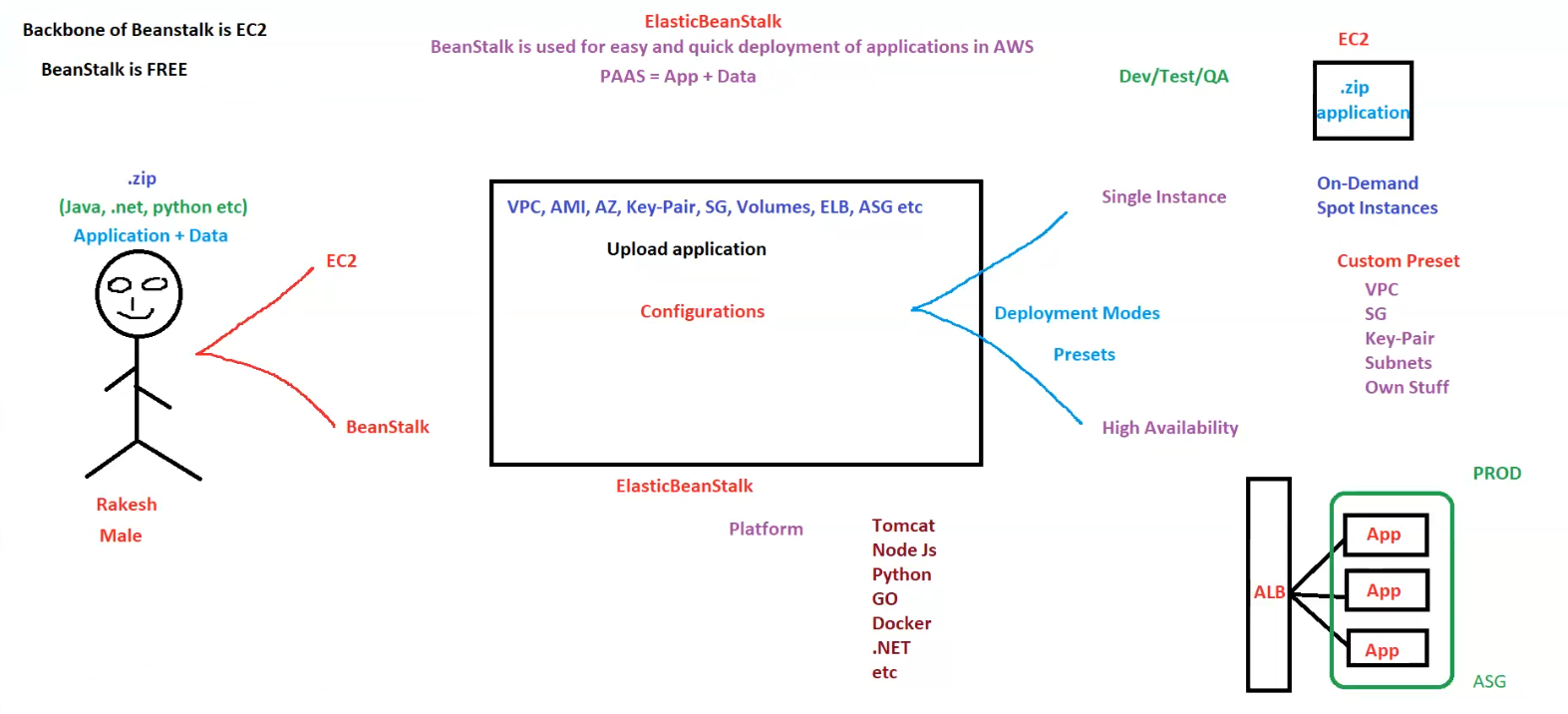
DevOps

[Elastic Bean Stalk 2](#_Toc193571490)

[AWS Elastic Beanstalk Overview 2](#_Toc193571491)

[AWS Elastic Beanstalk Architecture 4](#_Toc193571492)

# Elastic Bean Stalk



## AWS Elastic Beanstalk Overview

**1. What is Elastic Beanstalk?**

* Elastic Beanstalk is a **Platform-as-a-Service (PaaS)** offering by AWS.
* Used for **easy and quick deployment** of applications in AWS.
* Provides a **managed environment** to run applications with minimal manual infrastructure configuration.
* **Backbone of Elastic Beanstalk is EC2.**
* **Elastic Beanstalk is FREE**, but underlying resources (EC2, RDS, etc.) may incur costs.

**2. How Elastic Beanstalk Works**

1. **User uploads application & data** in a .zip format.
   * Supported languages/frameworks: **Java, .NET, Python, Node.js, Go, Docker, etc.**
   * This application package is deployed to an AWS environment.
2. **Elastic Beanstalk provisions the necessary AWS resources**, including:
   * **Networking & Security:** VPC, Security Groups (SG), Key-Pair, Subnets.
   * **Compute & Storage:** EC2 instances, AMIs, Availability Zones (AZ), Volumes.
   * **Scaling & Load Balancing:** Auto Scaling Group (ASG), Elastic Load Balancer (ELB).

**3. Deployment Modes in Elastic Beanstalk**

* **Single Instance Mode:**
  + Suitable for **development, testing, and QA environments**.
  + Uses **On-Demand or Spot Instances**.
* **High Availability Mode:**
  + Recommended for **production environments**.
  + Utilizes **Auto Scaling Group (ASG) and Application Load Balancer (ALB)** to ensure high availability.

**4. Elastic Beanstalk Deployment Presets**

* **Default Presets:**
  + AWS manages networking, compute, and scaling configurations.
* **Custom Presets:**
  + User-defined configurations including:
    - **VPC (Virtual Private Cloud)**
    - **Security Groups (SG)**
    - **Key-Pair**
    - **Subnets**
    - **Other custom network settings**

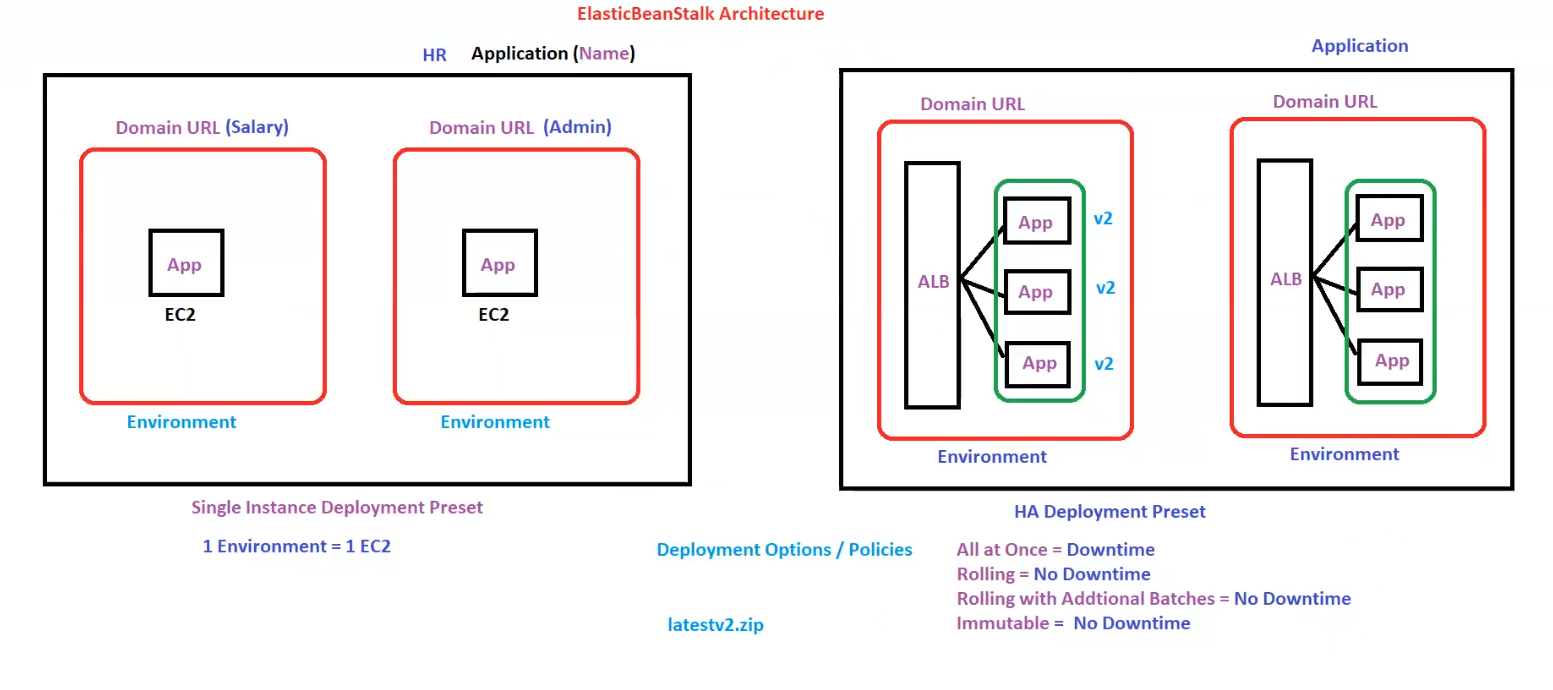
**5. Elastic Beanstalk Environments**

* **Dev/Test/QA Environment:**
  + Single instance deployment for testing.
  + Can use on-demand or spot instances.
* **Production (PROD) Environment:**
  + High availability setup with **Application Load Balancer (ALB) and Auto Scaling Group (ASG).**
  + Ensures scalability, fault tolerance, and redundancy.

**6. Summary of Elastic Beanstalk Features**

| **Feature** | **Description** |
| --- | --- |
| **Application Upload** | Deploy a .zip package with code & configurations |
| **Languages Supported** | Java, .NET, Python, Node.js, Go, Docker, etc. |
| **AWS Resources Managed** | EC2, VPC, Security Groups, Key-Pairs, Subnets, ELB, ASG, etc. |
| **Deployment Modes** | Single Instance (for Dev/Test), High Availability (for Prod) |
| **Presets** | Default (AWS Managed) & Custom (User-defined) |
| **Production Setup** | Load Balancer (ALB) + Auto Scaling Group (ASG) |

## AWS Elastic Beanstalk Architecture

**AWS Elastic Beanstalk Architecture**

**1. Overview**

Elastic Beanstalk provides an easy-to-use platform for deploying applications with different deployment presets. It manages infrastructure, scaling, and application hosting.

**2. Application & Environment Setup**

* **Application Name**: Example shown as "HR".
* **Environments**:
  + Each environment is associated with a **Domain URL**.
  + Two types of environments shown:
    - **Salary (Single Instance)**
    - **Admin (Single Instance)**

**Single Instance Deployment Preset**

* Each environment consists of **one EC2 instance**.
* This setup is commonly used for **development, testing, and small applications**.
* **Structure**:
  + 1 Environment = **1 EC2 instance**
  + The application runs on a single instance.
  + Example: Salary service and Admin service are deployed on separate EC2 instances.

**3. High Availability (HA) Deployment Preset**

* Used for **production environments** requiring scalability and fault tolerance.
* Includes **Application Load Balancer (ALB)** and **Auto Scaling Group (ASG)**.
* Multiple instances of the application run simultaneously.
* **Versioning Example**: Latest application version (e.g., latestv2.zip) is deployed.

**Structure**

* **ALB (Application Load Balancer)** distributes traffic among multiple application instances.
* Ensures **high availability and fault tolerance**.
* Each instance runs **version v2** of the application.

**4. Deployment Options / Policies**

Elastic Beanstalk supports various deployment strategies, each affecting downtime:

| **Deployment Method** | **Downtime Impact** |
| --- | --- |
| **All at Once** | Causes **Downtime** |
| **Rolling** | **No Downtime** (Gradual updates) |
| **Rolling with Additional Batches** | **No Downtime** (Extra instances added while updating) |
| **Immutable** | **No Downtime** (New instances are launched before switching traffic) |

**5. Summary of Elastic Beanstalk Deployment Presets**

| **Deployment Type** | **Features** | **Use Case** |
| --- | --- | --- |
| **Single Instance** | 1 EC2 per environment, simple setup | Dev/Test, Small Applications |
| **High Availability (HA)** | Multiple EC2 instances, ALB, ASG, No Downtime Deployment | Production, Scalable Apps |

**Key Takeaways**

* **Single instance deployments** are simple but lack redundancy.
* **HA deployments** use **ALB & ASG** for scalability and fault tolerance.
* **Deployment policies** allow for zero-downtime upgrades.
* Elastic Beanstalk simplifies application deployment while managing infrastructure.