

# Code Pipeline

- AWS CodePipeline is a continuous delivery service you can use to model, visualize, and automate the steps required to release your software. You can quickly model and configure the different stages of a software release process. CodePipeline automates the steps required to release your software changes continuously. For information about pricing for CodePipeline.
- CodePipeline is a *continuous delivery* service that automates the building, testing, and deployment of your software into production.
- Continuous delivery is a software development methodology where the release process is automated. Every software change is automatically built, tested, and deployed to production. Before the final push to production, a person, an automated test, or a business rule decides when the final push should occur. Although every successful software change can be immediately released to production with continuous delivery, not all changes need to be released right away.
- Continuous integration is a software development practice where members of a team use a version control system and frequently integrate their work to the same location, such as a main branch. Each change is built and verified to detect integration errors as quickly as possible. Continuous integration is focused on automatically building and testing code, as compared to *continuous delivery*, which automates the entire software release process up to production.
- You can use the CodePipeline console, the AWS Command Line Interface (AWS CLI), the AWS SDKs, or any combination of these to create and manage your pipelines.

## What can I do with CodePipeline?

You can use CodePipeline to help you automatically build, test, and deploy your applications in the cloud. Specifically, you can:

- **Automate your release processes:** CodePipeline fully automates your release process from end to end, starting from your source repository through build, test, and deployment. You can prevent changes from moving through a pipeline by including a manual approval action in any stage except a Source stage. You can release when you want, in the way you want, on the systems of your choice, across one instance or multiple instances.
- **Establish a consistent release process:** Define a consistent set of steps for every code change. CodePipeline runs each stage of your release according to your criteria.
- **Speed up delivery while improving quality:** You can automate your release process to allow your developers to test and release code incrementally and speed up the release of new features to your customers.
- **Use your favorite tools:** You can incorporate your existing source, build, and deployment tools into your pipeline. For a full list of AWS services and third-party tools currently supported by CodePipeline, see [Product and service integrations with CodePipeline](#).
- **View progress at a glance:** You can review real-time status of your pipelines, check the details of any alerts, retry failed stages or actions, view details about the source revisions used in the latest pipeline execution in each stage, and manually rerun any pipeline.
- **View pipeline history details:** You can view details about executions of a pipeline, including start and end times, run duration, and execution IDs.

## CodePipeline concepts

Modeling and configuring your automated release process is easier if you understand the concepts and terms used in AWS CodePipeline. Here are some concepts to know about as you use CodePipeline.

For an example of a DevOps pipeline, see [DevOps pipeline example](#).

The following terms are used in CodePipeline:

### Topics

- Pipelines
- Pipeline executions
- Stage operations

- Action executions
- Execution types
- Action types
- Artifacts
- Source revisions
- Triggers
- Variables

