



# AWS Step Functions

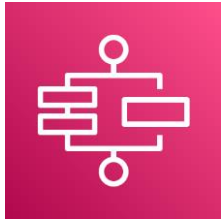


## At the core of the lesson

---

You will learn how to define AWS Step Functions and its uses.

## What is Step Functions?



AWS Step Functions

- Step Functions provides serverless orchestration for modern applications.
- Orchestration centrally manages a workflow by breaking it into multiple steps, adding flow logic, and tracking the inputs and outputs between the steps.



You can use AWS Step Functions to coordinate AWS services into serverless workflows. Workflows consist of a series of steps. The output of one step is the input to the next step.

As your applications run, Step Functions maintains the application state, tracking exactly which workflow step your application is in, and stores an event log of data that is passed between application components. That means if the workflow is interrupted for any reason, your application can pick up right where it left off.

## Using Step Functions

---

Step Functions gives you the ability to reuse components and use different services in your application. Step Functions also does the following:

- Coordinates existing AWS Lambda functions and microservices into applications
- Keeps application logic separated from implementation



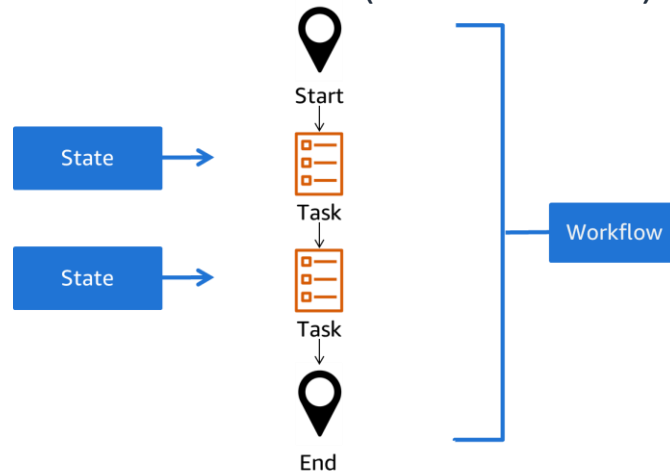
Step Functions coordinates your existing AWS Lambda functions and microservices into robust applications so that you can quickly rewire them into new compositions. The tasks in your workflow can run anywhere, including on instances, containers, functions, and mobile devices. Using Step Functions, you can quickly create distributed applications that leverage AWS services in addition to your own microservices.

Microservices are an architectural and organizational approach to software development where software consists of small independent services that communicate over well-defined APIs.

Step Functions keeps your application logic strictly separated from the implementation of your application. You can add, move, swap, and reorder steps without having to make changes to your business logic. Maintenance is simplified through this separation. You also gain modularity, scalability, and reuse of code.

## Core concepts

Step Functions is based on workflows (or state machines) and tasks.



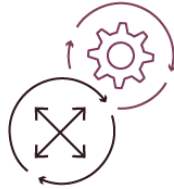
The workflows that you build with Step Functions are called *state machines*, and each step of your workflow is called a *state*. Tasks perform work, either by coordinating another AWS service or an application that you can host basically anywhere. You can reuse components, edit the sequence of steps, or swap out the code called by task states as your needs change.

## Benefits



### Productivity

Gives you the ability to connect and coordinate distributed components and microservices to quickly create applications



### Agility

Helps you diagnose and debug problems faster



### Resilience

Manages the operations and infrastructure of service coordination to help ensure availability at scale and under failure



©2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

6

Take a moment to become familiar with the benefits of Step Functions.

#### Productivity:

- Step Functions accelerates application development because you can define and manage the workflow of your application independently from its business logic.
- You can write your applications faster because Step Functions provides ready-made steps (states) and built-in error handling for your workflow.

#### Agility:

- Step Functions records a history each time it runs so that you can review all events in the sequence in one location.
- You can scale from one runtime to thousands of concurrent runtimes, especially when you use Step Functions with other serverless AWS resources. Examples of such resources include Lambda, Amazon Simple Storage Service (Amazon S3), and Amazon DynamoDB.
- With Step Functions, you pay for only what you use when you use it.

#### Resilience:

- Step Functions supports automatic error handling for graceful exits.
- It operates at scale, and you do not need to configure or manage its underlying resources.

## Features

---

Step Functions is a managed serverless service. Its main features include the following:

- Automatic scaling
- High availability
- Pay per use
- Security and compliance

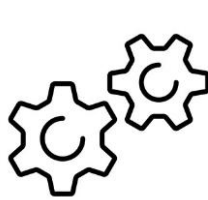


Here are more details about the features of Step Functions:

- Step Functions automatically scales the operations and underlying compute to run the steps of your application for you in response to changing workloads. Step Functions scales automatically to help ensure that the performance of your application workflow remains consistent as the frequency of requests increases.
- Step Functions has built-in fault tolerance and maintains service capacity across multiple Availability Zones in each Region to protect applications against individual machine or data center failures. This feature helps ensure high availability for both the service itself and for the application workflow that it operates.
- With Step Functions, you pay for each transition from one state to the next. Billing is metered by state transition, and you do not pay for idle time regardless of how long each state persists (up to 1 year). This feature keeps Step Functions cost-effective as you scale from a few calls to tens of millions.
- Step Functions is integrated with AWS Identity and Access Management (IAM) and recommends a least-privileged IAM policy for all of the resources used in your workflow. You can access Step Functions from VPC-initiated Lambda functions and other AWS services without traversing the public internet by using AWS PrivateLink.

## Use cases

Step Functions is useful for creating end-to-end workflows to manage jobs with dependent components and for dividing business processes into a series of steps.



Data processing



IT automation



Ecommerce



Web applications

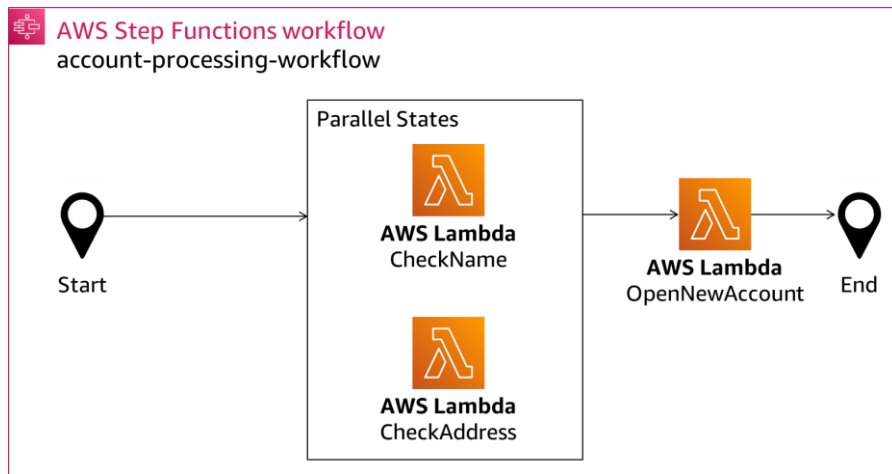


Here are some common ways that Step Functions are used:

- **Data processing:** You can use Step Functions to produce reports by consolidating data from multiple databases and validating, processing, and normalizing the data. You can also use it to refine and reduce large data sets into useful formats or coordinate multi-step analytics and machine learning workflows.
- **IT automation:** You can build tools for continuous integration and continuous deployment by using Step Functions. You can also synchronize data between source and destination S3 buckets by using Step Functions. You can use Step Functions to create event-driven applications that automatically respond to changes in infrastructure and handle repetitive processes in a workflow.
- **Ecommerce:** You can implement critical business processes, such as order fulfillment or inventory tracking, by using Step Functions. By using the visual workflows in Step Functions, other teams can sit down with developers to exchange ideas, identify optimizations, and shorten the processing time.
- **Web applications:** You can use Step Functions to combine Lambda functions to build a web-based application with a human approval. You can use it to implement a robust user registration process and sign-on authentication.



## Step Functions example



You can use Step Functions to implement a business process as a series of steps that make up a workflow. The individual steps in the workflow can invoke a Lambda function that has some business logic. This slide shows an example.

In this example of a banking system, a new bank account is created after validating a customer's name and address by using the account-processing-workflow AWS Step Functions workflow. The workflow begins with two Lambda functions—CheckName and CheckAddress—running in parallel as task states. Once both are complete, the workflow initiates the OpenNewAccount Lambda function. You can define retry and catch clauses to handle errors from task states. You can use predefined system errors or handle custom errors thrown by these Lambda functions in your workflow. Because your workflow code takes on error handling, the Lambda functions can focus on the business logic and have less code.

## Checkpoint questions

---

1. What are the features of Step Functions?
2. What are the benefits of Step Functions?
3. What is the difference between state machines, states, and tasks?



The answers to the questions are as follows:

1. What are the features of Step Functions?  
Automatic scaling, high availability, pay per use, and security and compliance
2. What are the benefits of Step Functions?  
Productivity, agility, and resilience
3. What is the difference between state machines, states, and tasks?  
The workflows you build with Step Functions are called state machines, and each step of your workflow is called a state. Tasks perform work.

## Key ideas

---



- Step Functions is a serverless orchestration service. You can use it to combine Lambda functions and other AWS services to build business-critical applications.
- With Step Functions, you can quickly create distributed applications that leverage AWS services in addition to your own microservices.



# Thank you

Corrections, feedback, or other questions?  
Contact us at <https://support.aws.amazon.com/#/contacts/aws-training>.  
All trademarks are the property of their owners.

©2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

12