5 Minutes to Enterprise JavaScript
With Red Hat OpenShift Application Runtimes

Lance Ball
Principal Software Engineer

John Clingan
Product Manager RHOAR

Wednesday, May 9 2018
NODE.JS
Is a Thing at Red Hat
So What Do You Mean?

- RHOAR - Red Hat OpenShift Application Runtime for Node.js 8.x LTS
- Supported Node.js RPMs and Runtime containers for Node.js 8.x LTS
- Community Node.js RPMs and Runtime containers for Node.js 9.x and 10.x
- Node.js tools and utilities for containerized deployment
- Node.js Core Committers
- Node.js Technical Steering Committee
OK… But Why Node.js?
In the Beginning, there was the Monolith

My App
The Monolith Begat Microservices
Small, discrete services that do one thing well - usually as REST over HTTP
This is what Node.js does well
But wait
My application is very complex!
Application complexity shifts from application code to the runtime platform.
Logging

µ svc A

µ svc B

µ svc C

µ svc D

µ svc E

µ svc F

µ svc G

µ svc H

µ svc J

Service Discovery

Routing

Elasticity

Tracing

Authn/z

Resilience
Application logic should be simple
Demo: set up minishift

```bash
$ minishift profile set 'summit-demo'
$ minishift config set cpus 2
$ minishift config set vm-driver virtualbox
$ minishift config set memory 4GB
$ minishift start
$ eval (minishift oc-env) # fish shell!
$ oc new-project summit-demo
```
Demo: create and deploy an application

$ mkdir myapp; and cd myapp
$ npm install -g express-generator
$ express .
$ code-insiders package.json # fish shell!
$ npx nodeshift --strictSSL=false --expose
Yeah and... now what?
Didn’t you say “Enterprise”?
μ-Services are not a panacea
Some services will fail
Causing more services to fail
eval(cascading failures) > dead application
Let’s get enterprisey
With RHOAR circuit breakers
RHOAR Demo
MICROSERVICES 2014 - Current

Support Services
- Smart Routing
- API Management
- Caching Service
- Configuration
- Messaging
- SSO
- Registry

Application Logic
- Client-side Load Balancing
- Service Registration
- Circuit Breaker
- Distributed Tracing

2014
RED HAT OPENSSHIFT APPLICATION RUNTIMES
Providing curated set of integrated runtimes and frameworks that standardizes Cloud Native App Dev

✓ Simplified development
✓ Strategic flexibility
✓ DevOps automation

YOUR APPS AND SERVICES

RED HAT OPENSSHIFT APPLICATION RUNTIMES
Launch Service
Reactive Vert.x  MicroProfile WildFly Swarm  Spring Boot Tomcat  JavaScript Node.js  Java EE JBoss EAP

APPLICATIONS LIFECYCLE MANAGEMENT
SUPPORTING MIDDLEWARE SERVICES
CONTAINER ORCHESTRATION & MANAGEMENT (KUBERNETES)

RED HAT OPENSSHIFT

YOUR INFRASTRUCTURE

Amazon Web Services
Microsoft Azure
Google Cloud
OpenStack
Datacenter
Laptop
EVOLUTION OF MICROSERVICES (2014 - FUTURE)

**Application Logic**
- Client-side Load Balancing
- Service Registration
- Circuit Breaker
- Distributed Tracing

**Support Services**
- Smart Routing
- API Management
- Caching Service
- Configuration
- Messaging
- SSO
- Registry

---

**Application Logic**
- Client-side Load Balancing
- Circuit Breaker

**Support Services**
- Distributed Tracing
- API Management
- Caching Service
- Messaging
- SSO

---

**Application Logic**
- Server-side Load Balancing
- Client-side Load Balancing
- Circuit Breaker

**Support Services**
- Registry
- Configuration
- Server-side Load Balancing

---

2014 | Current | Future
https://developers.redhat.com/launch
https://github.com/bucharest-gold/node-rpm
https://github.com/bucharest-gold/centos7-s2i-nodejs
THANK YOU

plus.google.com/+RedHat
linkedin.com/company/red-hat
facebook.com/redhatinc
twitter.com/RedHat
youtube.com/user/RedHatVideos