Introduction to DC/OS

Elizabeth K. Joseph
@pleia2
There was an eclipse today!
Elizabeth K. Joseph, Developer Advocate

- Developer Advocate at Mesosphere
- 15+ years working in open source communities
- 10+ years in Linux systems administration and engineering roles
- Founder of OpenSourceInfra.org
- Author of The Official Ubuntu Book and Common OpenStack Deployments
What are Cloud-Native Systems?

You no longer have a single server with everything running on it.

You have a multi-tier system with various layers and owners down the stack:

- Hardware
- Network
- Resource abstraction
- Scheduler
- Containers
- Virtual network
- Application
- ...
Cloud-native scopes

- Application
- Container
- Host
Cloud-Native with Containers

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
Apache Mesos: The datacenter kernel

http://mesos.apache.org/
Building block of the modern internet

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

http://mesos.apache.org/documentation/latest/powered-by-mesos/
THE BIRTH OF MESOS

CS262B
Ben Hindman, Andy Konwinski and Matei Zaharia create “Nexus” as their CS262B class project.

March 2010

TWITTER TECH TALK
The grad students working on Mesos give a tech talk at Twitter.

Spring 2009

MESOS PUBLISHED
Mesos: A Platform for Fine-Grained Resource Sharing in the Data Center is published as a technical report.

September 2010

APACHE INCUBATION
Mesos enters the Apache Incubator.

December 2010

DC/OS
April 2016

© 2017 Mesosphere, Inc. All Rights Reserved.
MULTIPLEXING OF DATA, SERVICES, USERS, ENVIRONMENTS

Typical Datacenter
siloed, over-provisioned servers, low utilization

Apache Mesos
automated schedulers, workload multiplexing onto the same machines
## Active Tasks

<table>
<thead>
<tr>
<th>Framework ID</th>
<th>Task ID</th>
<th>Task Name</th>
<th>Role</th>
<th>State</th>
<th>Started</th>
<th>Host</th>
</tr>
</thead>
<tbody>
<tr>
<td>62df48e-df4a-4309-94f0-73d5e94ab01e-0001</td>
<td>bus-demo_dashboard.37943816-8677-11e7-8432-425fcb45b8</td>
<td>dashboard.busdemo</td>
<td>slave_public</td>
<td>RUNNING</td>
<td>9 minutes</td>
<td>sandbox</td>
</tr>
<tr>
<td>62df48e-df4a-4309-94f0-73d5e94ab01e-0001</td>
<td>bus-demo_ingest.0999da65-8676-11e7-b432-425fcb45b8</td>
<td>ingest.busdemo</td>
<td>slave_public</td>
<td>RUNNING</td>
<td>9 minutes</td>
<td>sandbox</td>
</tr>
<tr>
<td>62df48e-df4a-4309-94f0-73d5e94ab01e-0004</td>
<td>broker-2_581647a0-6953-4c6a-af96-356d04535c38</td>
<td>broker-2</td>
<td>kafka-role</td>
<td>RUNNING</td>
<td>12 minutes</td>
<td>sandbox</td>
</tr>
<tr>
<td>62df48e-df4a-4309-94f0-73d5e94ab01e-0004</td>
<td>broker-1_d24b1885-860b-4ae9-9efb-502fcded5fe</td>
<td>broker-1</td>
<td>kafka-role</td>
<td>RUNNING</td>
<td>13 minutes</td>
<td>sandbox</td>
</tr>
<tr>
<td>62df48e-df4a-4309-94f0-73d5e94ab01e-0004</td>
<td>broker_0_eb077cd0-f416-4918-9cbb-1f5b1ea8c10d</td>
<td>broker-0</td>
<td>kafka-role</td>
<td>RUNNING</td>
<td>13 minutes</td>
<td>sandbox</td>
</tr>
<tr>
<td>62df48e-df4a-4309-94f0-73d5e94ab01e-0001</td>
<td>kafka.8a66774-8675-11e7-b432-425fcb45b8</td>
<td>kafka</td>
<td>slave_public</td>
<td>RUNNING</td>
<td>13 minutes</td>
<td>sandbox</td>
</tr>
<tr>
<td>62df48e-df4a-4309-94f0-73d5e94ab01e-0003</td>
<td>node-2_a9c28921-d7c1-4a32-8eb5-4f373b25665d</td>
<td>node-2</td>
<td>cassandra-role</td>
<td>RUNNING</td>
<td>14 minutes</td>
<td>sandbox</td>
</tr>
<tr>
<td>62df48e-df4a-4309-94f0-73d5e94ab01e-0003</td>
<td>node-2_a9c28921-d7c1-4a32-8eb5-4f373b25665d</td>
<td>node-2</td>
<td>cassandra-role</td>
<td>RUNNING</td>
<td>14 minutes</td>
<td>sandbox</td>
</tr>
</tbody>
</table>

**Cluster:** ejoseph-te4msh6
**Leader:** 10.0.5.257.5050
**Version:** 1.4.0
**Built:** 5 days ago by
**Started:** 53 minutes ago
**Elected:** 53 minutes ago

**Agents**
- Activated: 5
- Deactivated: 0
- Unreachable: 0

**Tasks**
- Staging: 0
- Starting: 0
- Running: 11
- Unreachable: 0
- Killing: 0
- Finished: 1
- Killed: 0
Mesos can’t run applications on its own.
A Mesos framework is a distributed system that has a scheduler.
Schedulers like Marathon start and keep your applications running. A bit like a distributed init system.
Learn more at https://mesosphere.github.io/marathon/
<table>
<thead>
<tr>
<th>Name</th>
<th>CPU</th>
<th>Memory</th>
<th>Status</th>
<th>Running Instances</th>
<th>Health</th>
</tr>
</thead>
<tbody>
<tr>
<td>bus-demo</td>
<td>0.2</td>
<td>4 GiB</td>
<td></td>
<td>2 of 2</td>
<td></td>
</tr>
<tr>
<td>cassandra</td>
<td>0.5</td>
<td>2 GiB</td>
<td>Running</td>
<td>1 of 1</td>
<td></td>
</tr>
<tr>
<td>kaika</td>
<td>1.0</td>
<td>1 GiB</td>
<td>Running</td>
<td>1 of 1</td>
<td></td>
</tr>
<tr>
<td>spark</td>
<td>1.0</td>
<td>1 GiB</td>
<td>Running</td>
<td>1 of 1</td>
<td></td>
</tr>
</tbody>
</table>
Containers

- Rapid deployment
- Some service isolation
- Dependency handling
- Container image repositories
CONTAINER ORCHESTRATION

CONTAINER SCHEDULING

RESOURCE MANAGEMENT

SERVICE MANAGEMENT
DC/OS brings it all together

● Resource management
● Task scheduling
● Container orchestration
● Logging and metrics
● Network management
● “Universe” catalog of pre-configured apps (including Apache Spark, Apache Kafka…), browse at http://universe.dcos.io/
● And much more https://dcos.io/
DC/OS is ...

- 100% open source (ASL2.0)
  + A big, diverse community
- An umbrella for ~30 OSS projects
  + Roadmap and designs
  + Documentation and tutorials
- Not limited in any way
- Familiar, with more features
  + Networking, Security, CLI, UI, Service Discovery, Load Balancing, Packages, ...
DC/OS Architecture Overview

Services & Containers

- HDFS
- Jenkins
- Marathon
- Cassandra
- Flink
- Spark
- Docker
- Kafka
- MongoDB
- +30 more...

DC/OS

- Container Orchestration
- Security & Governance
- Monitoring & Operations
- User Interface & Command Line

ANY INFRASTRUCTURE

- Physical Servers
- Virtual Servers
- Private Cloud
- Public Cloud Providers (Google, AWS, Azure)
Interact with DC/OS (1/2)

Web-based GUI

https://dcos.io/docs/latest/usage/webinterface/
Interact with DC/OS (2/2)

**CLI tool**

https://dcos.io/docs/latest/usage/cli/

**API**

https://dcos.io/docs/latest/api/
Catalog of Applications (Universe)
Install an Application

service
routing
email
high-availability
single-node
enterprise

service
GitLab service properties
NAME *
gitlab

CPUS *
1

MEM *
2048

ROLE *

HOST-VOLUME *
/srv/gitlab

HOST-SHARED-VOLUME *

CANCEL

REVIEW AND DEPLOY
Building a Real-World Pipeline
MODERN APPLICATION -> FAST DATA BUILT-IN

Use Cases:
- Anomaly detection
- Personalization
- IoT Applications
- Predictive Analytics
- Machine Learning

Data Ingestion → Data Ingestion

Message Queue/Bus → Analytics (Streaming)

Microservices ← Request/Response

Distributed Storage
The SMACK Stack

Use Cases:
- Anomaly detection
- Personalization
- IoT Applications
- Predictive Analytics
- Machine Learning

Devices

Sensors

Client

Message Queue/Bus

Analytics (Streaming)

Data Ingestion

Microservices

Distributed Storage

Request/Response

kafka

Spark

cassandra

© 2017 Mesosphere, Inc. All Rights Reserved.
Keeping things running: Day 2 Operations

**Metrics & Monitoring**
- Collecting metrics
- Routing events
- Downstream processing
  - Alerting
  - Dashboards
  - Storage (long-term retention)

**Logging**
- Scopes
- Local vs. Central
- Security considerations
Day 2 Operations con’t

Maintenance
- Cluster Upgrades
- Cluster Resizing
- Capacity Planning
- User & Package Management
- Networking Policies
- Auditing
- Backups & Disaster Recovery

Troubleshooting
- Debugging
  - Services
  - System
  - Access
- Tracing
- Chaos Engineering
Financial Transactions Demo

https://dcos.io/demos/
Financial Transactions Demo
Questions? Feedback?

Elizabeth K. Joseph
Twitter: @pleia2
Email: lyz@princessleia.com

@dcos
chat.dcos.io
users@dcos.io
/dcoss
/dcoss/examples
/dcoss/demos