

'Bad' Infrastructure



Exit



Log in to your 2016 Census



Thank you for participating in the Census. The system is very busy at the moment. Please wait for 15 minutes before trying again. Your patience and cooperation are appreciated. [code 9]



Building Web Apps on Google's Infrastructure

Presented by



Eric Jiang

*This presentation's code/slides can be found on
<https://github.com/lorderikir/googlecloud-techtalk>*



Google Cloud Platform

Talk Summary

1. Introduction to Google Cloud
2. What is Google App Engine
 - a. GAE Environments
 - b. What is Scaling and Why is it Important?
3. Deep-Dive
 - a. Deploying a simple API to Google App Engine
4. Other Tools

[NOTE]: You can play with Google Cloud Platform off your student accounts

Introduction

What is Google Cloud Platform?

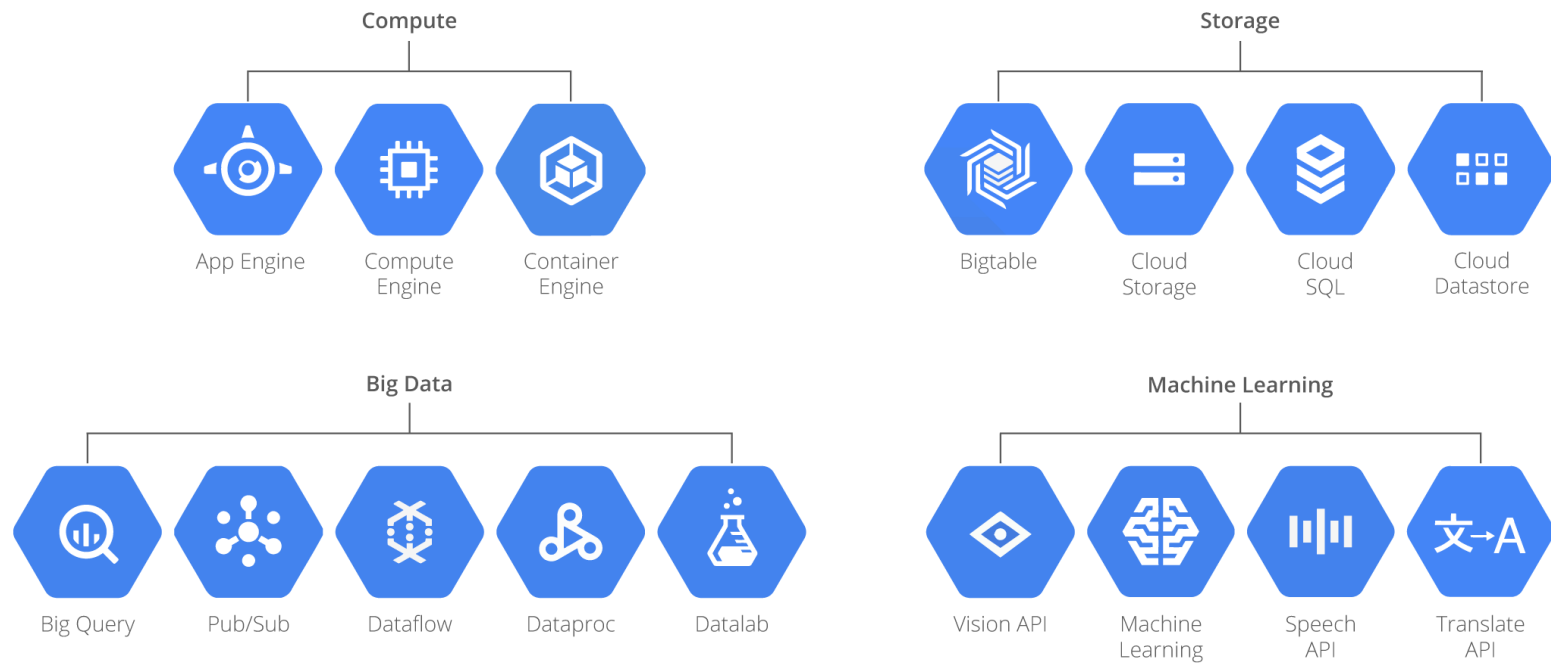
Google Cloud Platform lets you build and host applications and websites, store data, and analyze data on Google's scalable infrastructure.

Composes of many applications, such as:

- Google App Engine (GAE)
- Google Container Engine (GCE)
- Google DataStore
- Cloud ML (built off *TensorFlow*)
- and much more

Did you know that Firebase and [API.AI](#) are both on GCP

Google Cloud Platform



Google App Engine

- designed around the fact that *Google just can't send everyone into their datacentre(s) and update applications across their many datacenters*
- Built off Remote Deployments

Language	Environment
Java 7 (and Kotlin ¹)	Standard
Java 8	Standard (Beta)/Flexible
Node.js	Flexible
Python 2.7	Standard
Python 3.5	Flexible

¹ This for you Kotlin fans out there

Standard Environments run in a specialised environment. Though building the application is more constrained than other environments, it means scaling up is faster.

Flexible Environment applications run off a Docker container, it is designed for applications that receive constant traffic. When deployed they are Google Compute Engine VM²

² Because they run off Docker, you can write your own Dockerfile Configuration to deploy

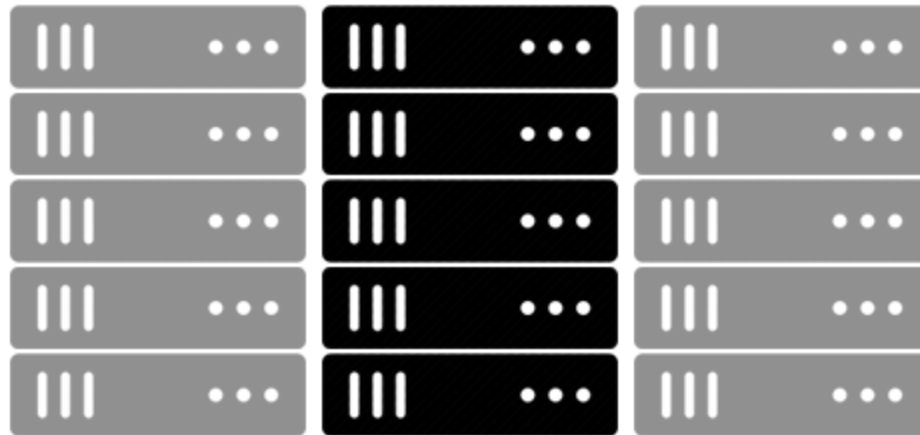
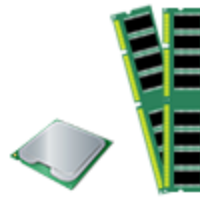
Horizontal vs Vertical Scaling

Me when I look at Scaling:





Vertical Scaling



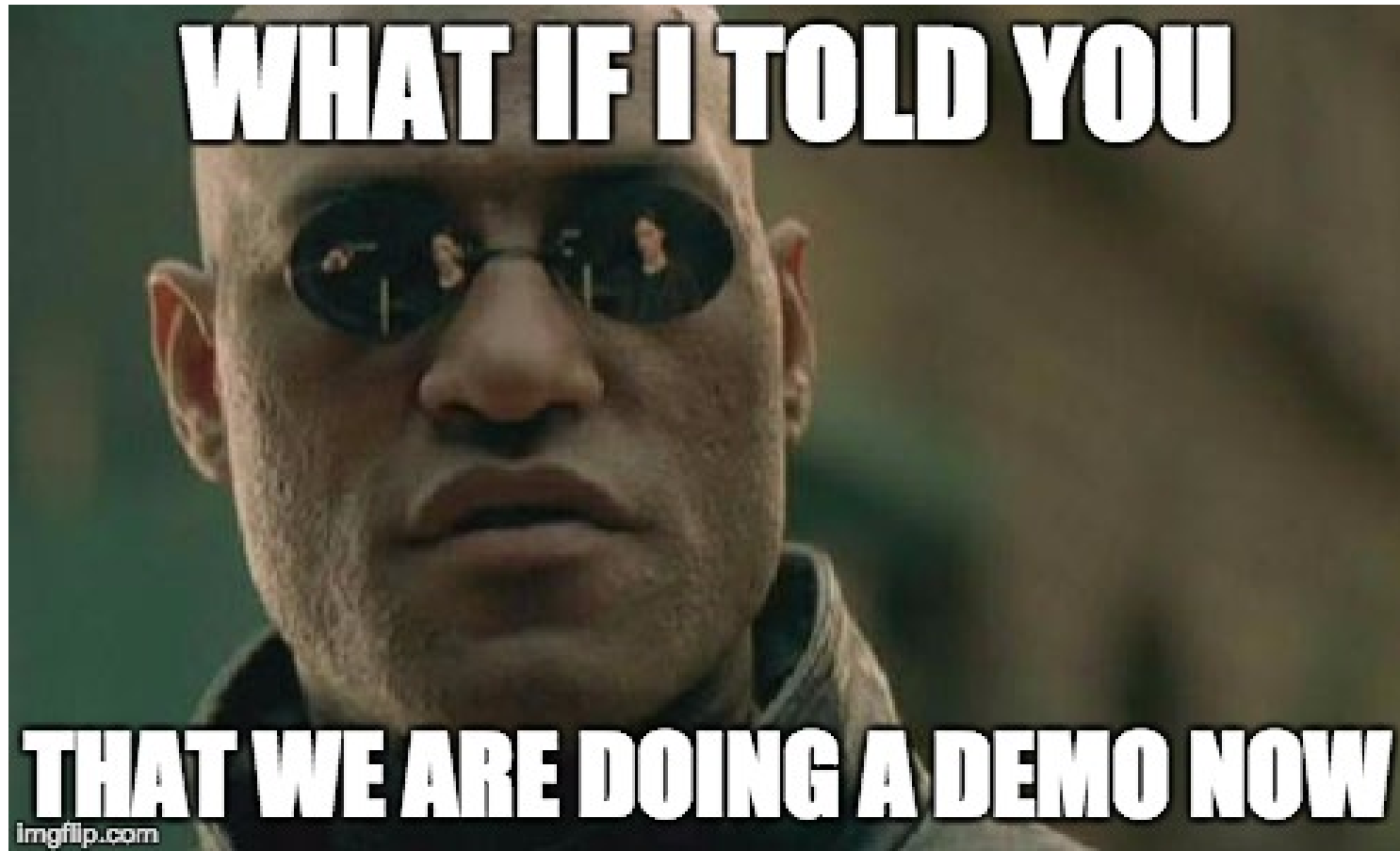
Horizontal Scaling



Benefits of Horizontal Scaling

- Dynamic scaling allows spinning up more instances and nodes faster, i.e. if you suddenly get a influx of traffic
- Vertical Scaling is limited to capacity of resources, simply adding more resources
- Just simplifying load testing isn't good enough, examples of this include Niantic (PokemonGo) and Australian Census 2016

Demo Section



Installing the SDK

1. Install the SDK over <https://cloud.google.com/sdk/downloads>
2. Authenticate Using `gcloud init` (login using your Monash Student Account)
3. You may need Java (JDK 1.8) and Maven (MVN) Installed if you are using the package provided.

If you are interested in developing on the framework provided I strongly suggest for you to read the docs.

Framework: <http://tinyurl.com/mplan-baseapi>

Deploying the App

- Reason's why we are using a Java Environment is that since its developed directly on the Standard Environment, it scales up way faster.

Other Tools Available on GCP that you play with

- Cloud ML (Google Cloud Machine Learning) which is built off TensorFlow
- Compute Engine - Google VMs
- Container Engine - built off Kubernetes and allows deployment of custom applications
- Cloud Storage - CDN provider of files (like *Amazon S3*)
- Network Balancer - for Load Balancing of traffic for your applications
- Cloud ML APIs such as Natural Language Processing, Data Loss Prevention, etc.

Questions