

---

# Web Development

## With Node/React/Docker/AWS

### an introduction

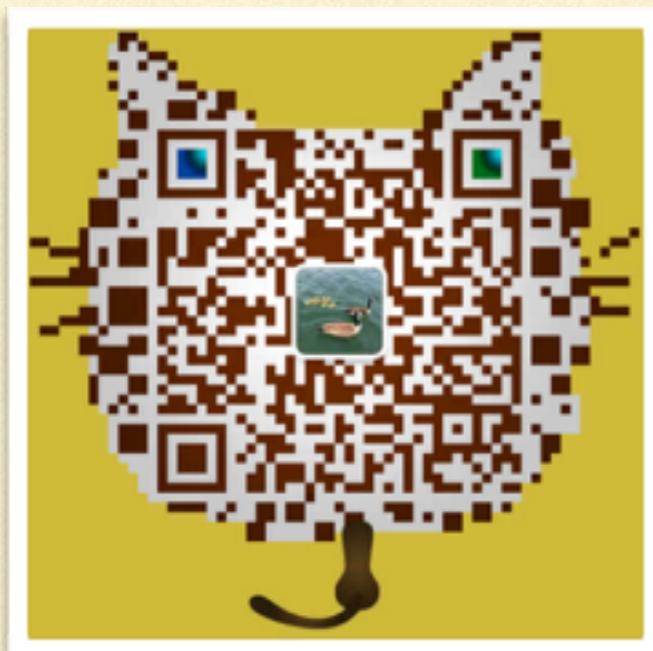
---

Created by: Yue Liu

Styled by: Jason Liu

8/9/2015

---



# Portfolio

All your data. All in one place.

TouchdownSpace now offers instant real-time bookings in most major cities and suburbs across the United States. Book with confidence 24/7!

Why book with us?  
We are the one true Underdog!

LinkedIn - <http://linkedin.com/in/leapon>

Github - <http://github.com/leapon>

---

# INDEX

NODE

BACKEND (70%)

REACT/BOOTSTRAP

FRONTEND (10%)

DOCKER

DEPLOYMENT (10%)

AWS

HOSTING (10%)

---

# WHAT IS NODE



NodeJS is server side Javascript

Runs on Google V8 engine

Created by Ryan Dahl in 2009



The screenshot shows the official Node.js website at <https://nodejs.org>. The page has a dark background with a navigation bar at the top featuring links for HOME, DOWNLOADS, DOCS, CONTRIBUTE, FOUNDATION, COMMUNITY, ABOUT, JOBS, and BLOG. Below the navigation, there is a brief introduction to what Node.js is, followed by a "Current Version: v0.12.7" message. A prominent green "INSTALL" button is centered, with "DOWNLOADS" and "API DOCS" buttons nearby. At the bottom, there is a "NODE.JS ON THE ROAD" section with a small icon and a link to "LEARN MORE".

[http://nodejs.org](https://nodejs.org)

# WHY USE NODEJS

## Success Stories.....



Rails to Node

- « Servers were cut to 3 from 30 »
- « Running up to 20x faster in some scenarios »
- « Frontend and backend mobile teams could be combined [...] »



Java to Node

- « Built almost twice as fast with fewer people »
- « Double the requests per second »
- « 35% decrease in the average response time »



---

## STRONG POINTS

- Stream-based real-time high-concurrent services
- One language for front and back end development

## WEAK POINTS

- CPU-intensive calculations
  - Not release version 1.0 yet
-

# WEB SERVER EXAMPLE

```
var http = require('http');

http.createServer(function (req, res) {
  res.writeHead(200, {'Content-Type': 'text/plain'});
  res.end('Hello World\n');
}).listen(1337, "127.0.0.1");

console.log('Server running at http://127.0.0.1:1337/');
```

```
→ test git:(master) ✘ node webserver.js
Server running at http://127.0.0.1:1337/
```

← → C 127.0.0.1:1337

Hello World

# CHARACTERISTICS

---

- RUNS IN SINGLE THREAD
  - EVENT-DRIVEN
  - CALLBACK HANDLING
  - HAS NON-BLOCKING IO
-

# I/O Costs

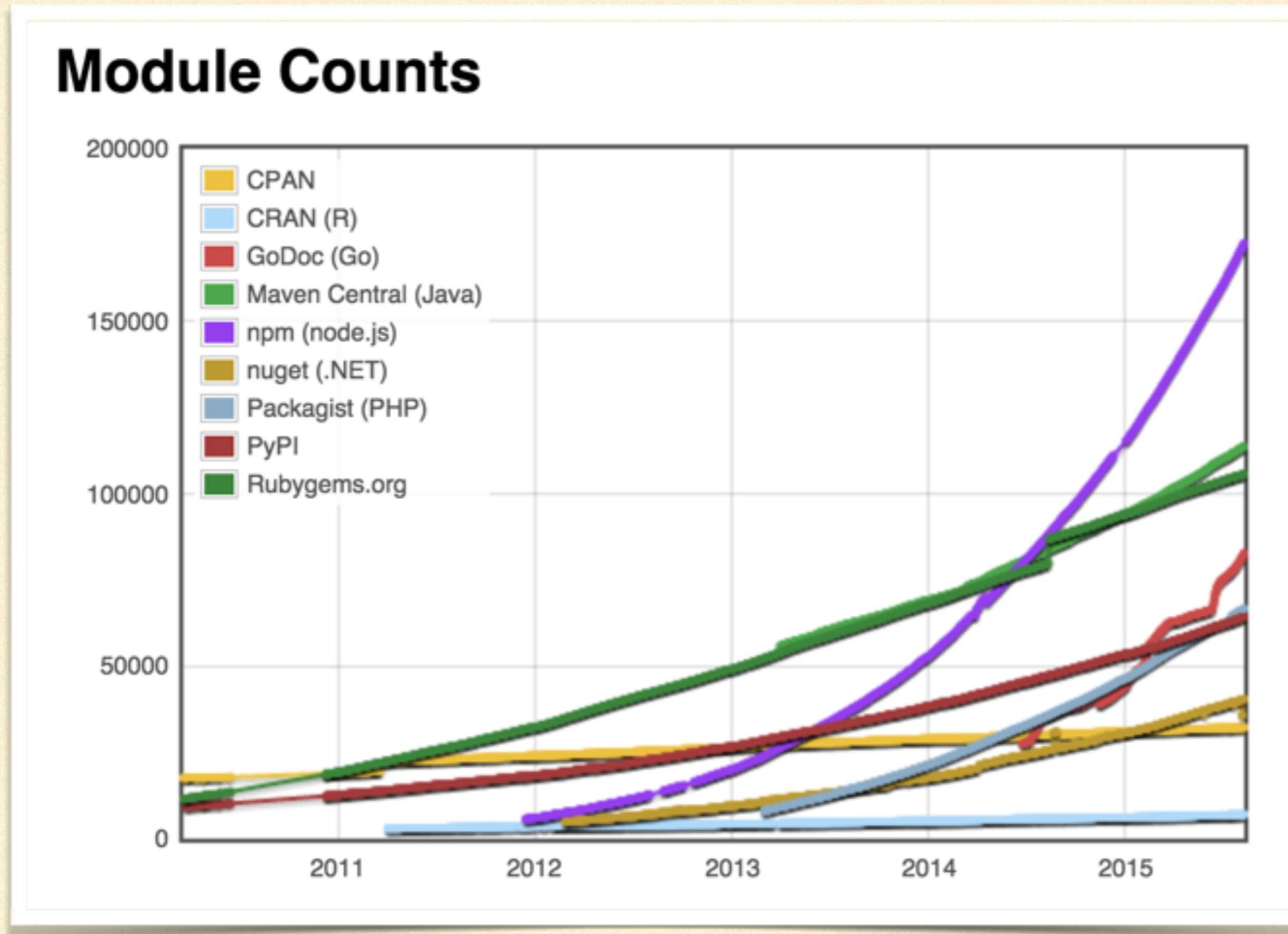
- L1: 3 cycles
- L2: 14 cycles
- RAM: 250 cycles
- DISK: 41,000,000 cycles
- NETWORK: 240,000,000 cycles

# Two approaches to I/O

```
// blocking I/O + threads
var urls = db.query("select * from urls"); // wait
urls.each(function (url) {
  var page = http.get(url); // wait
  save(page); // wait
});

// non-blocking I/O + event loop
db.query("select * from urls", function (urls) {
  urls.each(function (url) {
    http.get(url, function (page) {
      save(page);
    });
  });
});
```

# GROWTH OF NPM MODULES



source: <http://www.modulecounts.com/>

<http://npmjs.com> 71,780 packages on August 2015



find packages



sign up or log in



## most starred packages

### express

Fast, unopinionated, minimalist web framework

4.13.3 published 6 days ago by dougwilson

### request

Simplified HTTP request client.

2.60.0 published 3 weeks ago by simov

### browserify

browser-side require() the node way

11.0.1 published a week ago by substack

### commander

the complete solution for node.js command-line pro...

2.8.1 published 3 months ago by zhiyeliee

### underscore

JavaScript's functional programming helper library.

1.8.3 published 4 months ago by jashkenas

### forever

A simple CLI tool for ensuring that a given node scrip...

0.15.1 published a week ago by indexzero

### async

Higher-order functions and common patterns for as...

1.4.1 published 23 hours ago by aearly

### lodash

The modern build of lodash modular utilities.

3.10.1 published 5 days ago by jdalton

### socket.io

node.js realtime framework server

1.3.6 published 4 weeks ago by rauchg

### mongoose

Mongoose MongoDB ODM

4.1.1 published 5 days ago by vkarpov15

### bower

The browser package manager

1.4.1 published 4 months ago by sheerun

### chalk

Terminal string styling done right. Much color.

1.1.0 published a month ago by jbnicolai

### gulp

The streaming build system

3.9.0 published 2 months ago by phated

### grunt

The JavaScript Task Runner

0.4.5 published a year ago by cowboy

### pm2

Production process manager for Node.JS applicatio...

0.14.6 published 3 days ago by jshkurti

### mocha

simple, flexible, fun test framework

2.2.5 published 3 months ago by travisjeffery

### moment

Parse, validate, manipulate, and display dates

2.10.6 published 2 weeks ago by ichernev

### hapi

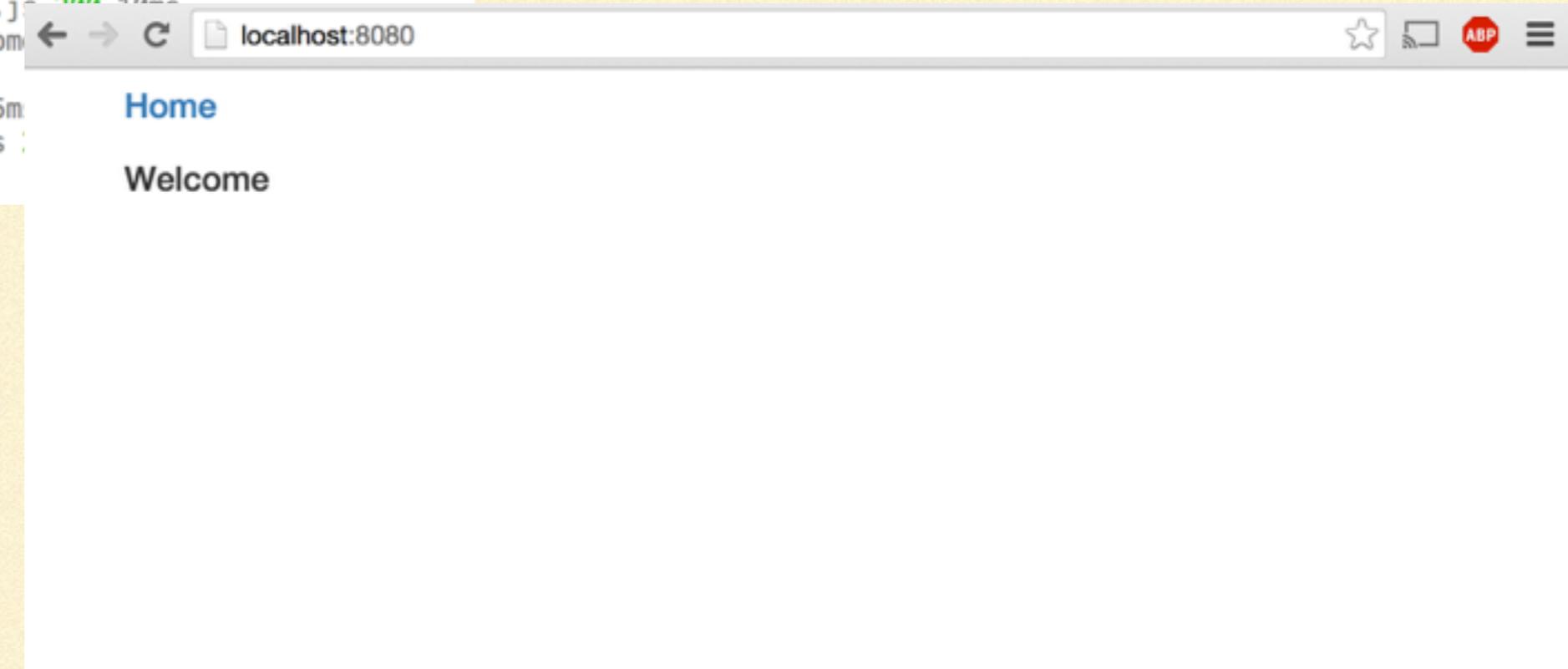
HTTP Server framework

8.8.1 published a week ago by hueniverse

# DEMO

```
→ leapbase git:(master) npm start  
  
> leapbase@0.1.0 start /Users/yliu0/git/leapbase  
> node ./site/bin/www  
  
use default setting  
Web server is listening on port 8080 ...
```

```
GET / 200 37ms - 978b  
GET /library/bootstrap/js/bootstrap.min.js 200 10ms  
GET /library/font-awesome/css/font-awesome.css 200 10ms  
GET /style/style.css 200 9ms - 88b  
GET /library/jquery/jquery.min.js 200 15ms  
GET /library/bootstrap/css/bootstrap.css 200 10ms
```



<https://github.com/leapon/leapbase>

---

# GET STARTED WITH NODE

<http://nodejs.org>

<http://expressjs.com>

<https://www.codecademy.com>

<http://nodeschool.io>

---

# WHAT IS REACTJS



- CREATED BY FACEBOOK
- IT IS THE VIEW IN THE MVC
- SIMPLE AND DECLARATIVE

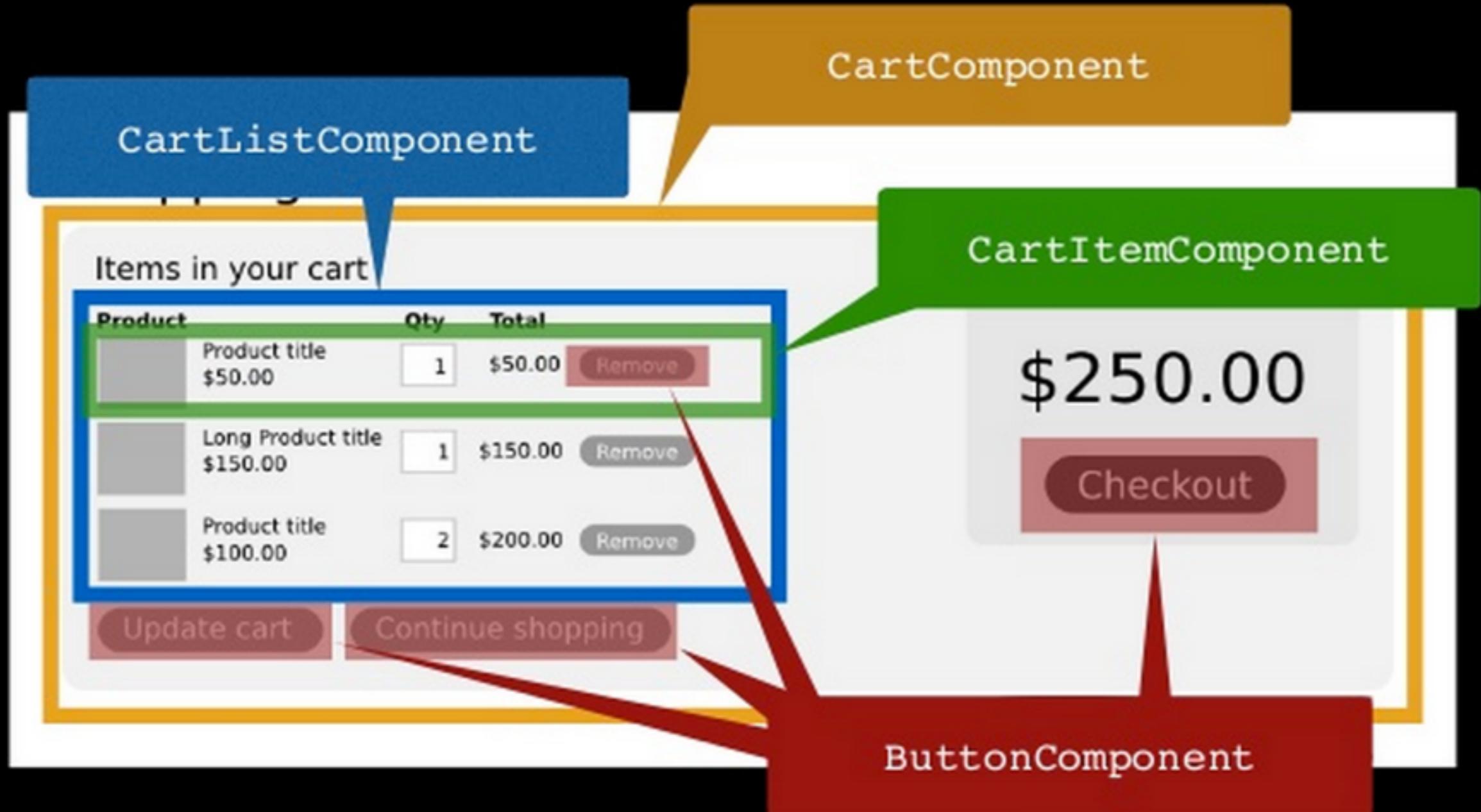
# WHY USE REACT

---

BREAK UI INTO COMPONENTS

COMPONENTS ARE

- COMPOSABLE
  - REUSABLE
  - MAINTAINABLE
  - TESTABLE
-



---

# FEATURES

- VIRTUAL DOM
  - JSX - COMPILES TO JAVASCRIPT
  - SERVER SIDE RENDERING
  - ONE-WAY DATA FLOW
-

# EXAMPLE JSX CODE

```
// tutorial1.js
var CommentBox = React.createClass({
  render: function() {
    return (
      <div className="commentBox">
        Hello, world! I am a CommentBox.
      </div>
    );
  }
});
React.render(
  <CommentBox />,
  document.getElementById('content')
);
```

# DEMO

A screenshot of a web browser window displaying a calendar interface. The address bar shows "localhost:8080". The header includes a logo, navigation icons, and links for "Project DNRDW", "Home", "Reference", and "About". Below the header, the title "Demo App" is displayed. A large heading "August 2015" is centered above a 7x7 grid representing the month's days. The grid has columns labeled "Sun", "Mon", "Tue", "Wed", "Thu", "Fri", and "Sat". The days are numbered sequentially from 26 to 31, with the 8th of August highlighted in a light gray color.

Sun	Mon	Tue	Wed	Thu	Fri	Sat
26	27	28	29	30	31	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31	1	2	3	4	5

A screenshot of a web browser window displaying a calendar interface. The address bar shows "localhost:8080". The header includes a logo, navigation icons, and a link for "Project DNRDW". Below the header, the title "Demo App" is displayed. A large heading "August 2015" is centered above a 7x7 grid representing the month's days. The grid has columns labeled "Sun", "Mon", "Tue", "Wed", "Thu", "Fri", and "Sat". The days are numbered sequentially from 26 to 31, with the 8th of August highlighted in a light gray color.

Sun	Mon	Tue	Wed	Thu	Fri	Sat
26	27	28	29	30	31	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31	1	2	3	4	5

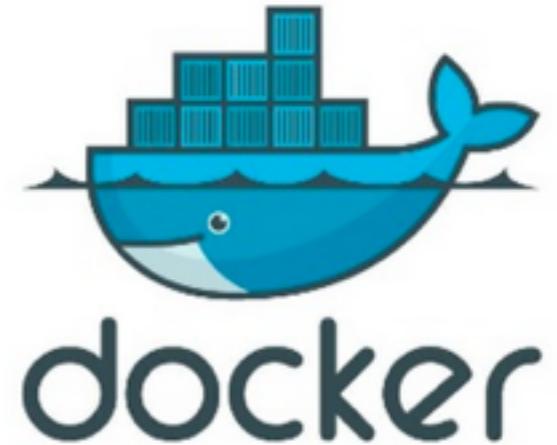
<https://github.com/leapon/docker-node-react-demo-webapp>

---

# REFERENCE

- <http://facebook.github.io/react/>
  - <https://github.com/facebook/react>
  - <https://facebook.github.io/flux/docs/overview.html>
-

# WHAT IS DOCKER



-OPEN SOURCE

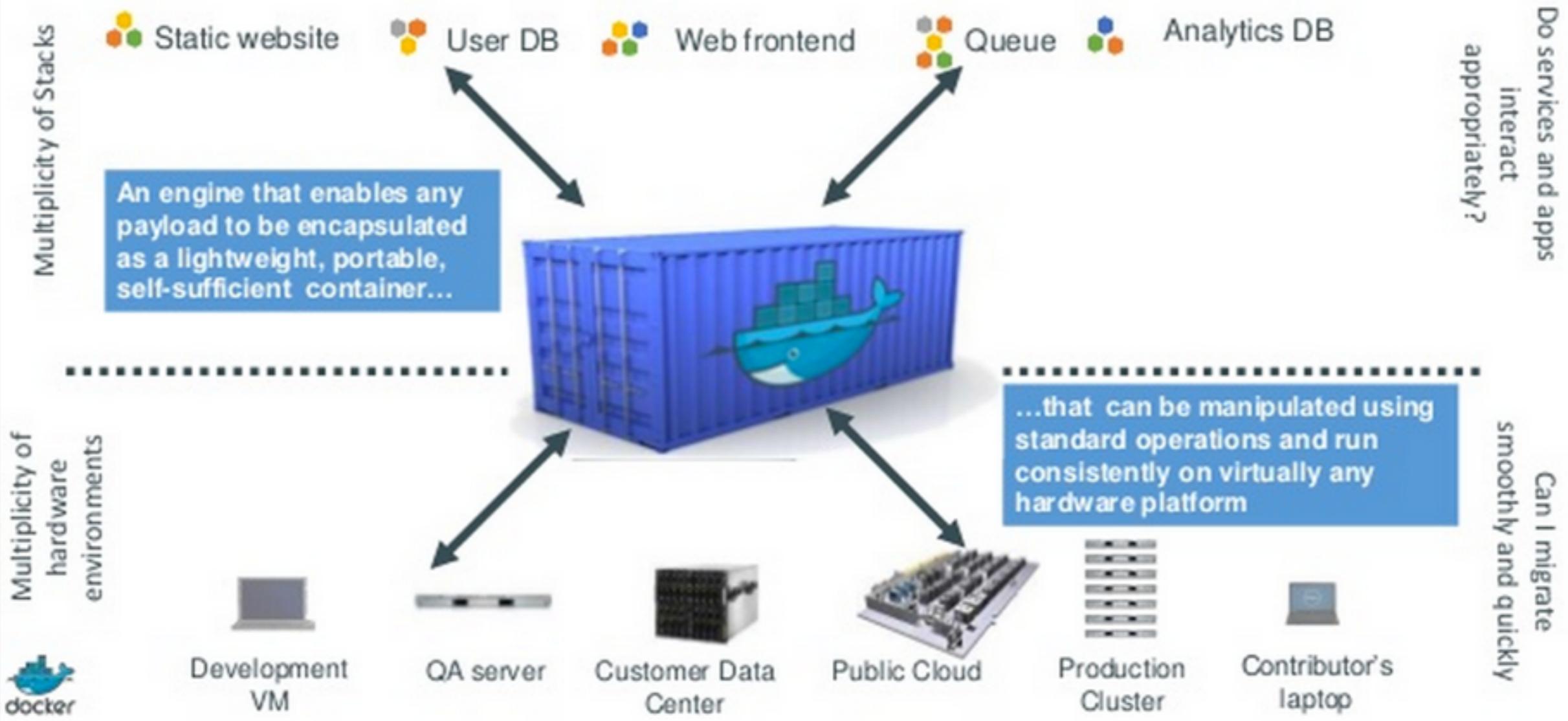
-WRITTEN IN GO

---

# WHY USE DOCKER

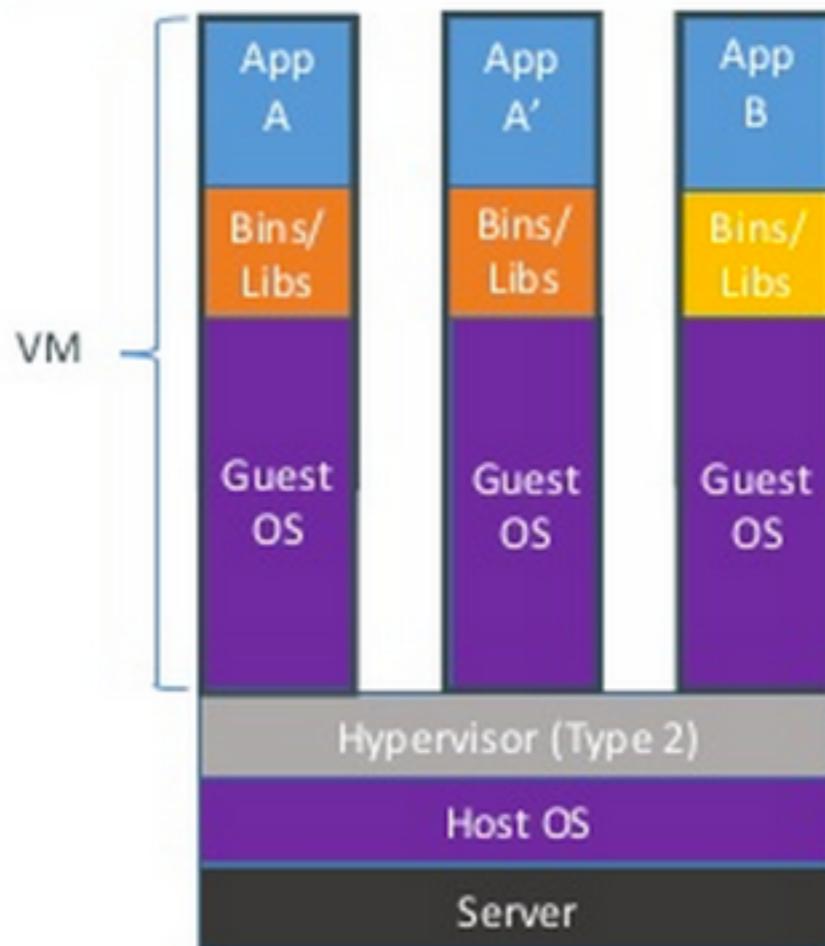
- BUILD ONCE, RUN EVERYWHERE
  - DEPLOY RELIABLY & CONSISTENTLY
  - TAKES ONLY SECONDS TO START
-

# Docker is a shipping container system for code



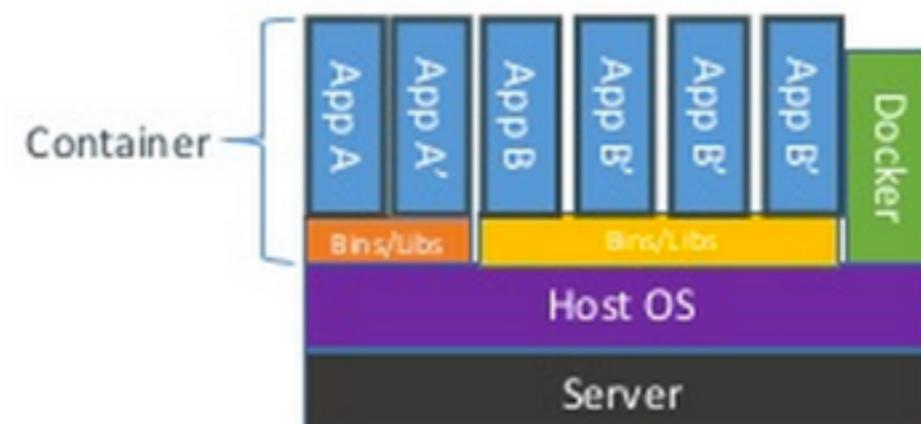
# DOCKER VS VM

## Containers vs. VMs



Containers are isolated,  
but share OS and, where  
appropriate, bins/libraries

...result is significantly faster deployment,  
much less overhead, easier migration,  
faster restart



# DOCKERFILE

---

```
# Dockerfile to create a dnrdw image
```

```
FROM google/nodejs
```

```
MAINTAINER leapon
```

```
# Add files to the image
```

```
RUN mkdir -p /opt/nodejs
```

```
ADD . /opt/nodejs
```

```
WORKDIR /opt/nodejs
```

```
# Install the dependencies modules
```

```
RUN npm install
```

```
# Expose environment variables
```

```
ENV LEAPBASE_HTTP_PORT=8000
```

```
# Expose the container port
```

```
EXPOSE 8000
```

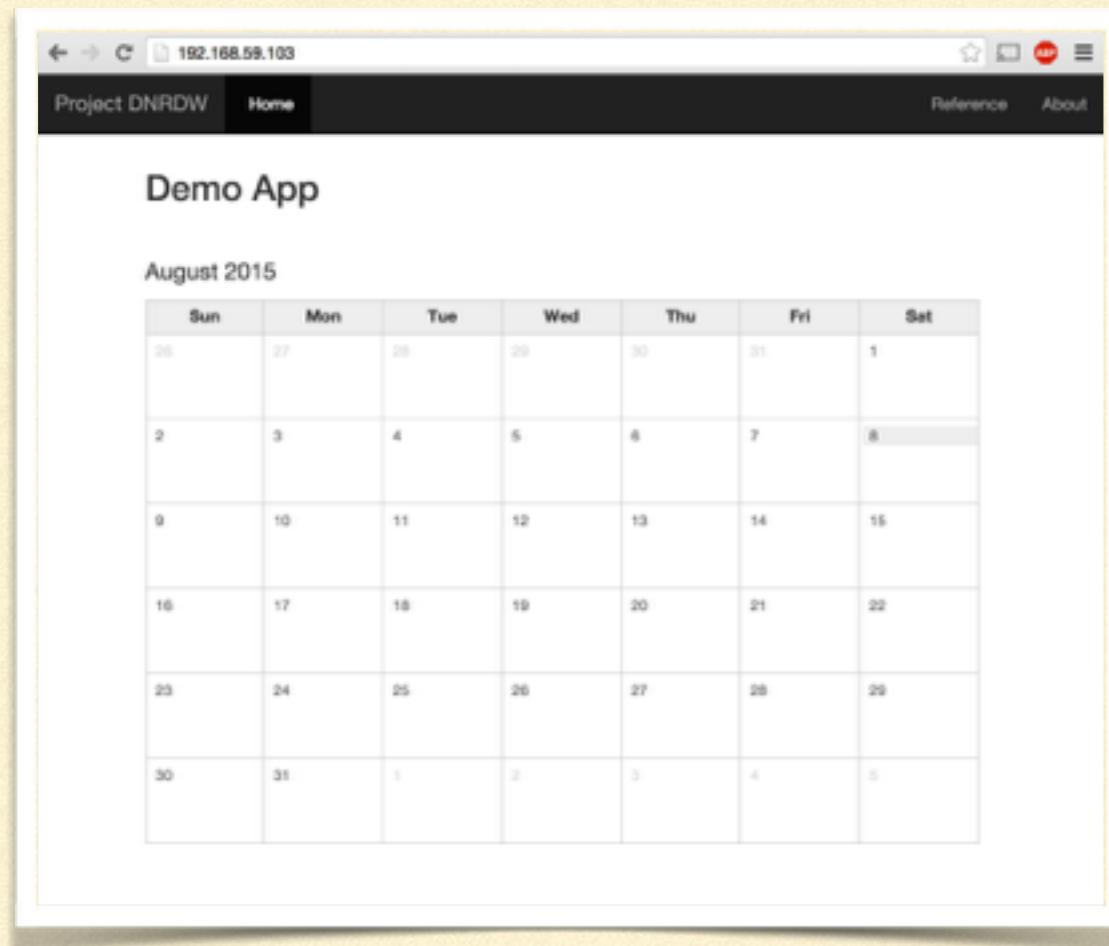
```
ENTRYPOINT ["npm", "start"]
```

---

# RUN DOCKER

```
# Create dnrdw docker image  
docker build --tag=leapon/dnrdw -f build/Dockerfile .
```

```
# Run docker container  
docker run --name=dnrdw -p 80:8000 -t -d leapon/dnrdw
```



---

# USE DOCKER HUB

`docker login`

`docker push leapon/dnrdw`

## LINKS

<https://www.docker.com>

<https://hub.docker.com>

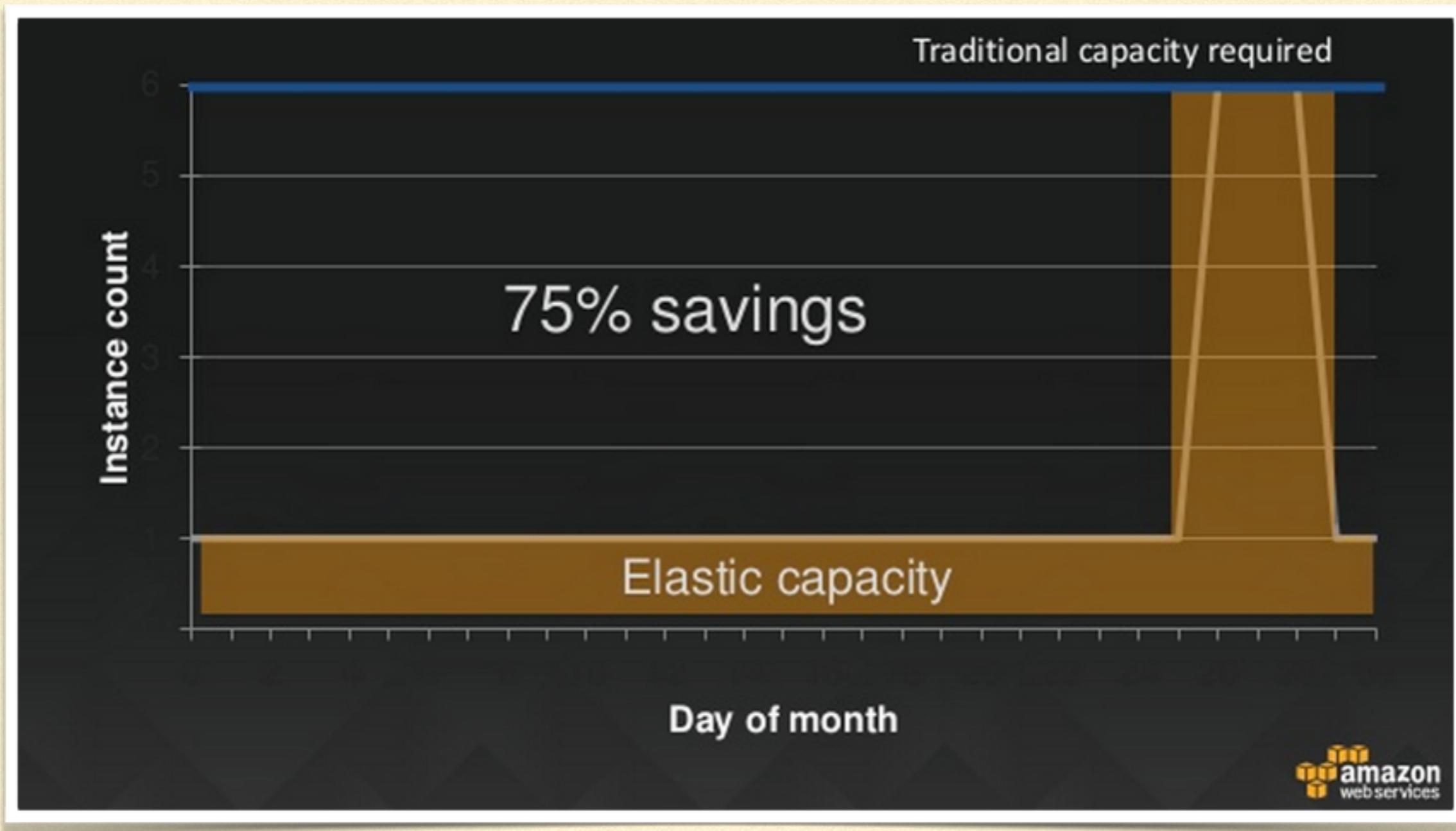
---

# AMAZON WEB SERVICES



- Pay as you need
- Scale up and down

# WHY USE AWS



# GLOBAL REACH

## AWS Regions



US East (Northern Virginia)  
US West (Northern California)  
Europe (Dublin)  
Asia Pacific (Singapore)  
Asia Pacific (Tokyo)



## AWS CloudFront Locations



Ashburn, VA / Dallas, TX / Jacksonville, FL / Los Angeles, CA / Miami, FL / Newark, NJ / New York, NY / Palo Alto, CA / Seattle, WA / St. Louis, MO / Amsterdam / Dublin / Frankfurt / London / Hong Kong / Tokyo / Singapore

---

# INFRASTRUCTURE

- COMPUTER: EC2
- STORAGE: S3 (OBJECT), EBS (BLOCK)
- NETWORK: VPC, ELB, ROUTE 53 (DNS)
- DATABASE: RDS (MYSQL, ORACLE, SQL SERVER)

# PLATFORM

- CLOUDFRONT, SQS, SNS, CLOUDFORMATION, ...
-

← → C https://console.aws.amazon.com/console/home?region=us-east-1

AWS Services Edit Leapon Liu N. Virginia Support

## Amazon Web Services

Compute

- EC2 Virtual Servers in the Cloud
- Lambda Run Code in Response to Events
- EC2 Container Service Run and Manage Docker Containers

Storage & Content Delivery

- S3 Scalable Storage in the Cloud
- Elastic File System PREVIEW Fully Managed File System for EC2
- Storage Gateway Integrates On-Premises IT Environments with Cloud Storage
- Glacier Archive Storage in the Cloud
- CloudFront Global Content Delivery Network

Database

- RDS MySQL, Postgres, Oracle, SQL Server, and Amazon Aurora
- DynamoDB Predictable and Scalable NoSQL Data Store
- ElastiCache In-Memory Cache
- Redshift Managed Petabyte-Scale Data Warehouse Service

Networking

- VPC Isolated Cloud Resources
- Direct Connect Dedicated Network Connection to AWS
- Route 53 Scalable DNS and Domain Name Registration

Administration & Security

- Directory Service Managed Directories in the Cloud
- Identity & Access Management Access Control and Key Management
- Trusted Advisor AWS Cloud Optimization Expert
- CloudTrail User Activity and Change Tracking
- Config Resource Configurations and Inventory
- CloudWatch Resource and Application Monitoring
- Service Catalog Personalized Catalog of AWS Resources

Deployment & Management

- Elastic Beanstalk AWS Application Container
- OpsWorks DevOps Application Management Service
- CloudFormation Templated AWS Resource Creation
- CodeDeploy Automated Deployments
- CodeCommit Managed Git Repositories
- CodePipeline Continuous Delivery

Analytics

- EMR Managed Hadoop Framework
- Kinesis Real-time Processing of Streaming Big Data
- Data Pipeline Orchestration for Data-Driven Workflows
- Machine Learning Build Smart Applications Quickly and Easily

Application Services

- SQS Message Queue Service
- SWF Workflow Service for Coordinating Application Components
- AppStream Low Latency Application Streaming
- Elastic Transcoder Easy-to-use Scalable Media Transcoding
- SES Email Sending Service
- CloudSearch Managed Search Service
- API Gateway Build, Deploy and Manage APIs

Mobile Services

- 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

Enterprise Applications

- WorkSpaces Desktops in the Cloud
- WorkDocs Secure Enterprise Storage and Sharing Service
- WorkMail PREVIEW Secure Email and Calendaring Service

## Resource Groups

A resource group is a collection of resources that share one or more tags. Create a group for each project, application, or environment in your account.

[Create a Group](#) [Tag Editor](#)

## Additional Resources

[Getting Started](#) See our documentation to get started and learn more about how to use our services.

[AWS Console Mobile App](#) View your resources on the go with our AWS Console mobile app, available from [Amazon Appstore](#), [Google Play](#), or [iTunes](#).

[AWS Marketplace](#) Find and buy software, launch with 1-Click and pay by the hour.

[AWS re:Invent - Register Now](#) Join us for keynote announcements, technical sessions, bootcamps and more.

## Service Health

All services operating normally.

Updated: Aug 08 2015 23:29:00 GMT-0400

[Service Health Dashboard](#)

https://console.aws.amazon.com/ec2/v2/home?region=us-east-1#Instances:sort=instanceId

AWS Services Edit Leapon Liu N. Virginia Support

EC2 Dashboard Events Tags Reports Limits

INSTANCES Instances Spot Requests Reserved Instances

IMAGES AMIs Bundle Tasks

ELASTIC BLOCK STORE Volumes Snapshots

NETWORK & SECURITY Security Groups Elastic IPs Placement Groups Key Pairs Network Interfaces

LOAD BALANCING Load Balancers

AUTO SCALING Launch Configurations Auto Scaling Groups

Launch Instance Connect Actions

Filter by tags and attributes or search by keyword

Name Instance ID Instance Type Availability Zone Instance State Status Checks Alarm Status Public DNS

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS
leapon	i-fe71b42c	t2.micro	us-east-1b	running	2/2 checks ...	None	ec2-52-3-90-30...

Instance: i-fe71b42c (leapon) Public DNS: ec2-52-3-90-30.compute-1.amazonaws.com

Description Status Checks Monitoring Tags

Instance ID	i-fe71b42c	Public DNS	ec2-52-3-90-30.compute-1.amazonaws.com
Instance state	running	Public IP	52.3.90.30
Instance type	t2.micro	Elastic IP	-
Private DNS	ip-172-31-14-14.ec2.internal	Availability zone	us-east-1b
Private IPs	172.31.14.14	Security groups	launch-web . view rules
Secondary private IPs		Scheduled events	No scheduled events
VPC ID	vpc-844eafe0	AMI ID	ubuntu-trusty-14.04-amd64-server-20150325 (ami-d05e75b8)
Subnet ID	subnet-f4102783	Platform	-
Network interfaces	eth0	IAM role	-
Source/dest. check	True	Key pair name	leapon-reactlet-aws
EBS-optimized	False	Owner	542714105552
Root device type	ebs	Launch time	August 2, 2015 at 11:26:14 AM UTC-4 (156 hours)
Root device	/dev/sda1	Termination protection	False
Block devices	/dev/sda1	Lifecycle	normal
		Monitoring	basic
		Alarm status	None

---

# RUN CONTAINER ON AWS

```
ssh -i ~/key/leapon-reactlet-aws.pem ubuntu@52.3.90.30
```

```
docker pull leapon/dnrdw
```

```
docker run --name=dnrdw -p 80:8000 -t -d leapon/dnrdw
```

```
docker logs -f dnrdw
```

---

# Demo App

August 2015

Sun	Mon	Tue	Wed	Thu	Fri	Sat
26	27	28	29	30	31	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31	1	2	3	4	5

# Q & A

