PROJECT: AWS code build and code deployment using Jenkins:

DOCUMENTATION: AWS code build and AWS code deploy

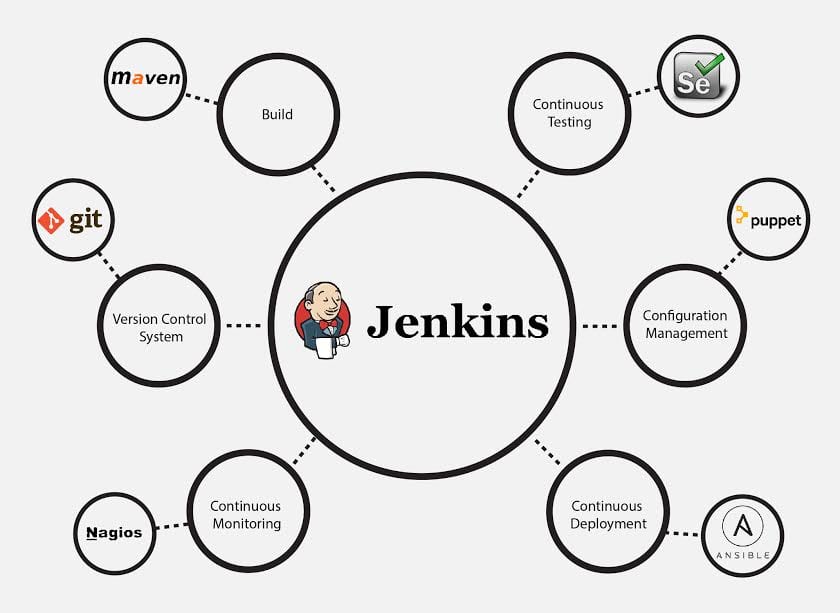
Jenkins is an open-source automation tool written in Java with plugins built for continuous integration. Jenkins is used to build and test your software projects continuously making it easier for developers to integrate changes to the project, and making it easier for users to obtain a fresh build. It also allows you to continuously deliver your software by integrating with a large number of testing and deployment technologies.

With Jenkins, organizations can accelerate the software development process through automation. Jenkins integrates development life-cycle processes of all kinds, including build, document, test, package, stage, deploy, static analysis, and much more.

Jenkins achieves Continuous Integration with the help of plugins. Plugins allow the integration of Various DevOps stages. If you want to integrate a particular tool, you need to install the plugins for that tool. For example Git, Maven 2 project, Amazon EC2, HTML publisher etc.

**Advantages of Jenkins include:**

* It is an open-source tool with great community support.
* It is easy to install.
* It has 1000+ plugins to ease your work. If a plugin does not exist, you can code it and share it with the community.
* It is free of cost.
* It is built with Java and hence, it is portable to all the major platforms



Aws code build:

AWS CodeBuild is a fully managed build service in the cloud. CodeBuild compiles your source code, runs unit tests, and produces artifacts that are ready to deploy. CodeBuild eliminates the need to provision, manage, and scale your own build servers. It provides prepackaged build environments for popular programming languages and build tools such as Apache Maven, Gradle, and more. You can also customize build environments in CodeBuild to use your own build tools. CodeBuild scales automatically to meet peak build requests.

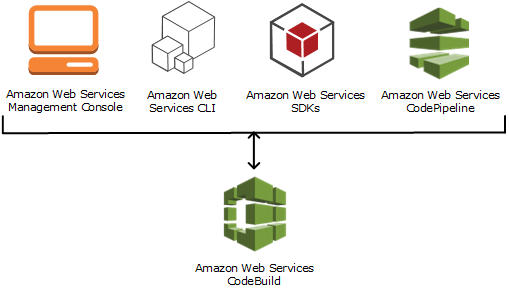
CodeBuild provides these benefits:

* **Fully managed** – CodeBuild eliminates the need to set up, patch, update, and manage your own build servers.
* **On demand** – CodeBuild scales on demand to meet your build needs. You pay only for the number of build minutes you consume.
* **Out of the box** – CodeBuild provides preconfigured build environments for the most popular programming languages. All you need to do is point to your build script to start your first build.

For more information, see [AWS CodeBuild](https://aws.amazon.com/codebuild/).

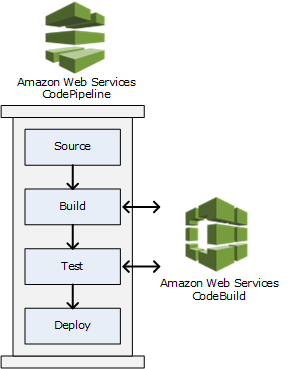
## How to run CodeBuild

You can use the AWS CodeBuild or AWS CodePipeline console to run CodeBuild. You can also automate the running of CodeBuild by using the AWS Command Line Interface (AWS CLI) or the AWS SDKs.



To run CodeBuild by using the CodeBuild console, AWS CLI, or AWS SDKs, see [Run AWS CodeBuild directly](https://docs.aws.amazon.com/codebuild/latest/userguide/how-to-run.html).

As the following diagram shows, you can add CodeBuild as a build or test action to the build or test stage of a pipeline in AWS CodePipeline. AWS CodePipeline is a continuous delivery service that you can use to model, visualize, and automate the steps required to release your code. This includes building your code. A pipeline is a workflow construct that describes how code changes go through a release process.



Aws code deploy:

CodeDeploy can deploy application content that runs on a server and is stored in Amazon S3 buckets, GitHub repositories, or Bitbucket repositories. CodeDeploy can also deploy a serverless Lambda function. You do not need to make changes to your existing code before you can use CodeDeploy.

CodeDeploy makes it easier for you to:

* Rapidly release new features.
* Update AWS Lambda function versions.
* Avoid downtime during application deployment.
* Handle the complexity of updating your applications, without many of the risks associated with error-prone manual deployments.

The service scales with your infrastructure so you can easily deploy to one instance or thousands.

CodeDeploy works with various systems for configuration management, source control, [continuous integration](https://aws.amazon.com/devops/continuous-integration/), [continuous delivery](https://aws.amazon.com/devops/continuous-delivery/), and continuous deployment. For more information, see [Product integrations](https://aws.amazon.com/codedeploy/product-integrations/).

The CodeDeploy console also provides a way to quickly search for your resources, such as repositories, build projects, deployment applications, and pipelines. Choose **Go to resource** or press the / key, and then type the name of the resource. Any matches appear in the list. Searches are case insensitive. You only see resources that you have permissions to view. For more information, see [Identity and access management for AWS CodeDeploy](https://docs.aws.amazon.com/codedeploy/latest/userguide/security-iam.html).

**Topics**

* [Benefits of AWS CodeDeploy](https://docs.aws.amazon.com/codedeploy/latest/userguide/welcome.html#benefits)
* [Overview of CodeDeploy compute platforms](https://docs.aws.amazon.com/codedeploy/latest/userguide/welcome.html#compute-platform)
* [Overview of CodeDeploy deployment types](https://docs.aws.amazon.com/codedeploy/latest/userguide/welcome.html#welcome-deployment-overview)
* [We want to hear from you](https://docs.aws.amazon.com/codedeploy/latest/userguide/welcome.html#welcome-contact-us)
* [CodeDeploy primary components](https://docs.aws.amazon.com/codedeploy/latest/userguide/primary-components.html)
* [CodeDeploy deployments](https://docs.aws.amazon.com/codedeploy/latest/userguide/deployment-steps.html)
* [CodeDeploy application specification (AppSpec) files](https://docs.aws.amazon.com/codedeploy/latest/userguide/application-specification-files.html)

## Benefits of AWS CodeDeploy

CodeDeploy offers these benefits:

* **Server, serverless, and container applications**. CodeDeploy lets you deploy both traditional applications on servers and applications that deploy a serverless AWS Lambda function version or an Amazon ECS application.
* **Automated deployments**. CodeDeploy fully automates your application deployments across your development, test, and production environments. CodeDeploy scales with your infrastructure so that you can deploy to one instance or thousands.
* **Minimize downtime**. If your application uses the EC2/On-Premises compute platform, CodeDeploy helps maximize your application availability. During an in-place deployment, CodeDeploy performs a rolling update across Amazon EC2 instances. You can specify the number of instances to be taken offline at a time for updates. During a blue/green deployment, the latest application revision is installed on replacement instances. Traffic is rerouted to these instances when you choose, either immediately or as soon as you are done testing the new environment. For both deployment types, CodeDeploy tracks application health according to rules you configure.
* **Stop and roll back**. You can automatically or manually stop and roll back deployments if there are errors.
* **Centralized control**. You can launch and track the status of your deployments through the CodeDeploy console or the AWS CLI. You receive a report that lists when each application revision was deployed and to which Amazon EC2 instances.
* **Easy to adopt**. CodeDeploy is platform-agnostic and works with any application. You can easily reuse your setup code. CodeDeploy can also integrate with your software release process or continuous delivery toolchain.
* **Concurrent deployments**. If you have more than one application that uses the EC2/On-Premises compute platform, CodeDeploy can deploy them concurrently to the same set of instances.