

Chapter 1

What is AWS Elastic Beanstalk?

AWS Elastic Beanstalk (EB) is a Platform-As-A-Service that allows developers to easily leverage on AWS for application deployment and management.



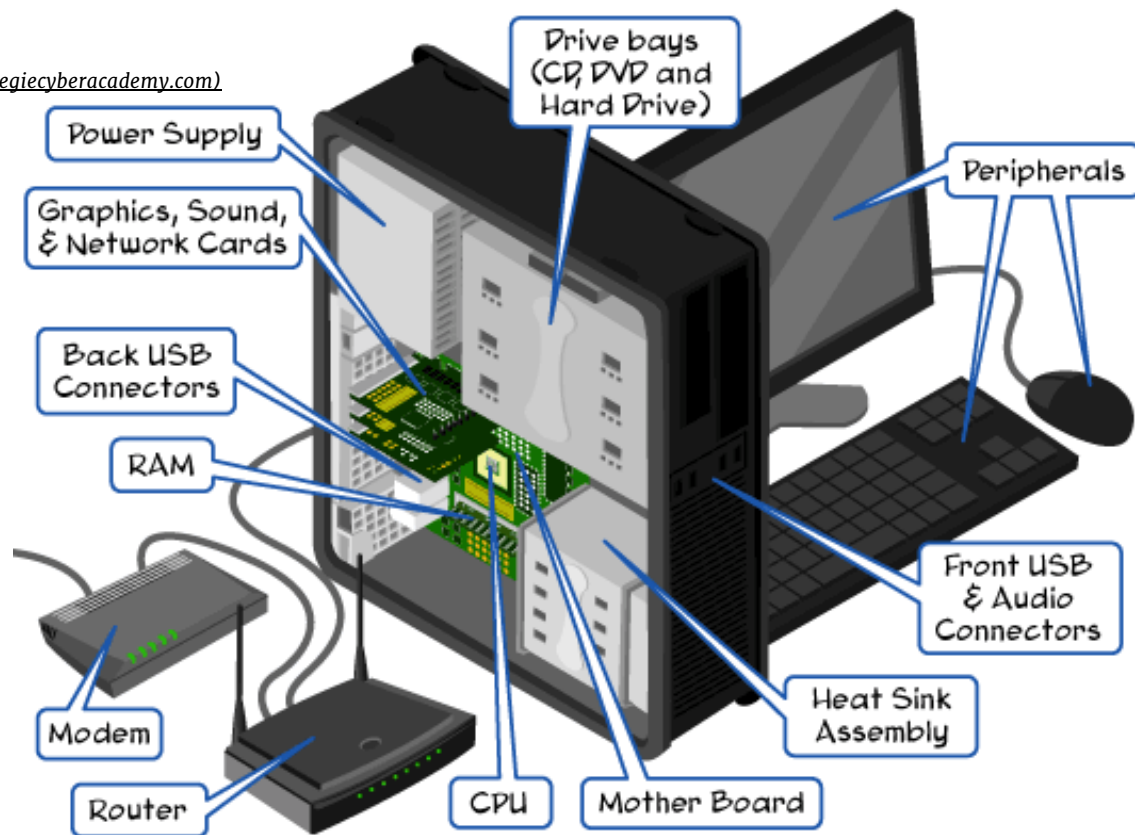
What is so useful about AWS Elastic Beanstalk?

A developer can deploy their applications on AWS without Elastic Beanstalk but that will mean spending time on selecting and piecing together services from the wide array of choices within AWS eco-system.

With Elastic Beanstalk, it abstracts away the infrastructure layer from the developers and allow them to quickly deploy and manage their applications on AWS.

Developers are in no way restricted by Elastic Beanstalk though. They still get granular configuration over their applications.

It's still too technical. Make it more layman.



Alright, so let's say we need to build a computer today. There are two ways we could go about accomplishing it:

1) Go to a computer warehouse.

The computer warehouse is large and confusing with rows after rows of shelves laid out in front of you. We feel overwhelmed with choices but it has all the parts that we need to build our computer so we soldier on.

We carefully navigate between the shelves to pick the CPU, SSD for storage, our operating system software, etc. Then we start to assemble them together to get the final product.

This draws parallel with building application on AWS without Elastic Beanstalk. You pick how powerful you want your EC2 server to be, what type of storage service and how large the storage space, etc. Then you start piecing them together to get your application up and running.

2) Go to a computer retail store.

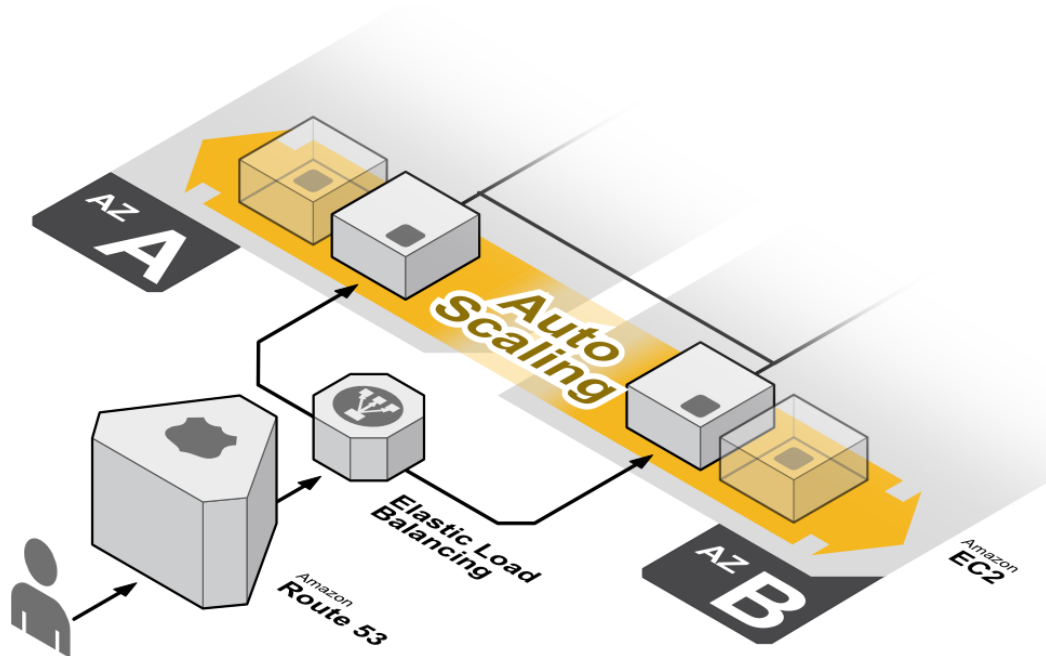
A computer retail store has a more consumer-friendly store front. It's way cleaner and air-conditioned. Along the alley, we can find salesmen are smiling at us, ready to take on any questions.

We tell the salesman our computing needs (e.g. gaming) and we walk out with a computer that can meet that needs. It is as simple as that. No picking up of manual or getting dirt on our hands.

And if we are not happy with the configuration, we have the freedom to upgrade the parts ourselves.

This is what devops' life will be like with Elastic Beanstalk.

Scale like a pro with Elastic Beanstalk



Out of the box, Elastic Beanstalk can give you a auto-scaling, load-balancing environment that are suitable to handle large traffic volume. With EB dashboard, you can easily manage your application by

- setting the maximum number of EC2 instances that can be triggered
- adding trigger rules to add/terminate instances to handle high load
- selecting proxy server such as nginx to cache static files
- enabling application health monitoring
- creating alarms

Why the need for this survival guide?

Despite the upsides that Elastic Beanstalk bring to developers, beginners seems to think that Elastic Beanstalk have a steep learning curve. Most of the time, we have a hard time learning new stuff over the internet because the lack of or bad documentation.

But contrary to popular belief, I found AWS Elastic Beanstalk to be more than sufficient.

In fact, the documentation are overflowing because there are so much Elastic Beanstalk can do. This in terms drowns out what beginners really need to know to get going with Elastic Beanstalk. And this was what exactly happened to me.

So first and foremost, this survival guide is written for myself as point of reference which cuts to the chase.

Secondly, I hope this guide will serve as a good starting point for developers who want to leverage on AWS without spending late-night frustration sieving through google for answers.

Elastic Beanstalk Survival Guide: Table Of Content

- **Chapter 1: [Introduction \(aws-elastic-beanstalk-survival-guide-introduction\)](#)**
- **Chapter 2: [Getting Started \(aws-elastic-beanstalk-getting-started-with-fundamentals\)](#)**
- **Chapter 3: [How to deploy application on AWS Elastic Beanstalk \(how-to-deploy-application-to-aws-elastic-beanstalk\)](#)**
- **Chapter 4: [How to configure AWS Elastic Beanstalk to scale \(configure-aws-elastic-beanstalk-to-scale\)](#)**
- **Chapter 5: [How to map custom domain name for AWS Elastic Beanstalk \(map-custom-domain-to-elastic-beanstalk-application\)](#)**

- **Chapter 6: How to configure SSL for AWS Elastic Beanstalk (configure-ssl-for-aws-elastic-beanstalk)**
- **Chapter 7: How to configure a Worker Environment for AWS Elastic Beanstalk (configure-worker-for-aws-elastic-beanstalk)**

Conclusion

As you can see from the TOC above, we will start by understanding the Elastic Beanstalk lingo first and then straight to deployment.

We will then refine the configuration for staging and production environments before wrapping it up with the one of the most googled topics for AWS Elastic Beanstalk application.

Included in some of the chapters are examples of non-mainstream but actual scenarios such as deploying environments across different AWS accounts, setting up multiple SSL for one environment etc.

So if you are interested in this *AWS Elastic Beanstalk Survival Guide* Series, remember to subscribe below for updates

RSS (<https://colintoh.com/blog/feed>)



Colin Toh | @p0larboy (<https://twitter.com/p0larBoy>)
Full-stack Javascript Developer at [2359 Media](http://2359media.com/) (<http://2359media.com/>)

8
Shares

Subscribe to my newsletter on Modern Web Development

Weekly delivery of web goodies - *Code Tips*,
5CWS and *Blog Updates*

What's your email?

SUBSCRIBE