Moving Renewable Energy Embedded Systems into the Cloud

Mark Heckler

Principal Technologist/Developer Advocate

Pivotal Software, Inc.

www.thehecklers.org
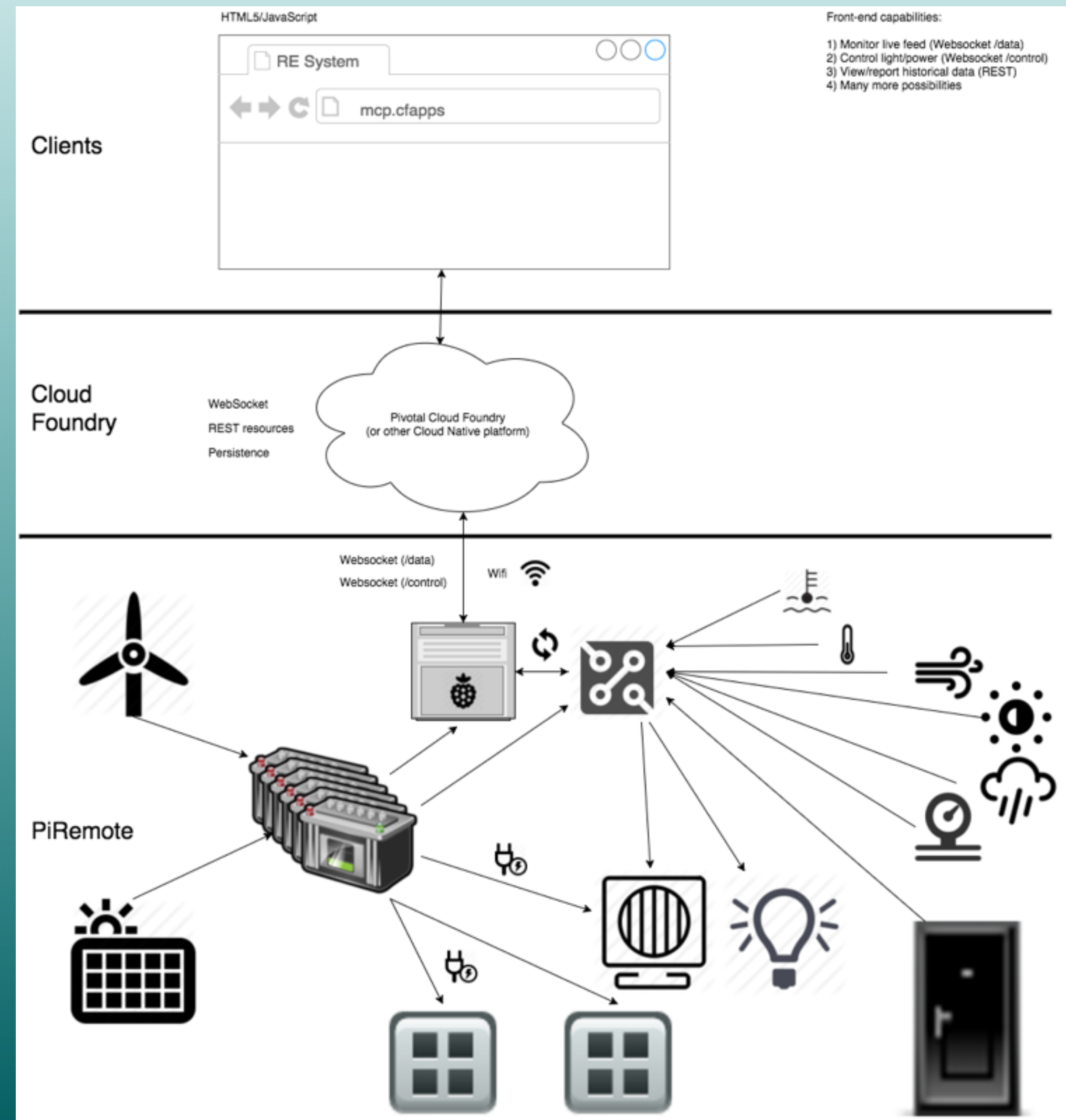
@MkHeck

Agenda at a Glance

- The Physical Side of IoT

- The Other 80%

- Let's build it!

- Questions

@MkHeck

The Physical Side of IoT: Requirements

- Easy to Set Up

- Versatile & Reliable Communication

- *Sufficient* Performance

- Expandable & Scalable

- Standards-based

- Easy to Deploy

@MkHeck

Architecture at a Glance

@MkHeck

Arduino for Sensor Duties

- Open source (hardware and software)

- Architecture permits easy replacement

- Shields up!

- Processing language similar to Java
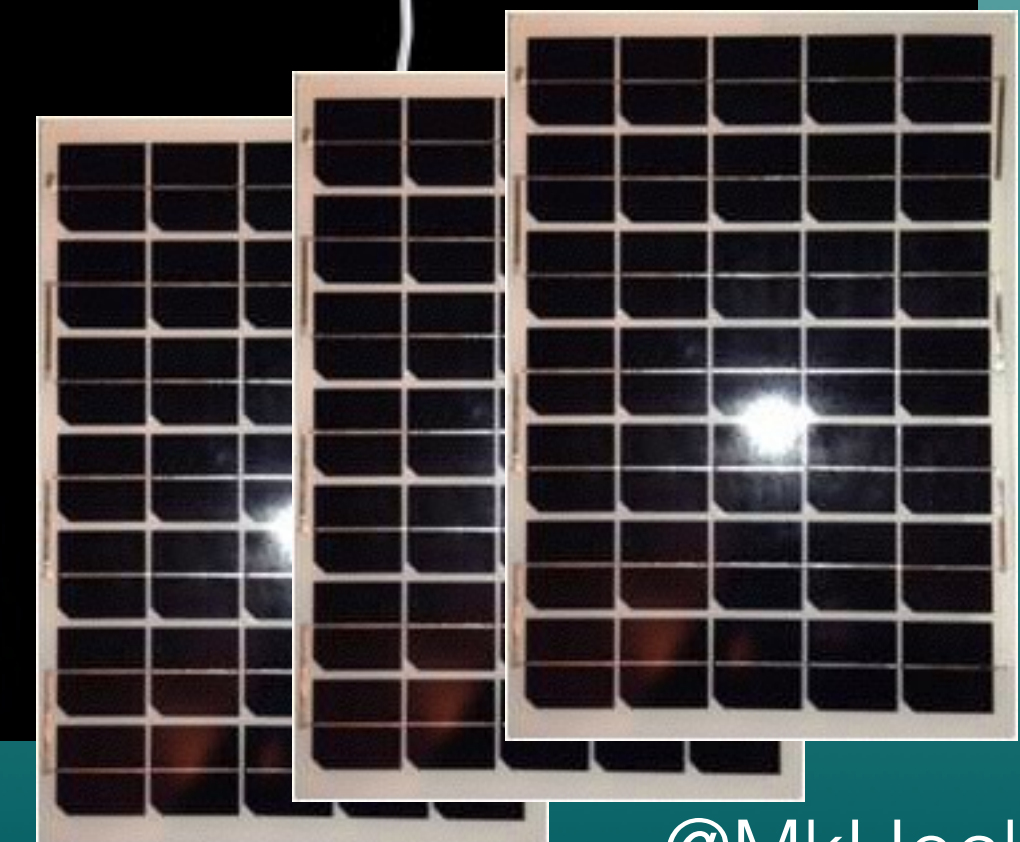
@MkHeck

Arduino for Sensor Duties

- Robust and durable

- Miserly power requirements

- Input power versatility: rated for 7-12v, but can take 6-20v

- Output power versatility: standard boards provide 3.3v & 5v

Raspberry Pi for Communication & Higher-level Logic

- Linux (ssh, vnc, versatility)

- Java

    - SE Embedded ships on Pi (Raspbian)

    - Library & tools support

- Web Standards

@MkHeck

"Things" as a Remote System Component

- Monitor Renewable Energy production/consumption

- Monitor environmental factors

- Powered by monitored system

- Low power consumption

The "Things"

@MkHeck

The Other 80%

- The **INTERNET** of Things

- Without connectivity…

NOW ENTERING the ISLAND OF MISFIT TOYS

The Other 80%

- Cloud application is vital to fully leverage IoT deployment

- Platform choices for this app

  - Versatile open source cloud

  - Open source development platform

  - SIMPLICITY

The Other 80%

- Spring brings all the toys

  - Boot (SUPERJAR!)

  - Actuator (health metrics/statistics)

  - Data REST, HATEOAS, HAL Browser (REST resources)

  - Data JPA (database backing)

  - Web, WebSocket

  - Cloud Connectors
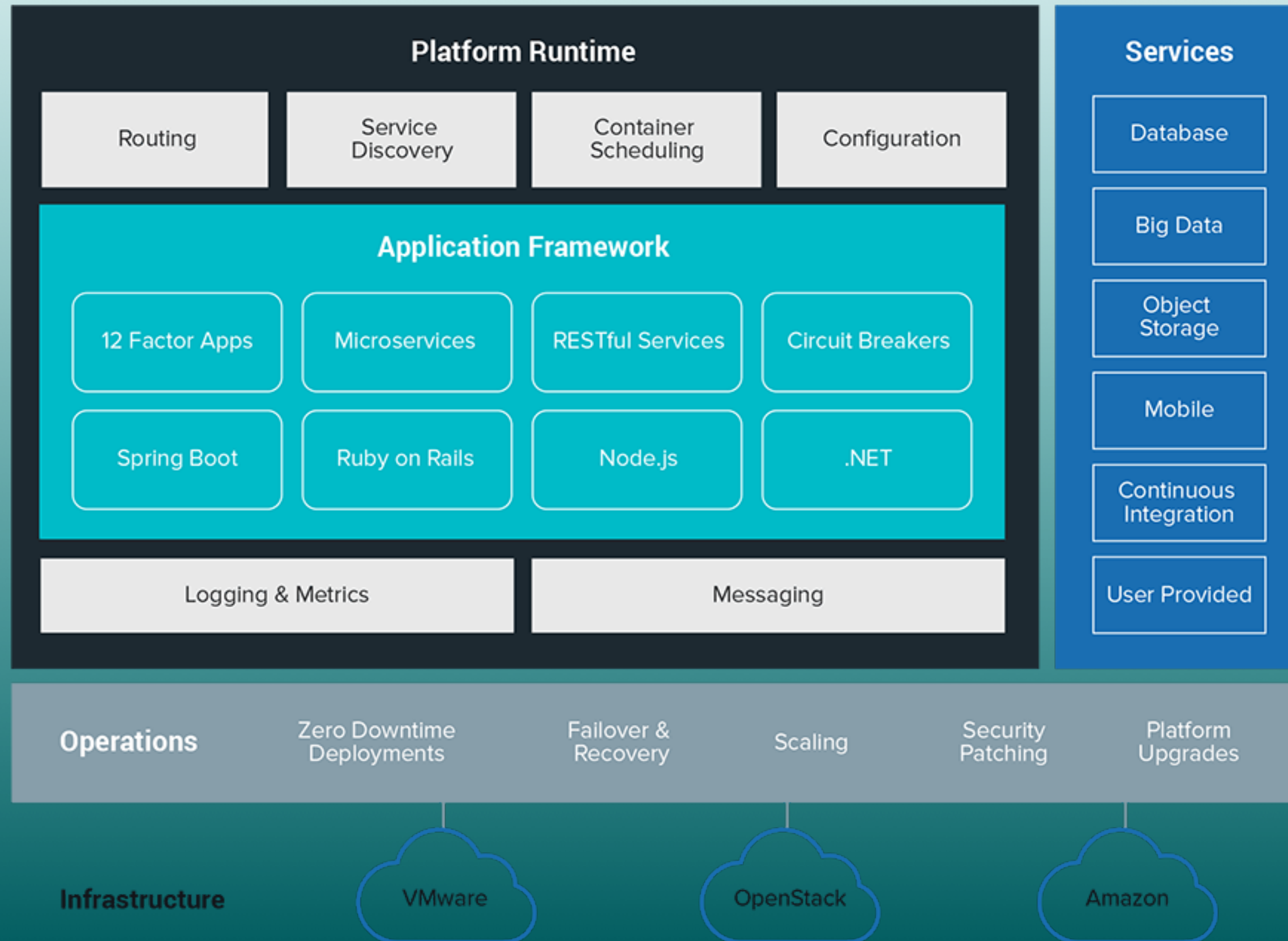
@MkHeck

Let's build it!

Thank You for Participating!

- Reference Links

  - 12 Factor apps: 12factor.net

  - Spring Initializr: start.spring.io

  - Cloud Foundry: cloudfoundry.org

  - Pivotal Web Services: run.pivotal.io

  - Code for this session: https://github.com/hecklerm/MCP

@MkHeck

Cloud Foundry

Platform Runtime

Routing	Service Discovery	Container Scheduling	Configuration

Application Framework

12 Factor Apps	Microservices	RESTful Services	Circuit Breakers
Spring Boot	Ruby on Rails	Node.js	.NET

Logging & Metrics	Messaging

Services

- Database
- Big Data
- Object Storage
- Mobile
- Continuous Integration
- User Provided

Operations

| Zero Downtime Deployments | Failover & Recovery | Scaling | Security Patching | Platform Upgrades |

Infrastructure

VMware   OpenStack   Amazon

@MkHeck

The End

Power Monitoring & Management in the Cloud

Temperature 25.0°C
Humidity 24.1%
Voltage 12.600V
Luminosity 0.00w

Wind Direction 225
Wind Speed 1.46mph
Rainfall 0.00in
Atmospheric Pressure 99735Pa

Mode: A
Power: 0
Status Lamp: 1
Windows: 1
Interior Light: 0

MCP Evolution

Mark Heckler, JavaOne 2015

@MkHeck