AWS Step Functions FAQs

General

Q: What is AWS Step Functions?

AWS Step Functions is a fully managed service that makes it easy to coordinate the components of distributed applications and microservices using visual workflows. Building applications from individual components that each perform a discrete function lets you scale and change applications quickly. Step Functions is a reliable way to coordinate components and step through the functions of your application. Step Functions provides a graphical console to arrange and visualize the components of your application as a series of steps. This makes it simple to build and run multi-step applications. Step Functions automatically triggers and tracks each step, and retries when there are errors, so your application executes in order and as expected. Step Functions logs the state of each step, so when things do go wrong, you can diagnose and debug problems quickly. You can change and add steps without even writing code, so you can easily evolve your application and innovate faster.

AWS Step Functions manages the operations and underlying infrastructure for you to help ensure your application is available at any scale.

Q: What are the benefits of designing my application using service orchestration to coordinate tasks?

Breaking an application into service components (or steps) ensures that the failure of one component does not bring the whole system down, that each component scales independently, and that components may be updated without requiring the entire system to be redeployed after each change. The coordination of service components involves managing execution dependencies, scheduling, and concurrency in accordance with the logical flow of the application. In such an application, developers may use service orchestration to do this and to handle failures.

Q: What are some use cases that can be solved with AWS Step Functions?

AWS Step Functions helps with any computational problem or business process that can be subdivided into a series of steps. Common use cases include:

* Document and Data Processing - Consolidate data from multiple databases into unified reports. Refine and reduce large data sets into useful formats.
* DevOps - Build tools for continuous integration and continuous deployment. Create event-driven applications that automatically respond to changes in infrastructure.
* E-commerce - Automate mission-critical business processes, such as order fulfillment and inventory tracking.
* Web Applications - Implement robust user registration processes and sign-on authentication.

Q: How does AWS Step Functions work?

With AWS Step Functions, you define your application as a state machine, a series of steps that together capture the behavior of the app. States in the state machine may be tasks, sequential steps, parallel steps, branching paths (choice), and/or timers (wait). Tasks are units of work, and this work may be performed by AWS Lambda functions, Amazon EC2 instances of any type, containers, or on premises servers—anything that can communicate with the Step Functions API may be assigned a task. The visual console automatically graphs each state in the order of execution, making it easy to design multi-step applications. The console highlights the real-time status of each step and provides a detailed history of every execution. Step Functions operates and scales the steps of your application and underlying compute for you to ensure your application executes reliably under increasing demand.

Q: When should I use AWS Step Functions vs Amazon SQS?

You should consider AWS Step Functions when you need to coordinate service components in the development of highly scalable and auditable applications. You should consider using [Amazon Simple Queue Service](https://aws.amazon.com/sqs/) (SQS), when you need a reliable, highly scalable, hosted queue for sending, storing, and receiving messages between services. Step Functions keeps track of all tasks and events in an application. Amazon SQS requires you to implement your own application-level tracking, especially if your application uses multiple queues. The Step Functions Console and visibility APIs provide an application-centric view that lets you search for executions, drill down into an execution’s details, and administer executions. Amazon SQS requires implementing such additional functionality. Step Functions offers several features that facilitate application development, such as passing data between tasks and flexibility in distributing tasks. Amazon SQS requires you to implement some application-level functionality. While you can use Amazon SQS to build basic workflows to coordinate your distributed application, you can get this facility out-of-the-box with Step Functions, alongside other application-level capabilities.

Q: When should I use AWS Step Functions vs AWS Batch?

[AWS Batch](https://aws.amazon.com/batch/) is a service that makes it easy to run batch computing workflows of any scale in the AWS cloud. You should use AWS Step Functions when you would like to build a distributed application as a series of steps, including sequential, parallel and/or branching logic (choices). You can use Batch to run jobs in your application and can use Step Functions to submit multiple Batch jobs with interdependencies. Use Batch when you would like automatic management of compute infrastructure in order to scale to the amount of batch processing required.

Q: When should I use AWS Step Functions vs Amazon Simple Workflow Service (SWF)?

You should consider using AWS Step Functions for all your new applications, since it provides a more productive and agile approach to coordinating application components using visual workflows. If you require external signals to intervene in your processes, or you would like to launch child processes that return a result to a parent, then you should consider [Amazon Simple Workflow Service](https://aws.amazon.com/swf/) (SWF). With Amazon SWF, instead of writing state machines in declarative JSON, you write a decider program to separate activity steps from decision steps. This provides you complete control over your orchestration logic, but increases the complexity of developing applications. You may write decider programs in the programming language of your choice, or you may use the Flow framework to use programming constructs that structure asynchronous interactions for you.

Q: What will happen to the existing Amazon Simple Workflow Service (SWF)?

AWS will continue to provide Amazon Simple Workflow (SWF) and to support all Amazon SWF customers.

Q: How does AWS Step Functions work with other AWS services?

AWS Step Functions works with several other AWS services, including AWS CloudFormation, AWS Lambda, Amazon EC2, Amazon EC2 Container Service (ECS), Amazon API Gateway, Amazon CloudWatch, AWS CloudTrail, and Auto Scaling.

The following are examples of some things you can do using various AWS services:

* Launch multiple related resources required for AWS Step Functions state machines and activities using AWS CloudFormation.
* Implement tasks using Lambda functions or programs running on EC2 or in ECS containers.
* Associate your Step Functions APIs with methods in an API Gateway API.
* Scale your EC2 task fleet for AWS Step Functions activities in response to CloudWatch metrics.

Q: How does AWS Step Functions work with AWS Lambda?

You can use [AWS Lambda](https://aws.amazon.com/lambda/) functions to perform tasks within your state machine. AWS Step Function passes input to the specified Lambda function, and then waits for the Lambda function to return a result. With Lambda, you can create tasks without provisioning or managing servers, in all languages supported by Lambda. To learn more about using Step Functions with Lambda, see our Step Functions [documentation](https://aws.amazon.com/documentation/step-functions/).

Q: How does AWS Step Functions work with Amazon EC2 and other compute resources?

All work in your state machine is done by tasks. A task may be an Activity, which can consist of any code in any language. Activities can be hosted on Amazon EC2, Amazon ECS, mobile devices—basically any computer that can communicate with the AWS Step Functions API. Activities long-poll Step Functions using API calls to request work, receive input data, do the work, and return a result.

Q: How does AWS Step Functions work with Amazon API Gateway?

Amazon API Gateway is a fully-managed service that makes it easy for developers to publish, maintain, monitor, and secure APIs at any scale. With just a few clicks in the AWS Management Console, you can associate your Step Functions APIs with methods in your API Gateway API, so that, when an HTTPS request is sent to an API method that you defined, API Gateway invokes your Step Functions API actions.

You can use an Amazon API Gateway API to start Step Functions state machines that coordinate the components of a distributed backend application. You can also integrate human activity tasks into the steps of your application, such as an approval requests and responses. You can even make serverless asynchronous calls to the APIs of services that your application uses.

Amazon API Gateway handles all of the tasks involved in accepting and processing hundreds of thousands of concurrent API calls, including traffic management, authorization and access control, monitoring, and API version management.

Q: How Does AWS Step Functions work with Amazon CloudWatch?

Amazon CloudWatch is a monitoring service for AWS services and applications that you run on AWS. Amazon CloudWatch collects and track metrics, sets alarms, and automatically reacts to changes in AWS Step Functions.

The following are examples of things you can do with Amazon CloudWatch and AWS Step Functions:

* Monitor Step Functions workflow metrics.
* Launch Step Functions workflows in response to CloudWatch Events.
* Generate alarms when Step Functions crosses metric thresholds.
* Schedule Step Functions workflows for time intervals as short as one minute using CloudWatch Events.

Q: Can I use AWS Step Functions with my on-premises resources?

Yes. AWS Step Functions applications can combine workers running in the datacenter with workers that run in the cloud. The workers in the datacenter continue to run as usual, along with any cloud-based workers.

Getting Started

Q: How do I get started with AWS Step Functions?

To start using AWS Step Functions, go to the [AWS Step Functions detail page](https://aws.amazon.com/step-functions/details/) and click the “Get Started for Free” button. If you do not have an Amazon Web Service account, you will be prompted to create one. After signing up, you can run a sample walkthrough in the AWS Step Functions Console which takes you through the steps of creating a state machine. Go to AWS Step Functions from the AWS Management Console. Next select the “Hello World” blueprint and add a “HelloWorld” Lambda function. You can now preview and run your state machine. To start using AWS Step Functions in your applications, please refer to the AWS Step Functions [documentation](https://aws.amazon.com/documentation/step-functions/).

Q: What language does AWS Step Functions use?

You may use any programming language to write an Activity, as long as you can communicate with AWS Step Functions using web service APIs. For convenience, you may use an AWS SDK in the language of your choosing. AWS Lambda supports code written in Node.js (JavaScript), Python, and Java (Java 8 compatible), and C# (using the .NET Core runtime). State machines are defined in JSON using [Amazon States Language](http://states-language.net/spec.html).

Security

Q: How do I authenticate users?

AWS Step Functions is integrated with [AWS Identity and Access Management](https://aws.amazon.com/iam/) (IAM). IAM policies can be used to control access to the Step Functions APIs.

Pricing and Billing

Q: How am I charged for using AWS Step Functions?

With AWS Step Functions you pay only for the transition from one step of your application workflow to the next, called a *state transition*. Billing is metered by state transition, regardless of how long each state persists (up to one year), making it cost-effective and easy to scale automatically from a few executions per day to tens of millions per month. Please see the [AWS Step Functions pricing page](https://aws.amazon.com/step-functions/pricing/) for details.