Mesosphere DCOS

Jan Repnak
jan.repnak@mesosphere.io

Agenda

- Containers
- Apache Mesos
- Marathon
- Datacenter Operating System (DCOS)
- Hands-On

LINUX CONTAINERS

Linux Containers

The why and the what

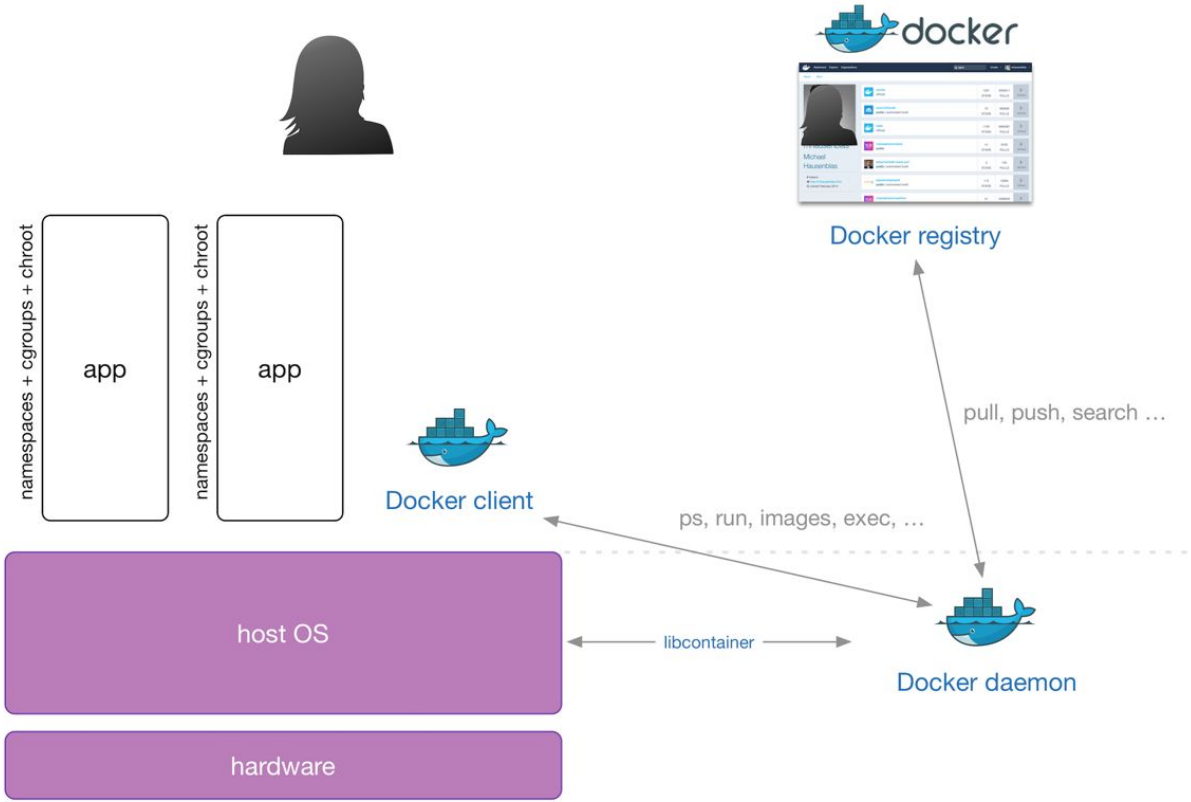
- Containers are not VMs …
  - … they are process groups for app-level dependency management
  - … they are lightweight (startup time, footprint, average runtime)
- security considerations
- pets vs cattle

Linux Containers

- **namespaces**
  - Isolate PIDs between processes
  - Isolate network resources (stacks, devices, etc.)
  - Isolate hostname/NIS (UTS)
  - Isolate filesystem mount (chroot)
  - Isolate inter process communication (IPC)
  - Isolate users/groups

- **cgroups**
  - https://sysadmincasts.com/episodes/14-introduction-to-linux-control-groups-cgroups

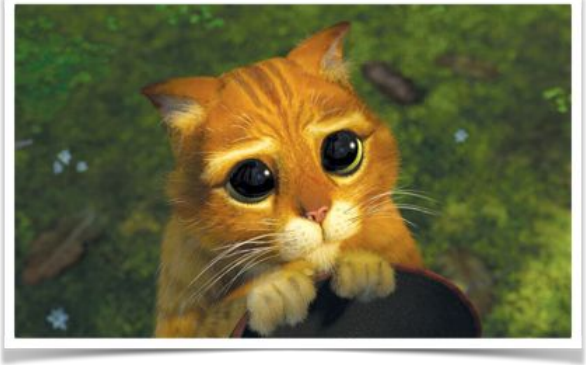
Docker



Pets vs Cattle

What Is This All About?

- **Pets:**
  treat machines as individuals that you give names and when they get ill you nurse them back to health.



- **Cattle:**
  anonymous, identical to each other; you assign numbers and when they get ill you get rid of it.

http://www.theregister.co.uk/2013/03/18/servers_pets_or_cattle_cern/

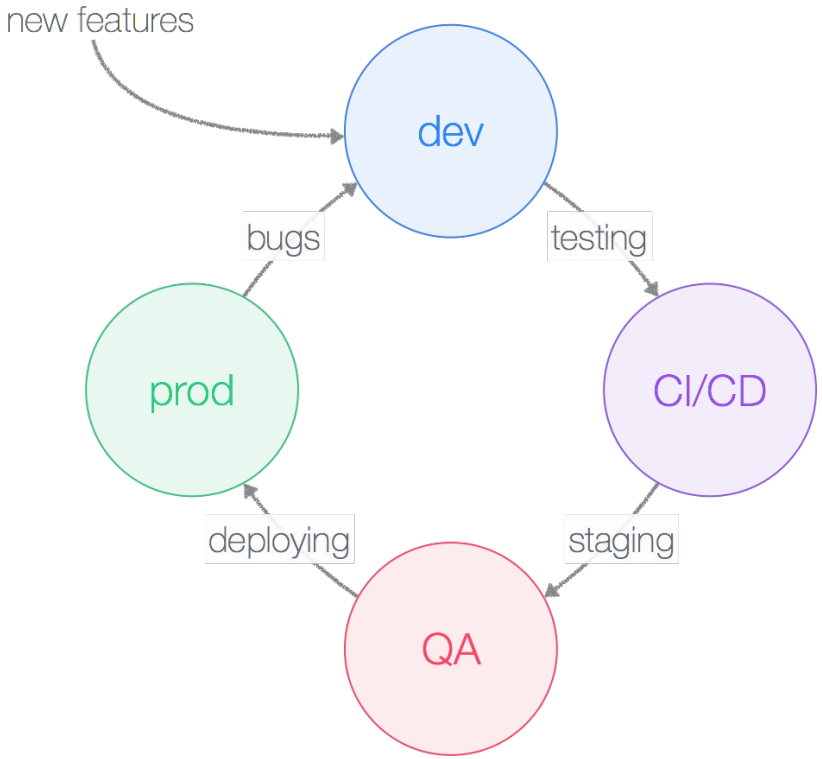
Consequences of going all-in with Cattle approach

- scale out on commodity hardware
- elasticity
- 'cheap' & 'simple'
- R U on pager duty? Just sleep through!

- social ≫ technology challenge
- new technical challenges such as service discovery

http://www.theregister.co.uk/2013/03/18/servers_pets_or_cattle_cern/

Container Life Cycle



Apache Mesos

MESOSPHERE HISTORY

2009

Apache Mesos built at UC Berkeley

- Core technology from AMPlab
- Corporate large-scale production deployments begin (e.g., Apple, Twitter, Salesforce)

2013

Mesosphere Founded

Key engineering leaders from Twitter, Airbnb - companies behind open-source tech

2014

First Mesosphere DCOS lighthouse customers

Tens of thousands of containers launched

Mesosphere is well funded

$50M by Tier 1 investors:
Andreessen Horowitz and Khosla Ventures

2015

Growing Ecosystem & DCOS General Availability

Now a breeze to install modern app services (e.g., Hadoop, Spark, Cassandra)

Expanded Operations

San Francisco (2013)
Hamburg (2014)
New York (2015)

WORKLOADS*

batch streaming PaaS

CHRONOS

Spark

STORM

kubernetes

MapReduce

hadoop

Flink

samza

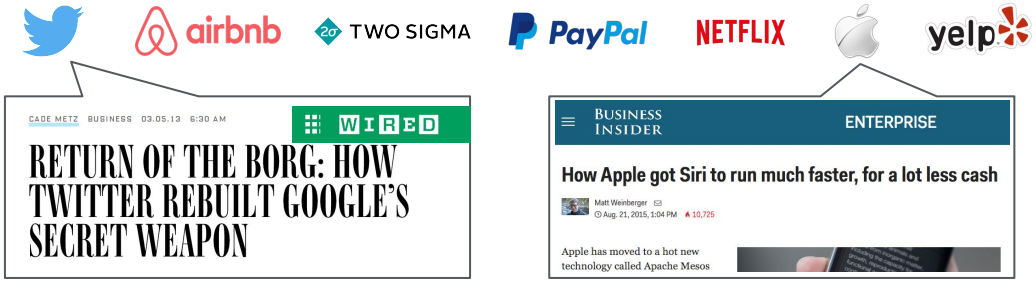
MARATHON

*) kudos to Timothy St. Clair, @timothysc

MESOS KERNEL APPLIES LESSONS FROM EARLY INNOVATORS

Production-proven Web Scale Cluster Managers

Borg/Omega	Tupperware/Bistro	Apache Mesos
~2001	*~2007*	*2010+*
Proprietary	Proprietary	Open Source (Apache license)



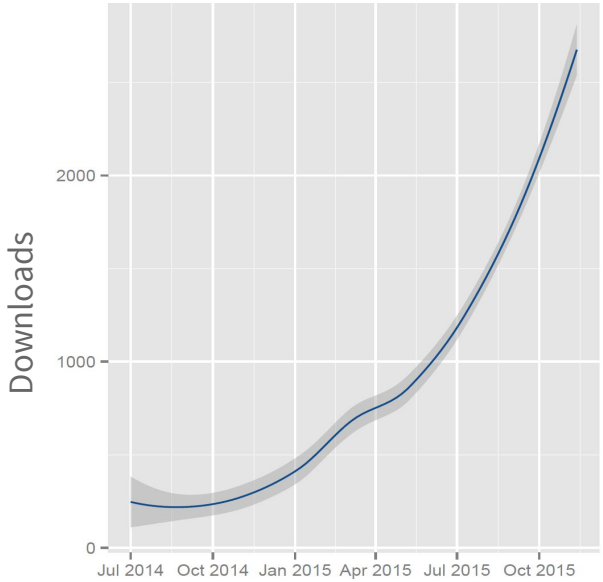
- Built at UC Berkeley AMPLab by **Ben Hindman** (Mesosphere Co-founder)
- Built in collaboration with Google to overcome some Borg Challenges
- Production proven at scale +80K hosts @ Twitter

MESOS IS THE IDEAL KERNEL FOR POWERING A DCOS

Designed to be flexible

- **Aggregates all resources** in the datacenter for modern apps
- **Intentionally simple** to enable massive scalability
- **Handles different types of tasks** - long running, batch, and real-time
- **Two-level scheduler architecture** enables multiple scheduling logics (a key challenge at Google)
- **Extensible** to work with new technologies

Gaining massive adoption



Mesos daily downloads, July 2014 - November 2015

airbnb
NETFLIX
PayPal
yelp
TWO SIGMA

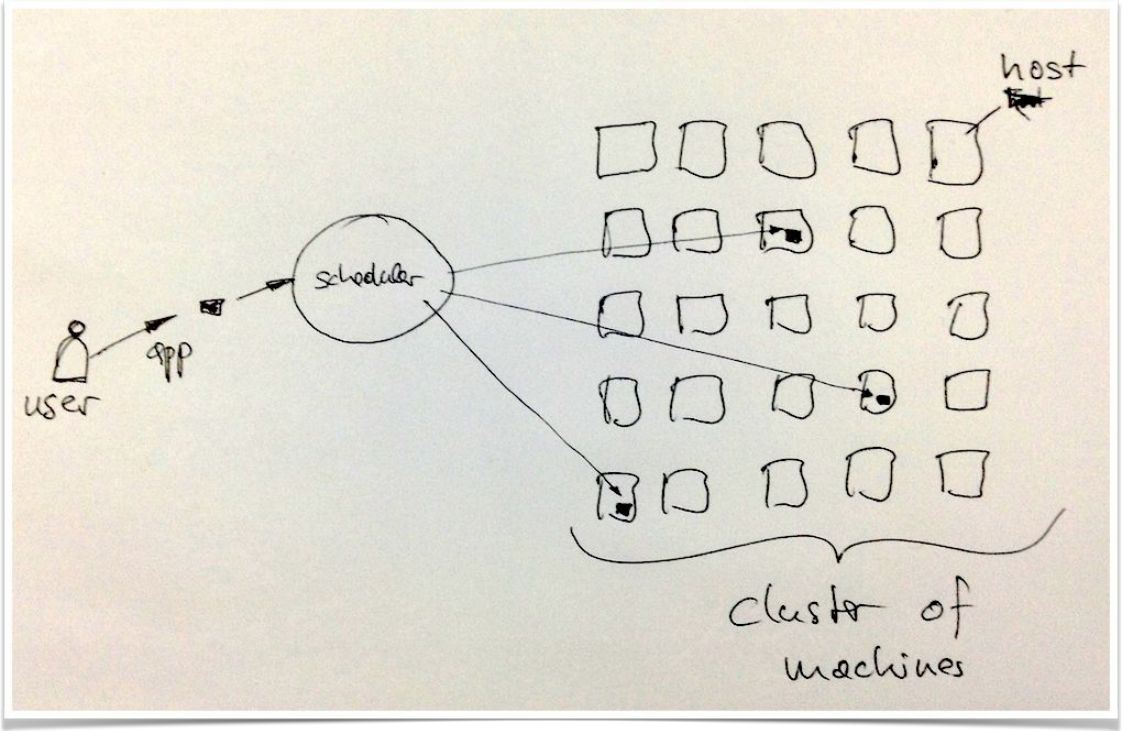
Pets vs Cattle

What Is This All About?

- A top-level ASF project
- A cluster resource negotiator
- Scalable to 10,000s of nodes but also useful for a handful of nodes
- Fault-tolerant, battle-tested
- An SDK for distributed apps
- Native Docker support

17

What does a scheduler do?



Resources

- resource == anything a task/executor consumes in order to do their work
- standard resources: cpu, mem, disk, ports
- at its core: DRF algorithm, for fair sharing across-resource types

Marathon

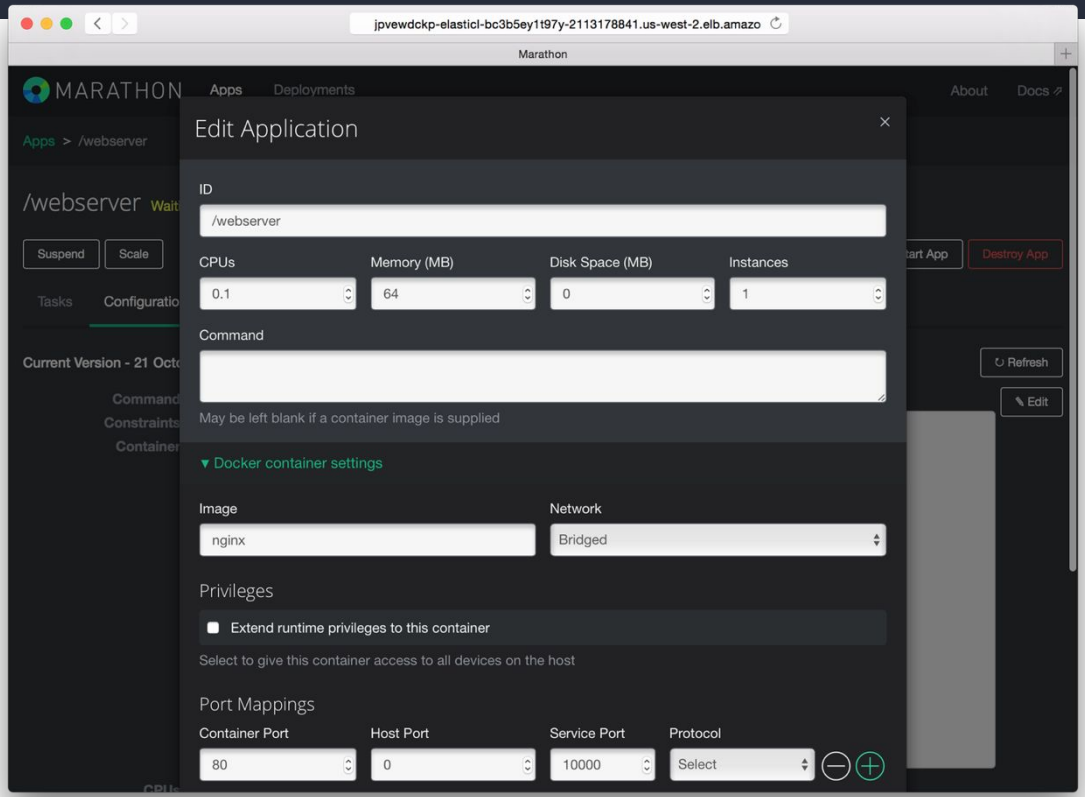
Marathon

- **An init System for datacenters**
  - starts instances of a long-running service somewhere in the cluster, for example, as Docker containers
  - restarts the instances if they crash
  - provides composition primitives
  - supports health checks
  - supports rolling upgrades

Marathon

- **Basics**
  - apps and groups
  - health checks
- **HTTP API**
  - curl, http
  - DCOS CLI
- **Team Player**
  - Integrates nicely into the DCOS ecosystem
  - Doesn’t try to solve everything itself

Marathon



Marathon

```
{
  "id": "webserver",
  "cmd": "python3 -m http.server 8080",
  "cpus": 0.5,
  "mem": 32.0,
  "container": {
    "type": "DOCKER",
    "docker": {
      "image": "python:3",
      "network": "BRIDGE",
      "portMappings": [
        { "containerPort": 8080, "hostPort": 0 }
      ]
    }
  },
  "acceptedResourceRoles": [
      "slave_public"
  ],
  "constraints": [
      [
          "hostname",
          "UNIQUE"
      ]
  ]
}
```

Service Discovery

Name	Consistency	Language	Registration	Lookup
ZooKeeper	strong	Java	client	bespoke clients
etcd	strong	Go	sidekick+client	HTTP API
Consul	strong	Go	automatic and through traefik (Consul backend)	DNS + HTTP/JSON API
Mesos-DNS	strong	Go	automatic and through traefik (Marathon back-end)	DNS + HTTP/JSON API
SkyDNS	strong	Go	client registration	DNS
WeaveDNS	strong	Go	auto	DNS
SmartStack	strong	Java	client registration	automatic through HAProxy config
Eureka	eventual	Java	client registration	bespoke clients

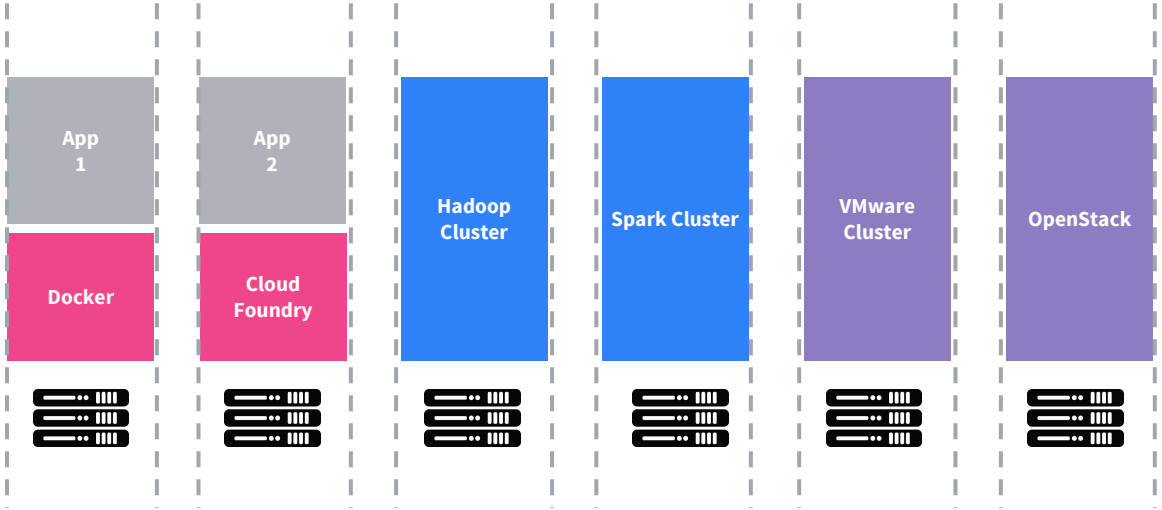
Datacenter Operating System (DCOS)

THE DATACENTER IS THE NEW SERVER

NEW FORM FACTOR FOR DEVELOPING AND RUNNING APPS

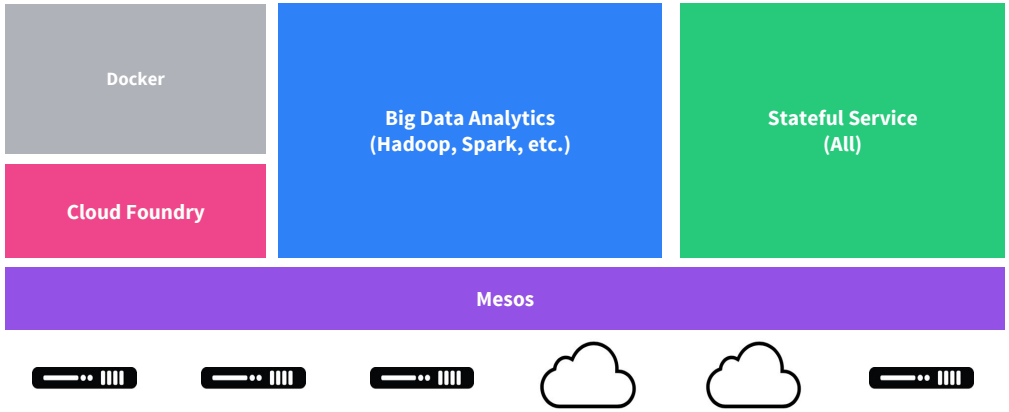
UNIT OF INTERACTION	PARTITION (LPAR)	SERVER	VIRTUAL MACHINE	DATACENTER
				
	MAINFRAME	**PHYSICAL (x86)**	**VIRTUAL**	**HYPERSCALE**
DEFINITIVE APPS AND OS	• DATA / TRANSACTION PROCESSING • UNIX, IBM OS/360	• ERP, CRM, PRODUCTIVITY, MAIL & WEB SERVER • LINUX, WINDOWS	• ERP, CRM, PRODUCTIVITY, MAIL & WEB SERVER • HYPERVISOR + GUEST OS	• BIG DATA, INTERNET OF THINGS, MOBILE APPS • **DATACENTER OPERATING SYSTEM (DCOS)**

TRADITIONAL DATACENTER



- Many “snowflakes”
- Management nightmare
- Lengthy cycles to deploy code
- Low utilization

MODERN DATACENTER



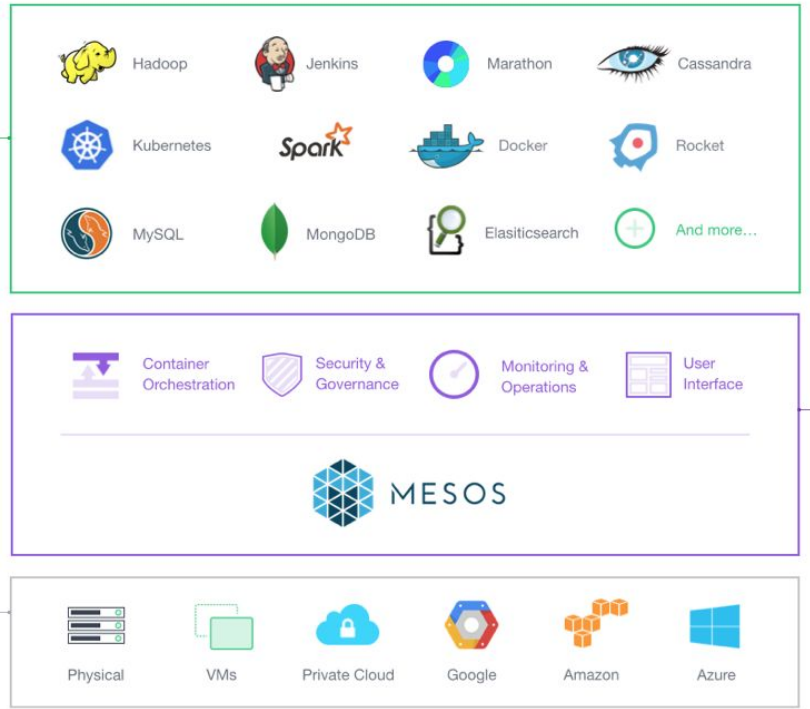
Deploys on-premise in cloud or both

- High performance and resource isolation
- Easy scalability and multi-tenancy
- Fault-tolerant and highly available
- Highly efficient with highest utilization
- Complete workload portability

Meet the Datacenter Operating System

Any Service or Container

Your favorite services, container formats, and those yet to come



Mesosphere DCOS

Runs distributed apps anywhere as simply as running apps on your laptop

Any Infrastructure

Build apps once on DCOS, and run it anywhere

DCOS is a Distributed Operating System

- kernel == Apache Mesos, scaling to 10,000 of nodes
- fault-tolerant in all components, rolling upgrades throughout
- containers first class citizens (LXC, Docker)
- local OS per node (container enabled)
- scheduling (long-lived, batch)
- service discovery, monitoring, logging, debugging

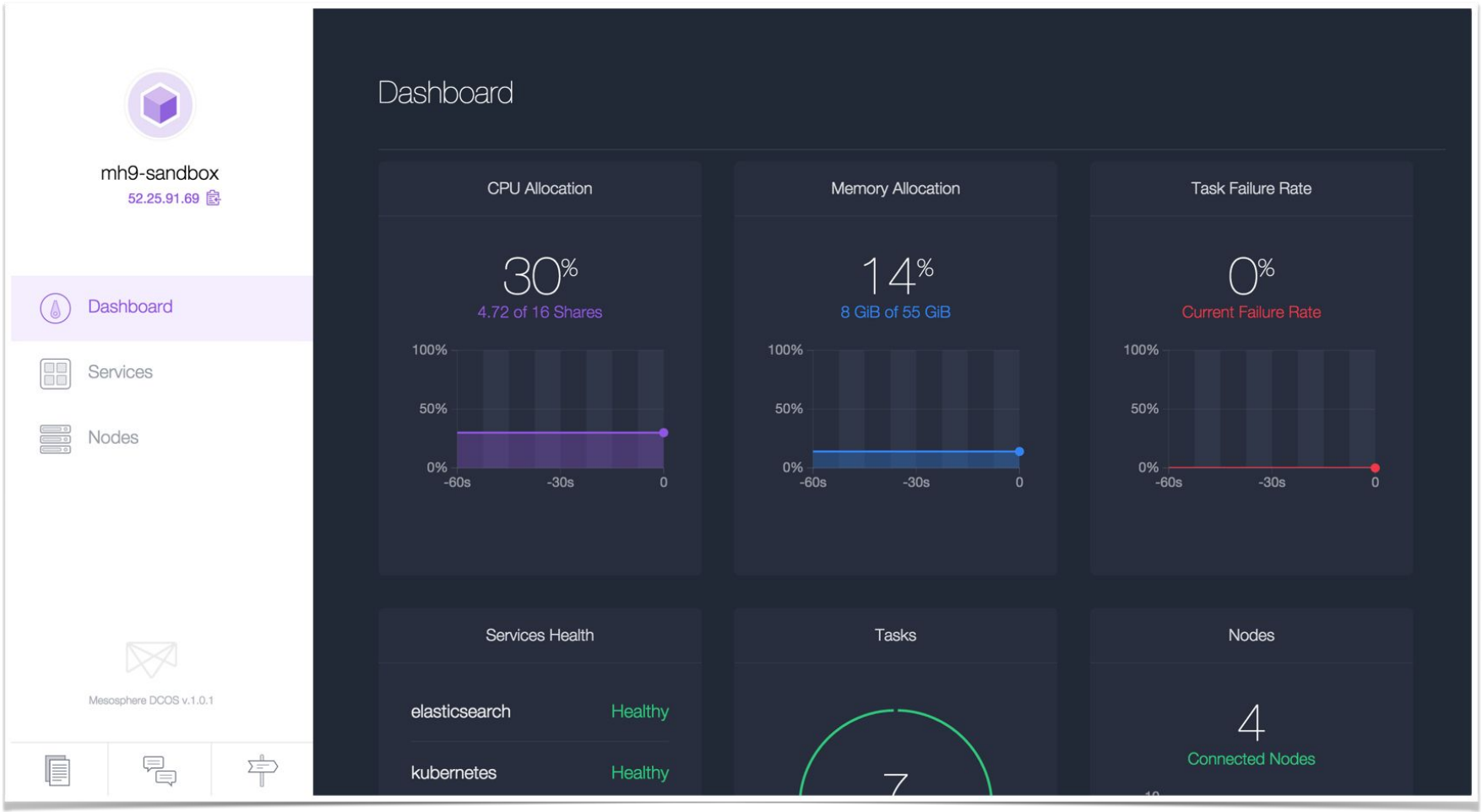
Local OS vs. Distributed OS

subsystem	function	local OS	datacenter OS
kernel	*central component of an OS abstracting hardware and managing OS services*	Linux kernel	Mesos
(short-term) dispatcher scheduler	*a CPU-scheduling and task-switching component of the OS*	dispatcher	Mesos and Mesos frameworks (e.g. Myriad, for YARN)
(long-term) job scheduler	*a long-term, job-oriented scheduling component of the OS*	cron, Control-M	Chronos
launch, bootstrap & config	*the start-up component of an OS used to launch user apps and services*	Sys V init, systemd	Marathon, Aurora, etcd, Zk
main memory	*component responsible for the management of the main memory (RAM)*	Linux VMM	Mesos, Tachyon
filesystem	*abstraction of devices like HDD/SSD to permanently store & retrieve data*	ext4, NTFS, HFS+	HDFS, MapR-FS, GlusterFS
access control	*components responsible for user ID, authentication and authorisation*	user/groups, AD/LDAP, PAM, Kerberos, ACLs	authentication support, MESOS-907, MESOS-910, MESOS-911
networking	*component for network configuration and management*	networkd, netstat	Mesos networking
logging & monitoring	*components to log OS-level and application data as well as to monitor its performance and health*	top, syslog, fluentd, monit	Mesos-DNS; Mesos logging + support for 3rd party performance and health monitoring via MESOS-780
IPC	*mechanism allowing different processes to communication with each other*	Linux IPC such as pipes	No Inter Framework Communications (IFC) as of now
user interface	*interface component allowing user to interact with OS*	bash shell, Windowing systems	Mesos CLI, Mesos dashboard
isolation	*service enabling process/application isolation*	cgroups, Docker	

http://bitly.com/os-vs-dcos

Benefits

- Run stateless services such as Web servers, app servers (via Marathon) and stateful services like Crate, Kafka, HDFS, Cassandra, ArangoDB etc. together on one cluster
- Dynamic partitioning of your cluster, depending on your needs (business requirements)
- Increased utilization (10% → 80% and more)


mh9-sandbox
52.25.91.69
Dashboard
Services
Nodes
Mesosphere DCOS v.1.0.1
Dashboard
CPU Allocation
30%
4.72 of 16 Shares
100%
50%
0%
-60s
-30s
0
Memory Allocation
14%
8 GiB of 55 GiB
100%
50%
0%
-60s
-30s
0
Task Failure Rate
0%
Current Failure Rate
100%
50%
0%
-60s
-30s
0
Services Health
elasticsearch
Healthy
kubernetes
Healthy
Tasks
7
Nodes
4
Connected Nodes

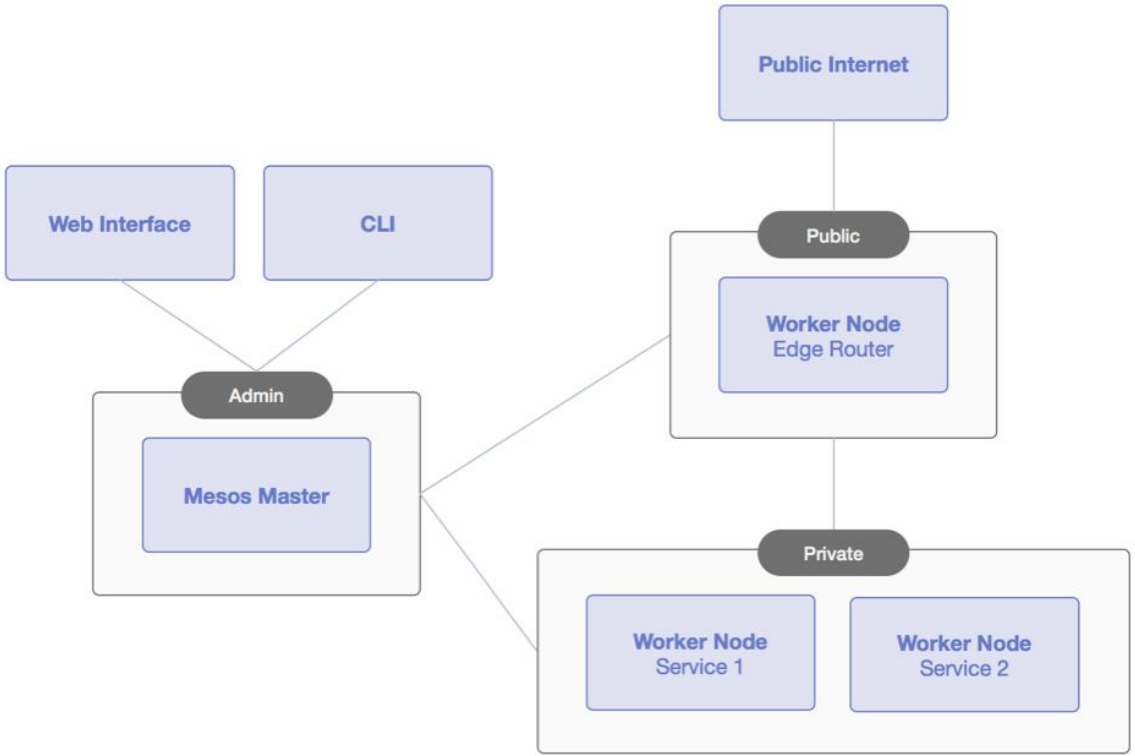
```
~/sandbox/dcos/ccm/dcos $ dcos package list
NAME           VERSION              APP             COMMAND  DESCRIPTION
elasticsearch  0.2.0                /elasticsearch  ---      DCOS implementation of the Mesos-Elasticsearch framework
kubernetes     v1.0.5-v0.6.4-alpha  /kubernetes     ---      Manage a cluster of Linux containers as a single system to accelerate Dev and simplify Ops.
~/sandbox/dcos/ccm/dcos $ dcos help
Command line utility for the Mesosphere Datacenter Operating
System (DCOS). The Mesosphere DCOS is a distributed operating
system built around Apache Mesos. This utility provides tools
for easy management of a DCOS installation.

Available DCOS commands:

       config          Get and set DCOS CLI configuration properties
       help            Display command line usage information
       marathon        Deploy and manage applications on the DCOS
       node            Manage DCOS nodes
       package         Install and manage DCOS packages
       service         Manage DCOS services
       task            Manage DCOS tasks

Get detailed command description with 'dcos <command> --help'.
~/sandbox/dcos/ccm/dcos $ _
```

DCOS Architecture



MICROSOFT AZURE CONTAINER SERVICE (ACS)

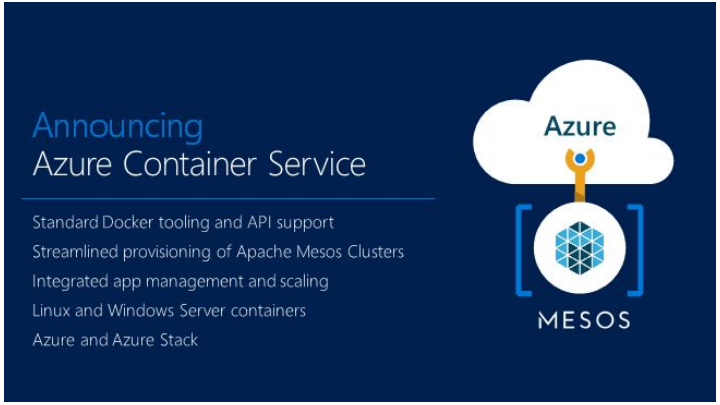
Challenges

- Needed a production grade native container service that would work on premises and on Azure, at massive scale
- Must easily integrate with Azure CI/CD, app management and auto scaling infrastructure
- Microsoft- and Linux-friendly technology

Mesosphere Solution

- After independent evaluation, MS team determined Mesos/Mesosphere was the right fit
- Currently integrating Mesosphere DCOS as the core technology for Azure Container Service

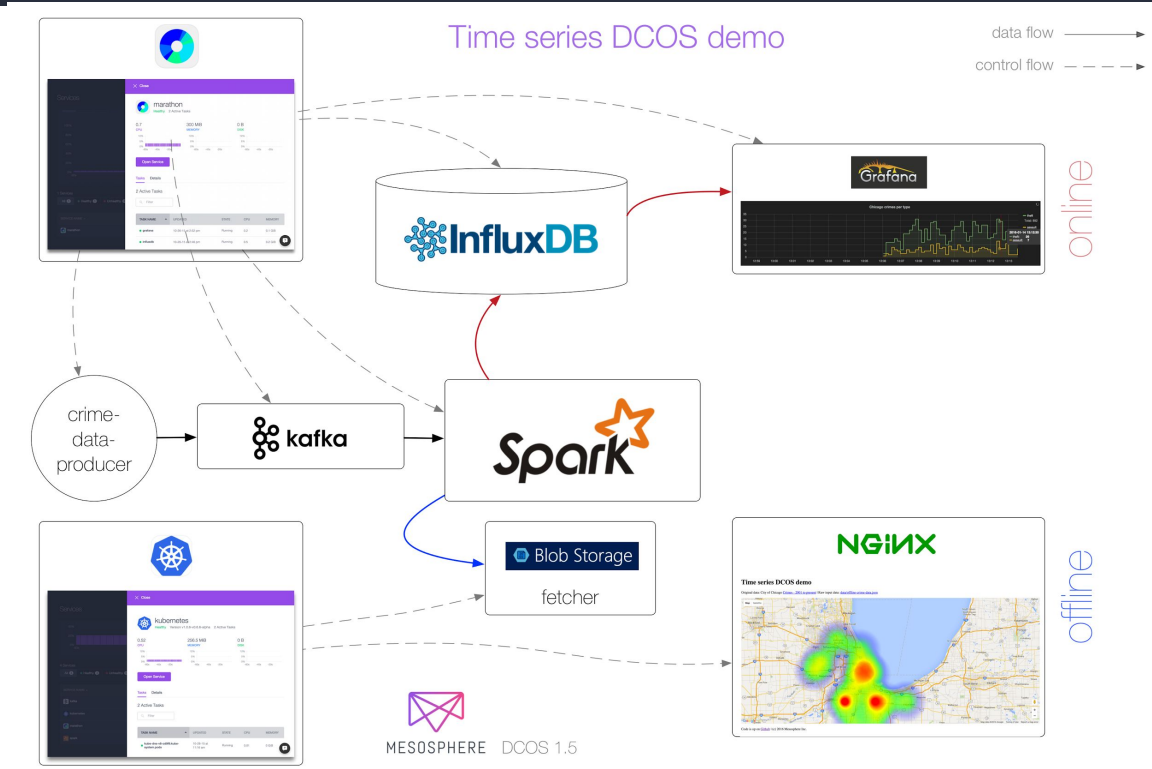
Microsoft



WEBINAR

DCOS ON AZURE WEBINAR
Tomorrow (Jan 26), 10:00 AM

http://bit.ly/dcosonazure



Resources

- **MESOSPHERE**
  https://mesosphere.com
- **DCOS COMMUNITY EDITION**
  https://mesosphere.com/product/
  https://docs.mesosphere.com/getting-started/tutorials/
- **MICROSOFT AZURE CONTAINER SERVICE**
  https://mesosphere.com/blog/2015/09/29/mesosphere-and-mesos-power-the-microsoft-azure-container-service/
- **APACHE MESOS FOR WINDOWS SERVER**
  https://mesosphere.com/blog/2015/08/20/mesos-everywhere-apache-mesos-for-windows-server/

Demo: Oinker

MESOSPHERE