0 bytes)' status of the 'GetFile' processor, indicating that the connection has been successfully created.

GetFile
GetFile 1.9.2
org.apache.nifi - nifi-standard-nar

In	0 (0 bytes)
Read/Write	0 bytes / 0 bytes
Out	0 (0 bytes)
Tasks/Time	0 / 00:00:00.000

ConvertRecord
ConvertRecord 1.9.2
org.apache.nifi - nifi-standard-nar

In	0 (0 bytes)	5 min
Read/Write	0 bytes / 0 bytes	5 min
Out	0 (0 bytes)	5 min
Tasks/Time	0 / 00:00:00.000	5 min

UpdateAttribute
UpdateAttribute 1.9.2
org.apache.nifi - nifi-update-attribute-nar

In	0 (0 bytes)	5 min
Read/Write	0 bytes / 0 bytes	5 min
Out	0 (0 bytes)	5 min
Tasks/Time	0 / 00:00:00.000	5 min

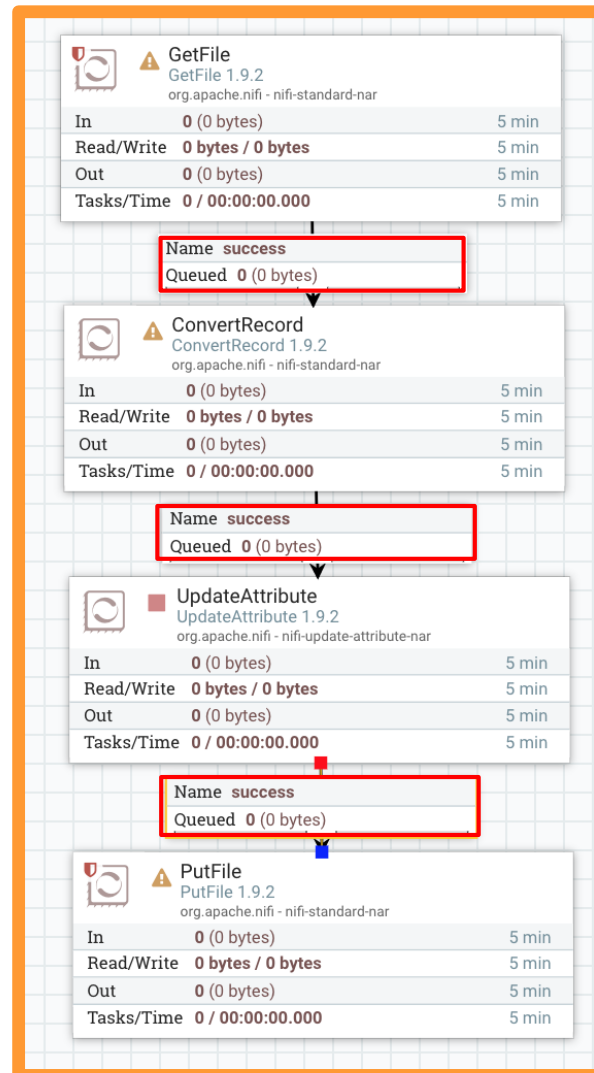
PutFile
PutFile 1.9.2
org.apache.nifi - nifi-standard-nar

In	0 (0 bytes)	5 min
Read/Write	0 bytes / 0 bytes	5 min
Out	0 (0 bytes)	5 min
Tasks/Time	0 / 00:00:00.000	5 min

コネクションが作成された

Apache NiFiでデータフローを作ろう！

2. データの流れ(コネクション)を設定する



4つ全部コネクションでつ
なぐとこんな感じ

Apache NiFiでデータフローを作ろう！

3. プロセッサの設定をする(GetFileプロセッサ)

Configure Processor

SETTINGS

SCHEDULING

PROPERTIES

COMMENTS

Name

ファイル取得

Automatically Terminate Relationships ?

☒ success
All files are routed to success

Id

29aa938e-016a-1000-8c42-593e72d9bc53

Type

GetFile 1.9.2

Bundle

org.apache.nifi - nifi-standard-nar

Penalty Duration ?

30 sec

Yield Duration ?

1 sec

Bulletin Level ?

WARN

CANCEL

APPLY

プロセッサを右クリックで
設定画面が開く

Apache NiFiでデータフローを作ろう！

3. プロセッサの設定をする(GetFileプロセッサ)

Configure Processor

SETTINGS SCHEDULING **PROPERTIES** COMMENTS

Required field

Property	Value
Input Directory	/opt/tmp/getfile
File Filter	[^\.].*\.csv\$
Path Filter	No value set
Batch Size	10
Keep Source File	true
Recurse Subdirectories	true
Polling Interval	0 sec
Ignore Hidden Files	true
Minimum File Age	0 sec
Maximum File Age	No value set
Minimum File Size	0 B
Maximum File Size	No value set

CANCEL APPLY

どこのディレクトリからファイルを取得するか

取得するファイルのフィルタリング

Apache NiFiでデータフローを作ろう！

3. プロセッサの設定をする(ConvertRecordプロセッサ)

Configure Processor

SETTINGS

SCHEDULING

PROPERTIES

COMMENTS

Name

データ変換(CSVtoJSON) ☒ Enabled

Id

29aace49-016a-1000-7b83-7dbcac652d38

Type

ConvertRecord 1.9.2

Bundle

org.apache.nifi - nifi-standard-nar

Penalty Duration ?

30 sec

Yield Duration ?

1 sec

Bulletin Level ?

WARN ▼

Automatically Terminate Relationships ?

☒ failure
If a FlowFile cannot be transformed from the configured input format to the configured output format, the unchanged FlowFile will be routed to this relationship

☐ success
FlowFiles that are successfully transformed will be routed to this relationship

CANCEL

APPLY

Apache NiFiでデータフローを作ろう！

3. プロセッサの設定をする(ConvertRecordプロセッサ)

Configure Processor

SETTINGS SCHEDULING **PROPERTIES** COMMENTS

Required field +

Property	Value
Record Reader	CSVReader →
Record Writer	JsonRecordSetWriter →
Include Zero Record FlowFiles	true

入出力に対応した
Controller Serviceを設定する

※Controller Service
各Processorから利用できる
共通のコンポーネントのこと

CANCEL APPLY

Apache NiFiでデータフローを作ろう！

3. プロセッサの設定をする(ConvertRecordプロセッサ)

◆ ConvertRecordで利用するController Serviceを設定

NiFi Flow Configuration

GENERAL

CONTROLLER SERVICES

	Name ▲	Type	Bundle	State	Scope	
	CSVReader	CSVReader 1.9.2	org.apache.nifi - nifi-re...	 Enabled	NiFi Flow	 
	JsonRecordSetWriter	JsonRecordSetWriter 1...	org.apache.nifi - nifi-re...	 Enabled	NiFi Flow	 

- CSVReader
- JsonRecordSetWriter

をそれぞれ設定

 Last updated: 13:27:45 JST

Listed services are available to all descendant Processors and services of this Process Group.

Apache NiFiでデータフローを作ろう！

3. プロセッサの設定をする(ConvertRecordプロセッサ)

- ◆ ConvertRecordで利用するController Serviceを設定(CSVReader)

Controller Service Details

SETTINGS | PROPERTIES | COMMENTS

Required field

Property	Value
Schema Access Strategy	Use 'Schema Text' Property
Schema Registry	No value set
Schema Name	\${schema.name}
Schema Version	No value set
Schema Branch	No value set
Schema Text	{ "name": "SchemaBook", "namespace": "nifi", "ty...
CSV Parser	Apache Commons CSV
Date Format	
Time Format	
Timestamp Format	
CSV Format	
Value Separator	
Treat First Line as Header	
Ignore CSV Header Column Names	

CSVReaderの設定画面

```
1 {  
2   "name": "SchemaBook", "namespace": "nifi", "type": "record", "fields": [  
3     { "name": "title", "type": "string" }, { "name": "author", "type": "string" }, { "name": "created", "type": "string" }  
4   ]  
}
```

スキーマ定義に従って CSV形式のデータを読み込む

OK

Apache NiFiでデータフローを作ろう！

3. プロセッサの設定をする(ConvertRecordプロセッサ)

- ◆ ConvertRecordで利用するController Serviceを設定(JsonRecordSetWriter)

Controller Service Details

JsonRecordSetWriterの設定画面

SETTINGS

PROPERTIES

COMMENTS

Name
JsonRecordSetWriter

Id
29b5be5a-016a-1000-d45d-1867fc639620

Type
JsonRecordSetWriter 1.9.2

Bundle
org.apache.nifi - nifi-record-serialization-services-nar

Supports Controller Service

- RecordSetWriterFactory 1.9.2 from org.apache.nifi - nifi-standard-services-api-nar

Referencing Components ⓘ

▼ Processors (1)

- データ変換(CSVtoJSON) *ConvertRecord*

OK

Apache NiFiでデータフローを作ろう！

3. プロセッサの設定をする(ConvertRecordプロセッサ)

- ◆ ConvertRecordで利用するController Serviceを設定(JsonRecordSetWriter)

Controller Service Details

SETTINGSPROPERTIESCOMMENTS

JsonRecordSetWriterの設定画面

Required field

Property	Value
Schema Write Strategy	Do Not Write Schema
Schema Cache	No value set
Schema Access Strategy	Inherit Record Schema
Schema Registry	
Schema Name	
Schema Version	
Schema Branch	
Schema Text	
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

データ入力時のスキーマ定義を継承
(CSVReaderで設定したスキーマ定義を利用)

OK

Apache NiFiでデータフローを作ろう！

3. プロセッサの設定をする(UpdateAttributeプロセッサ)

Configure Processor

SETTINGS

SCHEDULING

PROPERTIES

COMMENTS

Name

出力ファイル名設定

☒ Enabled

Automatically Terminate Relationships ?

☐ success

All successful FlowFiles are routed to this relationship

Id

29ab0f4e-016a-1000-5766-8f39e04afd5c

Type

UpdateAttribute 1.9.2

Bundle

org.apache.nifi - nifi-update-attribute-nar

Penalty Duration ?

30 sec

Yield Duration ?

1 sec

Bulletin Level ?

WARN

ADVANCED

CANCEL

APPLY

Apache NiFiでデータフローを作ろう！

3. プロセッサの設定をする(UpdateAttributeプロセッサ)

Configure Processor

SETTINGS

SCHEDULING

PROPERTIES

COMMENTS

Required field

+

Property	Value
Delete Attributes Expression	No value set
Store State	Do not store state
Stateful Variables Initial Value	No value set
Cache Value Lookup Cache Size	100
filename	? <code>\${filename:substringBeforeLast('.')}.json</code> 🗑

⚙ ADVANCED

CANCEL

APPLY

出力時のファイル名を設定
ここでは、GetFileで”filename”として
読み込まれていた値を使用

Apache NiFiでデータフローを作ろう！

3. プロセッサの設定をする(PutFileプロセッサ)

Configure Processor

SETTINGS

SCHEDULING

PROPERTIES

COMMENTS

Name

ファイル出力

Id

29ab4244-016a-1000-e6e3-1ebe6b209e67

Type

PutFile 1.9.2

Bundle

org.apache.nifi - nifi-standard-nar

Penalty Duration ?

30 sec

Bulletin Level ?

WARN

Enabled

Yield Duration ?

1 sec

Automatically Terminate Relationships ?

failure

Files that could not be written to the output directory for some reason are transferred to this relationship

success

Files that have been successfully written to the output directory are transferred to this relationship

CANCEL

APPLY

Apache NiFiでデータフローを作ろう！

3. プロセッサの設定をする(PutFileプロセッサ)

Configure Processor

SETTINGS SCHEDULING **PROPERTIES** COMMENTS

Required field +

Property	Value
Directory	/opt/tmp/putfile
Conflict Resolution Strategy	replace
Create Missing Directories	true
Maximum File Count	No value s
Last Modified Time	No value s
Permissions	No value s
Owner	No value s
Group	No value s

CANCEL APPLY

出力先のディレクトリを指定

Apache NiFiでデータフローを作ろう！

3. 完成！！

The screenshot displays the Apache NiFi web interface with a completed data flow. The top navigation bar includes the NiFi logo and various icons for navigation and actions. The status bar shows 0 processors, 0/0 bytes of data, and a timestamp of 16:32:25 JST. The left sidebar contains a 'Navigate' panel and an 'Operate' panel with a 'Multiple components selected' message and a 'DELETE' button.

The main canvas shows a data flow with four processors connected in a vertical sequence:

- ファイル取得 (File Get)**: GetFile 1.9.2, org.apache.nifi-nifi-standard-nar. In: 0 (0 bytes), Read/Write: 0 bytes / 0 bytes, Out: 0 (0 bytes), Tasks/Time: 0 / 00:00:00.000. Status: Name success, Queued 0 (0 bytes).
- データ変換(CSVtoJSON) (Data Conversion)**: ConvertRecord 1.9.2, org.apache.nifi-nifi-standard-nar. In: 0 (0 bytes), Read/Write: 0 bytes / 0 bytes, Out: 0 (0 bytes), Tasks/Time: 0 / 00:00:00.000. Status: Name success, Queued 0 (0 bytes).
- 出力ファイル名設定 (Output File Name Setting)**: UpdateAttribute 1.9.2, org.apache.nifi-nifi-update-attribute-nar. In: 0 (0 bytes), Read/Write: 0 bytes / 0 bytes, Out: 0 (0 bytes), Tasks/Time: 0 / 00:00:00.000. Status: Name success, Queued 0 (0 bytes).
- ファイル出力 (File Put)**: PutFile 1.9.2, org.apache.nifi-nifi-standard-nar. In: 0 (0 bytes), Read/Write: 0 bytes / 0 bytes, Out: 0 (0 bytes), Tasks/Time: 0 / 00:00:00.000.

The bottom status bar indicates 'NiFi Flow'.

Apache NiFiを使ってみよう！

実機デモ

おまけ : Apache MiNiFiについて

▶ MiNiFiとは？

- ▶ NiFiのサブプロジェクト
- ▶ Apache NiFiよりも軽い
- ▶ IoT機器に入れて動作させて、データを収集するなどの用途が想定されている
 - ▶ EX) センサーデータなど

まとめ

- ▶ Apache NiFiはオープンソースのデータフローオーケストレーションツール
- ▶ 各プラットフォームで動作する
- ▶ ブラウザ上からGUI操作にてデータフローの定義ができる
- ▶ 豊富なプロセッサ！

Apache NiFiはとても簡単に利用を開始することができます。
実現したいデータフローがあれば、ぜひ活用してみてください。

ご清聴ありがとうございました