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

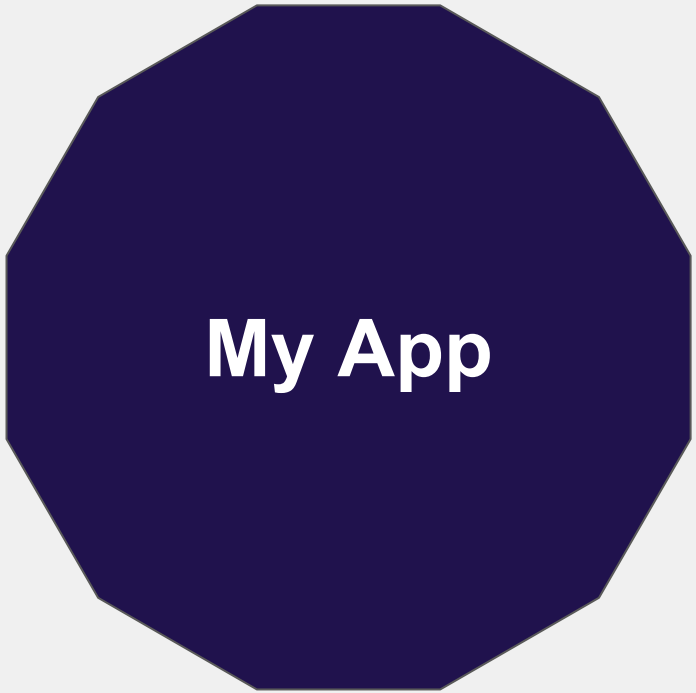
NODE.JS
Is a Thing at Red Hat

So What Do You Mean?

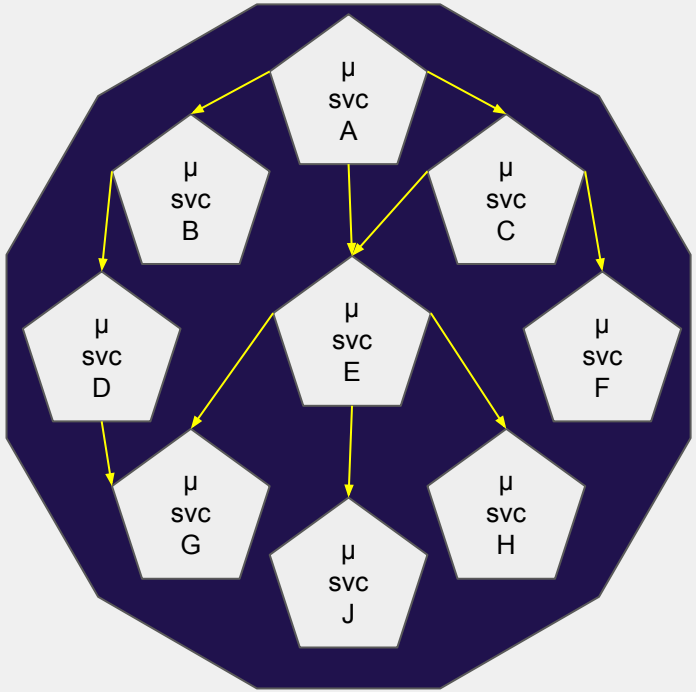
- RHOAR - Red Hat OpenShift Application Runtime for Node.js 8.x LTS
- Supported Node.js RPMs and Runtime containers for Node.js 8.x LTS
- Community Node.js RPMs and Runtime containers for Node.js 9.x and 10.x
- Node.js tools and utilities for containerized deployment
- Node.js Core Committers
- Node.js Technical Steering Committee

OK... But Why Node.js?

In the Beginning, there was the Monolith



The Monolith Begat Microservices

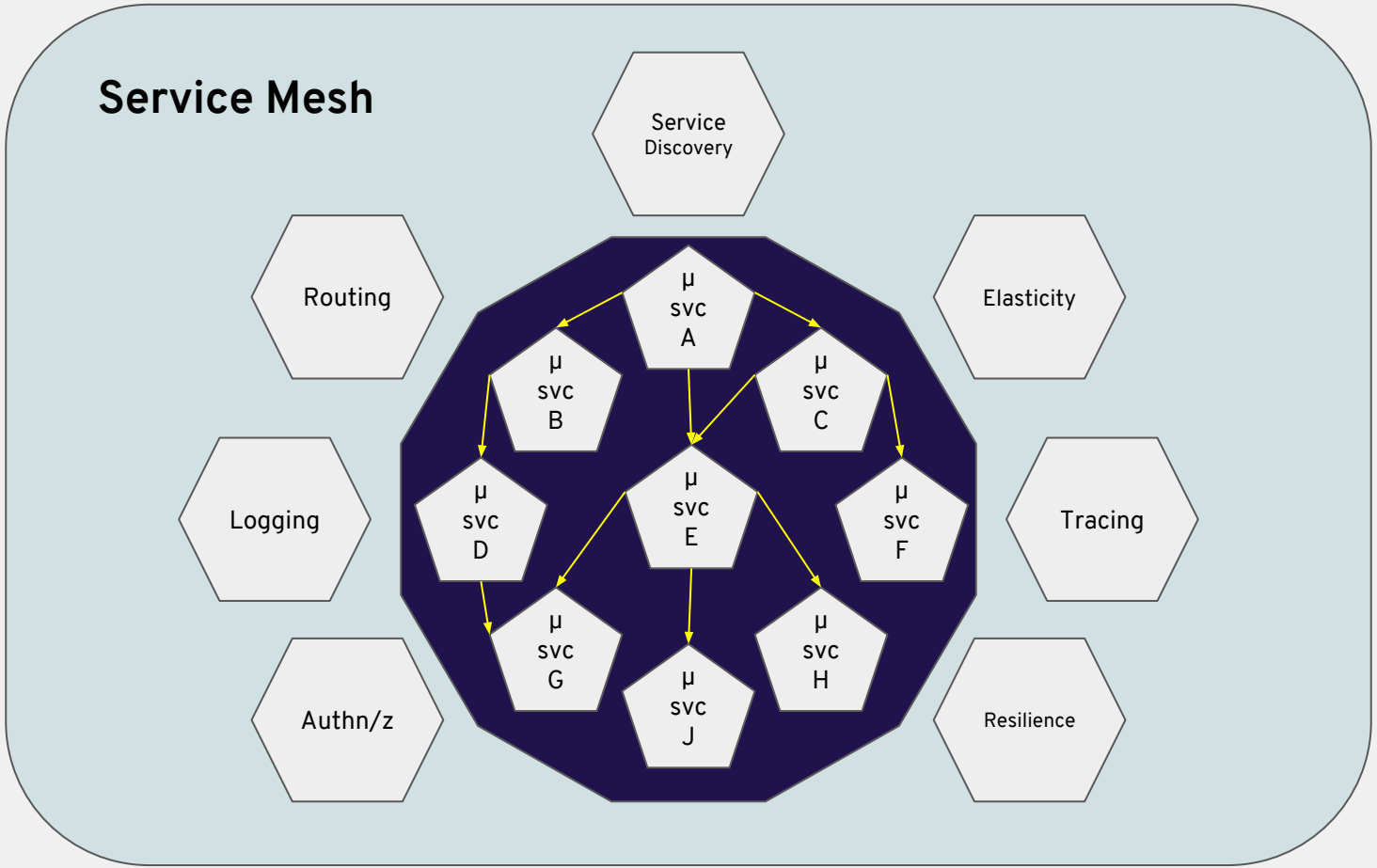


Small, discrete services that do one thing well - usually as REST over HTTP

This is what Node.js does well

But wait
My application is very complex!

Application complexity shifts from application code to the runtime platform


Service Mesh
Service Discovery
Routing
Elasticity
Logging
Tracing
Authn/z
Resilience
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

Application logic should be simple

Demo: set up minishift

```
$ 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
```

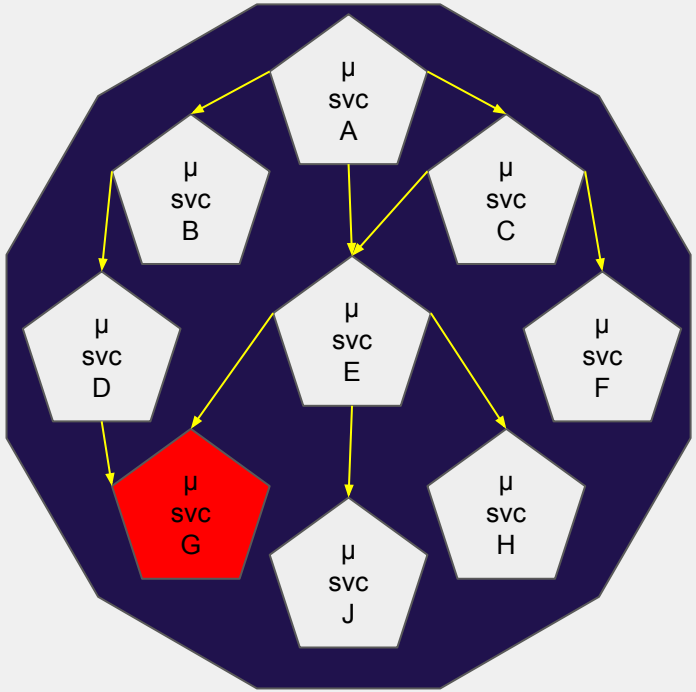
Demo: create and deploy an application

```
$ mkdir myapp; and cd myapp
$ npm install -g express-generator
$ express .
$ code-insiders package.json # fish shell!
$ npx nodeshift --strictSSL=false --expose
```

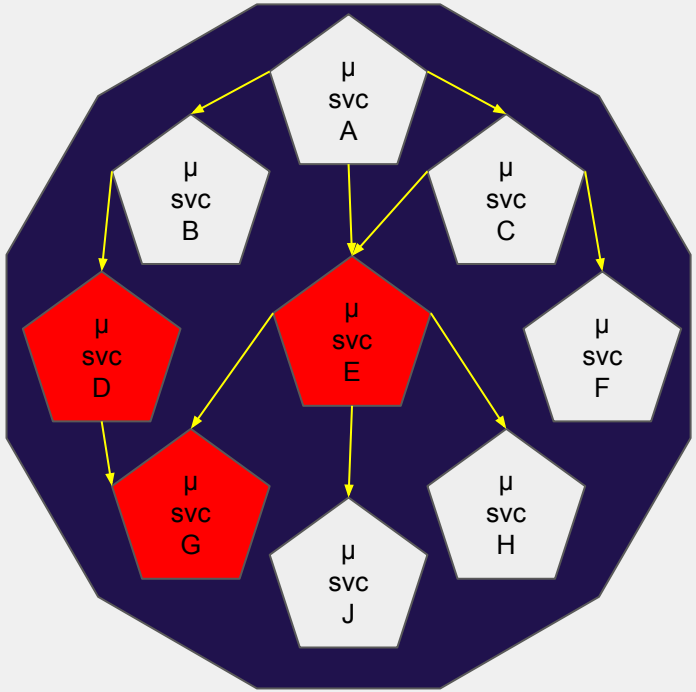
Yeah and… now what?

Didn’t you say “Enterprise”?

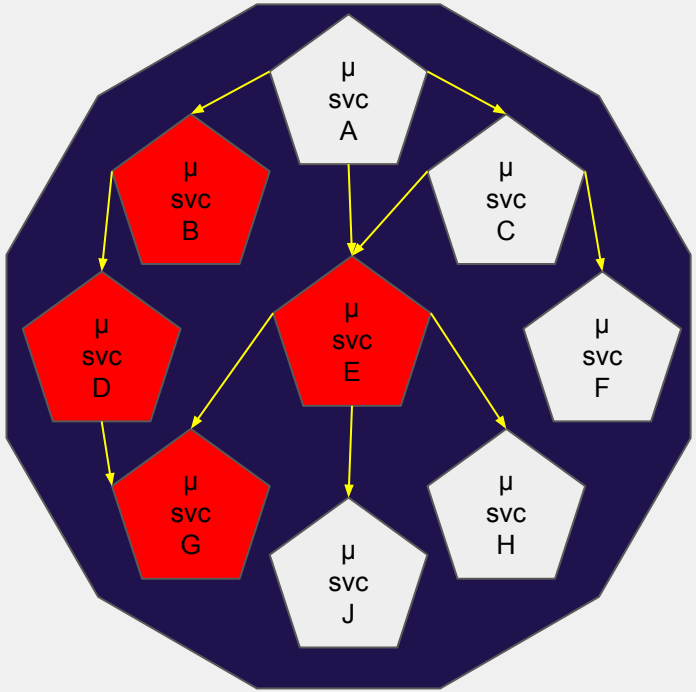
µ-Services are not a panacea



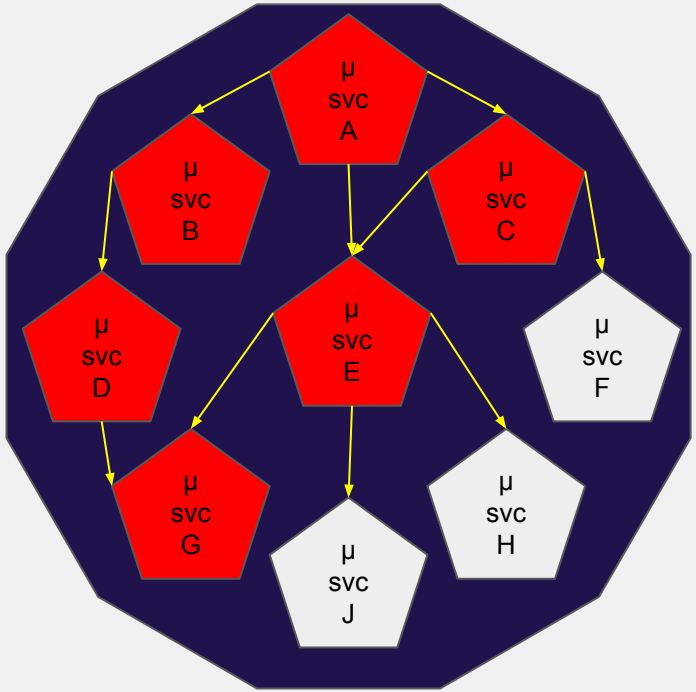
Some services will fail



Causing more services to fail



eval(cascading failures) > dead application



Let’s get enterprisey
With RHOAR circuit breakers

RED HAT DEVELOPER PROGRAM

Lance Ball

LAUNCH

Continuous application delivery,

built and deployed on OpenShift.

LAUNCH YOUR PROJECT

Supported Runtimes

WildFly SWARM

WildFly Swarm offers an innovative approach to packaging and running Java EE applications by packaging them with just enough of the server runtime to "java -jar" your application.

Learn more

VERT.X

Eclipse Vert.x is a tool-kit for building reactive applications on the JVM.

Learn more

Spring Boot makes it easy to create stand-alone, production-grade Spring based Applications that you can "just run".

Learn more

technology preview

RED HAT® FUSE

Red Hat® Fuse is a lightweight, flexible integration platform that uses Apache Camel at his core.

node JS

Node.js® is a JavaScript runtime built on Chrome's V8 JavaScript engine. Node.js uses an event-driven, non-blocking I/O model that makes it lightweight and efficient

Deployment type (1) — Mission (2) — Runtime (3) — Project Info (4) — Review (5) — Next Steps (6)

Deployment type

Use Continuous Delivery

To launch using OpenShift, click the *Use OpenShift* button. This will:

- In your GitHub namespace, create a repository containing your new project's code.
- Configure OpenShift to build and deploy your code on each push to your repository's master branch.

OpenShift Cluster* starter-ca-central-1

Configure tokens

Use OpenShift

Build and run locally

To launch manually, click on *I will build and run locally* and follow the instructions in the project's guide (found linked in the README). This will:

- Scaffold a project based on your chosen runtime
- Allow you to download the project as a ZIP file

I will build and run locally

Deployment type (1) — Mission (2) — Runtime (3) — Project Info (4) — Review (5) — Next Steps (6)

Mission

Missions are preconfigured, functioning applications that demonstrate a fundamental aspect of modern application development running in an environment similar to production. These can be used as a proof of technology demonstration, a teaching tool, or even a sandbox for understanding how to develop applications.

CRUD

Mission proficiency level: **Foundational**.

What the Relational Database Backend Booster Does
The Relational Database Backend booster expands on the REST API Level 0 booster to provide a basic example of performing *create*, *read*, *update* and *delete* (*CRUD*) operations on a PostgreSQL database using a simple HTTP API. *CRUD* operations are the four basic functions of persistent storage, widely used when developing an HTTP API dealing with a database.

Cache

Use a cache to improve the response time of applications

Circuit Breaker

Mission proficiency level: **Foundational**.

The *Circuit Breaker* mission demonstrates a generic pattern for reporting the failure of a service and then limiting access to the failed service until it becomes available to handle requests. This helps prevent cascading failure in other services that depend on the failed services for functionality.

Externalized Configuration

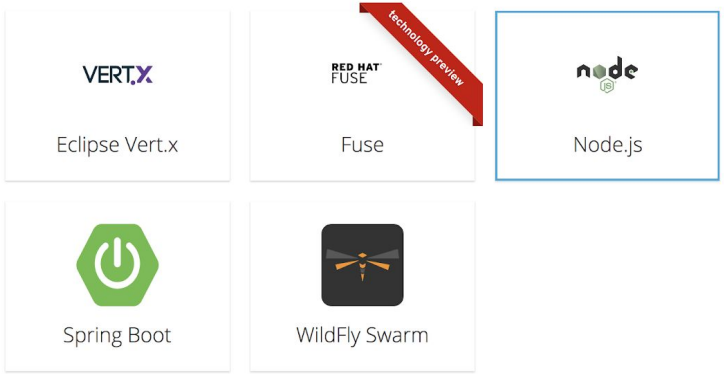
Mission proficiency level: **Foundational**.

The Externalized Configuration mission provides a basic example of using a ConfigMap to externalize configuration. *ConfigMap* is an object used by OpenShift to inject configuration data as simple key and value pairs into one or more Linux containers while keeping the containers independent of OpenShift.

Deployment type (1) — Mission (2) — Runtime (3) — Project Info (4) — Review (5) — Next Steps (6)

Runtime

We offer a choice of runtime frameworks to best fit your needs. WildFly Swarm delivers a microservices approach to Java EE, Eclipse Vert.x excels at reactive, asynchronous applications, and Spring Boot users may bring their projects to OpenShift as well.



Back Next

Deployment type (1) — Mission (2) — Runtime (3) — Project Info (4) — Review (5) — Next Steps (6)

Project Info

Runtime Version: 8.x (Community) / 8.x (RHOAR)

Name*: booster-circuit-breaker-nodejs

Version*: 1.0.0

Back | Next

Deployment type (1) — Mission (2) — Runtime (3) — Project Info (4) — Review (5) — Next Steps (6)

Review Summary

Deployment type

ZIP File

OpenShift Cluster: starter-ca-central-1

Mission

Circuit Breaker

Runtime

Node.js

Project Info

Runtime Version: 8.x (Community)

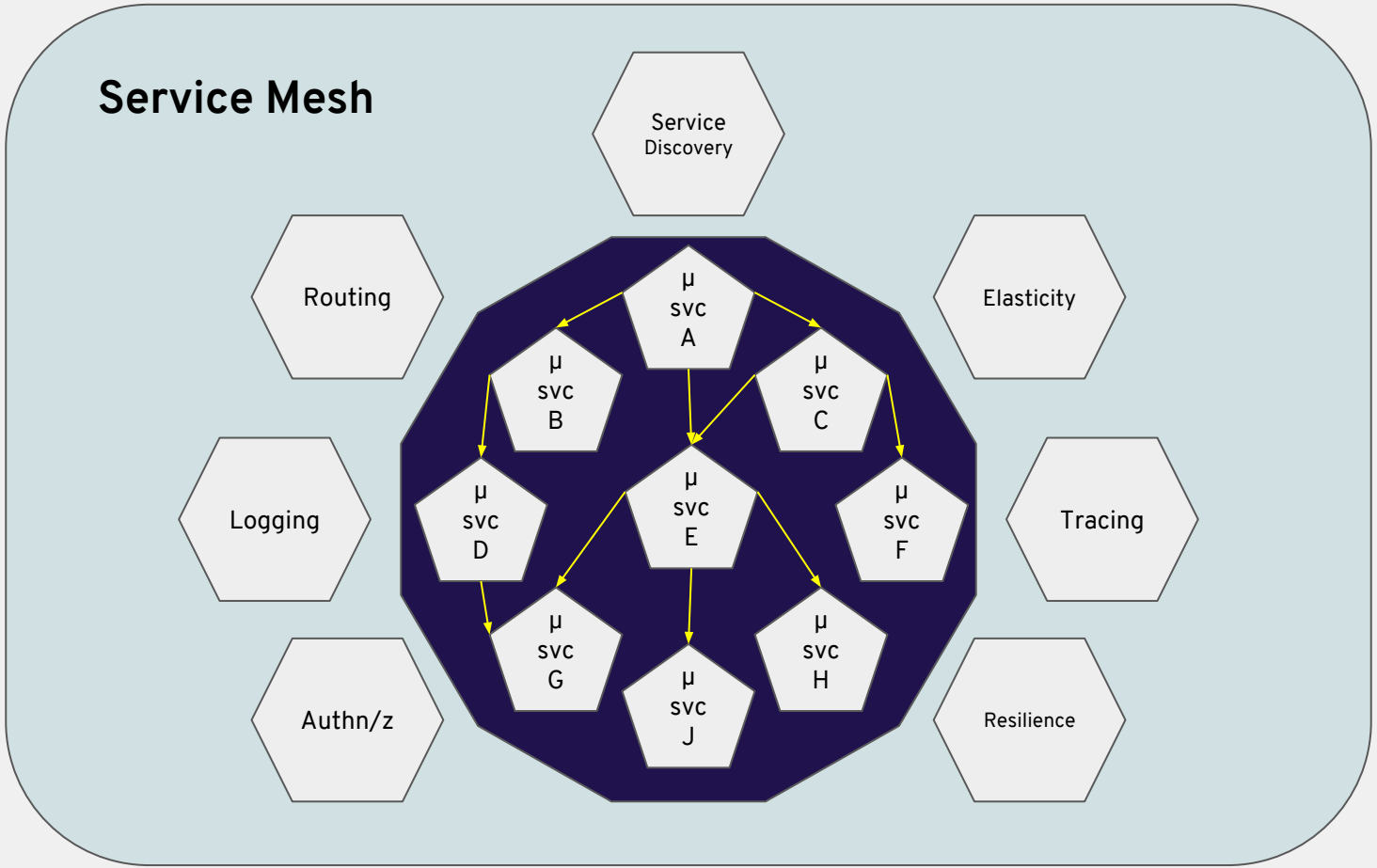
Name: booster-circuit-breaker-nodejs

Version: 1.0.0

Your project is available for download and is ready to build and deploy locally. Refer to the mission details in the Node.js Runtime Guide for more information on building, deploying, and interacting with your booster.

Back | Download as ZIP File

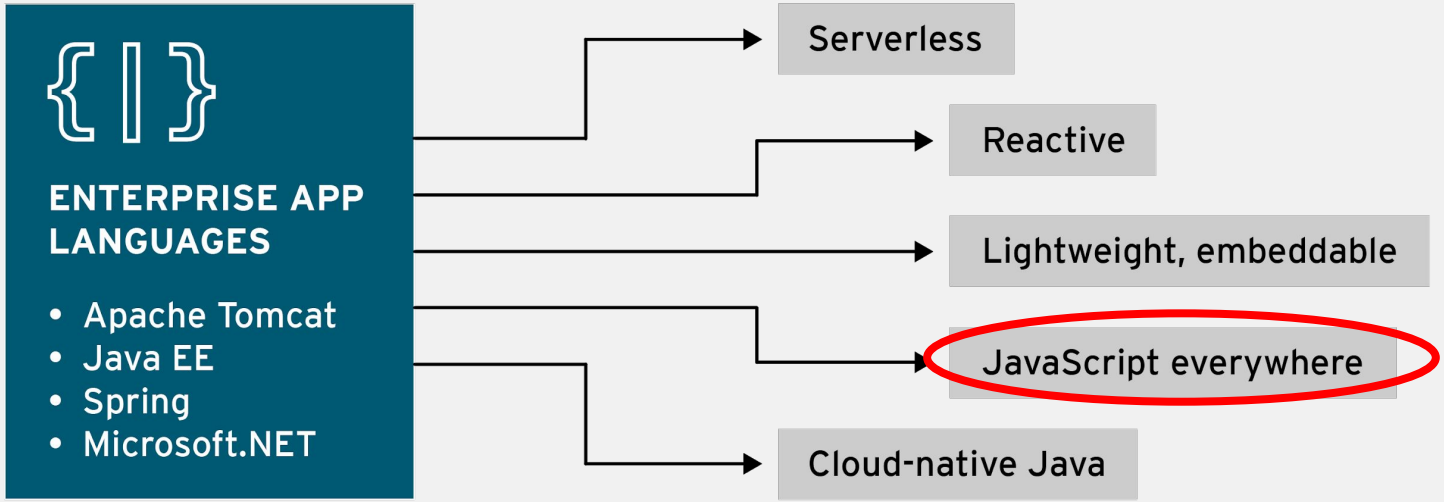
RHOAR Demo


Service Mesh
Service Discovery
Routing
Elasticity
Logging
Tracing
Authn/z
Resilience
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

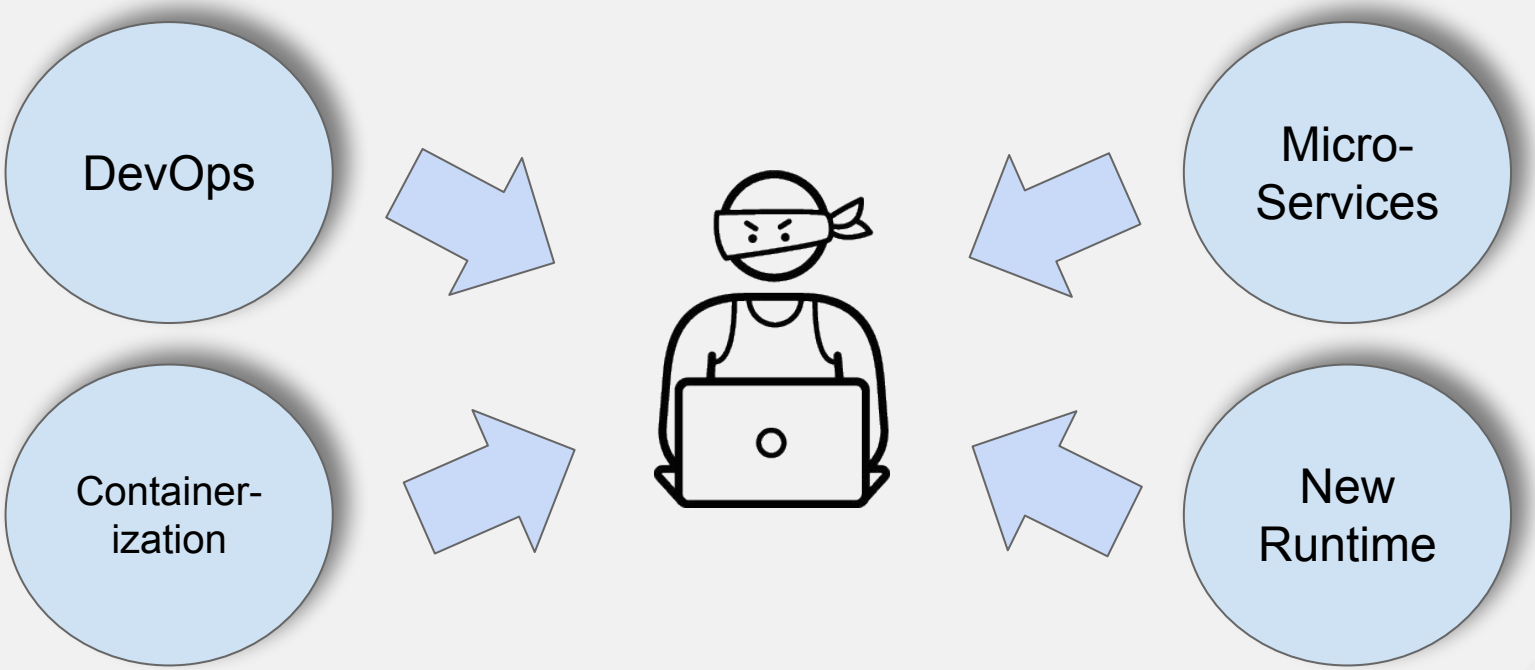
Commercial Break

DEVELOPERS DEMAND MORE OPTIONS

ENTERPRISES EXPAND USE OF LANGUAGES, FRAMEWORKS, & RUNTIMES

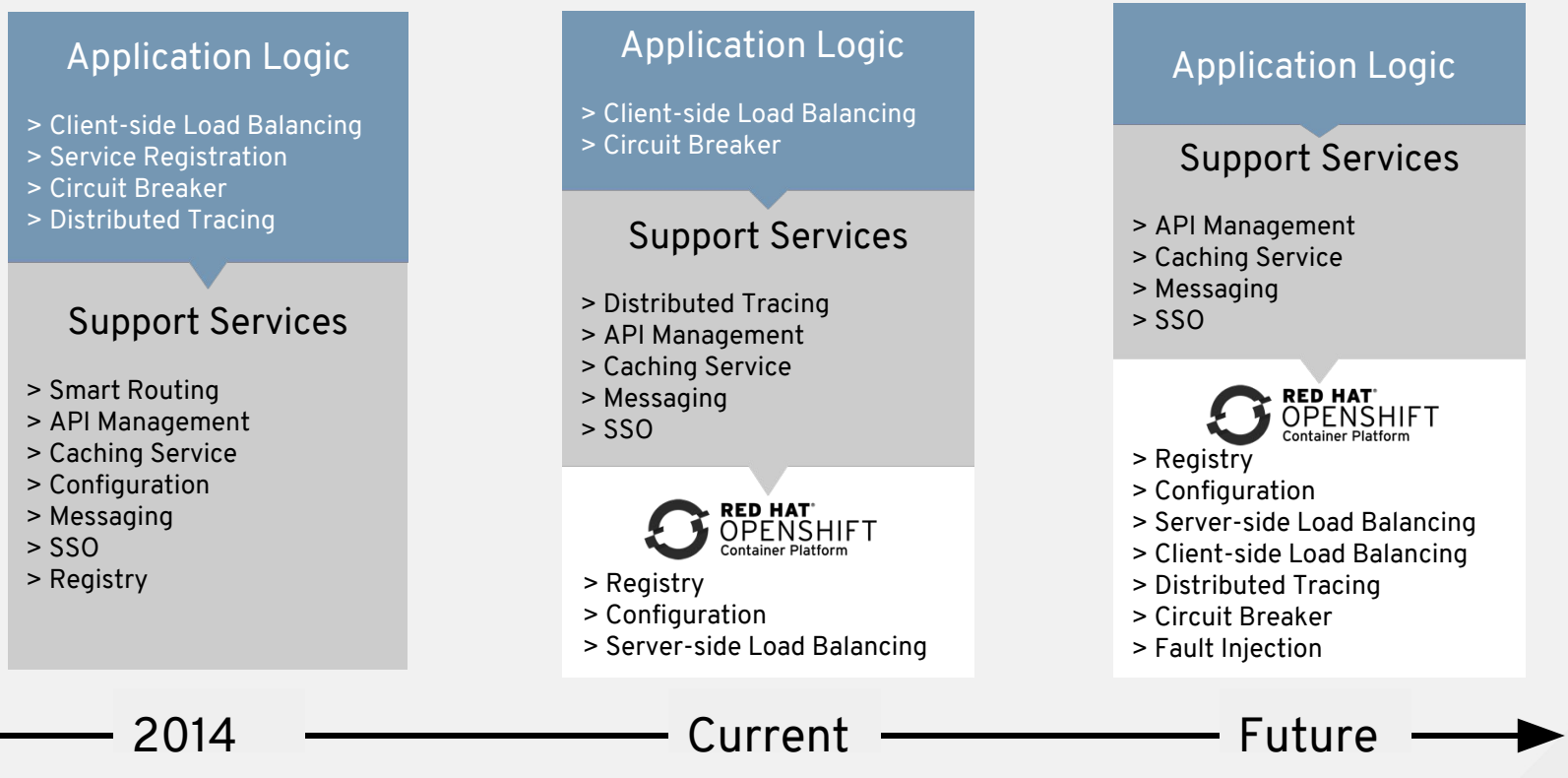


Developers are Asked to Manage a Lot of Change



Helping Developers find their Inner Peace

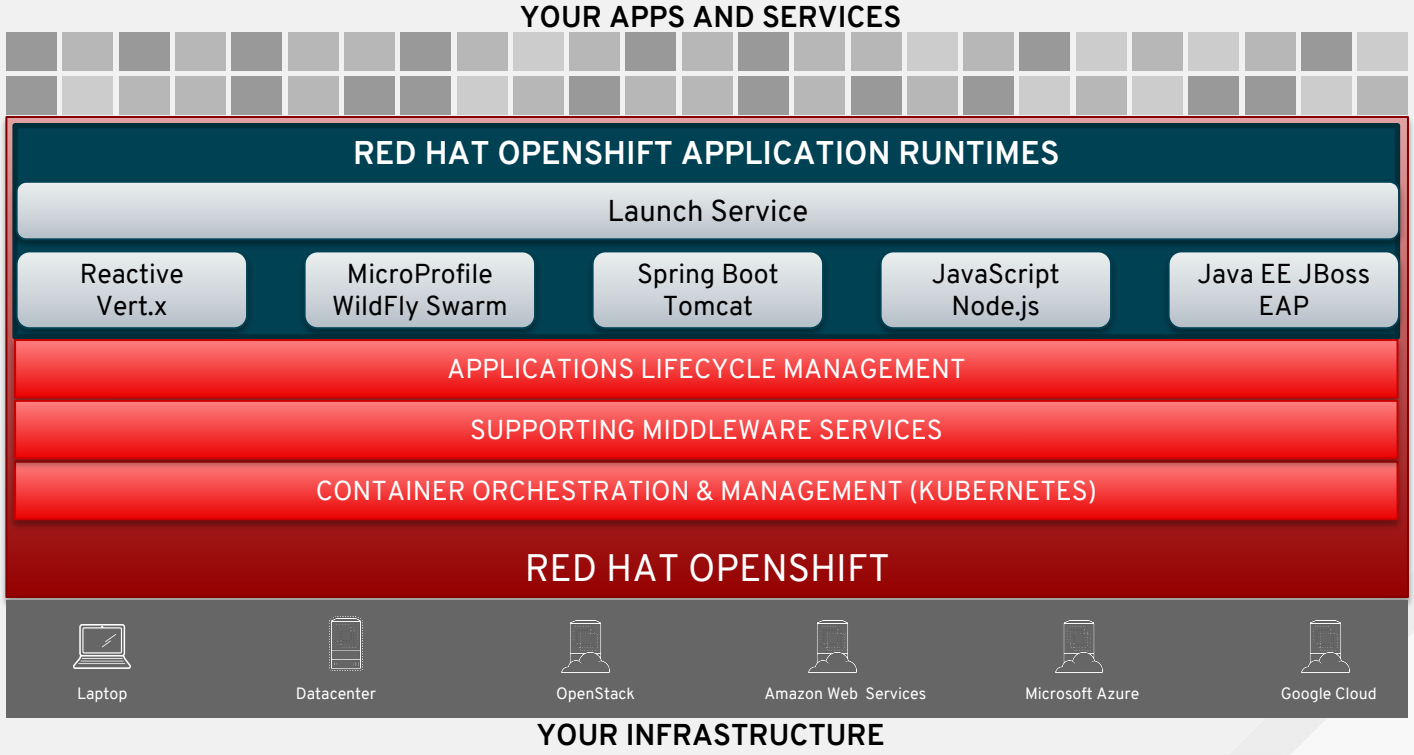
EVOLUTION OF MICROSERVICES (2014 - FUTURE)



RED HAT OPENSHIFT APPLICATION RUNTIMES

Providing curated set of integrated runtimes and frameworks *that standardizes Cloud Native App Dev*

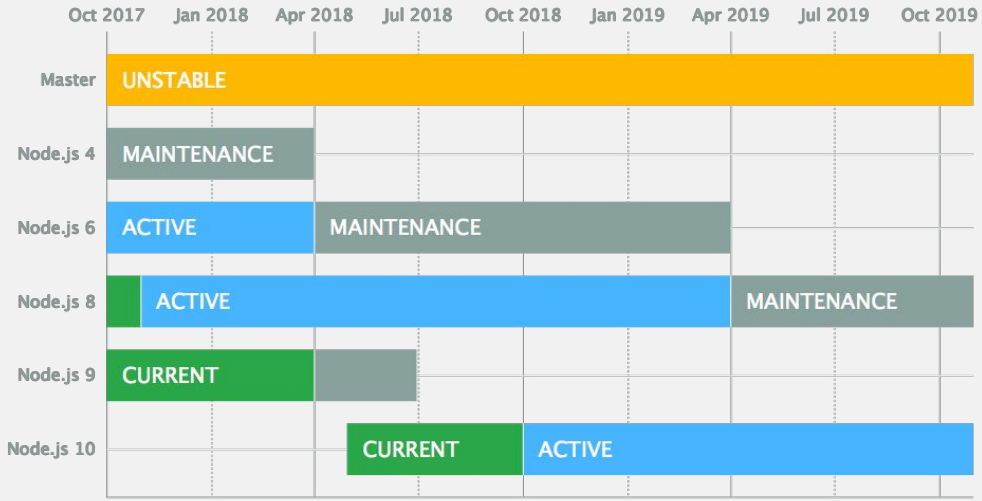
- ✓ Simplified development
- ✓ Strategic flexibility
- ✓ DevOps automation



Optimized for Kubernetes and DevOps

- Reduce complexity of build cloud native applications
- Supports kubernetes features that replace traditional standalone backing services
  - **Externalized Configuration:** Kubernetes ConfigMap
  - **Service Discovery**: Kubernetes Services (DNS)
  - **Load Balancing**: Kubernetes Replication Controller
  - **Auto-Restart**: Kubernetes Health Check w/MicroProfile Health Check API
  - **Metrics**: MicroProfile Metrics API with CNCF Prometheus
  - **Distributed Tracing**: Istio & OpenTracing with CNCF Jaeger

RHOAR Node.js Support and Lifecycle



- Support LTS releases
- Align with Node.js lifecycle
- Developer builds of non-LTS releases

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


facebook.com/redhatinc


linkedin.com/company/red-hat


twitter.com/RedHat


youtube.com/user/RedHatVideos