Container Orchestrators and DCOS
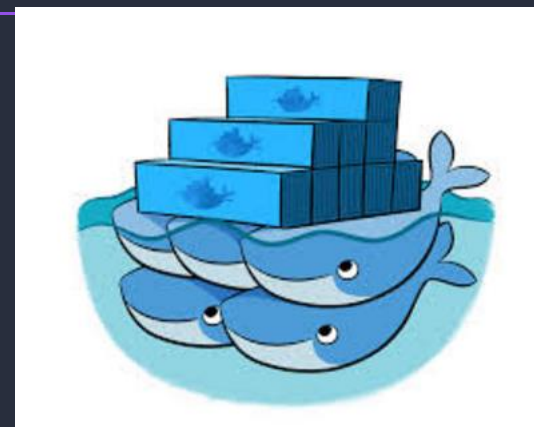
Timothy Chen

About me:

- Previously Mesosphere Lead Engineer, (ex-Microsoft/VMWare)

- Apache Mesos, Drill PMC

- Help maintain Apache Spark on Mesos

- Co-founder on new Container startup (still stealth)

Outline

- The Need of Container Orchestration

- What is DCOS?

- Breaking down DCOS

- Demo

The Open
Container Initiative

is an open governance structure for the
express purpose of creating open industry
standards around container formats and
runtime.

Web Application

Operating System

Hardware

SCALE & HIGH AVAILABILITY

Web App	Web App	Web App
Operating System	Operating System	Operating System
Hardware	Hardware	Hardware
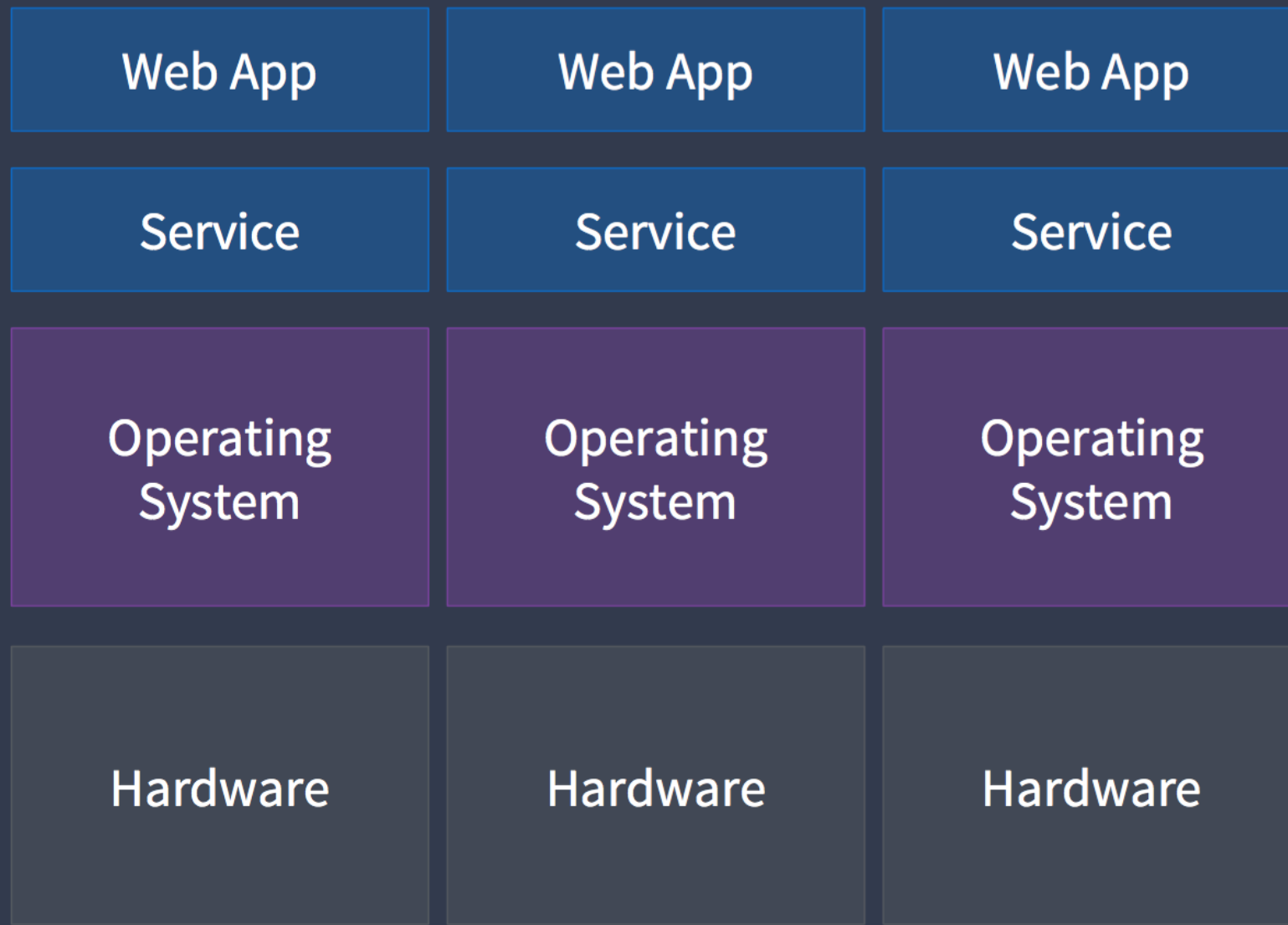
HARDWARE VIRTUALIZATION

Web App	Web App	Web App
Service	Service	Service
Operating System	Operating System	Operating System
Machine	Machine	Machine
Infrastructure		
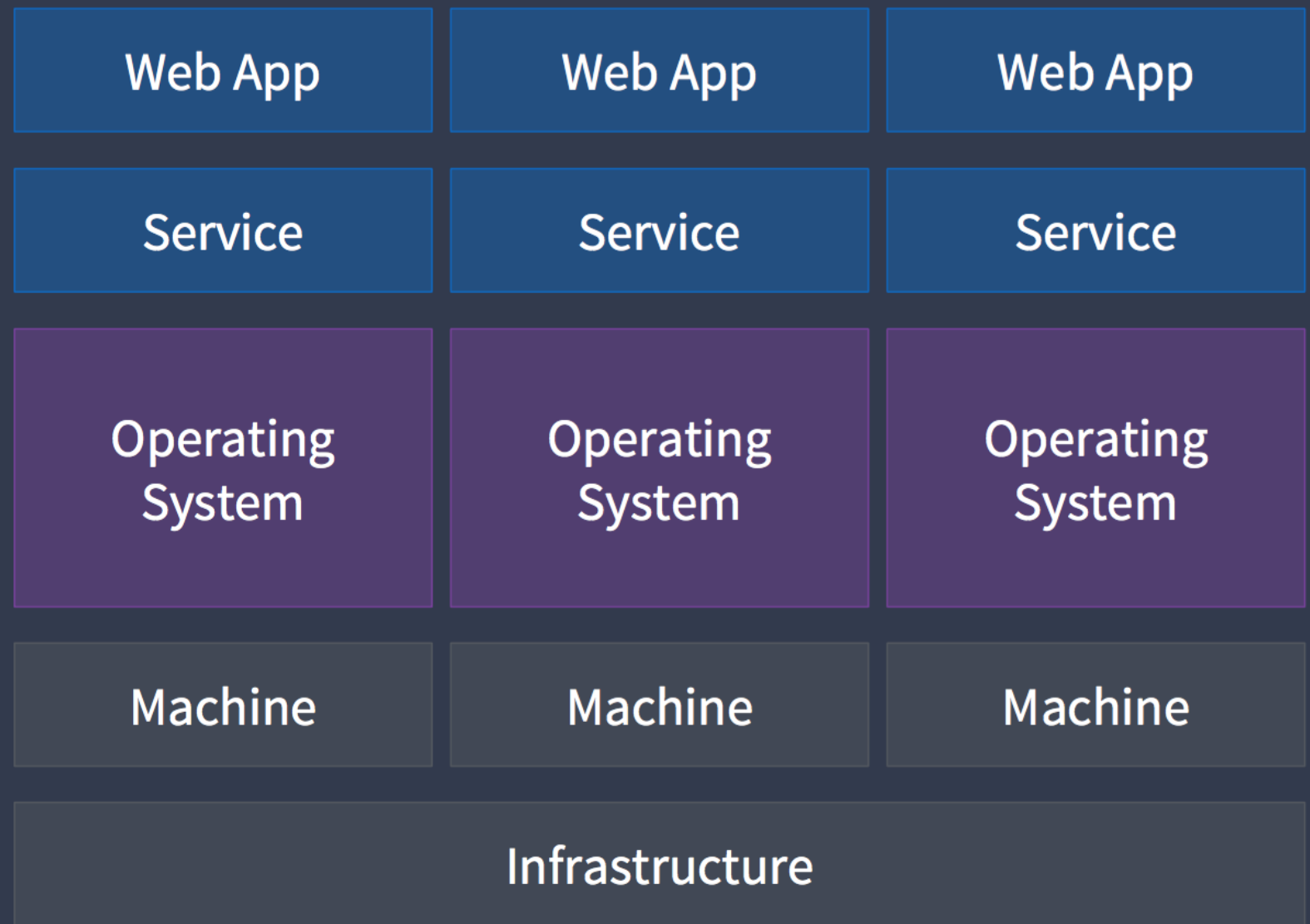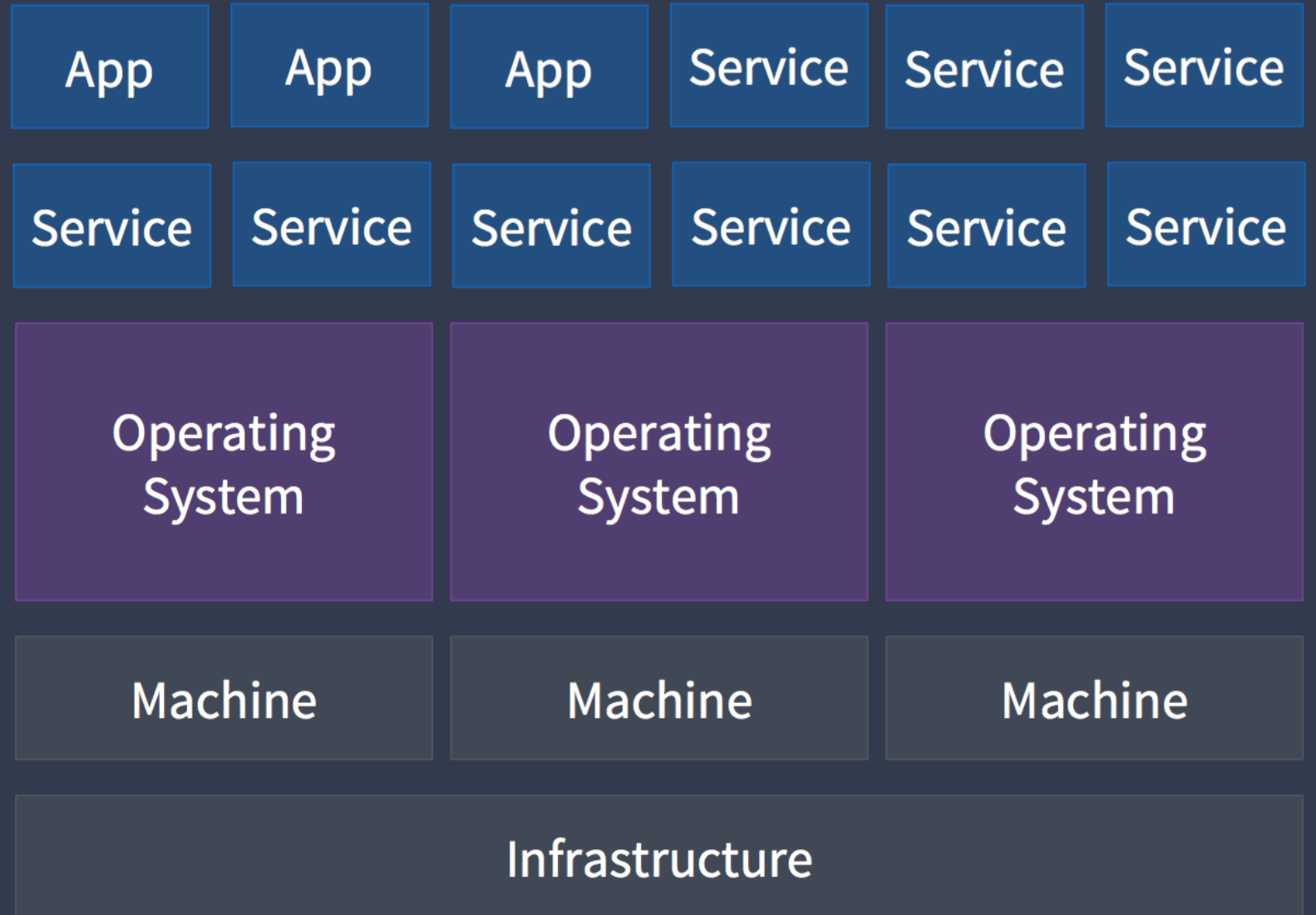
MICROSERVICES

CONTAINERIZED MICROSERVICES
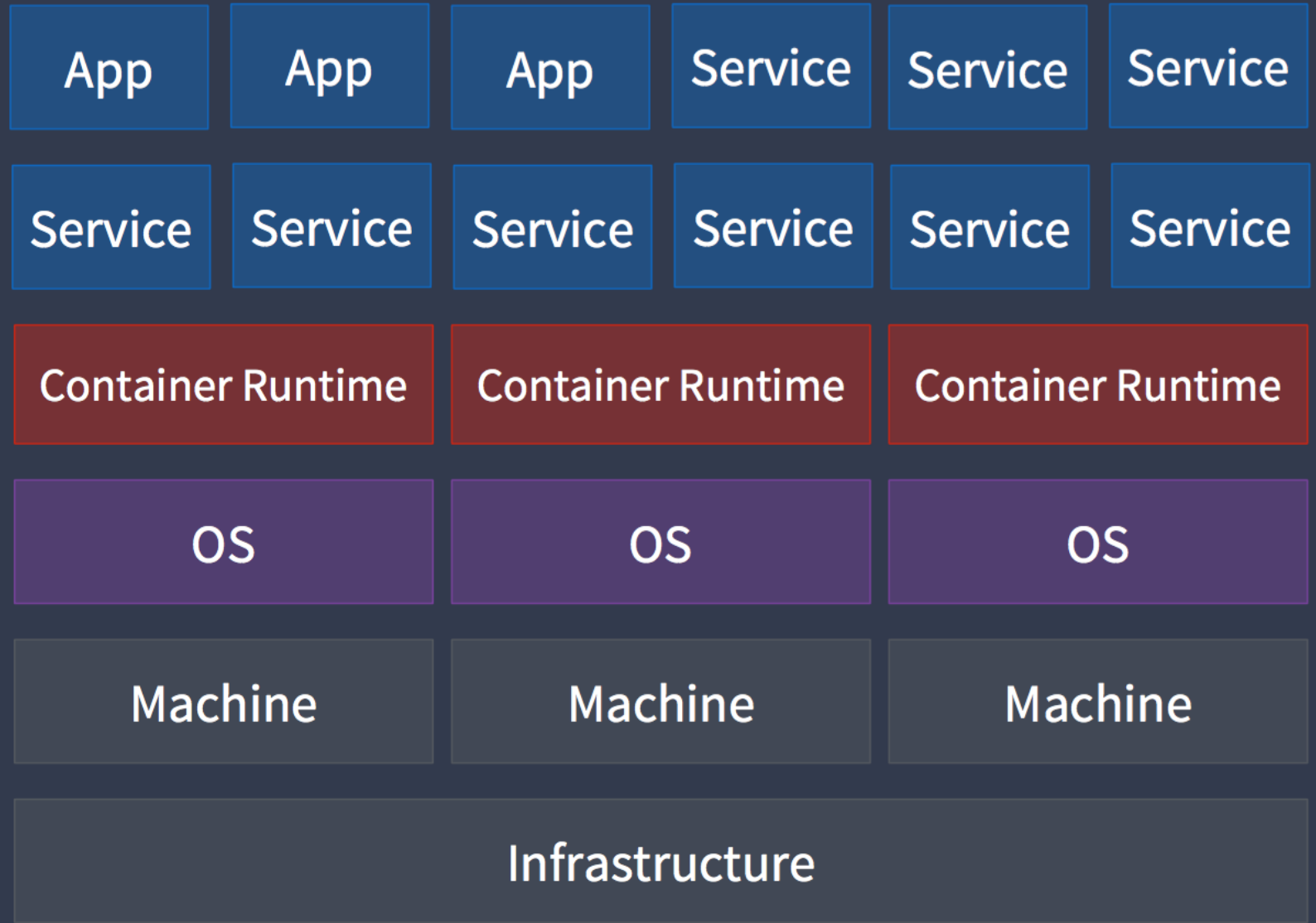
App	App	App	Service	Service	Service
Service	Service	Service	Service	Service	Service

Container Runtime	Container Runtime	Container Runtime
OS	OS	OS
Machine	Machine	Machine

Infrastructure

CONTAINER ORCHESTRATION

App	App	App	Service	Service	Service

Container Orchestration

Container Runtime	Container Runtime	Container Runtime
OS	OS	OS
Machine	Machine	Machine

Infrastructure
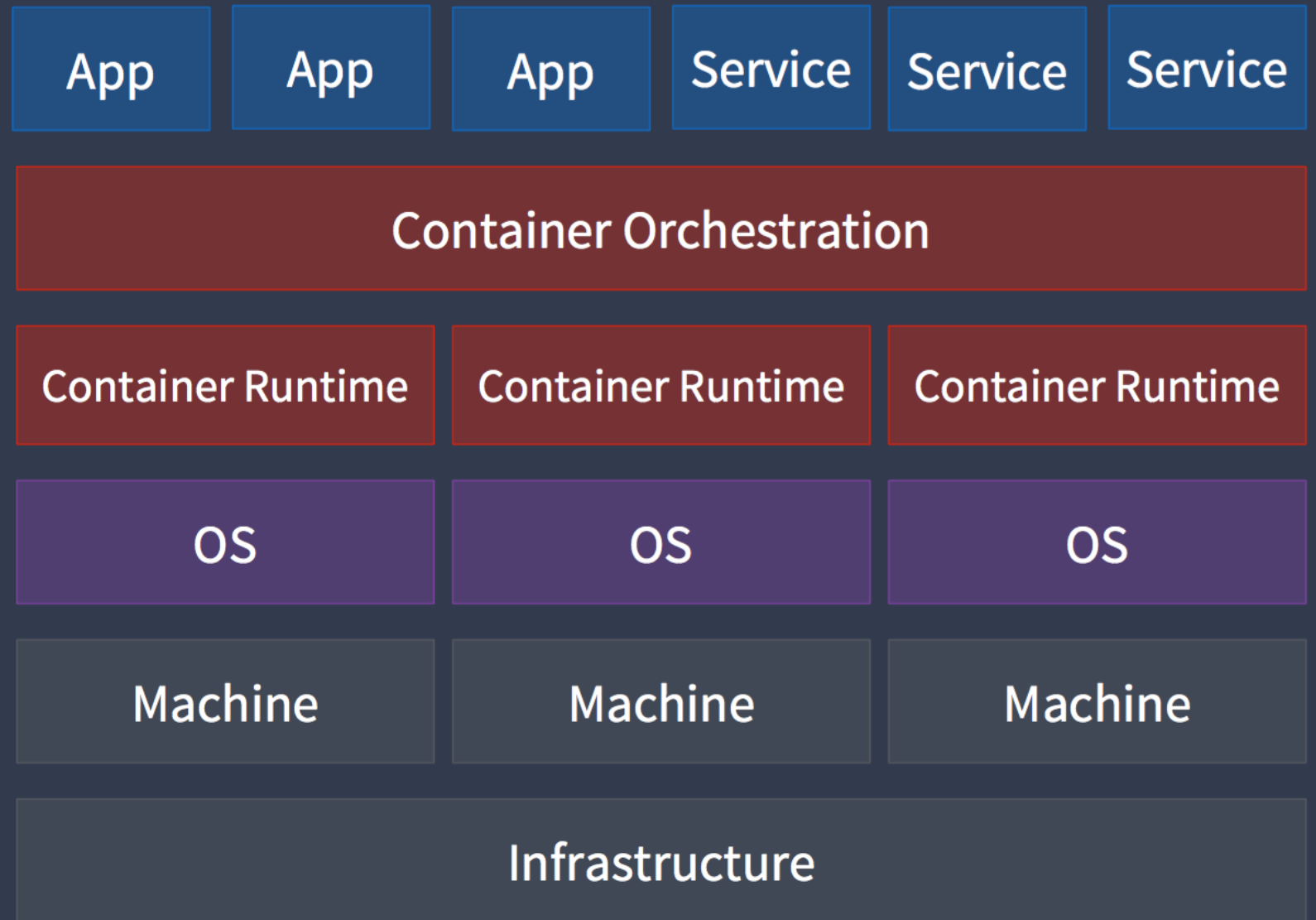
Facebook -> Tupperware
Google -> Borg / Omega
Yahoo -> YARN (Hadoop)
Twitter -> Mesos / Aurora
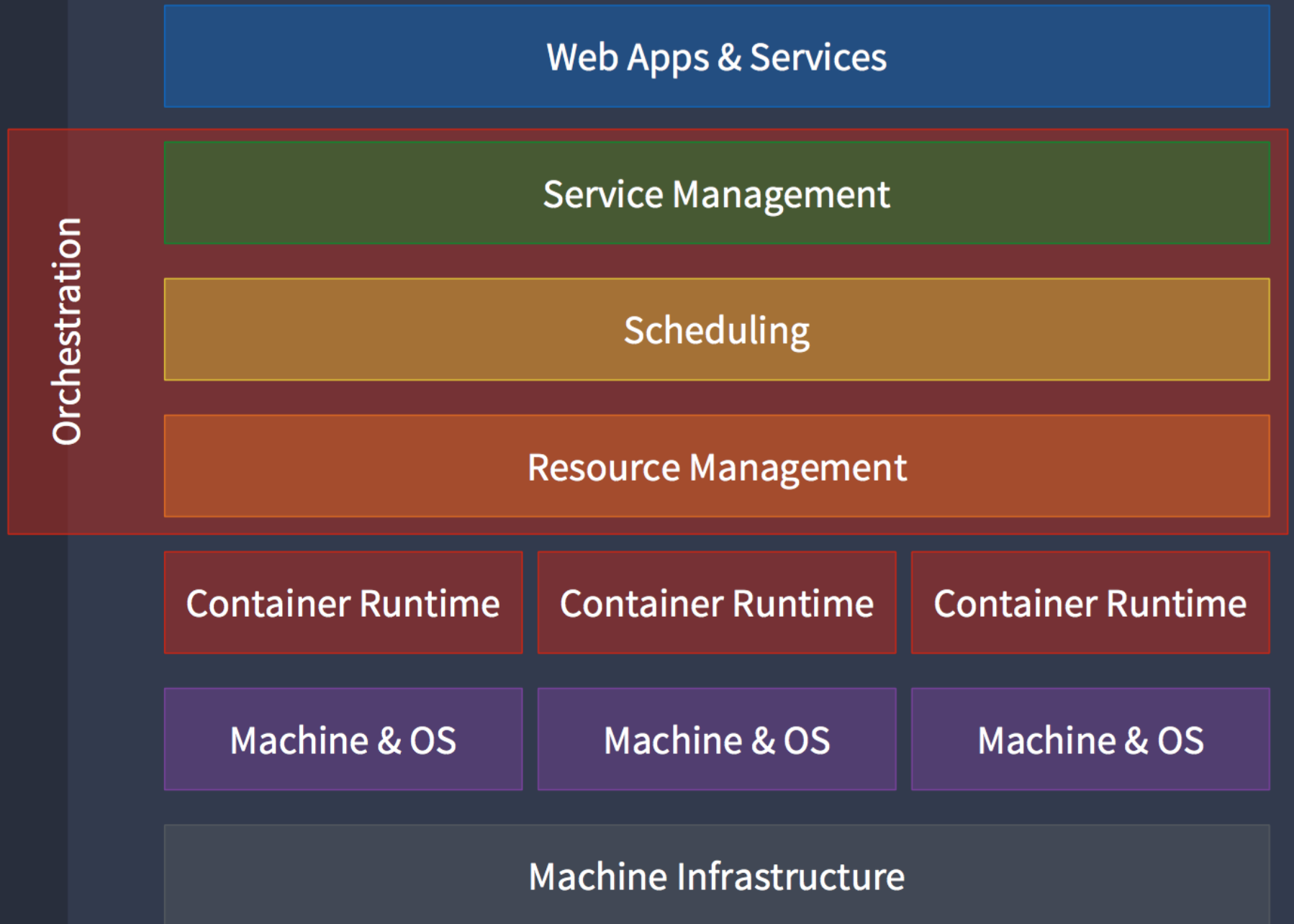
FUNCTIONAL CAPABILITIES

SCHEDULING

- Placement
- Replication/Scaling
- Resurrection
- Rescheduling
- Rolling Deployment
- Upgrades
- Downgrades
- Collocation

RESOURCE MANAGEMENT

- Memory
- CPU
- GPU
- Volumes
- Ports
- IPs

SERVICE MANAGEMENT

- Labels
- Groups/Namespaces
- Dependencies
- Load Balancing
- Readiness Checking

NON-FUNCTIONAL QUALITIES

SCALABILITY

Performance, Responsiveness, Efficiency

AVAILABILITY

Fault Tolerance, Robustness, Reliability, Resilience, Disaster Recovery

FLEXIBILITY

Format Support, Portability, Interoperability, Extensibility

USABILITY

Familiarity, Maintainability, Compatibility, Debuggability

PORTABILITY

Container Runtimes, Host OS, Hosted, Cloud, Bare-Metal

SECURITY

Auditability, Secrets Management, Encryption, Isolation

THENEWSTACK

Technology   Culture   Tutorials   eBooks   Podcasts   Events

BRIDGING REALITIES: ORCHESTRATION AND PROGRAMMABLE INFRASTRUCTURE

22 Jun 2016 9:10am, by Benjamin Ball and Alex Williams
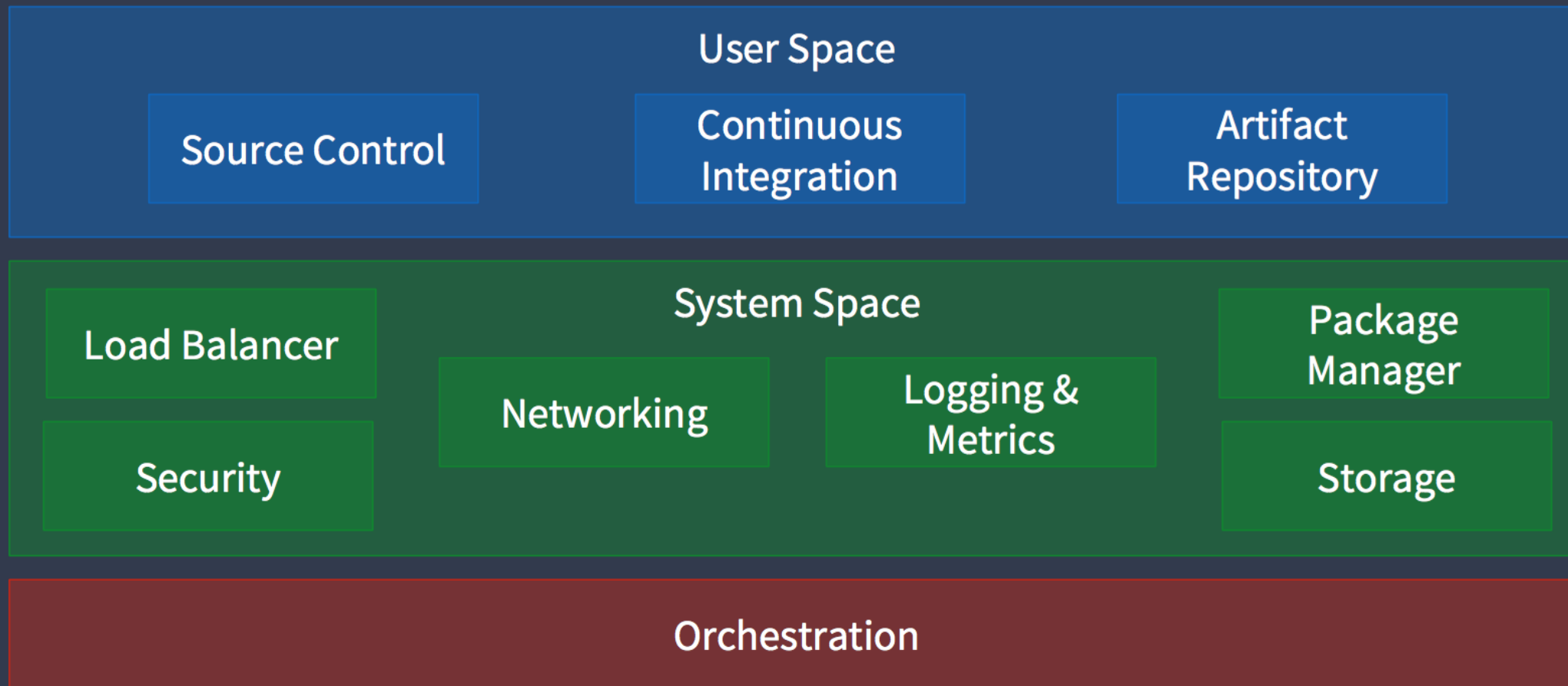
PAAS?

"Platform as a service (PaaS) is a category of cloud computing services that provides a platform **allowing customers to develop, run, and manage applications** without the complexity of building and maintaining the infrastructure typically associated with developing and launching an app."
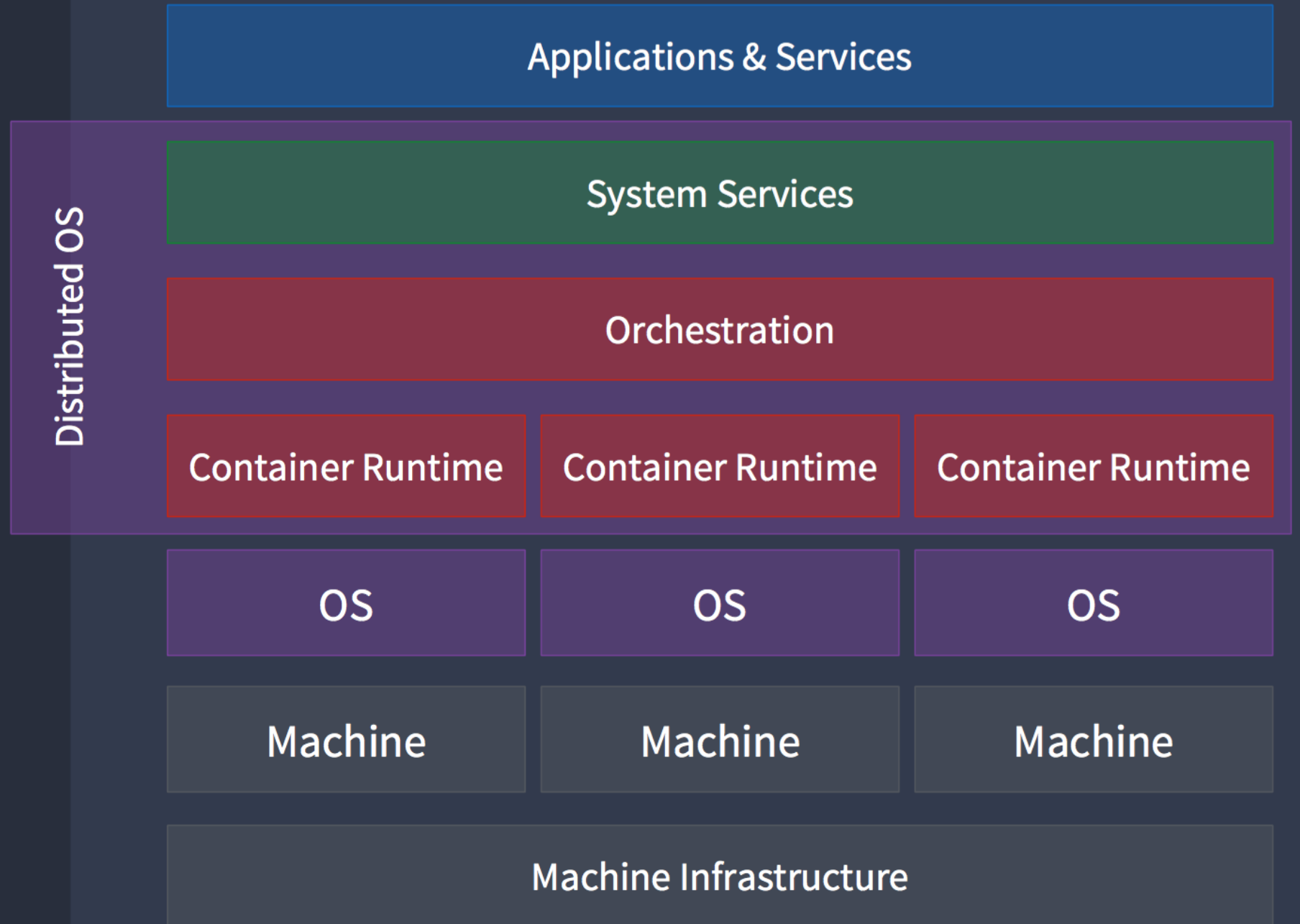
- Wikipedia

CONTAINER OPERATIONS

User Space

Source Control	Continuous Integration	Artifact Repository

System Space

Load Balancer

Security

Networking

Logging & Metrics

Package Manager

Storage

Orchestration

MEET THE
DATACENTER
OPERATING
SYSTEM

DC/OS

DC/OS IS A DISTRIBUTED OS

TRADITIONAL APPROACH

Container App 1	Container App 2	Big Data Analytics 1	Big Data Analytics 2	Stateful Service 1	Stateful Service 2
PaaS 1	PaaS 2				

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

DC/OS APPROACH

Complete workload portability

Container Apps (All)	Big Data Analytics (All)	Stateful Service (All)
PaaS (All)		

DC/OS

Proven in mission-critical deployments

Deploys on-premises or in cloud
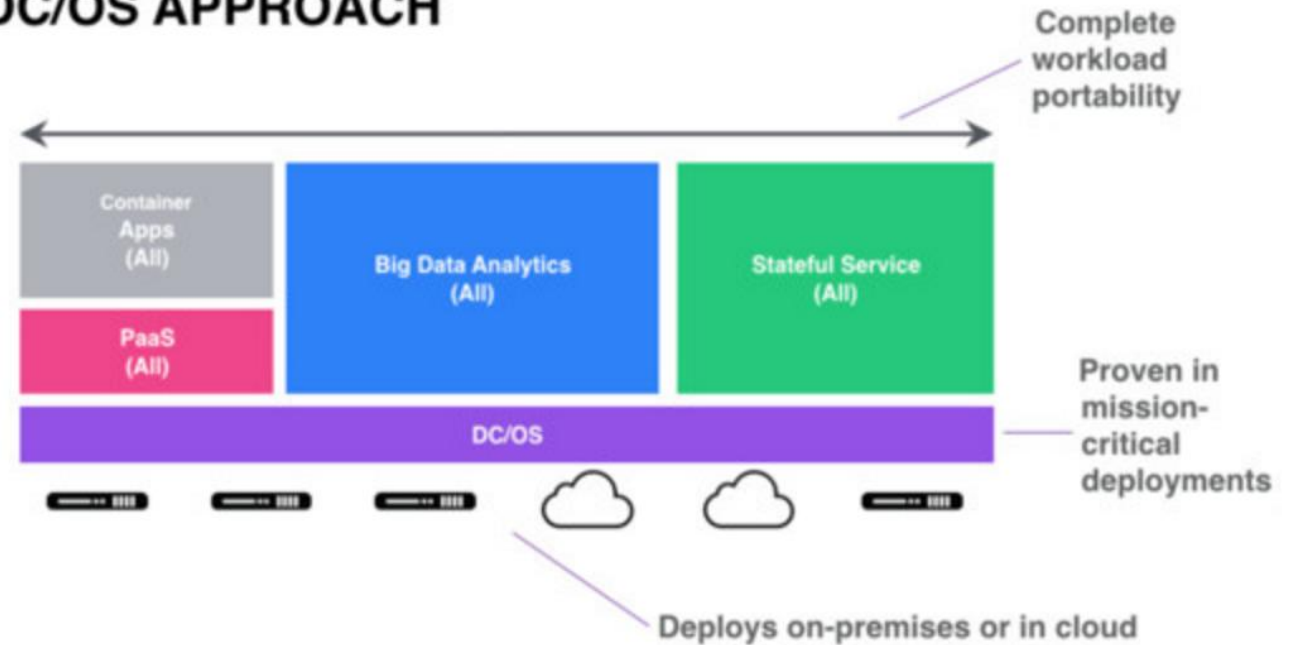
- High performance and resource isolation
- Easy scalability and multi-tenancy
- Fault tolerant and highly available
- Highly efficient with highest utilization

https://dcos.io
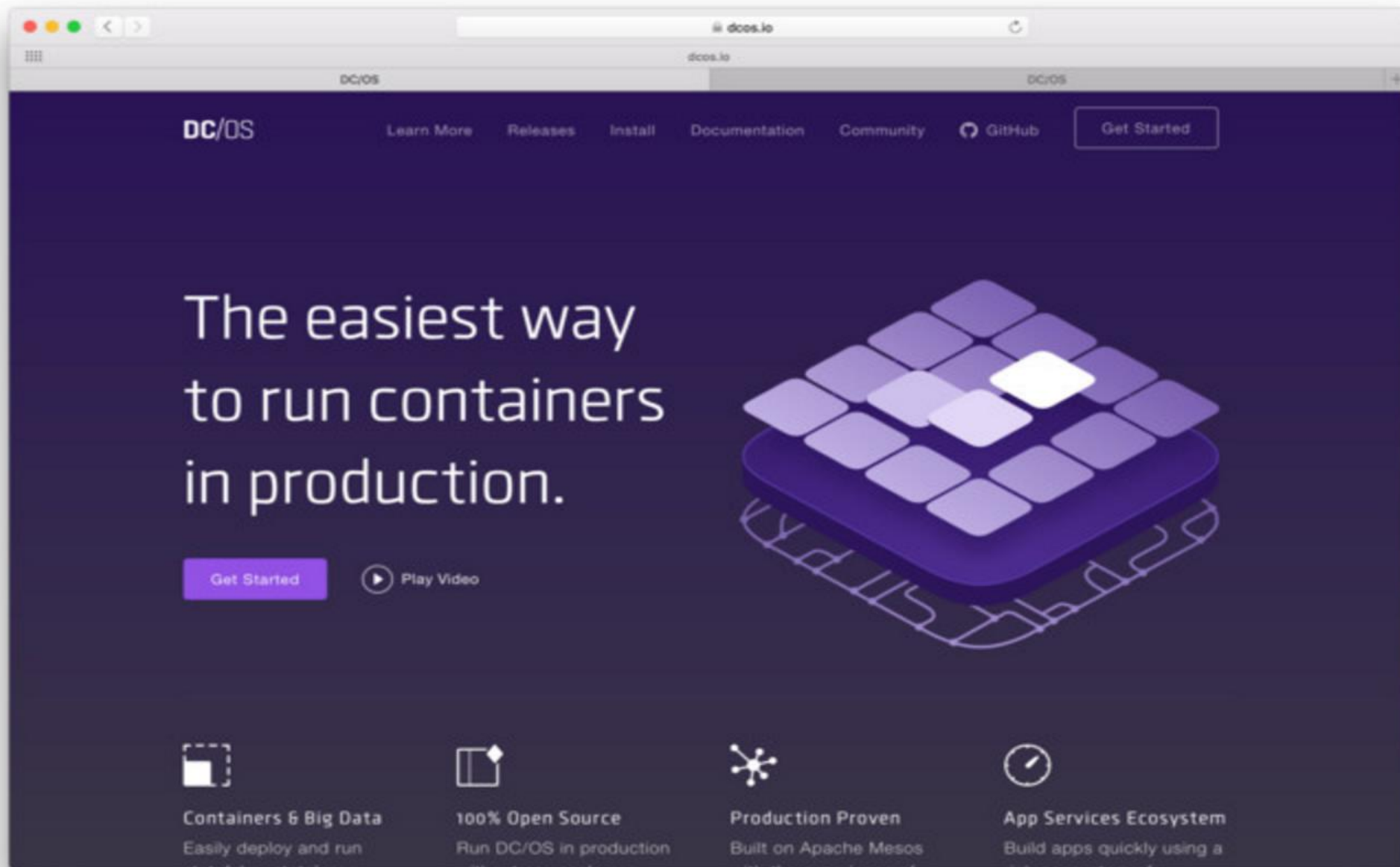
Why is DCOS awesome?

- Containers! (High resource utilization, etc)

- Extensible Isolation

- Public and Private Service repositories

- Cloud Agnostic Installer

- Web and Command Line Interfaces

- Service Discovery and Load Balancing
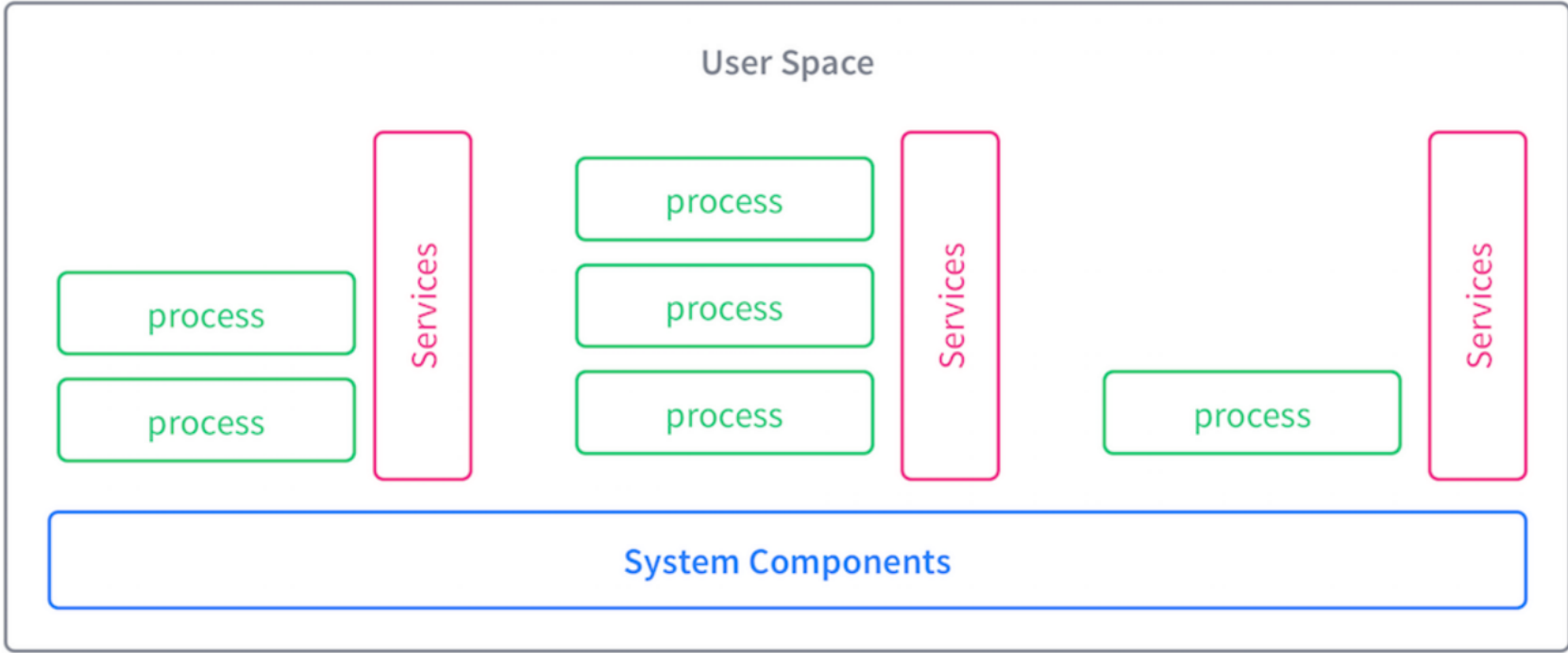
APACHE MESOS

- 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
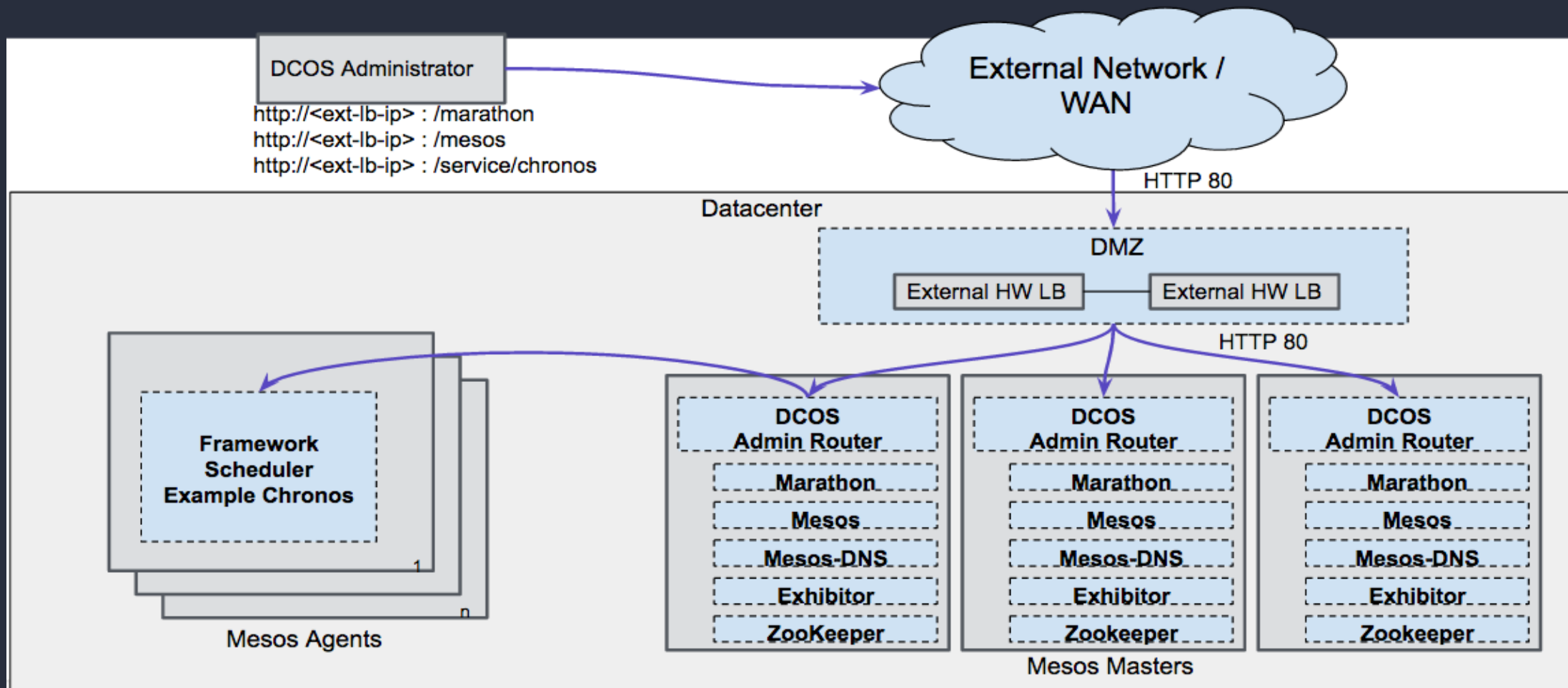- An SDK for distributed apps
- Native Docker support

mesos.apache.org

System Components

- Admin Router

- Mesos DNS

- Cosmos Service

- Marathon

- Diagnostic + Signal Service

- Minuteman Service

- DNS Proxy

- History Service

- Auth Service
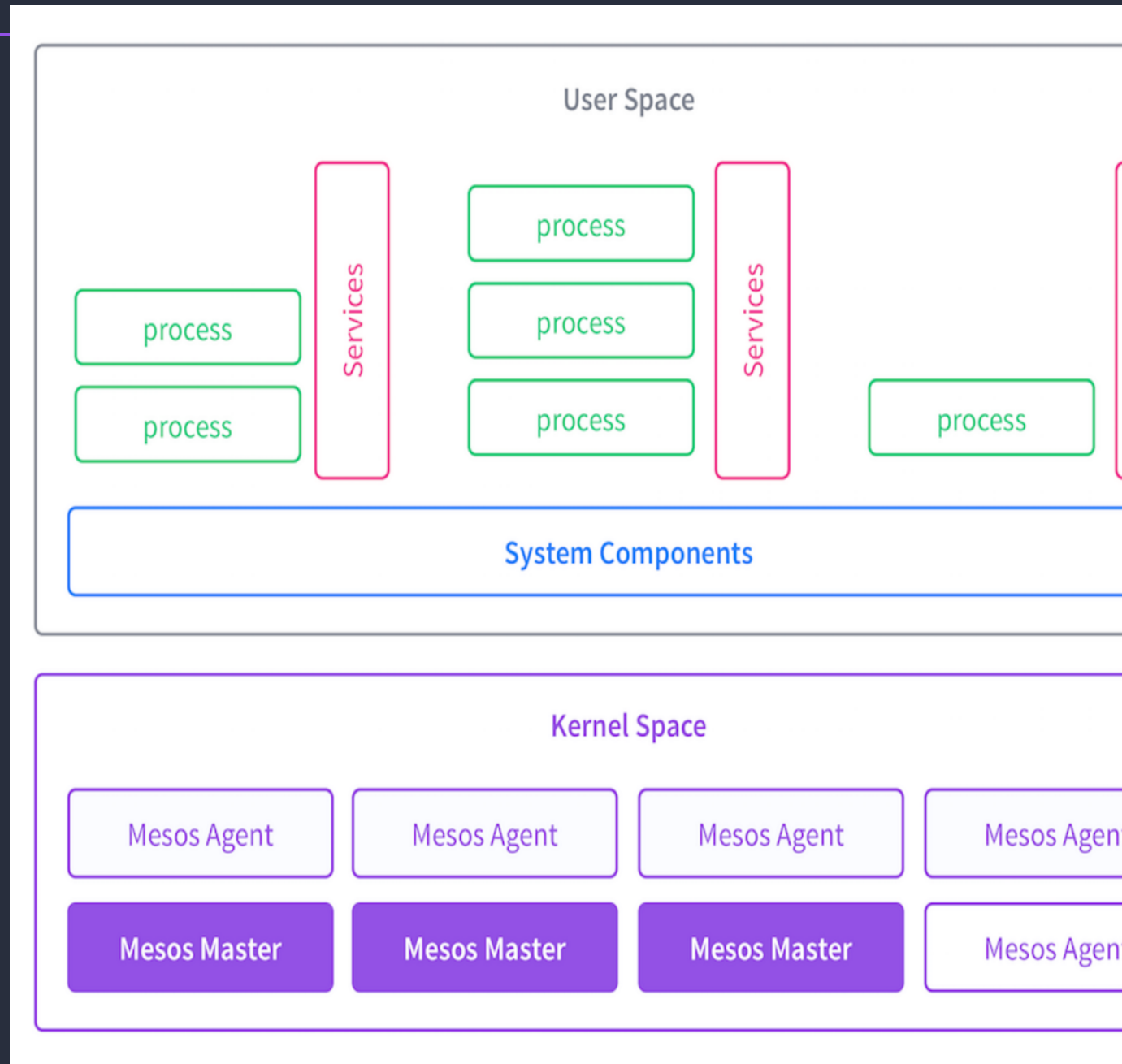
https://github.com/dcos/adminrouter

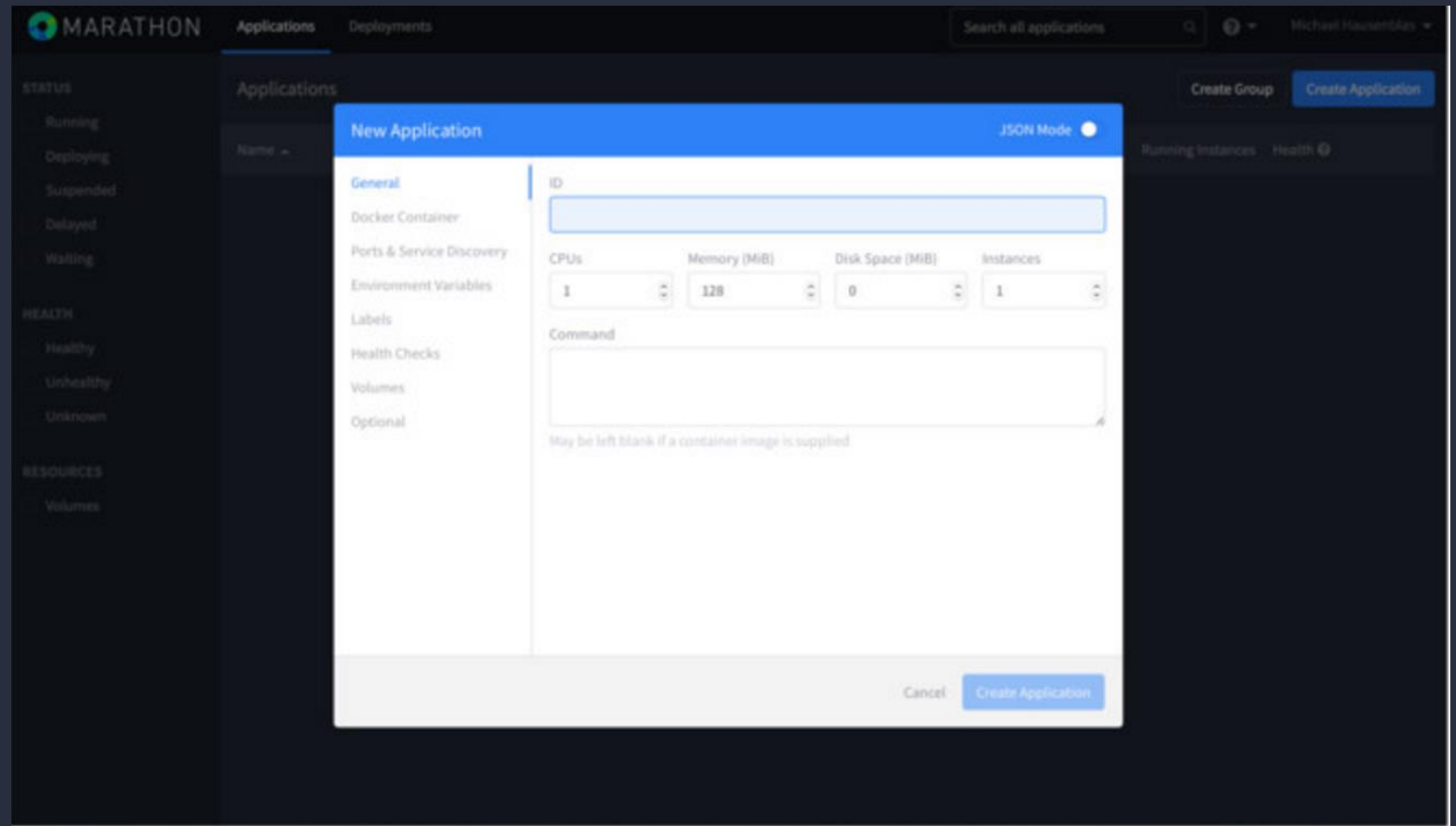| Component	Open Source Mesos Host : Port	With DCOS & Admin Router via HTTP
Mesos Master Web UI / API	\<mesos-master\> : 5050	\<mesos-master\> /mesos/
Mesos Agent UI / API	\<mesos-agent\> : 5051	\<mesos-master\> /slave/\<slave-id\>
Exhibitor UI / API	\<mesos-master\> : 8181	\<mesos-master\> /exhibitor/
Marathon UI / API	\<mesos-master\> : 8080	\<mesos-master\> /marathon/
Mesos-DNS API	\<mesos-master\> : 8123	\<mesos-master\> /mesos_dns/v1/...
Frameworks UI	\<framework-scheduler\> : xyz	\<mesos-master\> /service/\<service-id\>
DCOS History Svc API	N/A	\<mesos-master\> /dcos-history-service/history/...

Mesos DNS

https://github.com/mesosphere/mesos-dns
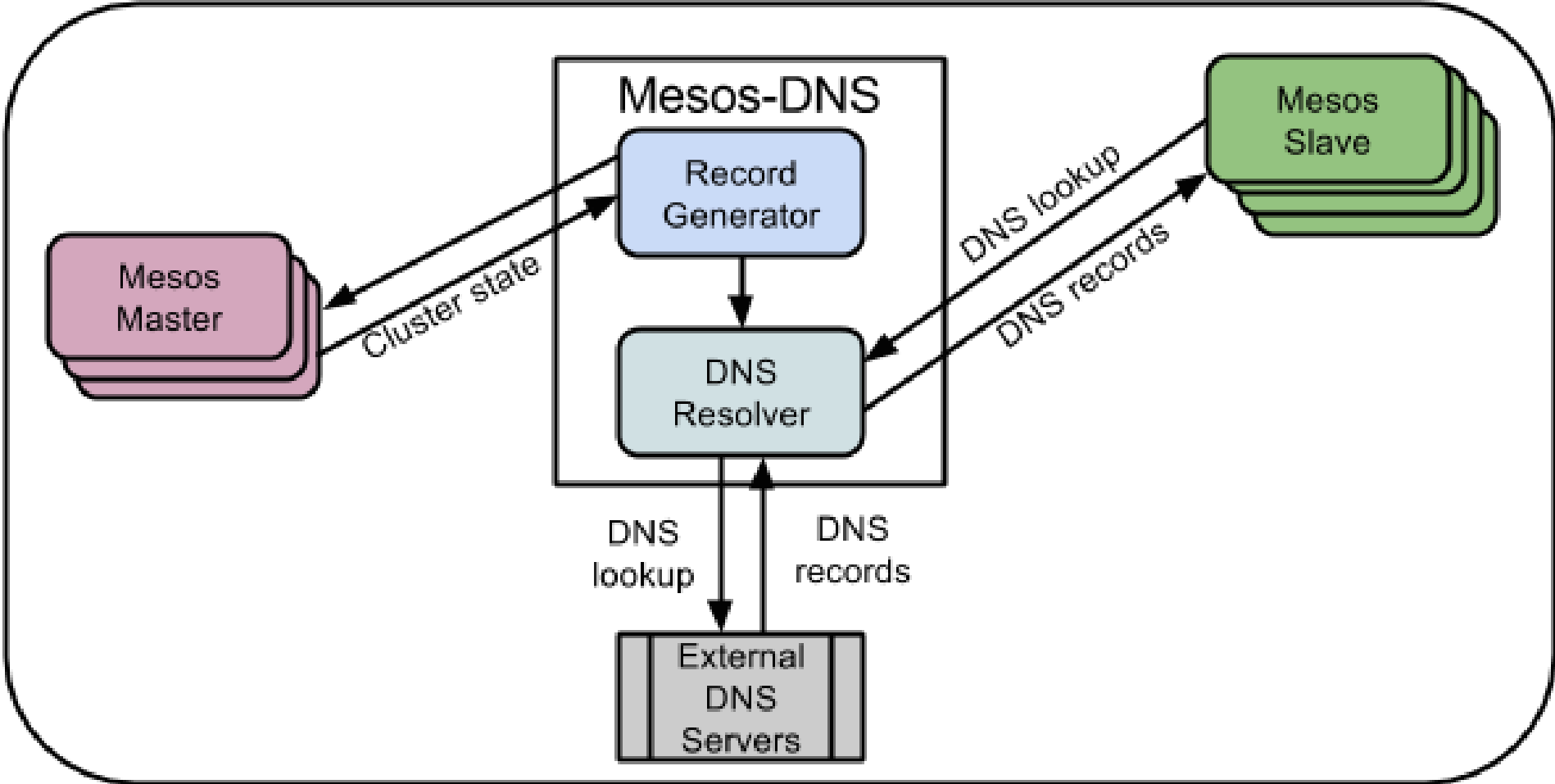
Cosmos

- **Package management server**

- **Handles packaging discovery, serving, install**

https://github.com/dcos/cosmos

Marathon

- DCOS Init system

- Deployment

- HA

- Health Checks

Diagnostics / Signal Service

- 3DT monitoring agent + REST API, collects node health and DC/OS specific services health

- Signal service forwards telemetry to consumer (SegmentIO, etc)

https://github.com/dcos/3dt

https://github.com/dcos/dcos-signal

Minuteman

- DC/OS Load balancing + Service discovery solution

- Distributed L4 load balancer with VIPs

https://github.com/dcos/minuteman

Install DC/OS

- Local Vagrant

- Cloud Provisioned (AWS, Azure)

- Custom installation (GUI, scriptable)

joel-enterprise
52.39.80.140

Dashboard
Services
Nodes
Network
Universe
System

Enterprise DC/OS v.1.7.0

Bootstrap superuser

CPU Allocation
0%
0 of 8 Shares

100%

50%

0%
-60s   -30s   0

Memory Allocation
0%
0 B of 27 GiB

100%

50%

0%
-60s   -30s   0

Task Failure Rate
0%
Current Failure Rate

100%

50%

0%
-60s   -30s   0

Services Health

marathon   Idle

View all Services

Tasks

0
Total Tasks

0   0
Tasks running   Tasks staging

Component Health

Admin Router   Healthy
Admin Router Reloader   Healthy
Admin Router Reloader Timer   Healthy
Cluster ID   Healthy
Diagnostics   Healthy
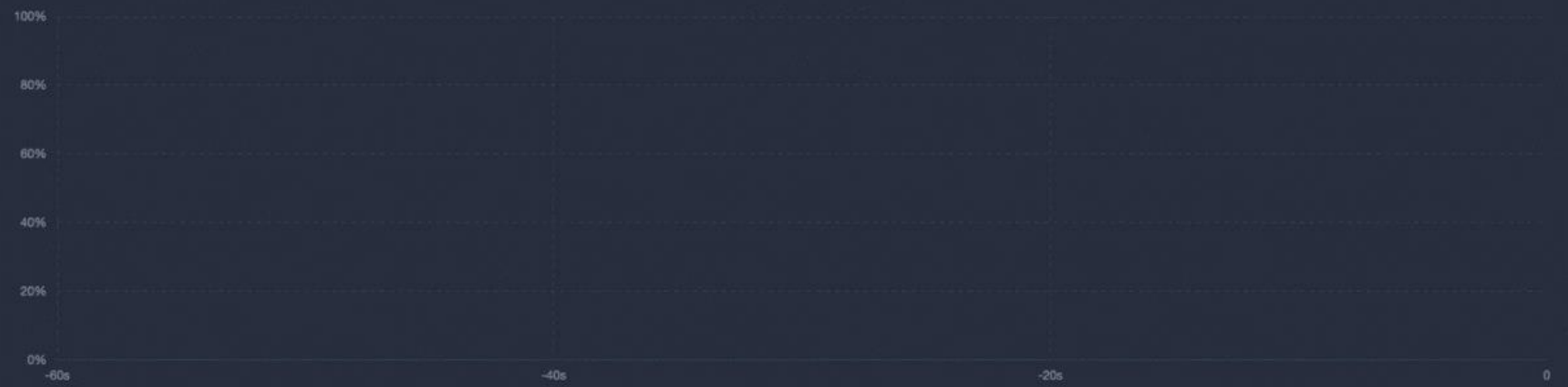
View all 29 Components

Nodes

2
Connected Nodes

10

Dashboard

Services

Nodes

Network

Universe

System

CPU   Memory   Disk

CPU Allocation Rate
1 Total Services

100%

80%

60%

40%

20%

0%

-60s   -40s   -20s   0

1 Services

All ① | ● Healthy ⓪ | ● Unhealthy ⓪ | 🔍 Filter

SERVICE NAME ▲	HEALTH	TASKS	CPU	MEM	DISK
🔵 marathon	Idle	0	0	0 B	0 B

Bootstrap superuser

Dashboard

Services

Nodes

Network

Universe

System

1 Virtual IPs

VIRTUAL IP ▲

1.1.1.1:30000

SUCCESSES

342

Bootstrap superuser

‹ **Back**

1.1.1.1:30000

Virtual IP

Backends

Successes and Failures per Minute ▾
3 Total Backends

ⓘ

-30m:
● Successes: 3412

3000

2000

1000

0.00

-60m   -50m   -40m   -30m   -20m   -10m

Requests

● Successes   ● Failures

3 Backends

🔍 Filter

BACKEND NAME ▲	SUCCESSES	FAILURES	P99 LATENCY
10.0.2.164:24745	1823	3	2.4ms
10.0.2.167:2293	1180	0	4.11ms
10.0.2.168:4707	1602	0	2.55ms

DC/OS Packages

- A package:

    - Mesos framework (Marathon JSON)

    - CLI

    - Metadata (Description, Icon)

- Universe stores all packages index and versions

- (http://github.com/mesosphere/universe)

- DCOS CLI contacts Cosmos to un/install packages

Dashboard

Services

Nodes

Network

Universe

System

Enterprise DC/OS v.1.7.0

Bootstrap superuser

Packages Installed

Search

Selected Packages

arangodb
0.3.0

Install Package

cassandra
1.0.0-2.2.5

Install Package

chronos
2.4.0

Install Package

jenkins
0.2.3

Install Package

kafka
1.0.0-0.9.0.1

Install Package

marathon
0.15.3

Install Package

spark
1.6.1-6

Install Package

Community Packages

Demo!

http://github.com/mesosphere/tweeter

Next big question moving forward…

How do we run containers much more efficiently at scale?

Thanks!

@tnachen

tnachen@gmail.com