

Saga Configuration

Last Modified on 12/19/2025 5:24 am EST

Overview

The saga controls the process, it handles the steps in the queue and instructs the system to perform the next step. The saga runs the steps in the process through predefined business logic configured in the saga YAML files. You can define a saga each for processing a different document type and different workspace. You can change the steps in the saga, but not the entire saga.

Saga steps

1. Download: The system downloads the collected data.
2. Validation: The system validates that a request is not a repeating request.
3. Extraction: The system extracts data from the collected files.
4. Business process: The system runs the steps in the process in an order, and after each process ends, it moves to the next step to be performed.
5. Move to next queue: The system continues to the next queue after getting a message from the saga.

Configuration

For each type of document, you have a dedicated saga YAML file.

- saga-backup-file: To define steps for processing backup files.
- saga-invoice-image-file: To define steps for processing invoice image files.
- saga-pod-file: To define steps for processing the POD files.
- saga-packing-list-file: To define steps for processing the packaging list.

Sample YAML format

Below is the sample YAML format for saga-backup-file.

```

kind: document
metadata:
  name: deduction/v1/documents/saga/saga-backup-file
spec:
  steps:
    - name: Download Backup File
      version: 200
      queueName: download-backup-file
      queueDelaySeconds: 10
      onSuccess: Validation Backup File
      onFailure: null

    - name: Validation Backup File
      version: 250
      queueName: validation-backup-file
      queueDelaySeconds: 45
      onSuccess: Extract Backup File
      onFailure: null

    - name: Extract Backup File
      version: 300
      queueName: extract-backup-file
      queueDelaySeconds: 45
      onSuccess: Persist Backup File Data
      onFailure: null

    - name: Persist Backup File Data
      version: 400
      queueName: persist-backup-file-data
      queueDelaySeconds: 10
      onSuccess: null
      onFailure: null

```

Parameter	Description
Name	The step name.
Version	The step order number.
queueName	The message bus queue from which the steps are performed.
queueDelaySeconds	The delay on which each step is performed in the process.
onSuccess onFailure	The step to be performed after the current step success and failure. If the next step is null, then it is the last step of the saga, or the system is instructed to skip the next steps.

For example, in the above template:

```

- name: Validation Backup File
  version: 250
  queueName: validation-backup-file
  queueDelaySeconds: 45
  onSuccess: Extract Backup File
  onFailure: null

```

If the validation Backup File is successful, then the Extract Backup File step is performed, and on failure nothing is performed and the process ends.