



PLATFORM DEEP DIVE / Plugins / Custom Plugins / Documents plugin / Using the plugin / splitting-a-document

Contents

- PLATFORM DEEP DIVE / Plugins / Custom Plugins / Documents plugin /
Using the plugin / Splitting a document
 - Sending the request
 - Receiving the reply

PLATFORM DEEP DIVE / Plugins / Custom Plugins / Documents plugin / Using the plugin / Splitting a document


You can split a document into multiple parts using the Documents Plugin. This feature is useful, for example, when a user uploads a bulk scanned file that needs to be separated into separate files.


Sending the request


To split a document, follow these steps:

1. Create a process and add a **Kafka send event node** and a **Kafka receive event node**. These nodes are used to send the request and receive the reply.
2. Configure the first node, Kafka send event node by adding a **Kafka send action**.

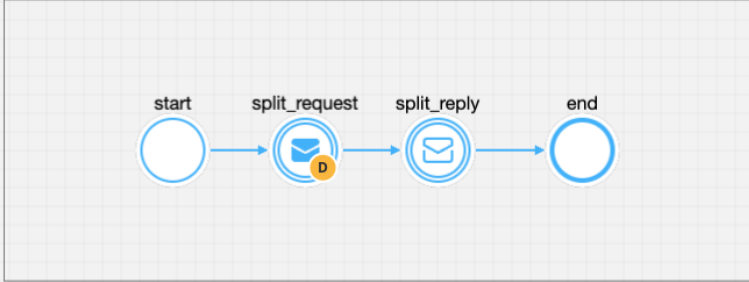
FLOWX.AI



split_docs (DRAFT) 

Changes saved Auto arrange 

Search by node name 


Default



Node: **split_request** (ID: 727240)  

Node Config Actions

Actions +

split_requ... + 

Action Edit

ID: 728253


Name

split_request

Order

1

Timer Expression

Kafka Send Action 

☒ Automatic ☐ Manual

☒ Mandatory ☐ Optional

Save

3. Specify the **Kafka topic** to which you want to send the request.

Parameters

Custom

From integration

Topics

```
ai.flowx.in.document.split.v1
```

4. Fill in the body message request:

Message

```
1 {  
2   "fileId": 4742,  
3   "parts": [  
4     {  
5       "documentType": "BULK",  
6       "customId": "1234_759769",  
7       "pagesNo": [1,2]  
8     }  
9   ]  
10 }
```

Advanced configuration

Show Headers ☒

```
1 {"processInstanceId": "${processInstanceId}}}
```

- **fileId**: The ID of the file to be split.
- **parts**: A list containing information about the expected document parts.
 - **documentType**: The document type.
 - **customId**: The client ID.

- **shouldOverride**: A boolean value (true or false) indicating whether to override an existing document if one with the same name already exists.
- **pagesNo**: The pages that you want to separate from the document.

! INFO

You can customize the Kafka topic names by overwriting the following environment variables during deployment:

`KAFKA_TOPIC_DOCUMENT_SPLIT_IN` - default value:

`ai.flowx.in.qa.document.split.v1` - this is the topic that listens for the request from the engine

`KAFKA_TOPIC_DOCUMENT_SPLIT_OUT` - default value:

`ai.flowx.updates.qa.document.split.v1` - this is the topic on which the engine expects the reply

The above examples of topics are extracted from an internal testing environment. When setting topics for other environments, follow this pattern:

`ai.flowx.updates.{{environment}}.document.split.v1`.

! CAUTION

The Engine listens for messages on topics with specific names. Make sure to use an outgoing topic name that matches the pattern configured in the Engine.

Receiving the reply

You can view the response by accessing the Audit log menu. The reply will be sent to the Kafka topic specified in the Kafka receive event node.

Node: **split_reply** (ID: 728352)

Node Config

General Config

Node name

split_reply

Can go back? ☒

Swimlane

Default



Stage



Response Timeout

Response Timeout (PT30S)

Data stream topics

Custom

From integration

Topic Name

ai.flowx.updates.qa.document.split.v1

Key Name

receiveReply



Add stream

The response body will contain the following values:

- **docs**: A list of documents.
 - **customId**: The client ID.
 - **fileId**: The ID of the file.
 - **documentType**: The document type.
 - **minioPath**: The storage path for the document.
 - **downloadPath**: The download path for the document.
 - **noOfPages**: The number of pages in the document.

Audit log details



Event: process instance, message receive, 25 Oct 2022 at 2:36 PM

Url: ai.flowx.updates.qa.document.split.v1

Body:

```
1  {"docs":[{"customId":"1234_759769","fileId":4743,"documentType":"BULK",
    "documentLabel":null,
    "minioPath":"qualitance-dev-paperflow-qa-process-id-759770/1234_759769/
    4743_BULK.pdf","downloadPath":"internal/files/4743/download",
    "noOfPages":2,"error":null}], "error":null}
```

Here's an example of the response JSON:

```
{
  "docs": [
    {
```



```
"customId": "1234_759769",  
"fileId": 4743,  
"documentType": "BULK",  
"documentLabel": null,  
"minioPath": "qualitance-dev-paperflow-qa-process-id-  
759770/1234_759769/4743_BULK.pdf",  
"downloadPath": "internal/files/4743/download",  
"noOfPages": 2,  
"error": null  
}  
],  
"error": null  
}
```

Was this page helpful?