



## Serverless x GenAI BKK Workshop

Generative AI at scale: Serverless workflows for enterprise-ready apps

[Pre-requisite] Enable foundation model access in Amazon Bedrock

[Pre-requisite] Configuring the front-end application

### ► Playground

### ▼ Use cases

#### ► Building a RAG pipeline

#### ▼ Document extraction and summarization

##### ► Intelligent document processing with Generative AI

##### ▼ Scaling with serverless workflows

Building the workflow

[Verifying the workflow execution](#)

High level Code Walkthrough

Scheduling using Amazon EventBridge Scheduler (Optional)

Additional summarization techniques (Optional)

Summary

### ► Workshop Cleanup

### ▼ AWS account access

[Open AWS console \(us-west-2\)](#)

[Get AWS CLI credentials](#)

Exit event

[Event dashboard](#) > [Use cases](#) > [Document extraction and summarization](#) > [Scaling with serverless workflow](#)

## Verifying the workflow execution

In this section, you are going to first run the workflow, verify the execution in the Step Functions console and validate the results by accessing the Amazon DynamoDB database.

### Run the workflow

1. Access [Amazon S3 console](#) in another tab and search for S3 bucket containing the string `s3useruploadbucket`. Note the name.
  - Take a moment to explore the contents of the bucket. There are two PDF files and a JSON file. Download the JSON file and check the content. You will notice a JSON array with two items in the `manifest.json`. For the workshop, we have only two items. But, you can have millions of items in the JSON array. Distributed map iterates on the items, runs them as configured.
2. Select **Execute** from the Step Functions designer studio.
3. If you are not on the State Machines page, choose State machines on the left side hamburger menu icon and select the `DocumentProcessingWorkflow`.
4. In the Start execution popup, paste the following input. You are running the workflow with this input. These input values are substituted for the corresponding variable at run time. Replace the `{{INPUT_BUCKET}}` with the bucket name in step 1.

```
1 {
2   "input_bucket": "{{INPUT_BUCKET}}",
3   "key": "manifest.json"
4 }
```



### View the Workflow execution in Step Functions Console

1. In the execution details page, select Distributed Map state in the Graph View, then select **Details** tab.

**Graph view**

**Map**

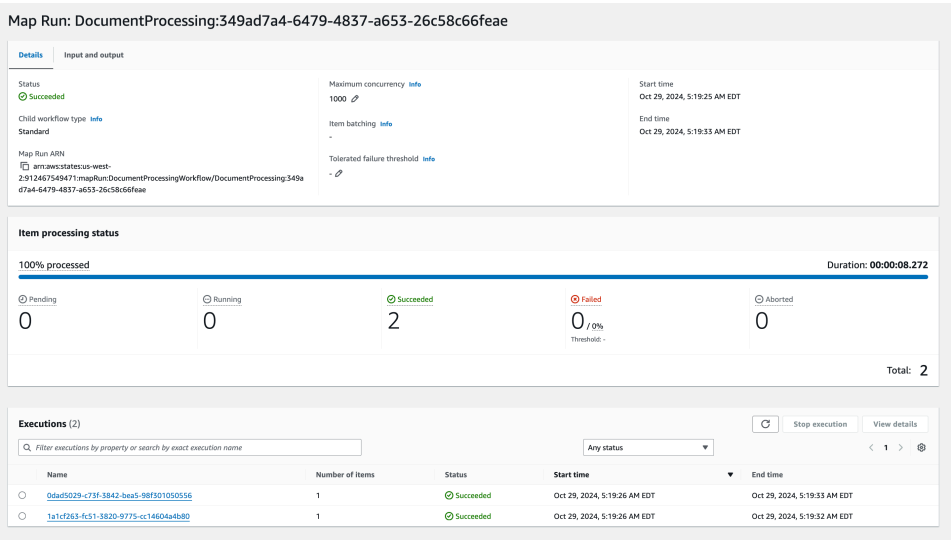
Map Run [Map Run](#) [Test state](#)

Input/Output	Variables	Details	Definition	Events
<b>Status</b> Succeeded	<b>Type</b> Map	<b>Processing mode</b> DISTRIBUTED <a href="#">Learn more</a>	<b>Duration</b> 00:00:21.238	
<b>Resource</b> <a href="#">Map Run</a>	<b>Started After</b> 00:00:00.026			

**Items overview**

Total	Pending
2	0
Succeeded	In progress
2	0

2. Select **Map Run** link to view details of the Distributed Map execution.
3. This page provides a summary of the Distributed Map job.

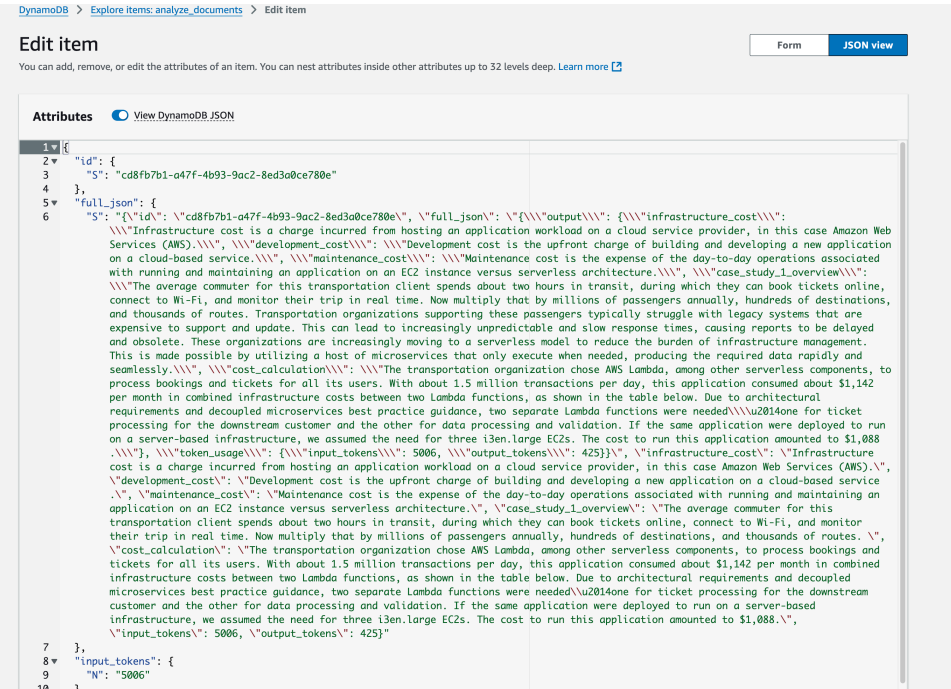


8. We can see that 2 child workflows executions completed successfully with 0 failures. If you had 10 entries in the manifest.json, you will see 10 executions here as it loops through every item in the manifest.json array to process the document.
9. On the Map Run execution page, explore the State input/output information by selecting the corresponding tabs. If you have trouble viewing the input, enable the Advanced view option at the top of the page.

Verify the results

1. Navigate to [Amazon DynamoDB](#) in your AWS Console. Make sure you are in the correct region.
2. If you're not already on the Tables page, select Tables from the hamburger menu on the left. You should see two tables: analyze\_documents and document\_summaries. Let's inspect the analyze\_documents table first, where you should find 2 records.

Select **Explore table items** on top right. Inspect one of the extracted JSON from the DynamoDB Table. In the next section, we will walk through the code to explain how the document was extracted in this format.



1. Navigate to the document\_summaries table and inspect the summary for this document.

DynamoDB > Explore items: document\_summaries > Edit item

Edit item

FormJSON view

You can add, remove, or edit the attributes of an item. You can nest attributes inside other attributes up to 32 levels deep. [Learn more](#)

Attributes

Add new attribute

Attribute name	Value	Type	
id - Partition key	4d62ce7e-7399-4e52-aaaa-2abee87dae5e	String	
file_key	serverless-document.pdf	String	Remove
input_tokens	4695	Number	Remove
output_tokens	172	Number	Remove
summary	- The document discusses the cost considerations for a transportation organization that is evaluating running its ticket booking and data processing applications on AWS Lambda versus a server-based EC2 infrastructure.	String	Remove

CancelSaveSave and close

Summary

You have come to the end of the document extraction and summarization module. In this module, you created a workflow with distributed map, learnt how to perform document extraction and summarization at scale using Step Functions Distributed Map.

✔

**Congratulations!** You used Step Functions Distributed Map state to extract and summarize documents at scale.

Previous

Next

https://catalog.us-east-1.prod.workshops.aws/event/dashboard/en-US/workshop/40-usecases/70-doc-summarization/70-scaling-with-workflows/2-...

3/3