Container Platforms as Equalizers: Running Health Services Across the World

PRAEKELT·ORG

Jamie Hewland

KubeCon + CloudNativeCon Seattle

11 December 2018

00

Introduction

Me & Praekelt.org

About me

🌍 Johannesburg, South Africa

👨🏼‍💻 Site Reliability Engineer (SRE)

💁🏼‍♂️ Tech Ambassador

🐦 @jayhewland
🐙 @JayH5

Our Mission

Praekelt.org uses mobile technology to solve some of the world's largest social problems.

PRAEKELT·ORG

Our Technologies

We build open-source, scalable platforms that allow anyone with a mobile phone to access vital information and essential services—putting wellbeing in the palm of their hands.

PRAEKELT·ORG

Nonprofit projects

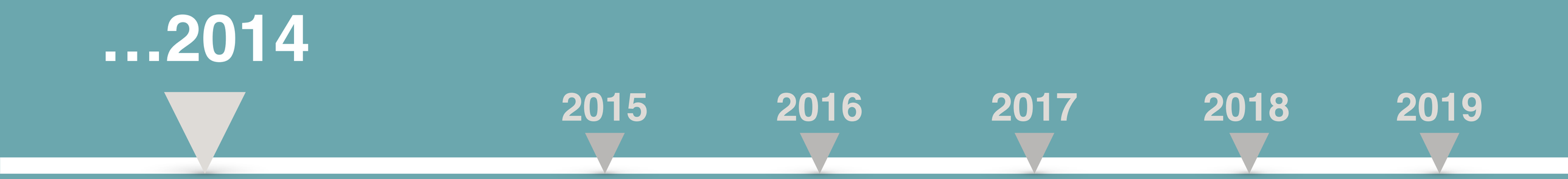
- **Projects developed through partnerships with funders**
- **Several different projects with different funders at any one time**
- **Projects vary in size**
  - Multi-year, national-scale
  - Short pilot projects, studies

01

Before containers

The Problem

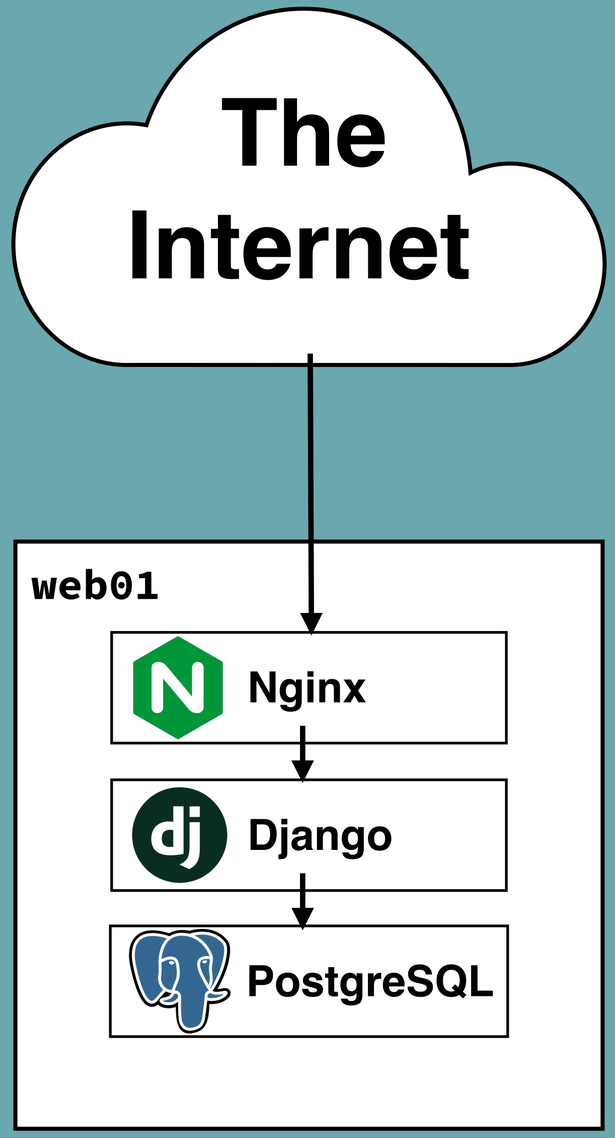
Timeline

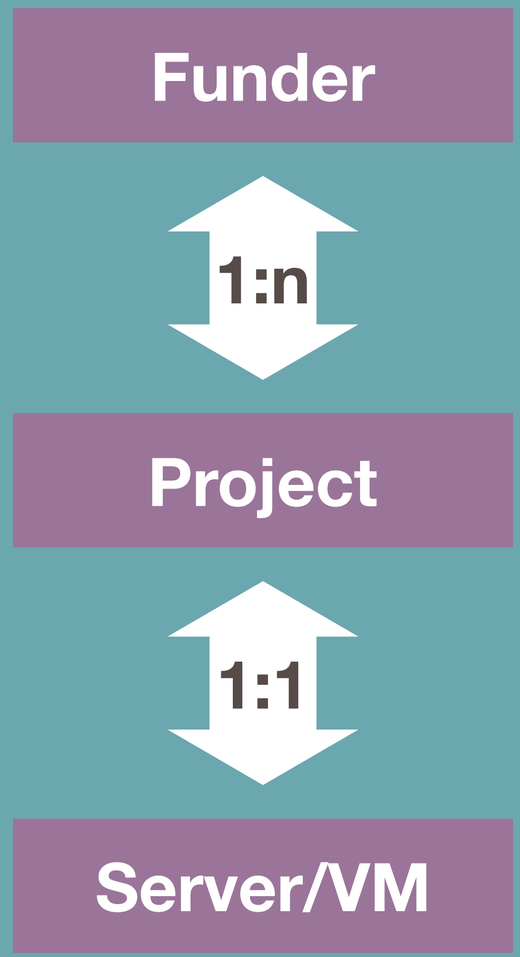


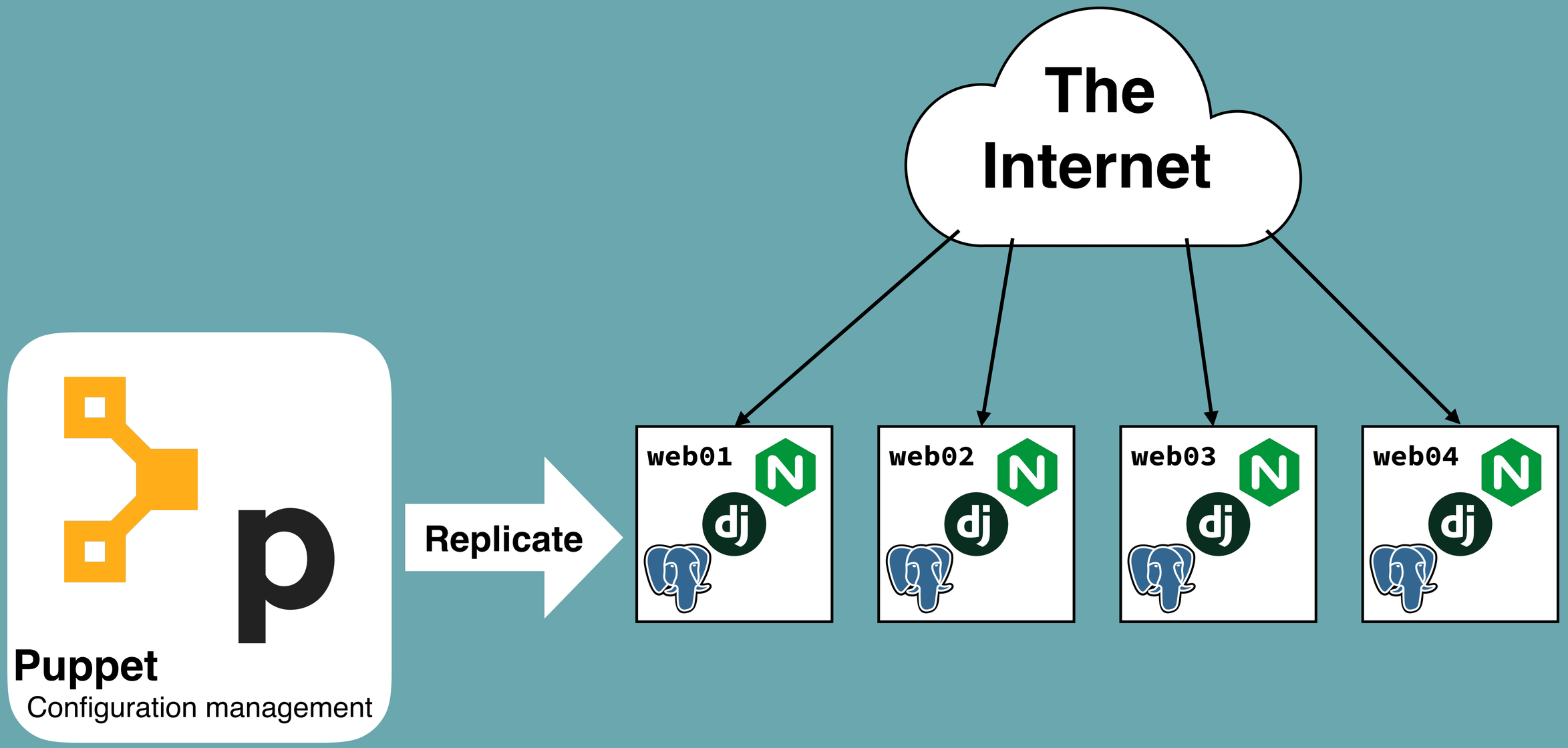
Youth

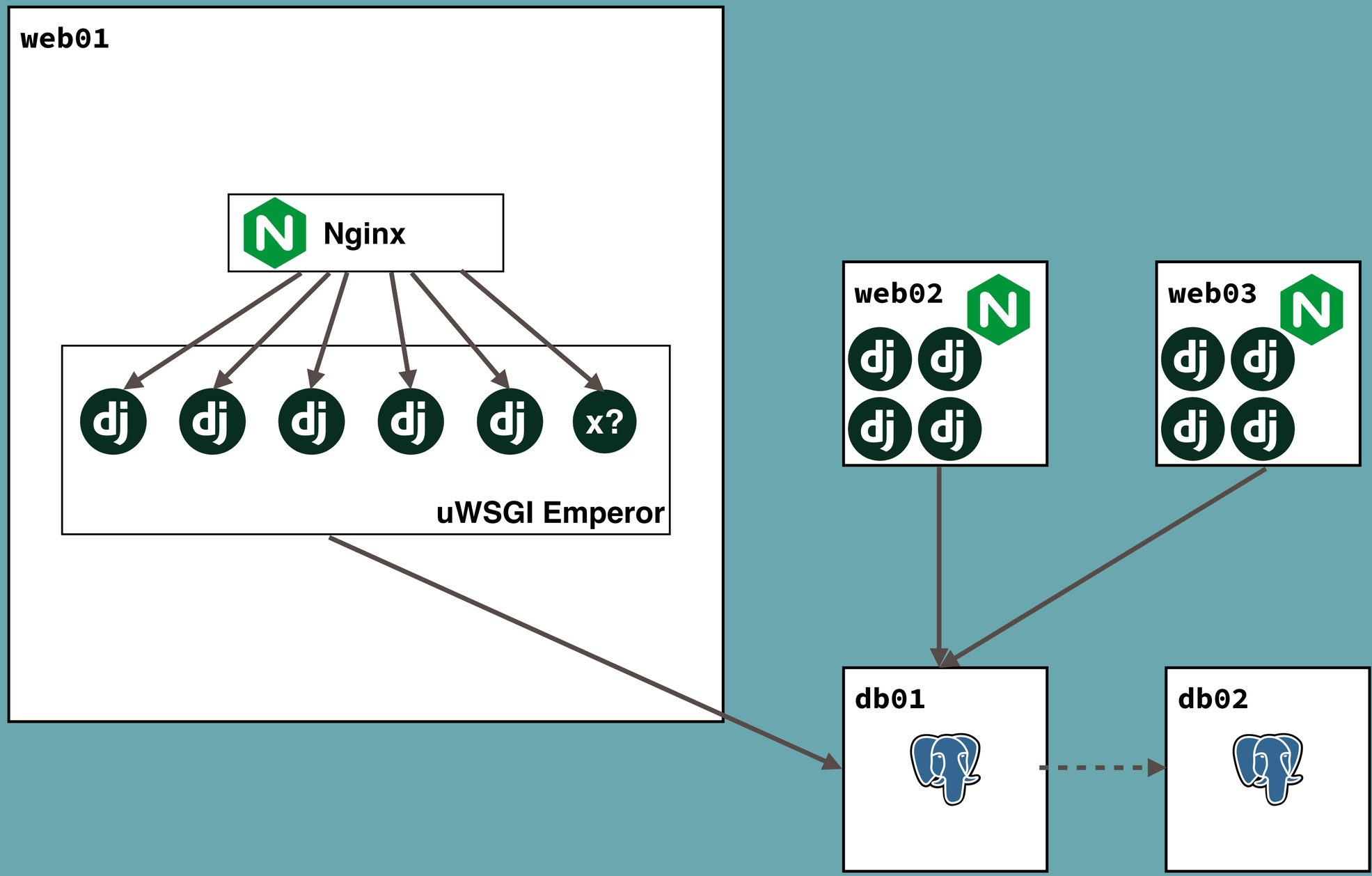
1. Running more sites more efficiently

- Mobi-sites & social media
- Education, sexual & reproductive health


The Internet
web01
Nginx
Django
PostgreSQL


Funder
1:n
Project
1:1
Server/VM


The Internet
Puppet
Configuration management
Replicate
web01
web02
web03
web04


web01
Nginx
dj
dj
dj
dj
dj
x?
uWSGI Emperor
web02
dj
dj
dj
dj
web03
dj
dj
dj
dj
db01
db02

For *Youth* we wanted to:

- **Make more efficient usage of resources**
- **Automate recovery of downed servers**
- **Make it easier to deploy code**

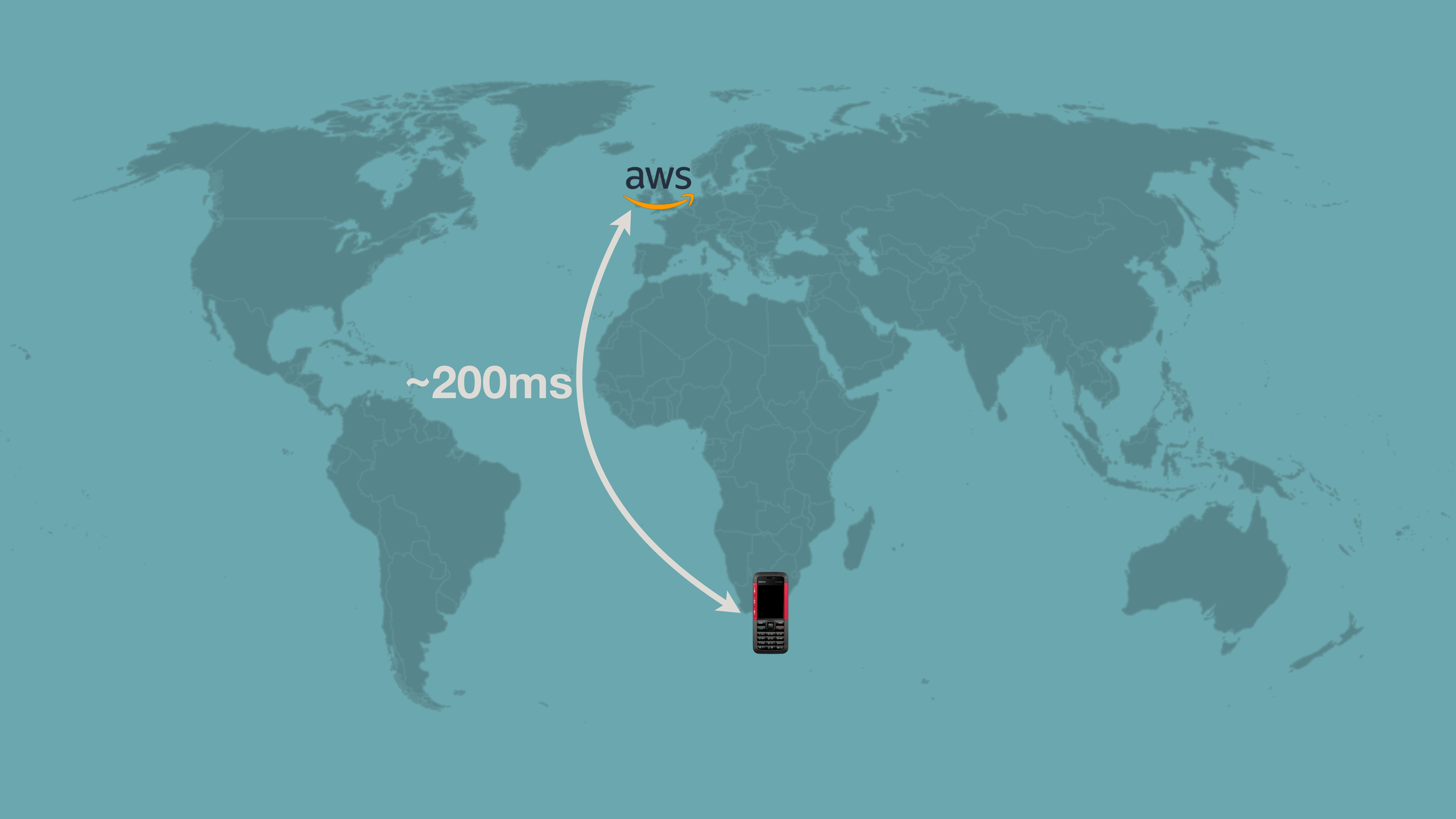
Health

2. Running apps closer to their users

- Messaging (SMS, USSD, WhatsApp)
- Maternal health & ECD

Vumi messaging platform

- Tools to integrate with carriers, aggregators
- SMS & USSD (“Star Menu”)
- Develop message flows in a web UI or JavaScript
- Fancy message store based on Riak in AWS in Ireland


aws
~200ms

Ping times to AWS data centres from Johannesburg

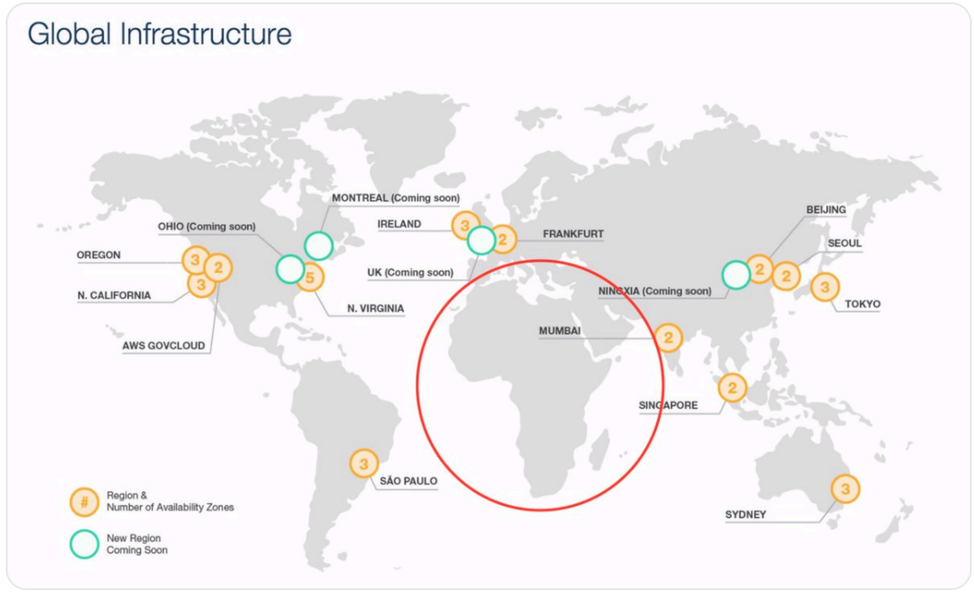
www.cloudping.info

Region	Latency
US-East (Virginia)	260 ms
US East (Ohio)	282 ms
US-West (California)	489 ms
US-West (Oregon)	316 ms
Canada (Central)	306 ms
Europe (Ireland)	195 ms
Europe (London)	180 ms
Europe (Frankfurt)	179 ms
Europe (Paris)	197 ms
Asia Pacific (Mumbai)	377 ms
Asia Pacific (Osaka-Local)	490 ms
Asia Pacific (Seoul)	618 ms
Asia Pacific (Singapore)	1024 ms
Asia Pacific (Sydney)	472 ms
Asia Pacific (Tokyo)	424 ms
South America (São Paulo)	469 ms
China (Beijing)	416 ms
China (Ningxia)	608 ms
AWS GovCloud (US)	591 ms

Richard Stanley
@datarichness

Follow

Hi @awscloud, just thought I'd help you figure out where to put your next CDN and availability zone.



6:24 PM - 24 Sep 2016

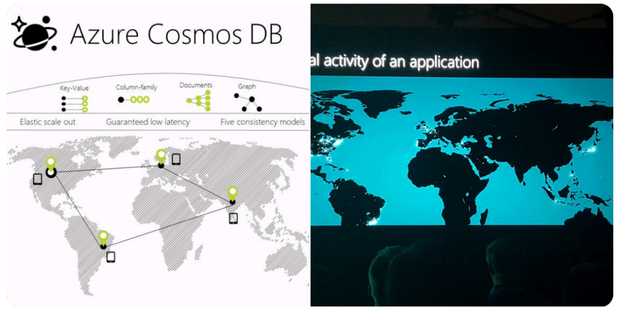
130 Retweets **130** Likes

5 130 130

Simon de Haan
@smn

Following

MS talking about Cosmos DB with global connectivity and low latency to where users are. Fascinating but ... 🤔 Africa? #VelocityConf



7:46 PM - 22 Jun 2017

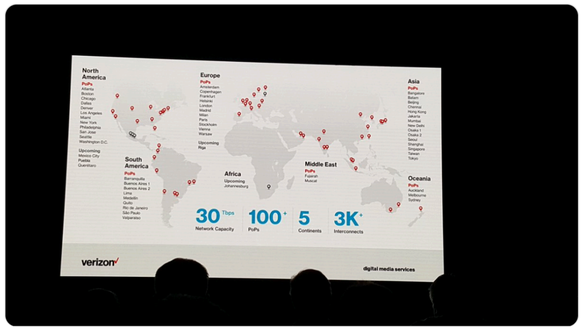
5 Retweets **8** Likes

1 5 8

Simon de Haan
@smn

Following

👋 Africa, "upcoming" ... #VelocityConf



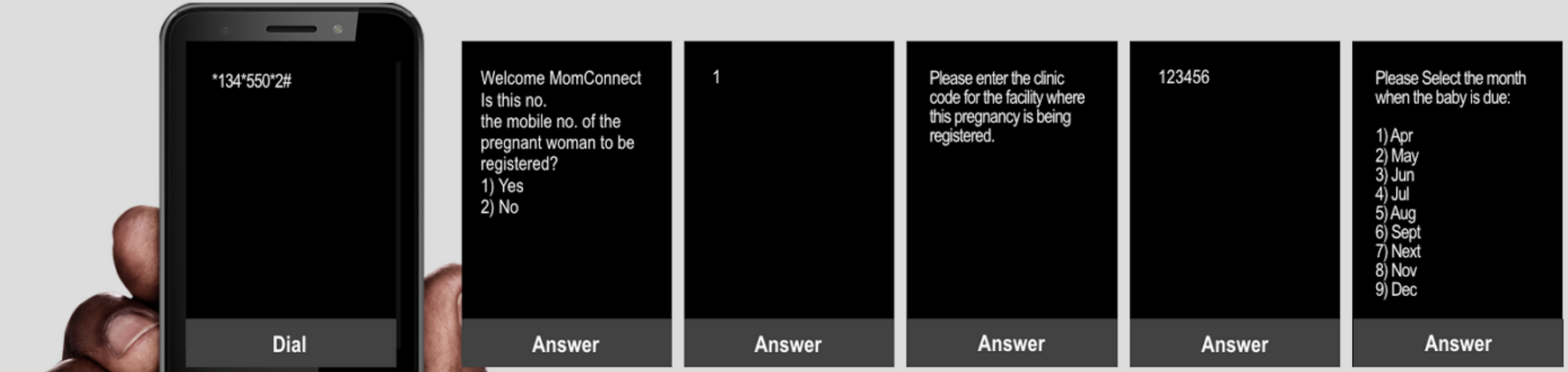
7:22 PM - 22 Jun 2017

2 Retweets **1** Like

2 2 1

Unstructured Supplementary Service Data (USSD)

- Session-based and latency sensitive
- 180s max duration, typically billed per 20s



bit.ly/USSDSMS

For *Health* we wanted to:

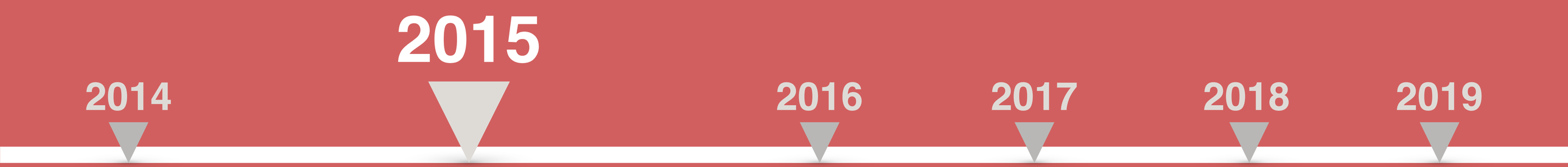
- **Host more services closer to users (lower latency)**
- **Keep data in-country (as part of national health system)**
- **Scale up our platform to support more users**

02

Containers

And their orchestration

Timeline



2015 at Praekelt.org

***Youth:* Free Basics**

- Launched in many countries simultaneously
- Incubator with 100 new sites

internet.org by facebook

***Health:* MomConnect**

- Services in SA taking off
- POPI legislation

momconnect

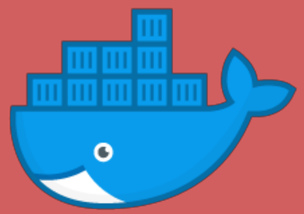
2015 in Cloud Native

Standardisation

- Kubernetes reaches version 1.0 along with formation of CNCF
- Docker at version 1.4-1.9, Open Container Project (eventually *OCI*) formed

kubernetes

CLOUD NATIVE
COMPUTING FOUNDATION



docker

OPEN CONTAINER INITIATIVE

We chose Mesos/Marathon

- **“Simpler” than Kubernetes**
  - Fewer upfront decisions
  - Fewer independent components
  - No buy-in to networking model necessary
  - Marathon app = run *n* instances of container image *x*
- **Emphasis on things we wanted**
  - Resource constraints the default
  - High-availability

```
{
  "id": "/tuneme",
  "instances": 2,
  "cpus": 0.1,
  "mem": 256,
  "container": {
    "type": "DOCKER",
    "docker": {"image": "praekeltfoundation/molo-tuneme:42f355e"},
    "portMappings": [{"containerPort": 8000}]
  },
  "env": {
    "DATABASE_URL": "postgres://tuneme:secret@prd-shared-db01/tuneme",
    ...
  },
  "labels": {
    "HAPROXY_0_VHOST": "tuneme.org,www.tuneme.org",
    "HAPROXY_GROUP": "external"
  }
}
```

03

Seed

In-country maternal health platform

Timeline

What is a Seed project?

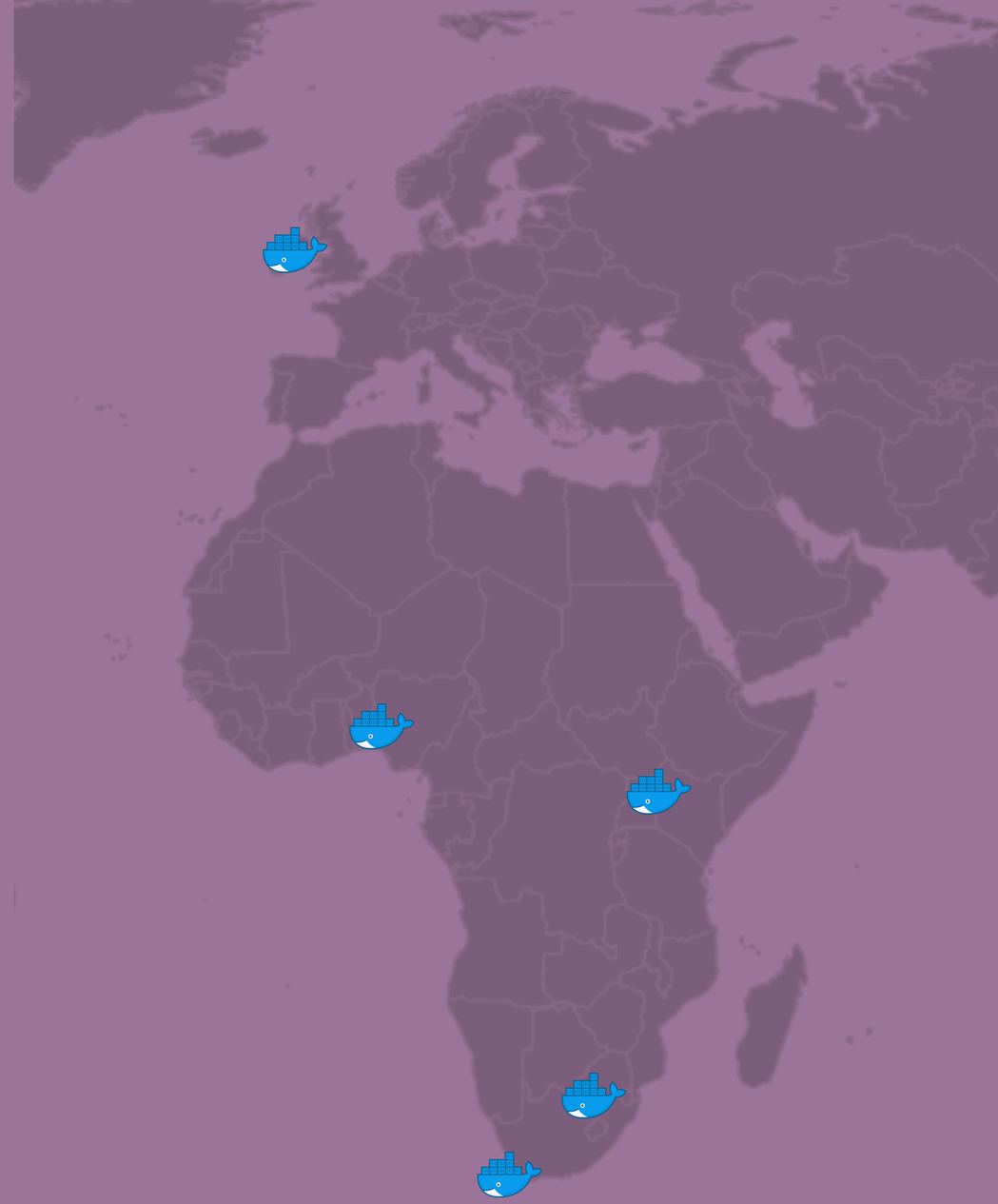
- **Government** endorsement
- **Multi-stakeholder** consortium
- Using **open source** technologies
- With room and budget for **co-design**
- Using **feedback loops** to improve service delivery
- Integrated into **national information systems**
- Has potential for a **national footprint within a year**
- With an explicit view to **handover within a year** of implementation

Container orchestration

Hoped that container orchestration would help because…

- **High level of automation => high level of self-sufficiency?**
- **Able to support a “national-scale” platform**
- **Common platform between different countries**

Container portability



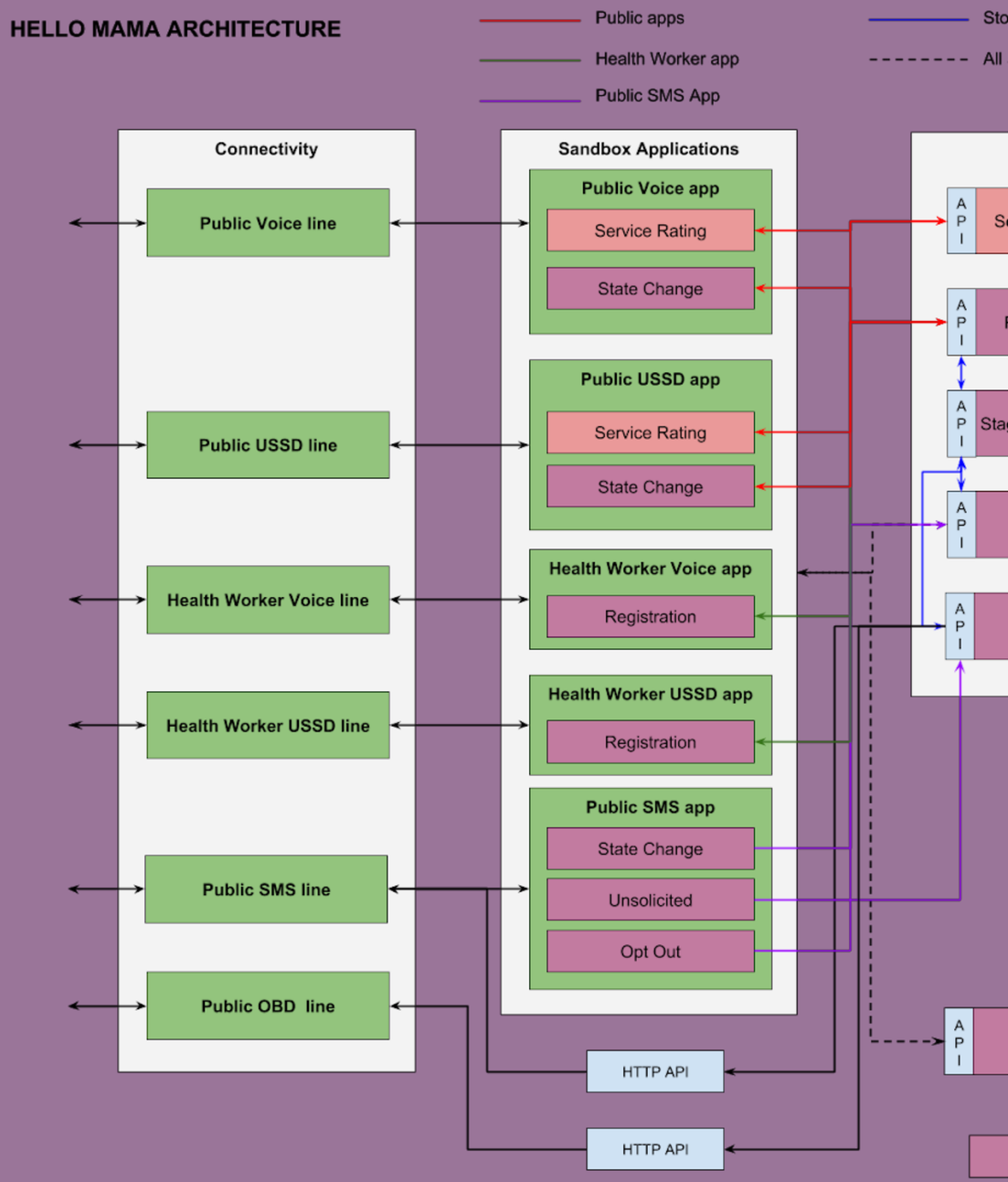
Containers allowed us to…

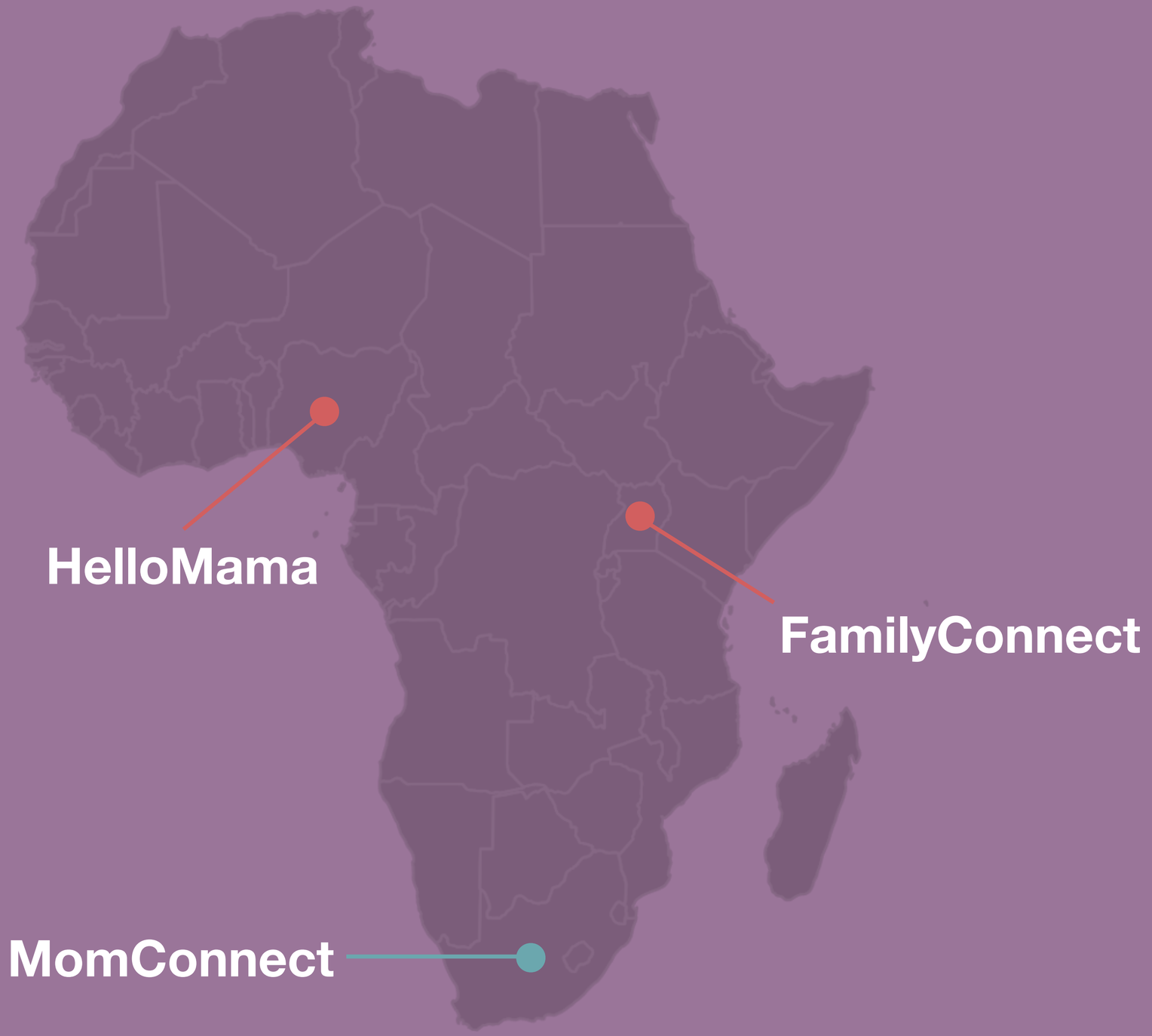
- Get an MVP running in a new country quickly
- Migrate easily between hosting providers
- Treat different hosting environments as the same/similar

In retrospect Seed was hugely ambitious

- Microservices
- In-country hosting
- Container orchestration

Spent too many “innovation tokens”




HelloMama
FamilyConnect
MomConnect

High-availability for what?

In Nigeria…

- One public IP accessible from one host
- Frequent network outages
- Persistent storage issues (errors lead to RO-filesystems)
- Windows-only remote desktop connection
- System clocks changing underneath VMs

High-availability for what?

“The cluster service was rebooted to bring up the highly available VMs.”

High-availability for what?

In Uganda…

- **Physical servers hosted by partner in office**
- **Starts with 2 hosts**
- **Dial-up-like internet speeds**
  - (Please try make your container images small)
- **Servers eventually moved to Gov. datacenter & 3rd added**

Timeline

Peak Mesos

- **Johannesburg Mesos/Marathon cluster peaks at 30 nodes**
  - ~255GB RAM, 120 cores, ~900 containers
  - Bare-metal. VMs on self-managed XenServer
- **Move to OSS Mesosphere DC/OS**
  - Like a *distribution* for Mesos, with lots of extras, more stability
- **Team of 4 SREs managing clusters in 4 countries**

The Nigerian & Ugandan platforms have now been handed over to local partners

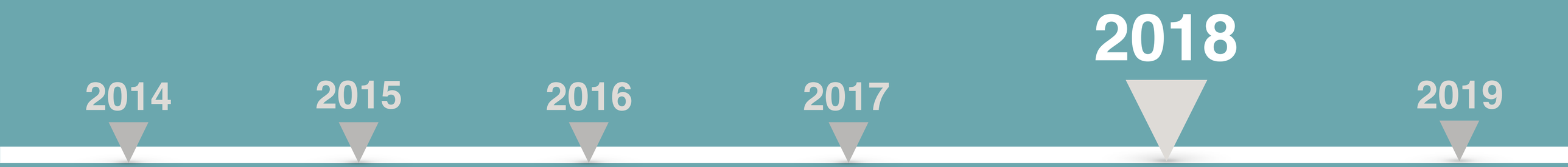
bit.ly/SeedRetro

04

Lessons learned

Reflecting on Seed

Timeline



Infrastructure for handovers

Did we do the best we could have?

- Over-estimated scale of projects
- Common platform benefitted *us*, but did it benefit those inheriting it?
- When you have a container orchestrator-shaped hammer, everything looks like a nail

Infrastructure for handovers

What would the ideal system for handing over look like?

- **Container orchestration?** ***Possibly not.***
- **Distributed system or an old-school web server?**
- **Can we get portability without container orchestration?**
- **How much are we willing to give up?**

Co-designing infrastructure

- **Historically only did co-design with end-users**

If we're developing infrastructure to *hand over* to others…

…then the inheritors of that infrastructure are *also* end-users.

- **Shouldn't dictate what technology others must use without their input**

GlobalMoms

05
Kubernetes, Spinnaker, & beyond

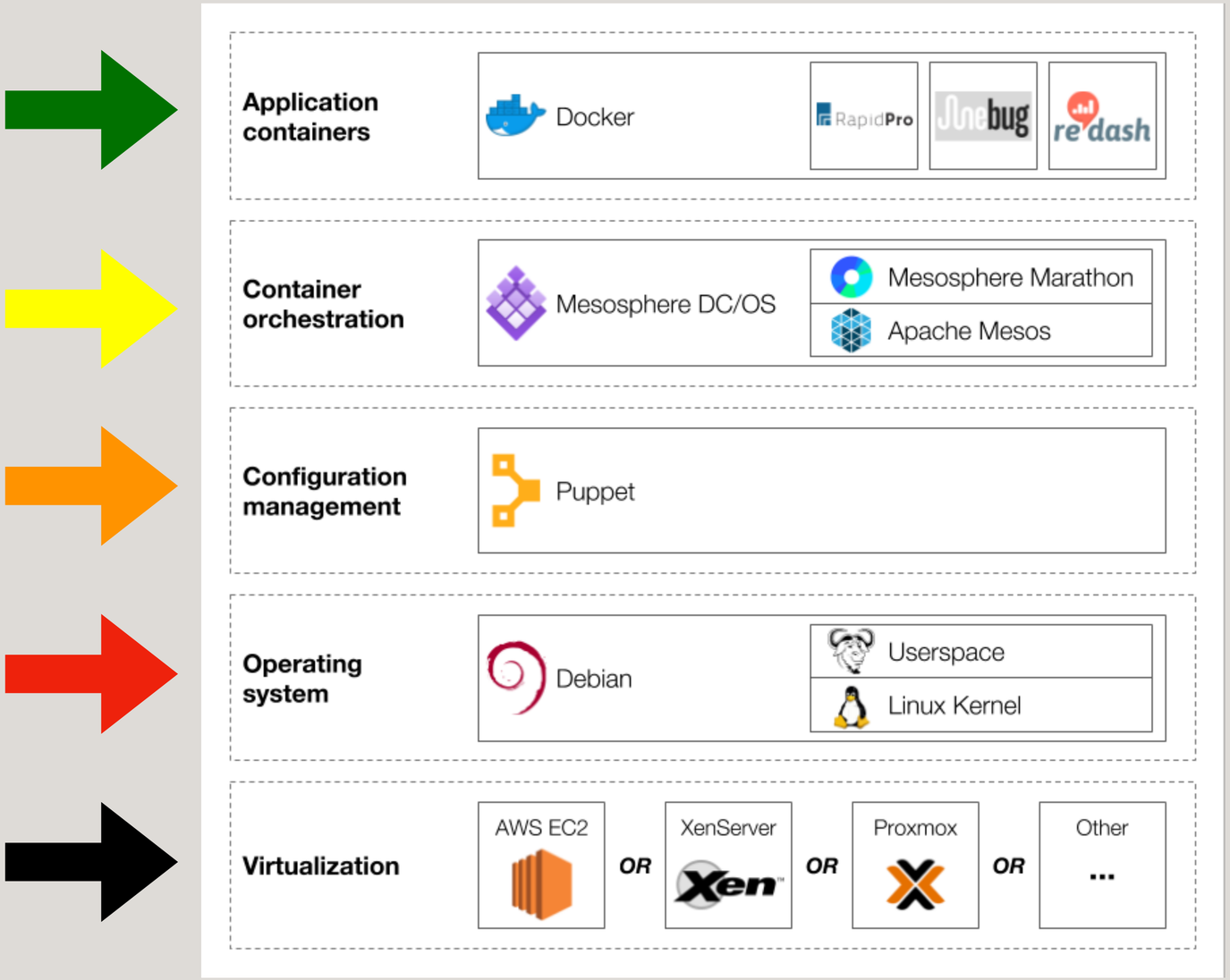
Looking forward…

Timeline

2014 2015 2016 2017 2018 **2019+**

When you have to manage every layer…

…you can't afford to add more of them



Where we can use cloud…

And where we can’t…

Kubernetes

- **Increasingly hard to argue that it's not the *de-facto* standard**
- **The killer feature is the community & ecosystem**
  - Global & local (South African) community
  - Building things we wouldn't need to if we used Kubernetes
- **Docker images "the price of admission to modern platforms such as Kubernetes" — *paid***

Building too much stuff for Mesos

Load-balancing

HTTPS certificates

Secrets & secure introduction

Persistent storage

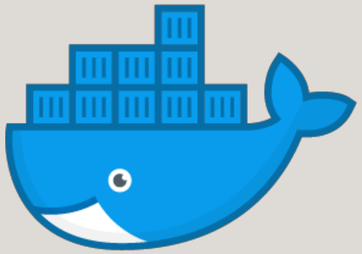
Config file management

…

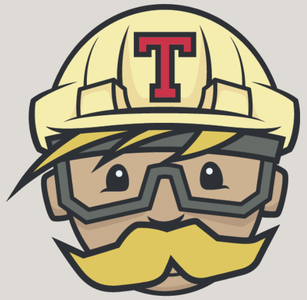
Kubernetes

- **Still more complicated than we'd like**
  - Many technology decisions to make
  - Moves very fast
- **No *de-facto* "distribution" yet**
  - Waiting for *"the Ubuntu of distributions"* to (possibly) use in not-the-Cloud
- **Strategy:**
  - Use managed Cloud services where we can
  - Use the simplest everything (no service meshes for us)

GitHub



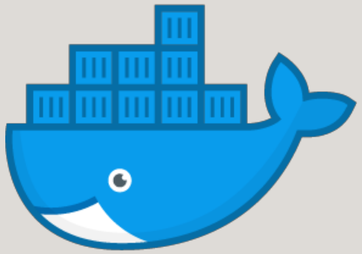
Docker Hub



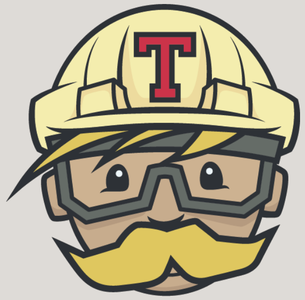
Travis CI

Mesosphere DC/OS

GitHub



Docker Hub



Travis CI

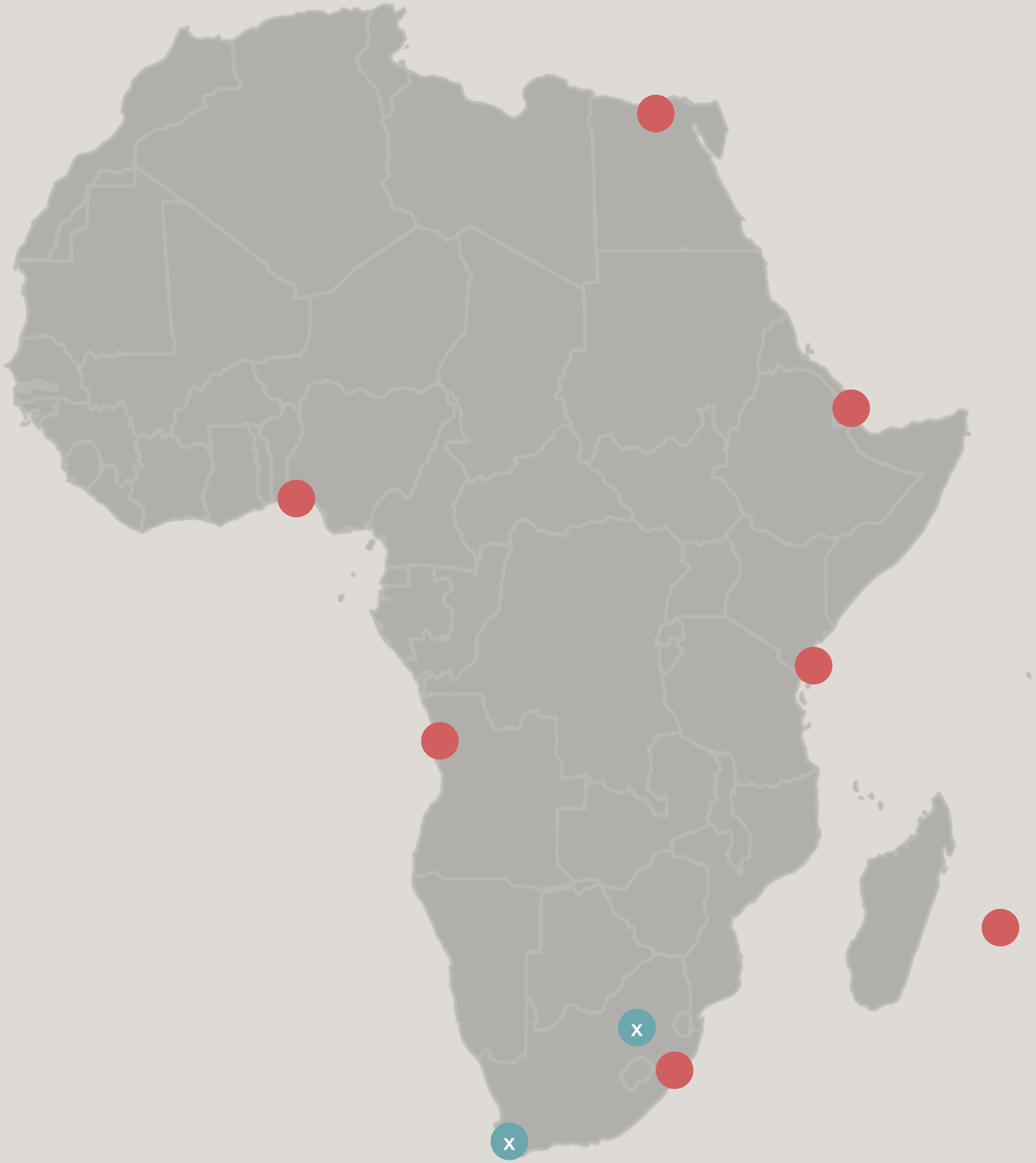
Mesosphere DC/OS

Kubernetes

More cloud coming to Africa

- Cloud datacenter *(TBC)*
- Edge or CDN PoP

(Azure, AWS, Google, Cloudflare, Fastly)



Thank you.

@jayhewland
jamie@praekelt.org
@praekeltorg

Want to read more about this?
medium.com/mobileforgood

Special thanks to the Linux Foundation

PRAEKELT·ORG