RICE®

Web Development

COMP 431 / COMP 531

Web Hosting

Scott E Pollack, PhD

March 22, 2016

Part II – Back End Development

- Homework Assignment 6 (Draft Back-End)
  - Due THURSDAY 3/24

Assignment 6: Draft Backend Web App

- Frontend Application on Heroku
  - Talks to dummy server
  - Landing, Main, Profile view
  - It **ALL** works
  - Upload images on posts, profile picture, etc
- End-to-End Tests
  - There's a list of scenarios to write
- Site Review
  - Your site will be reviewed by COMP531 students
- Backend Application on Heroku
  - Node server
  - `GET /status`
  - `PUT /status`
  - `GET /statues/:users`
  - `GET /posts/:id`
  - `POST /post`
  - Stubs for everything else
- Unit Tests
  - Update status
  - Add a post
- Graded by robot

http://localhost:80

- Our app is hosted "locally" and is technically accessible from the web
- Normally we want it somewhere else

Perhaps a web hosting service
They provide space on a server

Here would be good

Web Hosting

- Free
  - you get what you pay for
  - ~~Geocities~~, NeoCities, Weebly, WordPress, Blogger, …
- Shared($5/mo)
  - one box lots of clients
- Virtual Dedicated ($15/mo)
  - you get a VM on an otherwise shared box
  - Root, can be managed or unmanaged
- Dedicated ($100/mo)
  - Your own physical box
  - Root, can be managed or unmanaged
- Cloud
  - VMs all the way

Your choice of operating system
Windows or Linux

You get an account

Always FTP
sometimes SSH (Linux)
or RemoteDesktop (Windows)

Upload files
html, js, css, php, cgi

They run Apache

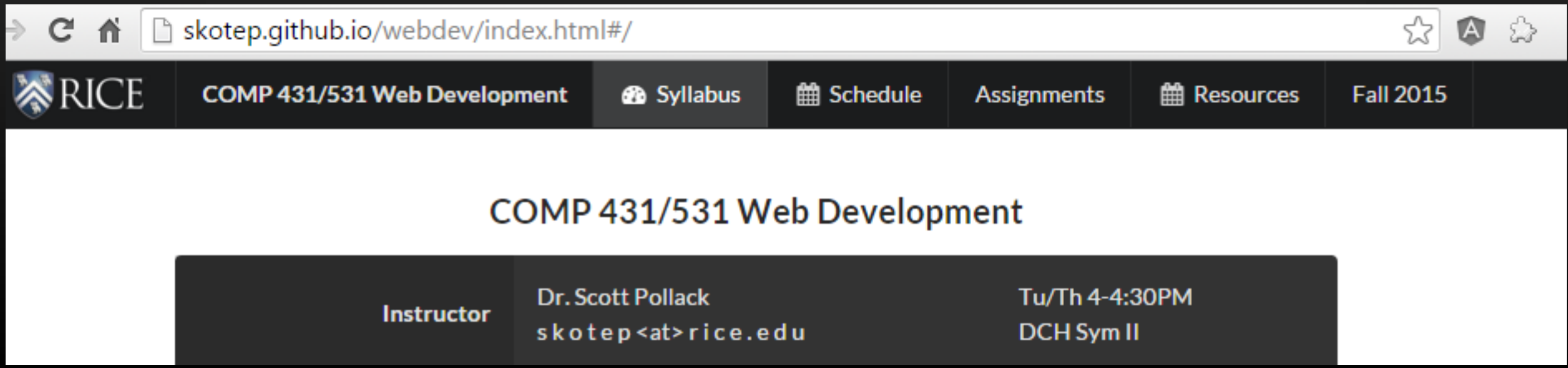
You might run Tomcat, etc
from a dedicated box

Static Hosting

GitHub Pages

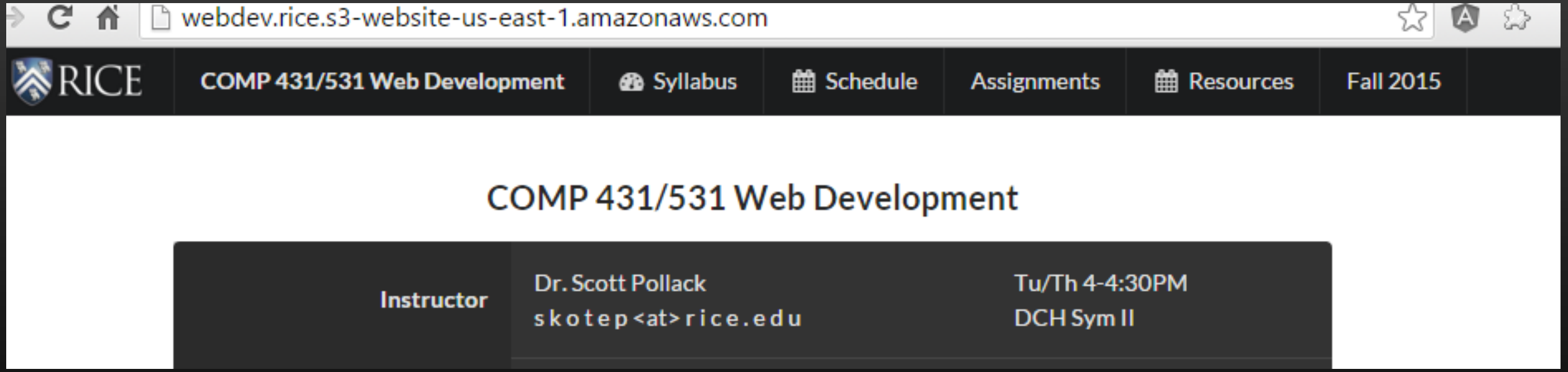
Websites for you and your projects.

Hosted directly from your GitHub repository. Just edit, push, and your changes are live.



AWS Documentation » Amazon Simple Storage Service (S3) » Developer Guide » Hosting a Static Website on Amazon S3

Hosting a Static Website on Amazon S3

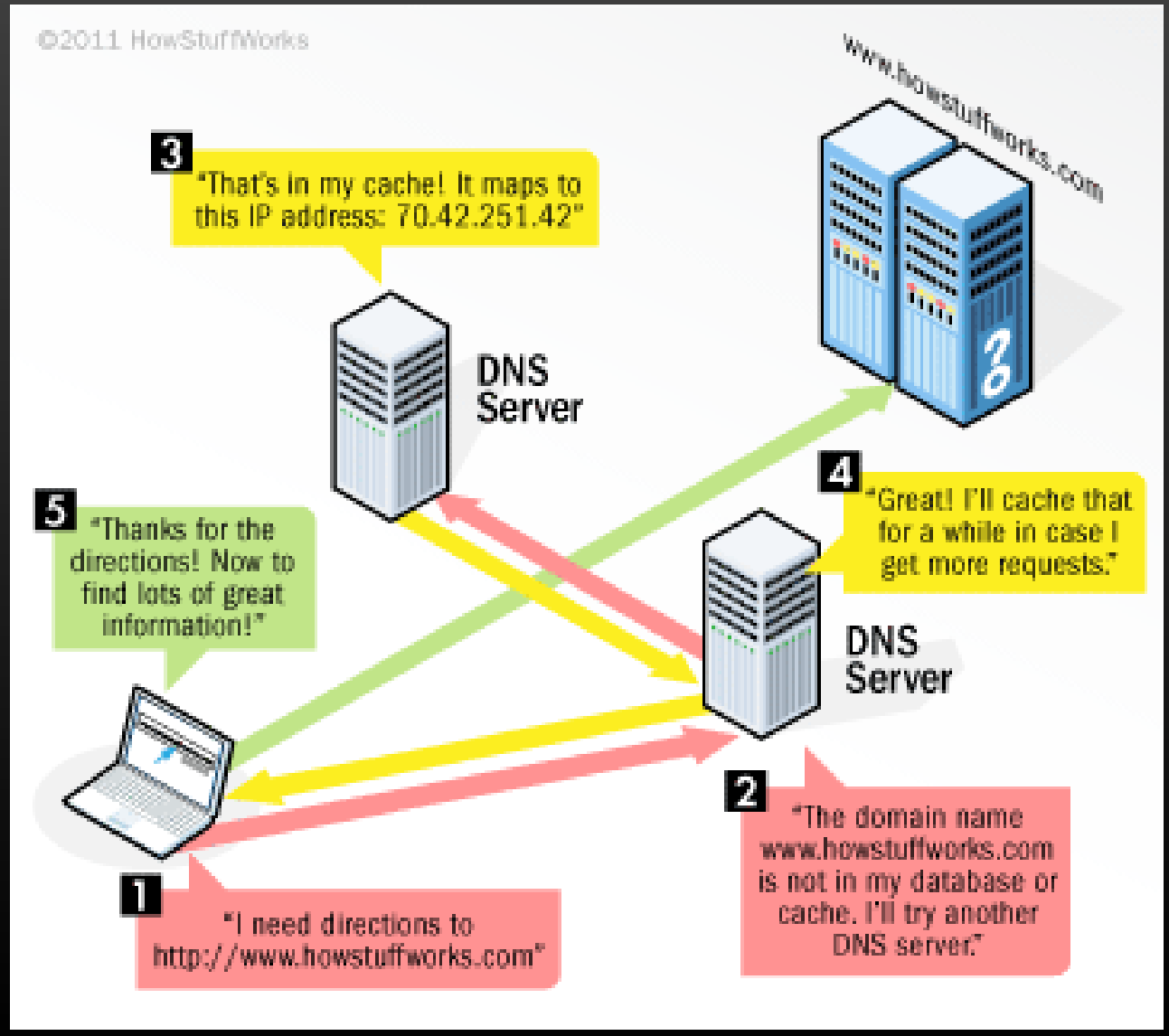


Web Hosting

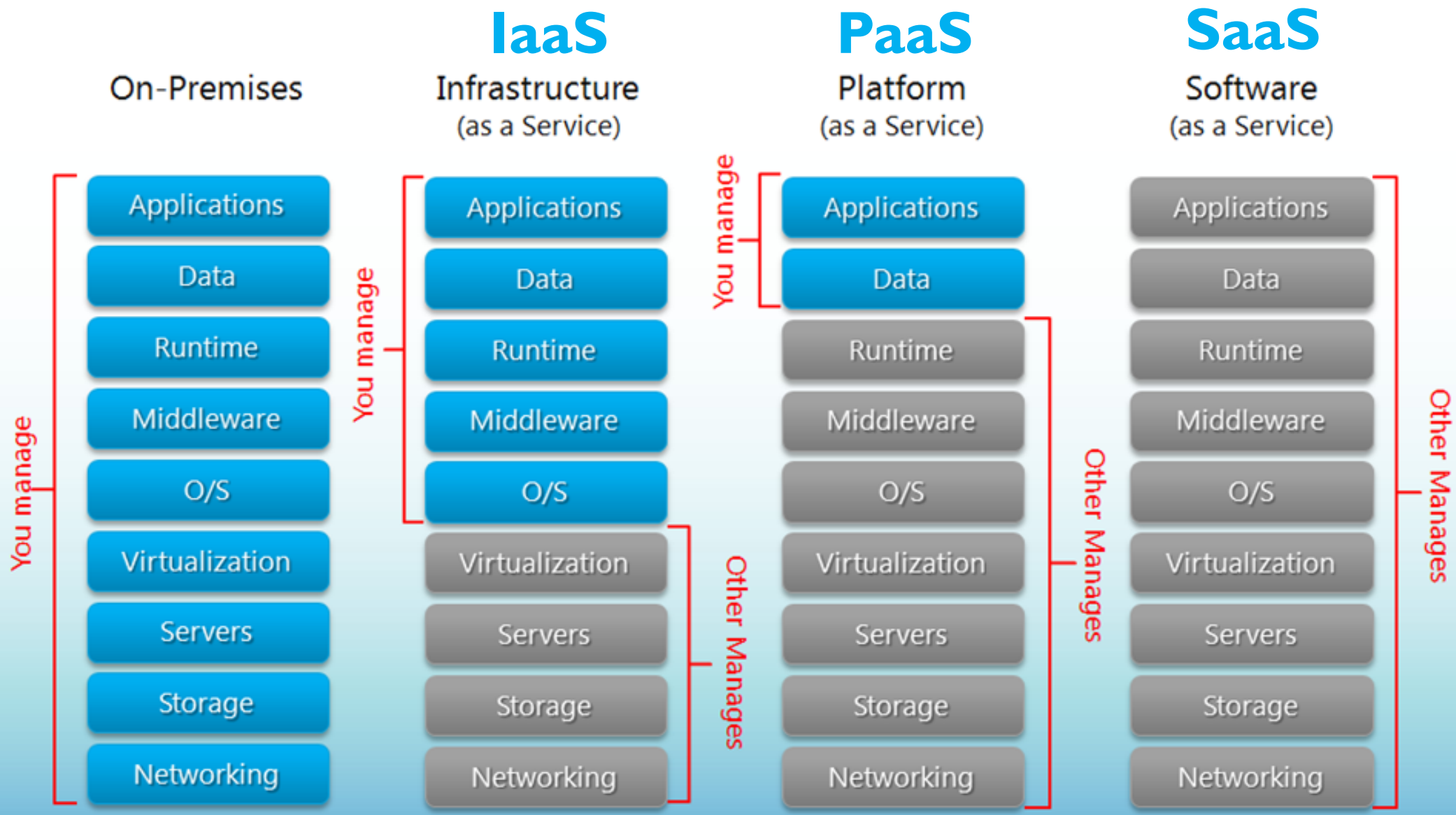
Control Center

Domain Name System

- Links "name" to "IP address"
- The web hosting service provides us an IP address
- We register a name for this IP address with their DNS
  - Typically hosters will do both

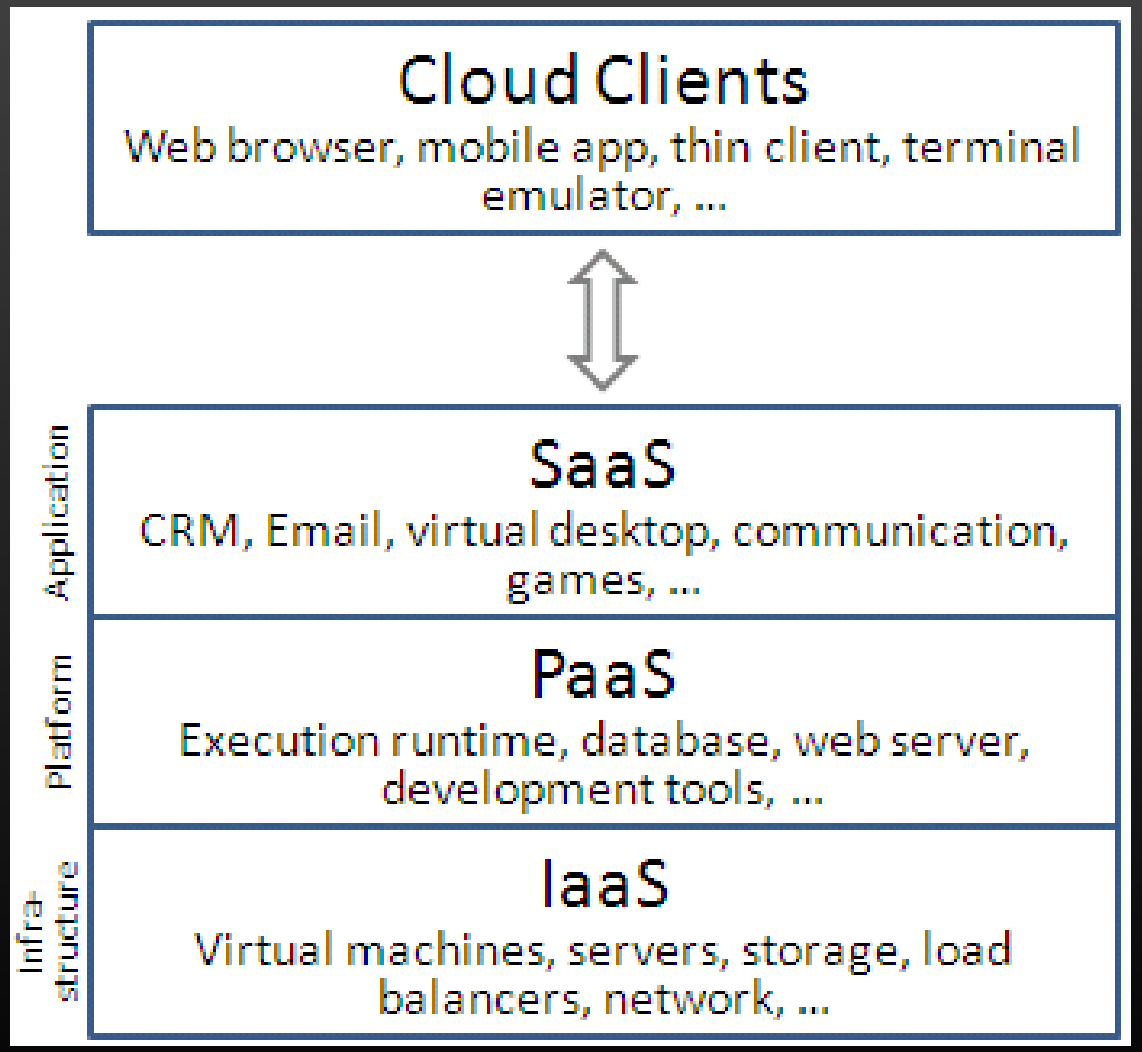


Separation of Responsibilities



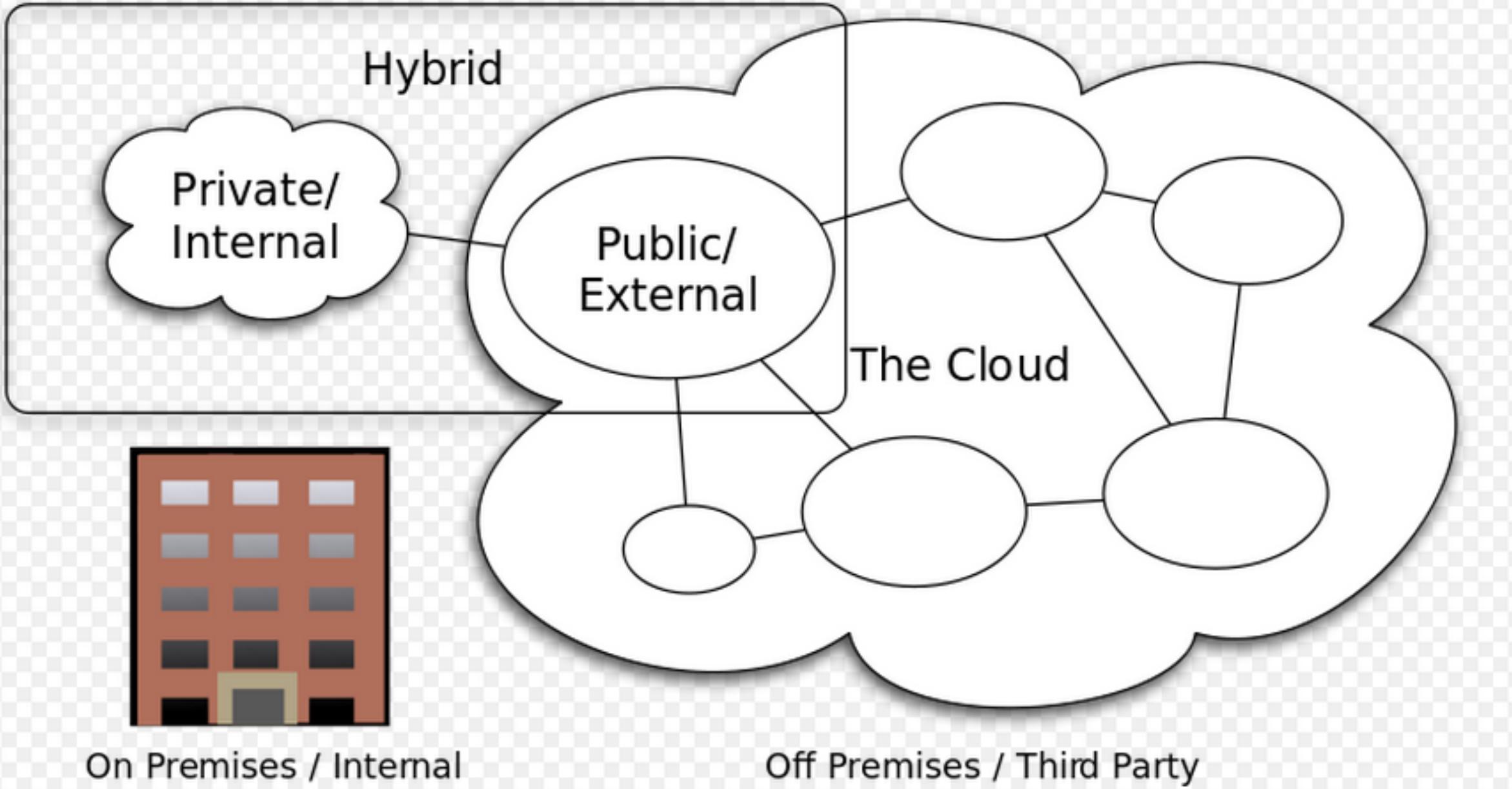
Cloud Hosting (PaaS)

- Web hosting services act as SaaS
  - they provide you a running server
- Cloud hosting is PaaS
  - They provide you a platform
- We therefore run our own server on their box (a VM)
  - Sometimes a canned server
- As a service, they provide
  - Resiliency, redundancy, scalability, uptime, etc…



PaaS Providers

OPENSHIFT



Cloud Computing Types

Amazon Web Services

Compute

EC2
Virtual Servers in the Cloud

EC2 Container Service
Run and Manage Docker Containers

Elastic Beanstalk
Run and Manage Web Apps

Lambda
Run Code in Response to Events

Storage & Content Delivery

S3
Scalable Storage in the Cloud

CloudFront
Global Content Delivery Network

Elastic File System PREVIEW
Fully Managed File System for EC2

Glacier

Developer Tools

CodeCommit
Store Code in Private Git Repositories

CodeDeploy
Automate Code Deployments

CodePipeline
Release Software using Continuous Delivery

Management Tools

CloudWatch
Monitor Resources and Applications

CloudFormation
Create and Manage Resources with Templates

CloudTrail
Track User Activity and API Usage

Config
Track Resource Inventory and Changes

OpsWorks
Automate Operations with

Internet of Things

AWS IoT BETA
Connect Devices to the cloud

Mobile Services

Mobile Hub BETA
Build, Test, and Monitor Mobile apps

Cognito
User Identity and App Data Synchronization

Device Farm
Test Android, Fire OS, and iOS apps on real devices in the Cloud

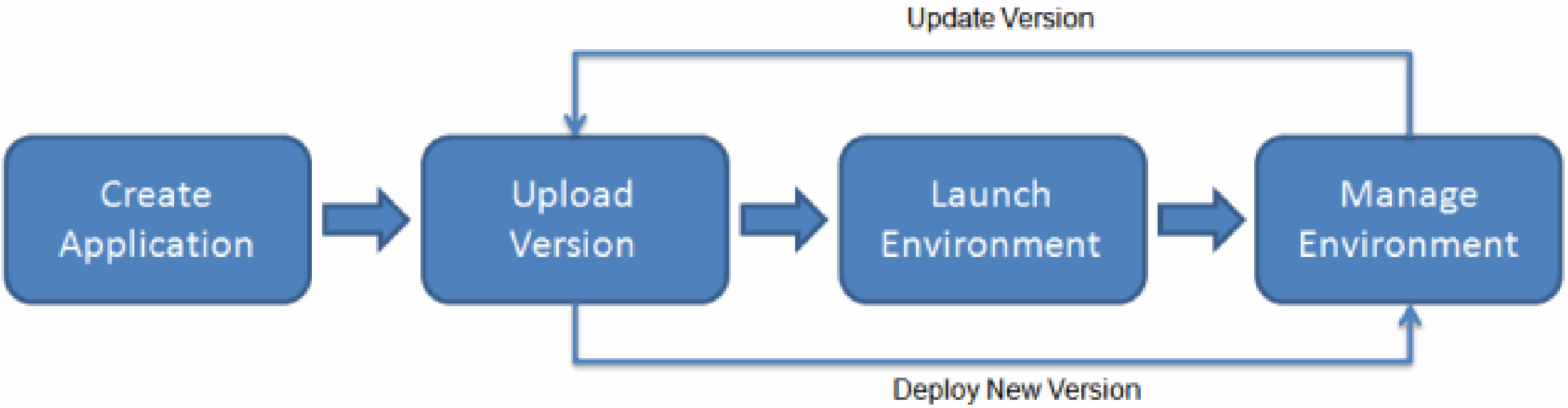
Mobile Analytics
Collect, View and Export App Analytics

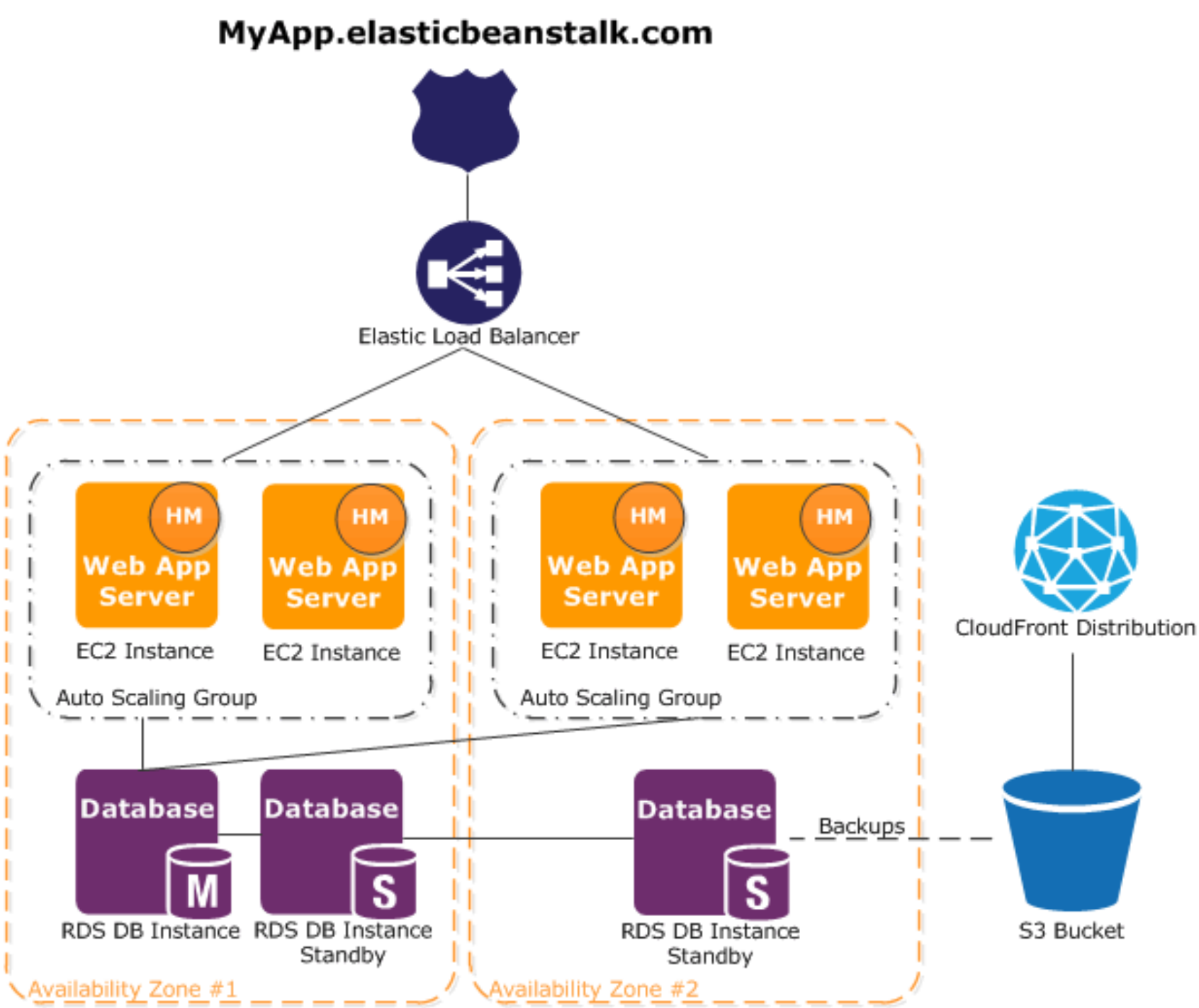
SNS
Push Notification Service

Application Services

API Gateway
Build, Deploy and Manage APIs

PaaS Usage in Practice




MyApp.elasticbeanstalk.com
Elastic Load Balancer
HM
Web App Server
EC2 Instance
HM
Web App Server
EC2 Instance
Auto Scaling Group
HM
Web App Server
EC2 Instance
HM
Web App Server
EC2 Instance
Auto Scaling Group
CloudFront Distribution
Database
M
RDS DB Instance
Database
S
RDS DB Instance Standby
Database
S
RDS DB Instance Standby
Backups
S3 Bucket
Availability Zone #1
Availability Zone #2

Amazon Web Services

Free Tier*

As part of AWS's Free Tier, new AWS customers can get started with Amazon EC2 for free. Upon sign-up, new AWS customers receive the following EC2 services each month for one year:

- 750 hours of EC2 running Linux, RHEL, or SLES t2.micro instance usage
- 750 hours of EC2 running Microsoft Windows Server t2.micro instance usage
- 750 hours of Elastic Load Balancing plus 15 GB data processing
- 30 GB of Amazon Elastic Block Storage in any combination of General Purpose (SSD) or Magnetic, plus 2 million I/Os (with Magnetic) and 1 GB of snapshot storage
- 15 GB of bandwidth out aggregated across all AWS services
- 1 GB of Regional Data Transfer

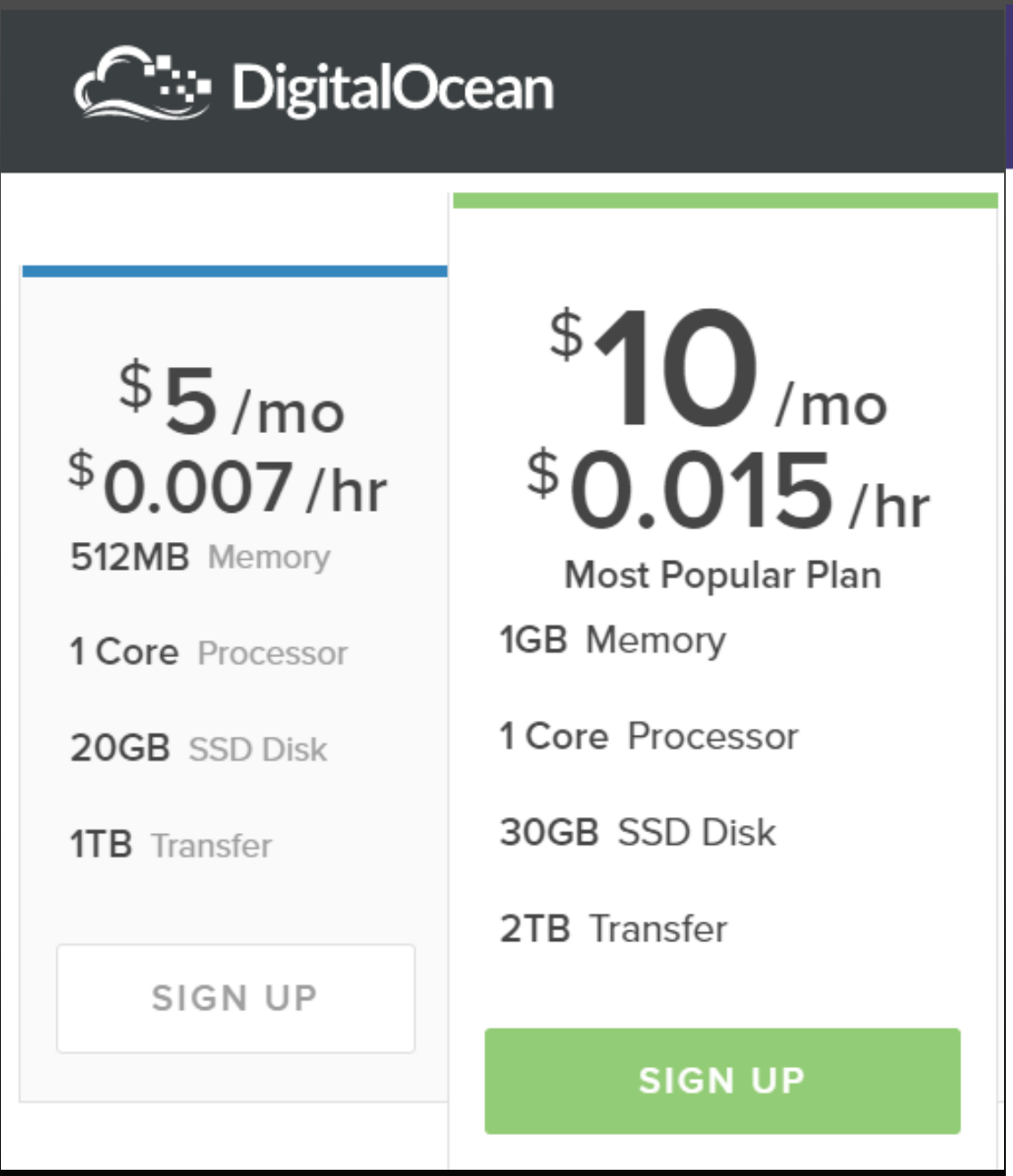
Amazon S3 Pricing

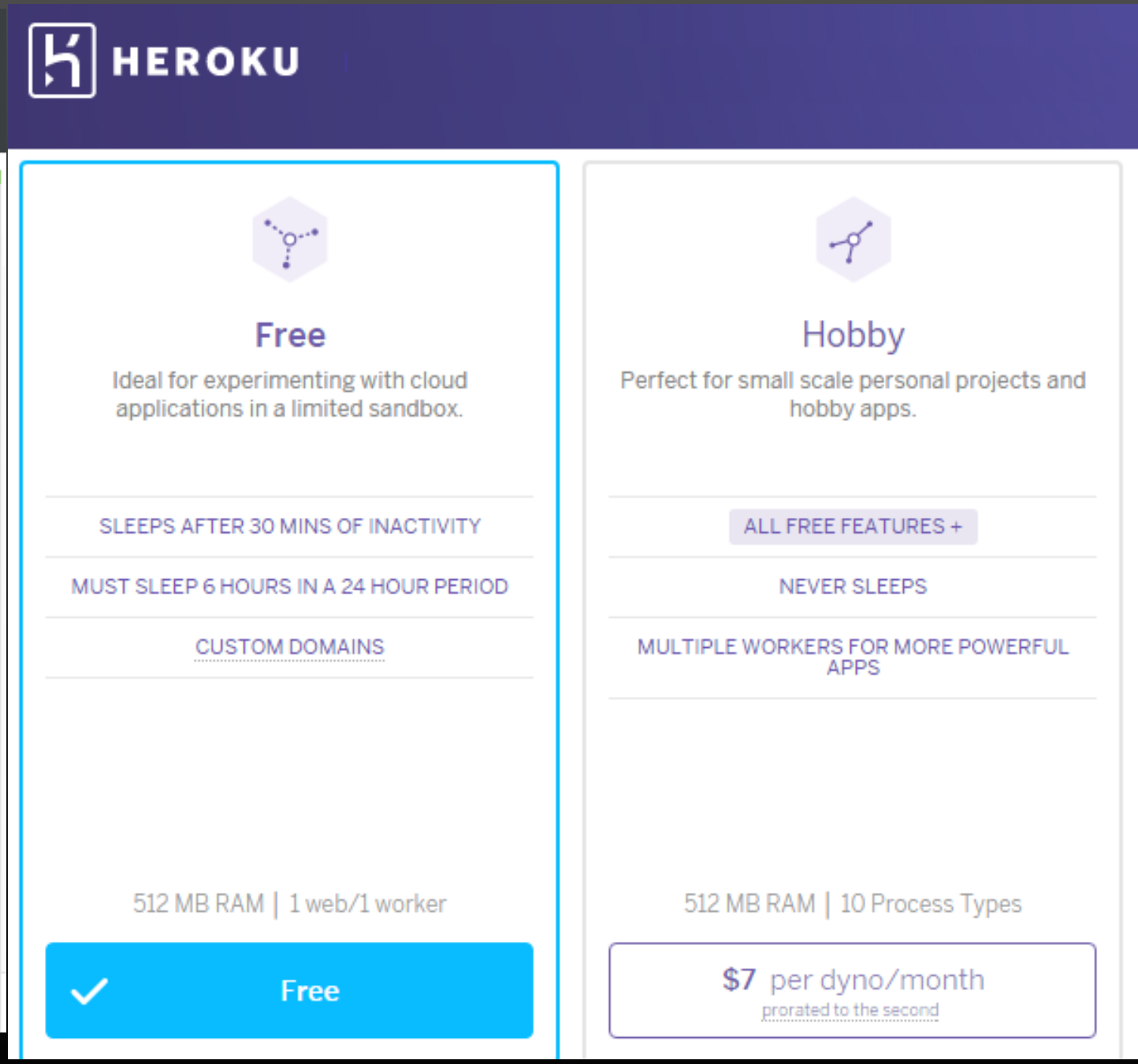
	Standard Storage
First 1 TB / month	$0.0300 per GB

Amazon EC2 Pricing

On-Demand Instance Prices

	vCPU	ECU	Memory (GiB)	Linux/UNIX Usage
t2.micro	1	Variable	1	$0.013 per Hour
t2.small	1	Variable	2	$0.026 per Hour
t2.medium	2	Variable	4	$0.052 per Hour
t2.large	2	Variable	8	$0.104 per Hour
m4.large	2	6.5	8	$0.126 per Hour
m4.xlarge	4	13	16	$0.252 per Hour
m4.2xlarge	8	26	32	$0.504 per Hour


DigitalOcean
$5 /mo
$0.007 /hr
512MB Memory
1 Core Processor
20GB SSD Disk
1TB Transfer
SIGN UP
$10 /mo
$0.015 /hr
Most Popular Plan
1GB Memory
1 Core Processor
30GB SSD Disk
2TB Transfer
SIGN UP


HEROKU
Free
Ideal for experimenting with cloud applications in a limited sandbox.
SLEEPS AFTER 30 MINS OF INACTIVITY
MUST SLEEP 6 HOURS IN A 24 HOUR PERIOD
CUSTOM DOMAINS
512 MB RAM | 1 web/1 worker
Free
Hobby
Perfect for small scale personal projects and hobby apps.
ALL FREE FEATURES +
NEVER SLEEPS
MULTIPLE WORKERS FOR MORE POWERFUL APPS
512 MB RAM | 10 Process Types
$7 per dyno/month
prorated to the second

In-Class Exercise: Host Your Backend on Heroku

Go to your backend app directory

here you should be able to run your app by typing: node index.js -OR- npm start

Now, do this:

```
> git init
> heroku create ${optional app name}
> echo web: node index.js > Procfile
> echo node_modules >> .gitignore
> echo npm-debug.log >> .gitignore
> git add . && git commit
> git push heroku master
> heroku ps:scale web=1
```

README.json

```
{
  "netid":"sep1",
  "backend": "https://..."
}
```

Navigate to your new Heroku hosted app!

Turnin README.json

COMP431-S16:inclass-19

Download and run

https://www.clear.rice.edu/comp431/sample/RiceBookServer/test-backend.py

```
> python test-backend.py README.json
```