AWS Elastic Beanstalk

What is Elastic Beanstalk?

AWS Elastic Beanstalk is an easy-to-use service for deploying and scaling web applications and services developed with Java, .NET, PHP, Node.js, Python, Ruby, Go, and Docker on familiar servers such as Apache, Nginx, Passenger, and IIS.

We may say AWS Elastic Beanstalk is an orchestration service offered by Amazon Web Services used to set up your application architecture.

AWS Elastic Beanstalk makes it even easier for developers to quickly deploy and manage applications in the AWS Cloud. Developers simply upload their application, and Elastic Beanstalk automatically handles the deployment details of capacity provisioning, load balancing, auto-scaling, and application health monitoring.

Why AWS Elastic Beanstalk?

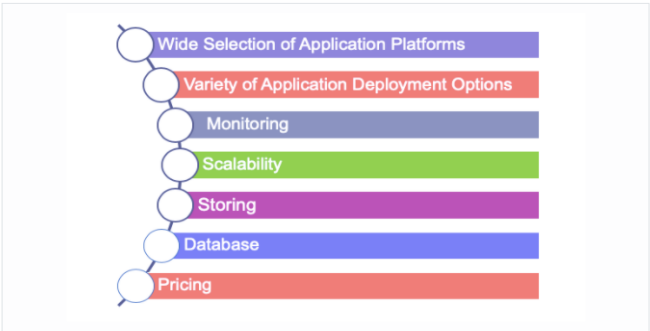
Before and After AWS Elastic Beanstalk

Thanks to the AWS Elastic Beanstalk, you can create your custom application architecture. In this environment, you can locate a number of EC2 instances, S3, Simple Notification Service (SNS), CloudWatch, Autoscaling, Elastic Load Balancers, and Databases. Then you can deploy and manage your applications within minutes in the AWS Cloud

It manages these items for you and also provides you to update your software running on it.

But if you don't want to use AWS Elastic Beanstalk you need to create and manage the same system which AWS Elastic Beanstalk handles in harmony.

Features of AWS Elastic Beanstalk



Features of AWS Elastic Beanstalk

- **Application Platforms:**

Elastic Beanstalk offers a wide selection of application platforms. AWS Elastic Beanstalk allows software applications written in a variety of common languages and frameworks such as Java,.NET, Node.js, PHP, Mysql, Python, Go, and Docker to build the web application.

- **Application Deployment Options:**

AWS Elastic Beanstalk allows you to deploy your code through the **AWS Management Console**, **Elastic Beanstalk Command Line Interface**, **Visual Studio**, and **Eclipse**.

- **Monitoring:**

AWS Elastic Beanstalk provides a useful user interface for tracking and controlling the performance of the applications. Elastic Beanstalk gathers more than 40 main metrics and characteristics to assess the quality of the application.

It is also integrated with Amazon CloudWatch and AWS X-Ray. You can leverage the monitoring dashboard to view key performance metrics such as latency,

- **Management and Updates:**

You can choose to have AWS Elastic Beanstalk automatically update to the latest version of your Elastic Beanstalk environment using Managed Platform Updates.

- **Scaling:**

AWS Elastic Beanstalk uses Elastic Load Balancing and Auto-Scaling to dynamically scale the device in and out depending on the particular needs of your device. In addition, thanks to the multiple availability zones option, you can run your applications in more than one zone

- **Storing:**

AWS Elastic Beanstalk stores your application files and, optionally, server log files in Amazon S3. Optionally, you may configure Elastic Beanstalk to copy your server log files every hour to Amazon S3.

If you delete the Elastic Beanstalk so delete files and log in the S3 as well. Because deleting Elastic Beanstalk doesn't ensure to delete the S3 bucket associated with Elastic Beanstalk automatically.

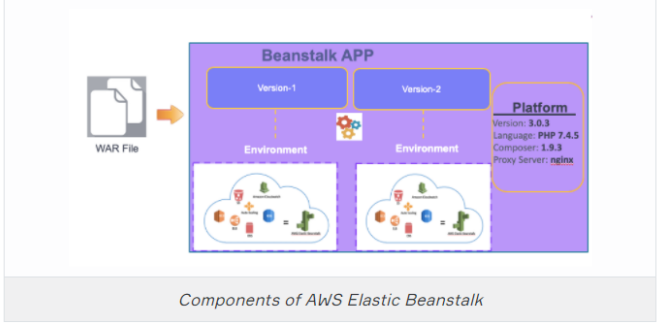
- **Database:**

AWS Elastic Beanstalk does not restrict you to any specific data persistence technology. You can choose to use Amazon Relational Database Service (Amazon RDS) or Amazon DynamoDB, or use Microsoft SQL Server, Oracle, or other relational databases running on Amazon EC2.

- **Pricing:**

AWS Elastic Beanstalk service is free to use. There is no additional charge for AWS Elastic Beanstalk–**you pay only for the AWS resources actually used to store and run your application**.

Components of AWS Elastic Beanstalk



Components of AWS Elastic Beanstalk

- **Application:**

In Elastic Beanstalk, an application version refers to a specific, labeled iteration of deployable code for a web application. An application version points to an Amazon Simple Storage Service (Amazon S3) object that contains the deployable code, such as a Java WAR file. An application version is part of an application.

- **Environment:**

An environment is a **collection of AWS resources** running an application version. When you create an environment, Elastic Beanstalk provisions the resources needed to run the application version you specified.

- **Environment Tier:**

When you launch an Elastic Beanstalk environment, you first choose an environment tier. The environment tier designates the type of application that the environment runs, and determines what resources Elastic Beanstalk provisions to support it. An application that serves HTTP requests runs in a **Web Server Environment Tier**. An environment that pulls tasks from an Amazon Simple Queue Service (Amazon SQS) queue runs in a **Worker Environment Tier**.

- **Environment Configuration:**

An environment configuration identifies a collection of parameters and settings that define how an environment and its associated resources behave. When you update an environment's configuration settings, Elastic Beanstalk automatically applies the changes to existing resources or deletes and deploys new resources (depending on the type of change).

- **Saved Configuration:**

A saved configuration is a template that you can use as a starting point for creating unique environment configurations.

- **Platform:**

A platform is a combination of an operating system, programming language runtime, web server, application server, and Elastic Beanstalk components. You design and target your web application to a platform. Elastic Beanstalk provides a variety of platforms on which you can build your applications.

Supported platform versions:

- Docker
- Multicontainer Docker
- Preconfigured Docker
- Go
- Java SE
- Tomcat
- .NET Core on Linux
- .NET on Windows Server
- Node.js
- PHP
- Python
- Ruby

Complementary Lesson about AWS Elastic Beanstalk