

# Elastic Beanstalk

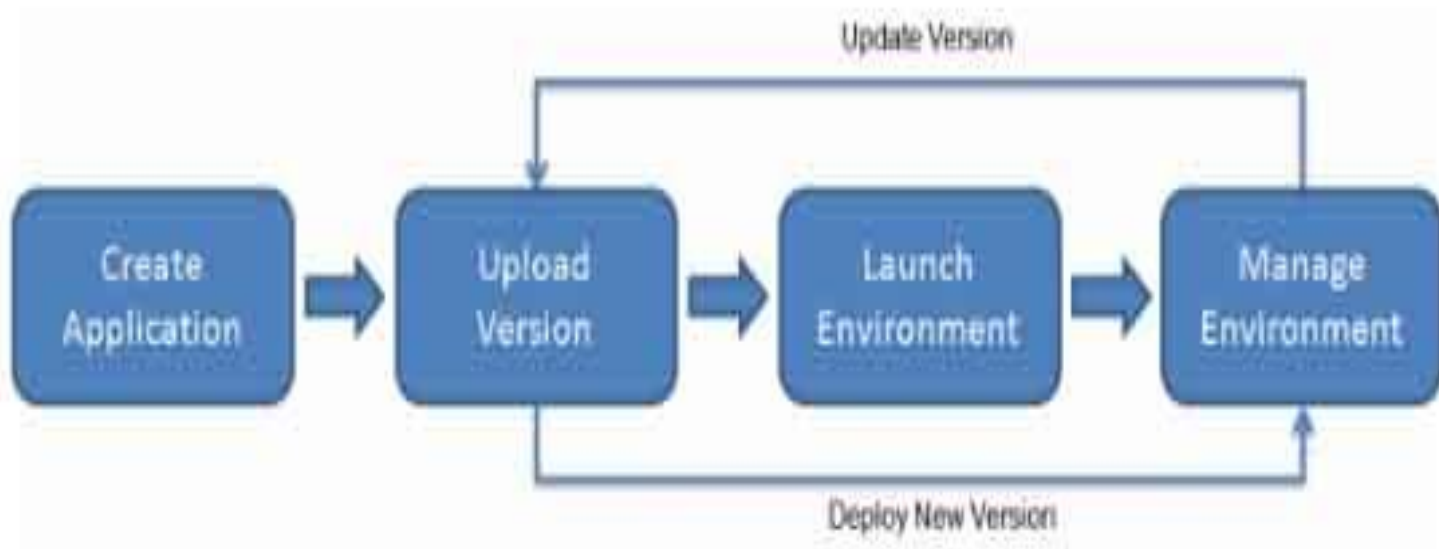
- AWS Elastic Beanstalk is an easy-to-use service for deploying and scaling web applications and services developed with Java, .NET, PHP, Node.js, Python, Ruby, Go, and Docker on familiar servers such as Apache, Nginx, Passenger, and IIS.
- You can simply upload your code and Elastic Beanstalk automatically handles the deployment, from capacity provisioning, load balancing, auto-scaling to application health monitoring. At the same time, you retain full control over the AWS resources powering your application and can access the underlying resources at any time.
- There is no additional charge for Elastic Beanstalk - you pay only for the AWS resources needed to store and run your applications.

# AWS Elastic Beanstalk Benefits

- Easy to start with
- Autoscaling options
- Developer productivity
- Customization
- Cost-effective
- Management and updates

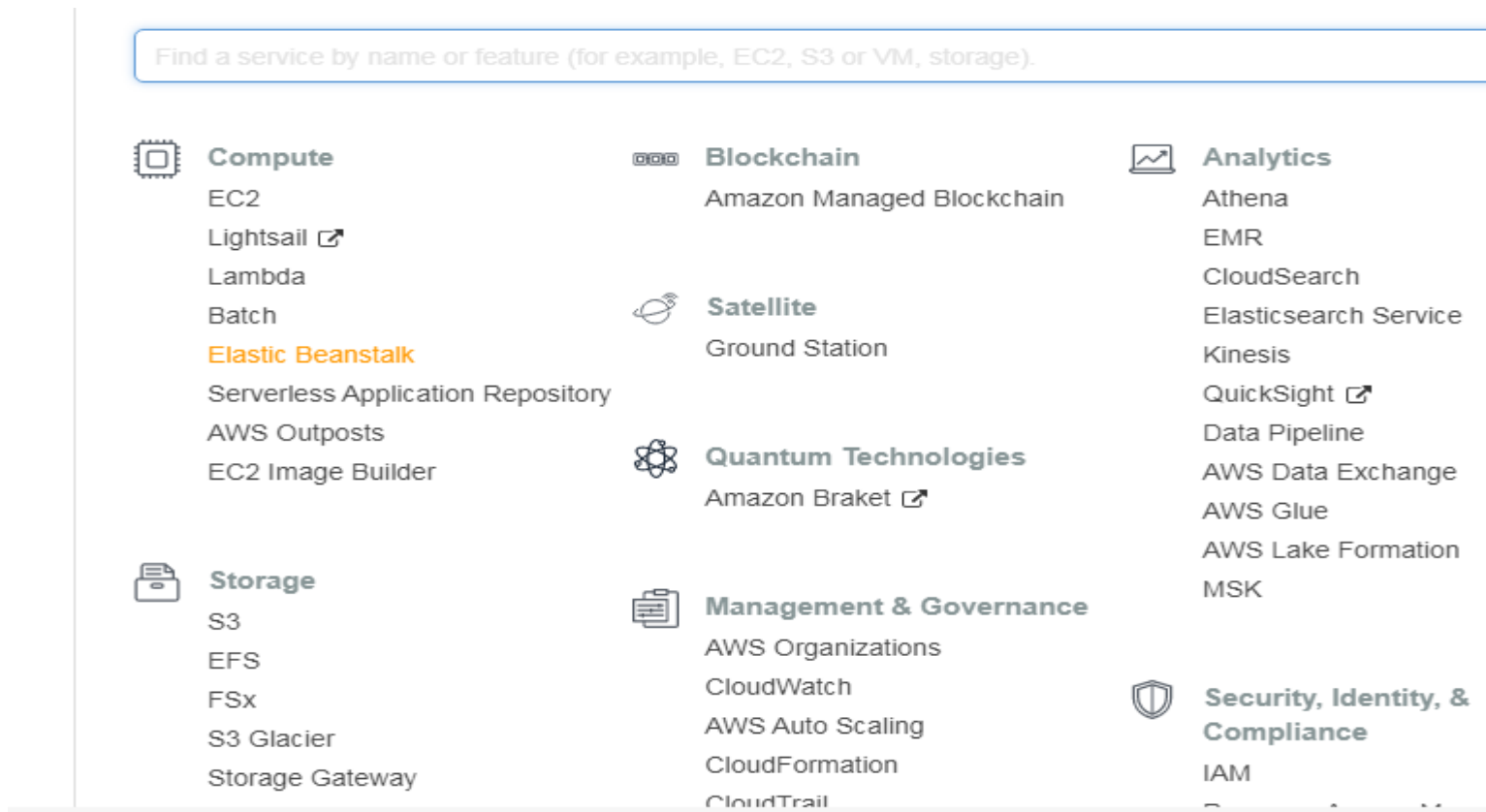


# AWS Elastic Beanstalk Steps



# Lab: Deploy PHP code

- 1) Keep PHP complete application in local system
- 2) The PHP files should be in zip format
- 3) Open AWS ---Services—Elasticbean stalk --





# Lab: Deploy PHP code

4) Click on create a new environment

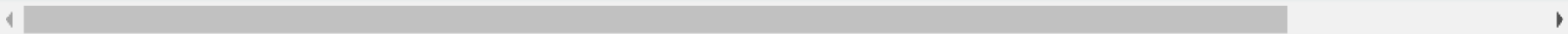
Elastic Beanstalk > Environments

## All environments

 Actions ▼ Create a new environment

< 1 > 

Environment name ▲	Health ▼	Application name ▼	Date created ▼	Last modified ▼	URL ▼	Running versions ▼	Platform ▼
Empty							
No environments to display.							







# Lab: Deploy PHP code

## 5) Select webserver environment

Elastic Beanstalk > Create environment

### Select environment tier

AWS Elastic Beanstalk has two types of environment tiers to support different types of web applications. Web servers are standard applications that listen for and then process HTTP requests, typically over port 80. Workers are specialized applications that have a background processing task that listens for messages on an Amazon SQS queue. Worker applications post those messages to your application by using HTTP.

- ☒ Web server environment  
Run a website, web application, or web API that serves HTTP requests.  
[Learn more](#) 
- ☐ Worker environment  
Run a worker application that processes long-running workloads on demand or performs tasks on a schedule.  
[Learn more](#) 

Cancel

Select

## Lab: Deploy PHP code

Give Application name—select the platform – PHP –Upload the file  
—next—create

Wait for 5 mins –Environment creation will be completed

Copy the obtained URL and paste in new tab(URL link/PHPfilename)

Actions ▾

Overview

Refresh



Health

Ok

Causes

Running Version

nodejs-sample.zip

Upload and Deploy



Configuration

64bit Amazon Linux 2016.03  
v2.1.0 running Node.js

Change

W

Recent Events

Show All

Time	Type	Details
2016-05-06 12:21:30 UTC-0400	INFO	Successfully launched environment: pmd1215-env
2016-05-06 12:20:32 UTC-0400	INFO	Environment health has transitioned from Pending to Ok. Initialization in progress on 1 instance. 0 out of 1 instance completed (running for 4 minutes).
2016-05-06 12:19:32 UTC-0400	INFO	Added instance [i-0f77fb2d3eeaf5ec9] to your environment.
2016-05-06 12:19:26 UTC-0400	INFO	Created CloudWatch alarm named: awseb-e-ebi2jjhtug-stack-AWSEBCloudwatchAlarmHigh-1M3Q9OKFX8Y1B