ContainerSolutions

THE SOFTWARE CIRCUS IS COMING TO TOWN

Learning Apache Mesos with minimesos

Frank Scholten & Viktor Sadovnikov
May 6th 2016

Frank Scholten

- Senior software engineer at
- Creator of minimesos
- CS graduate at UNIVERSITEIT TWENTE.
- Open Source
  - ➔ Apache Whirr (now in attic)
  - ➔ Apache Mahout

@Frank_Scholten

Viktor Sadovnikov

- 1985 - first keyboard touch
- 1988 - first line of code
- 1993 - first computer at home
- 1994 - first paid code line
- 2001 - first code to test another code
- 2005 - first build server runs
- 2007 - first automated deployments

@sadovnikov

Agenda

● Intro + presentation	1:00 PM	- 2:00 PM
● Exercises	2:00PM	- 2:30 PM
● Break	2:30PM	- 2:45 PM
● Exercises	2:45PM	- 3:45 PM
● Break	3:45PM	- 4:00 PM
● Experiment and Q&A	4:00PM	- 4:45 PM
● Wrap up	4:45PM	- 5:00 PM
● Beer!	5:00 PM	

Outline

- What is Apache Mesos?
- How does Mesos work?
- Technical details

- What is minimesos?
- CLI & Java API
- Upcoming features

What is Mesos?

“Apache Mesos abstracts CPU, memory, storage, and other compute resources away from machines (physical or virtual), enabling fault-tolerant and elastic distributed systems to easily be built and run effectively”

- Created in Berkeley AMP lab by Ben Hindman as part of PhD thesis
- https://www.cs.berkeley.edu/~alig/papers/mesos.pdf
- Top level Apache project since 2013

Powered by Mesos

http://mesos.apache.org/documentation/latest/powered-by-mesos/ lists 91 organisation

Why Mesos?

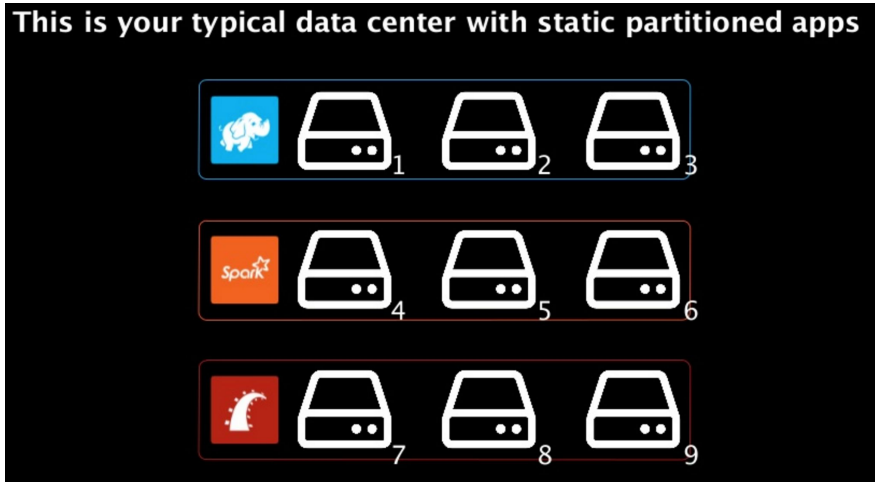
Run multiple technologies, AKA frameworks, on a single Mesos cluster

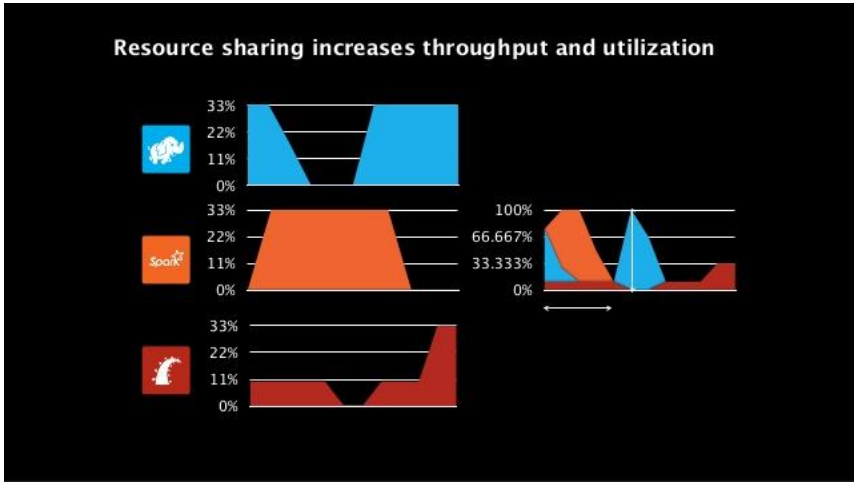
Blog post by Phil Winder @ CS blog

http://container-solutions.com/reasons-use-apache-mesos-frameworks

Why Mesos?

Cost savings through improved utilization



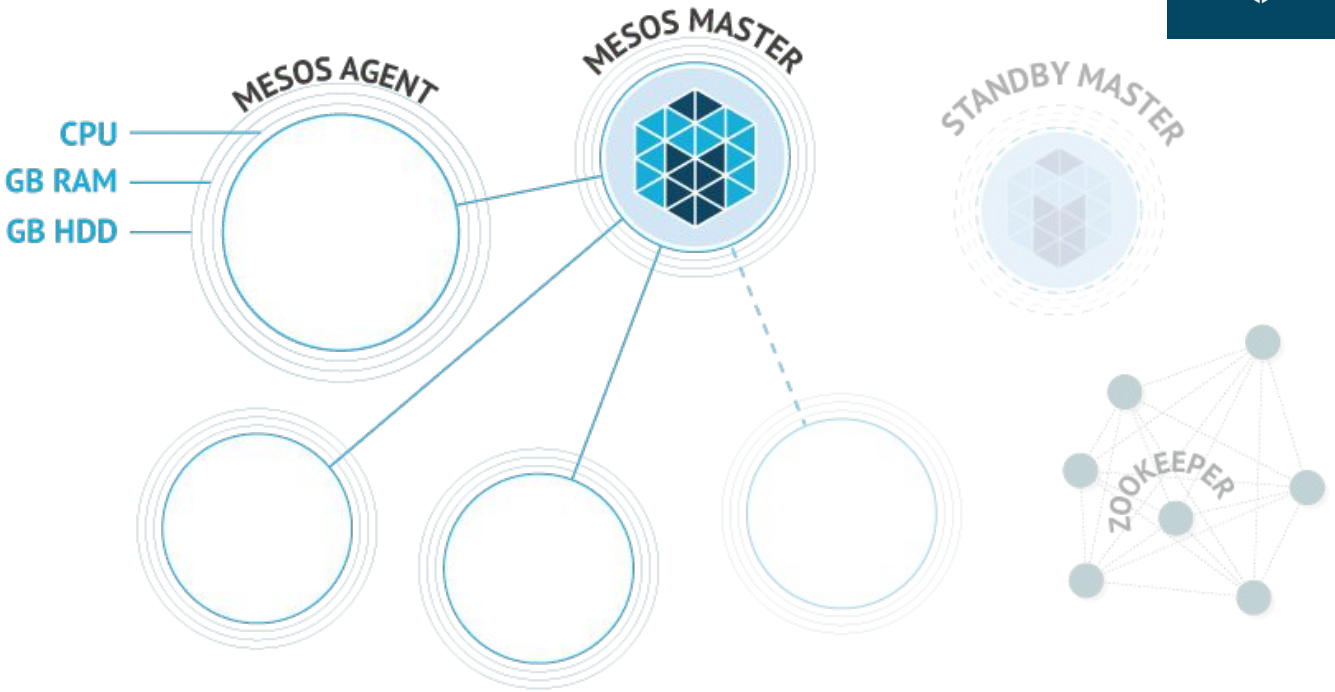


Why Mesos?

Simplify deployment

- Schedule Docker containers and regular processes
- Reuse Mesos primitives for creating distributed systems
- *“Program against your datacenter like it’s a single pool of resources”*

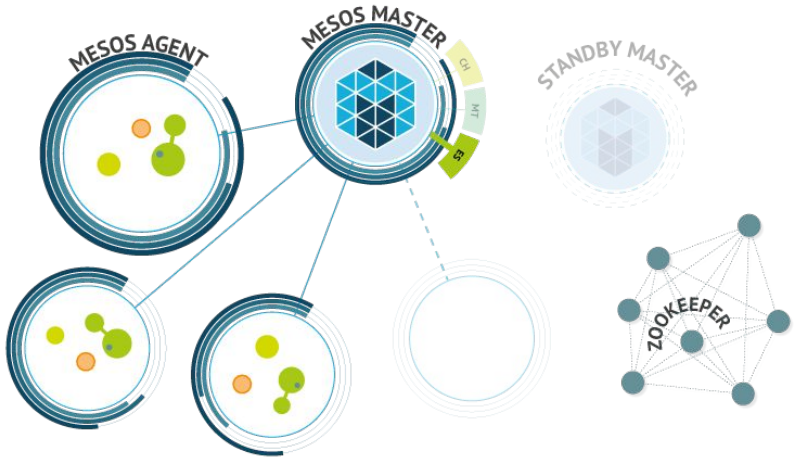
Mesos Architecture



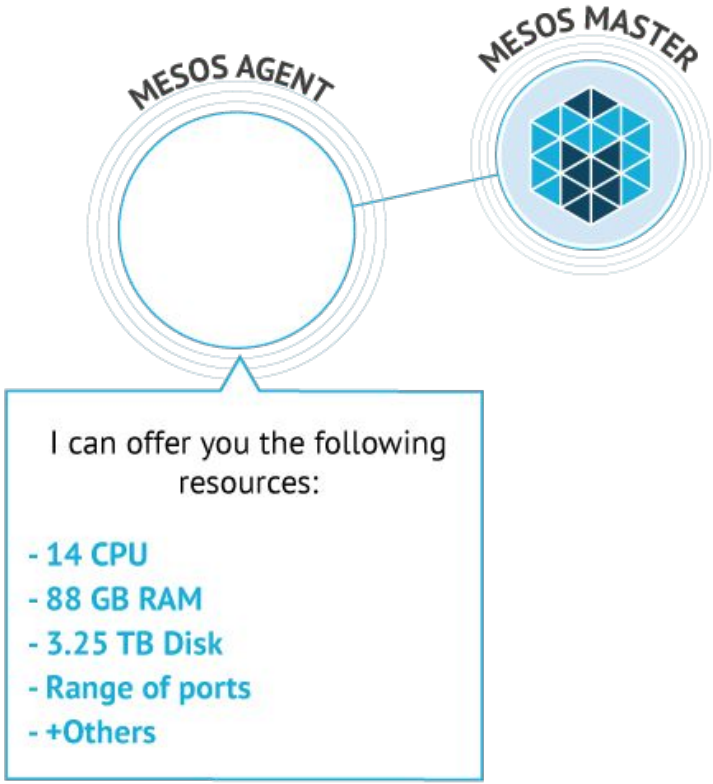
Mesos Architecture

Quick run through Mesos architecture and terminology

- Mesos Master
- Mesos Agents
- ZooKeeper
- Frameworks
- Framework Scheduler
- Framework Executors
- Framework Tasks

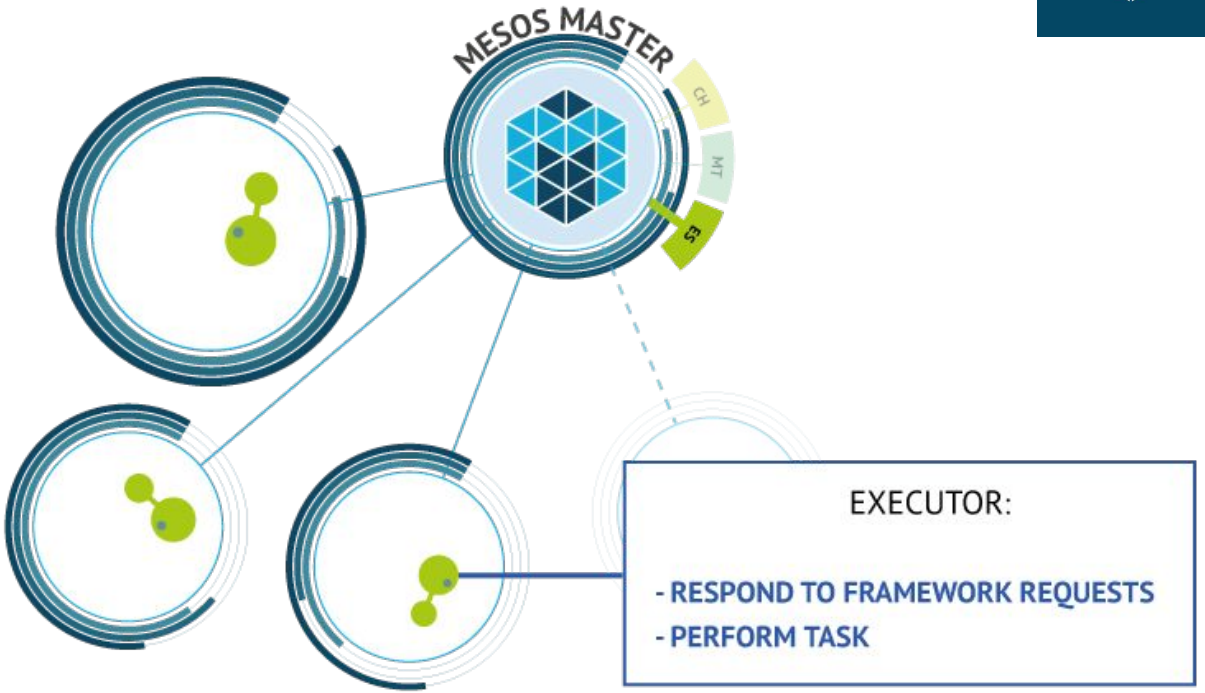


Mesos Resources

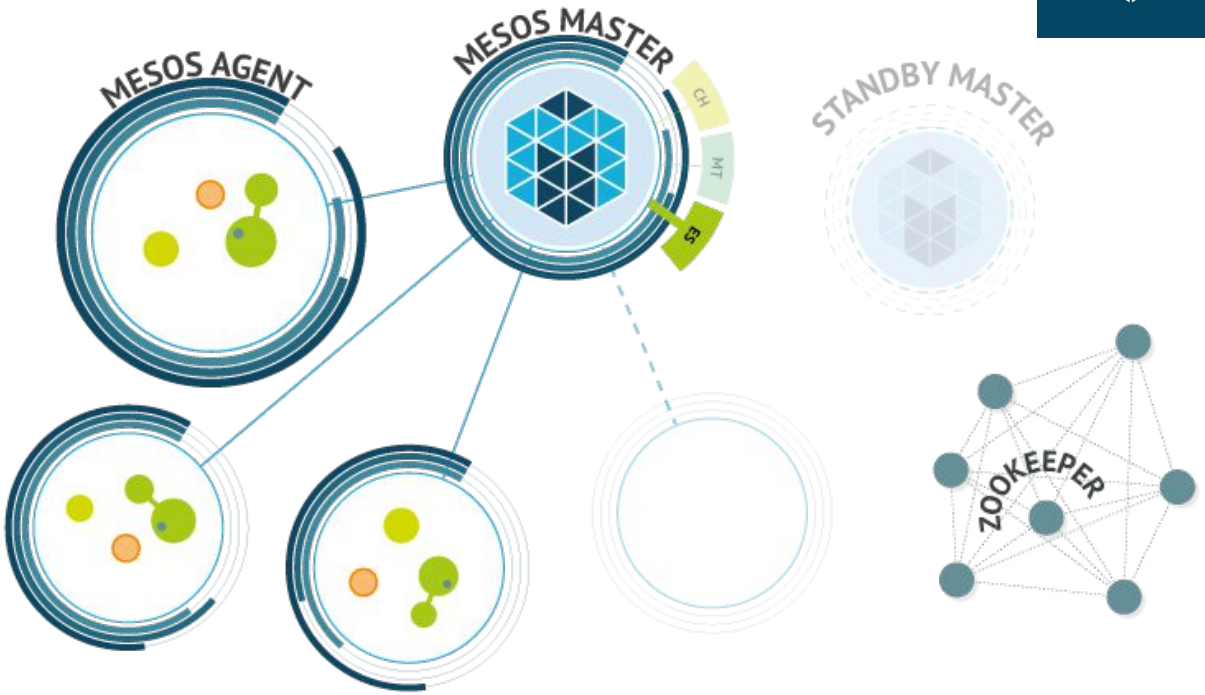


Mesos Scheduler

Mesos Executor



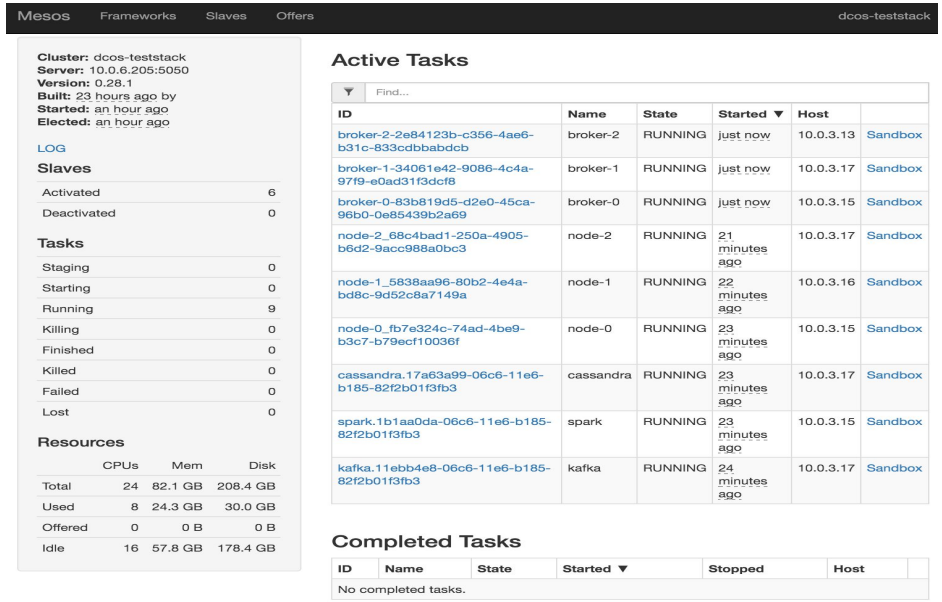
Overview



Mesos UI

http://$MASTER:5050

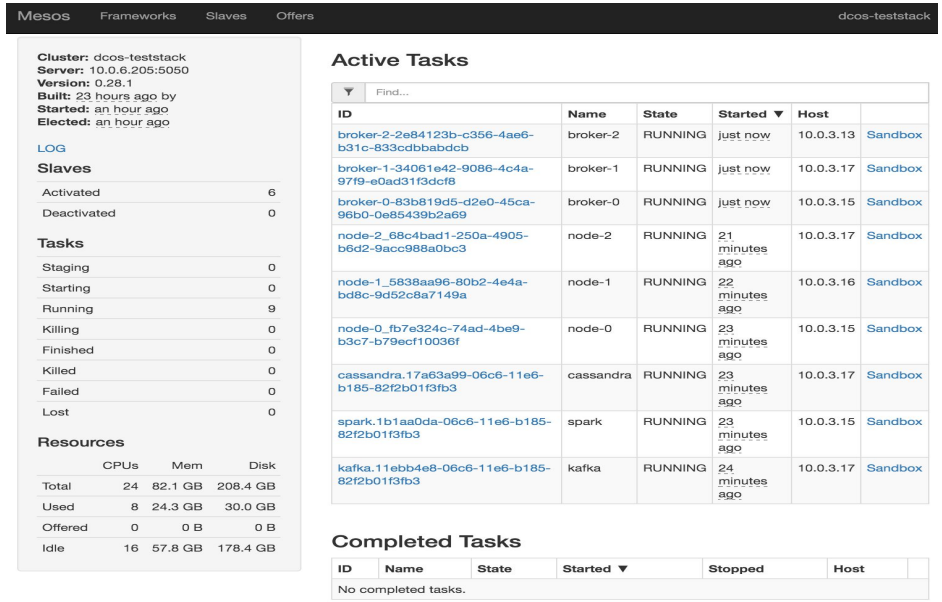
- Frameworks
- Agents
- Offers
- Tasks
- Logs



Mesos Tasks

Status

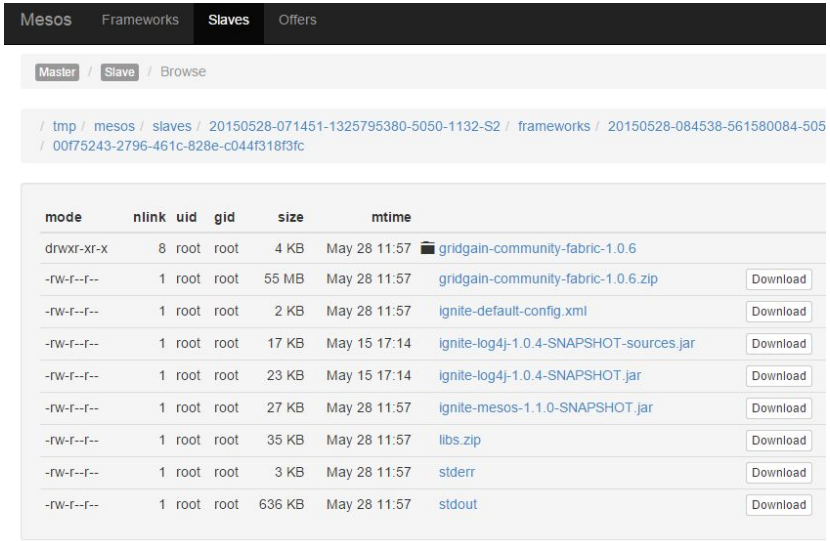
- TASK_STAGING
- TASK_STARTING
- TASK_RUNNING
- TASK_FINISHED
- TASK_KILLING
- TASK_KILLED
- TASK_FAILED
- TASK_LOST
- TASK_ERROR



Mesos task sandbox

Directory with task-specific resources

- Mesos Fetcher downloads resources before task starts
- Configuration files
- Binaries
- `stdout` and `stderr` logs



http://mesos.apache.org/documentation/latest/fetcher
http://mesos.apache.org/documentation/latest/sandbox

Mesos State

- Cluster-wide available at *http://$MASTER:5050/state.json*

```
{
  "cluster": "testcluster",
  "git_sha": "2dd7f7ee115fe00b8e098b0a10762a4fa8f4600f",
  "leader": "master@172.17.0.4:5050",
  // ... SNIP ...
}
```

- Agent info available at *http://$AGENT:5051/state.json*

```
{
  // ... SNIP ...
  "hostname": "172.17.0.5",
  "git_tag": "0.25.0",
  "master_hostname": "172.17.0.4",
  "attributes": {},
  "id": "4adbc43c-20c4-4a5e-bebf-99c0c4d20f8c-S0"
}
```

Mesos Concepts

- Isolator
- Containerizer
- Persistent volumes
- Dynamic reservations
- Modules and hooks
- Leader election
- Maintenance
- IP per container

http://mesos.apache.org/documentation/latest/index.html

Mesos implementation

- Written in C++
- Based on Actor model via libprocess library
- Internal communication is done via Google protocol buffers

Blog post by Frank Scholten @ CS blog

http://container-solutions.com/how-protocol-buffers-are-used-in-mesos-framework-development

- Resource isolation via cgroups
- Many components are pluggable
- UI written in Angular

Mesos Frameworks

Marathon

A cluster-wide init and control system for services in cgroups or Docker containers

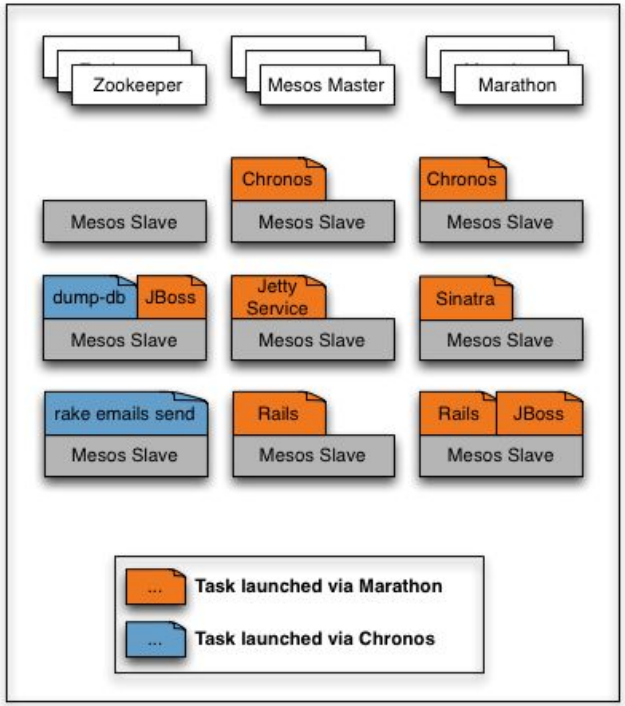
Download Marathon v0.15.2

v0.15.2 SHA-256 Checksum • v0.15.2 Release Notes

Chronos

A fault tolerant job scheduler for Mesos which handles dependencies and ISO8601 based schedules

Download Chronos v2.4.0

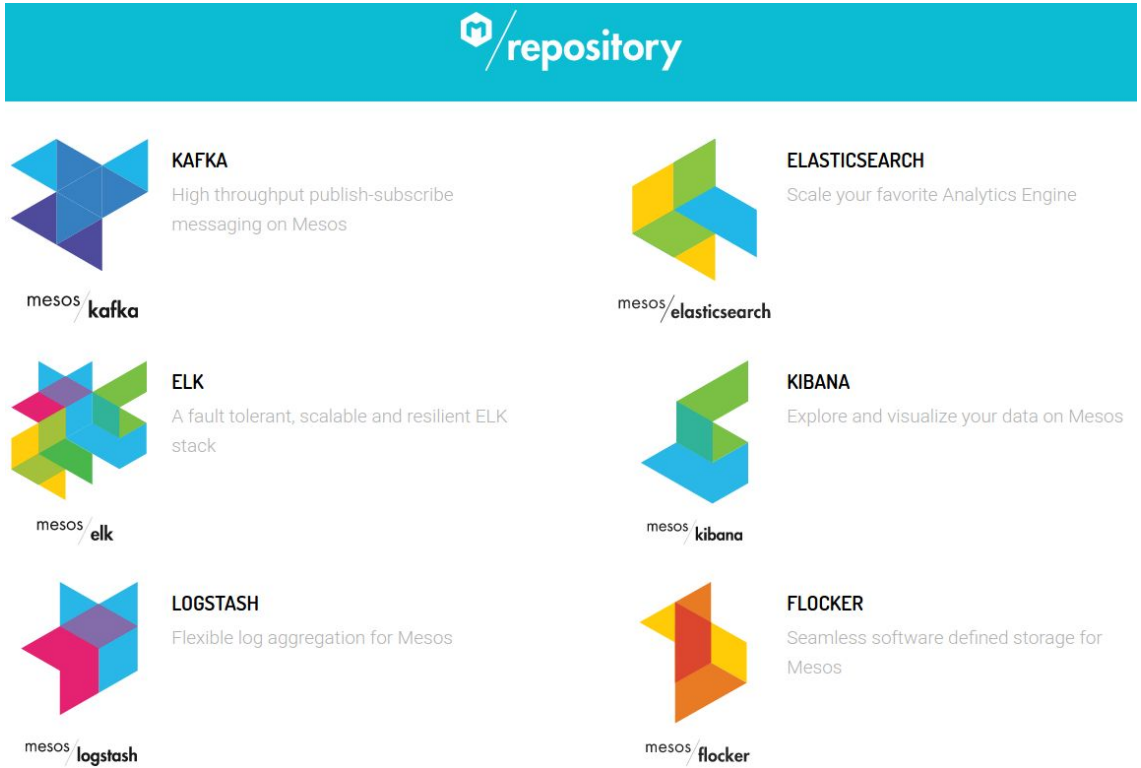


Marathon

Framework for deploying long running services and Docker containers

- Automatic restart
- Scaling
- Healthchecks
- Placement Constraints
- API
- Created by Mesosphere
- Comes out of the box in minimesos

Mesos Frameworks



Apache MESOS

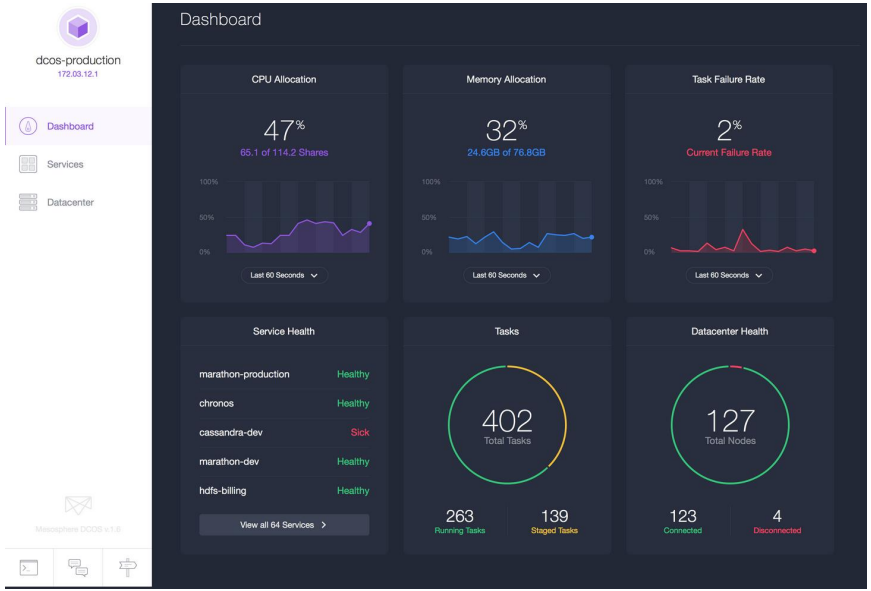
Repository of Mesos frameworks

http://mesosframeworks.com

DC/OS

- Datacenter Operating System
- Distribution for Apache Mesos
- Created by Mesosphere
- Open Sourced in April
- UI, cli & DC/OS packages

https://dcos.io



Creating a Mesos cluster

- Install Mesos Master, Mesos Agents and create a Zookeeper ensemble
- Provision a cluster with our Terraform module

  https://github.com/ContainerSolutions/terraform-mesos

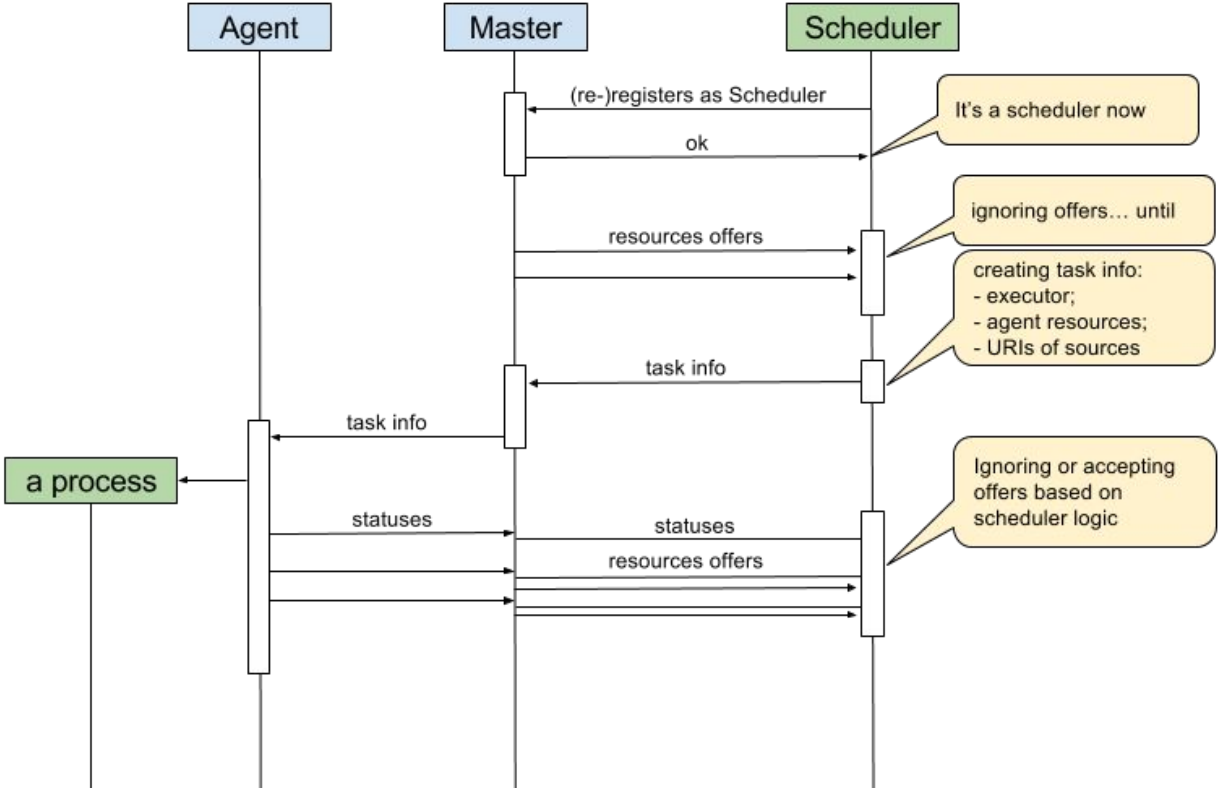
  Blog post by Jaroslav Holub @ CS blog

  http://container-solutions.com/how-to-set-up-mesos-on-google-cloud-with-terraform

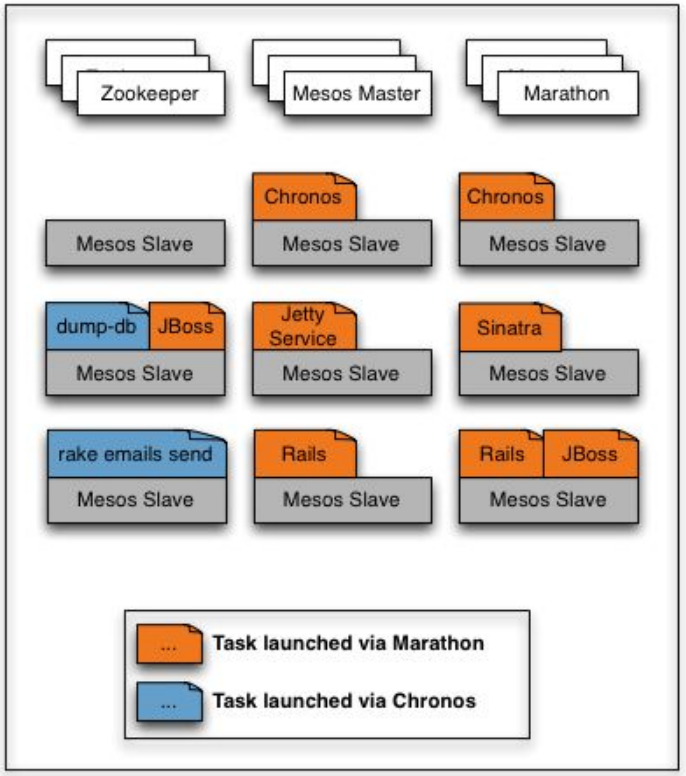
- Use minimesos! ;-)

scheduler workflow

Sequence Diagram



Deploying frameworks via Marathon



minimesos

Why minimesos?

- developed a number of frameworks: ElasticSearch, LogStash, Kibana
- taking minutes to deploy and to ruin someone else's tests
- costs of having Google Cloud clusters almost per developer
- debugging is nearly impossible - rely on logs

mesos elk

ELK

A fault tolerant, scalable and resilient ELK stack

mesos kibana

KIBANA

Explore and visualize your data on Mesos

mesos logstash

LOGSTASH

Flexible log aggregation for Mesos

mesos flocker

FLOCKER

Seamless software defined storage for Mesos

mesos elasticsearch

ELASTICSEARCH

Scale your favorite Analytics Engine

Why minimesos?

- Build classes
- Deploy framework

  (somewhere far away)
- …
- Check logs
- `TASK_LOST` :-(
- ....Yawn!

- Long feedback loop
- Hard to do end-to-end feature testing
- Many components, many possible failures

minimesos

Why minimesos?

Needs

- unit and integration tests as local builds and CI part
- running cluster locally

Solutions

- minimesos Java API
- minimesos CLI

minimesos CLI. Part I

- Installation

```
curl -sSL https://minimesos.org/install | sh
```

- Start default cluster

```
minimesos init; minimesos up
```

- Destroy running cluster

```
minimesos destroy
```

minimesos CLI. Part II

- Initializing configuration file

```
minimesos init
```

- Installing an application through Marathon

```
minimesos install --marathonFile tasks/task.json
```

- Retrieving Mesos state

```
minimesos state
```

minimesos CLI. Demo

- `curl -sSL https://minimesos.org/install | sh`
- `minimesos up`
- Mesos Master and Marathon UI
- `docker ps; minimesos destroy; docker ps -a`
- current directory in IntelliJ
- `minimesos init`
- configuration file editing
- `minimesos up; docker ps` and Mesos Master UI
- app.json and app.sh for Marathon
- `minimesos install --marathonFile tasks/app.json`
- Marathon and Master UI; kill a process; Marathon and Master UI
- es.json and `minimesos install --marathonFile tasks/es.json`
- Marathon, Master and ElasticSearch UI

minimesos Java API

```
@ClassRule
public static final MesosClusterTestRule RULE = MesosClusterTestRule.fromFile("src/test/resources/configFiles/minimesosFile-mesosClusterTest");

public static MesosCluster CLUSTER = RULE.getMesosCluster();


@Test
public void mesosClusterCanBeStarted() throws Exception {
  MesosMaster master = CLUSTER.getMaster();
  JSONObject stateInfo = master.getStateInfoJSON();
  assertEquals(3, stateInfo.getInt("activated_slaves"));
}
```

minimesos Java API. Demo

```
@BeforeClass
public static void startScheduler() throws Exception {

    String ipAddress = CLUSTER.getMasterContainer().getIpAddress();

    LOGGER.info("Starting Scheduler, connected to " + ipAddress);
    SchedulerContainer scheduler = new SchedulerContainer(CONFIG.dockerClient, ipAddress);

    // Cluster now has responsibility to shut down container
    CLUSTER.addAndStartContainer(scheduler);

    LOGGER.info("Started Scheduler on " + scheduler.getIpAddress());
}

@Test
public void testNodeDiscoveryRest() {

    long timeout = 120;
    DockerContainersUtil util = new DockerContainersUtil(CONFIG.dockerClient);

    final Set<String> ipAddresses = new HashSet<>();
    Awaitility.await("9 expected executors did not come up").atMost(timeout, TimeUnit.SECONDS).until(() -> {
        ipAddresses.clear();
        ipAddresses.addAll(util.getContainers(false).filterByImage(Configuration.DEFAULT_EXECUTOR_IMAGE).getIpAddresses());
        return ipAddresses.size() == 9;
    });

    HelloWorldResponse helloWorldResponse = new HelloWorldResponse( ipAddresses, Arrays.asList(8080, 8081, 8082), timeout );
    assertTrue("Executors did not come up within " + timeout + " seconds", helloWorldResponse.isDiscoverySuccessful());

}
```

minimesos. Upcoming changes

0.8.1 - Released this week

- Minimesos IP address tokens for each role
- New commands: ps & uninstall

0.9.0 - Coming soon

- Weave Net service discovery

a bit more distant releases

- REST API and clients in several languages
- distributed minimesos cluster

Exercises

Exercises (30 minutes)

- Let’s try out minimesos in a browser using Katacoda!

- Go to https://minimesos.org/try
- Now run a few scenarios from Katacoda on your laptop

15 minute Break

Exercises (60 minutes)

- Clone https://github.com/ContainerSolutions/learning-mesos-with-minimesos-workshop
- Check out README.md for exercises
- (suggestion) Work in pairs

15 minute Break

Experimentation

Experiment! Some suggestions...

- Bonus exercise: Installing ELK on minimesos
- Deploy your own applications
- Try out Mesos Starter, Mesos Framework or minimesos Maven plugin

  https://github.com/ContainerSolutions/mesos-starter
  https://github.com/ContainerSolutions/mesosframework
  https://github.com/ContainerSolutions/minimesos-maven-plugin

- Hack on minimesos!

  https://github.com/ContainerSolutions/minimesos/issues

Wrap up and Q&A

Questions?

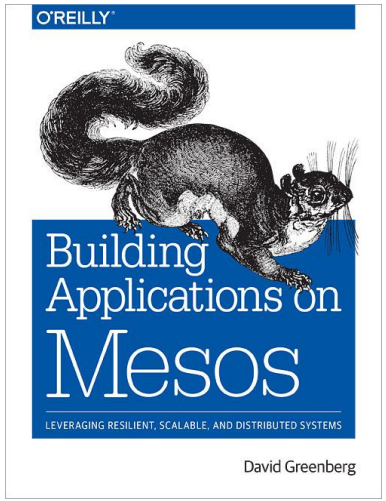
How to contribute to minimesos

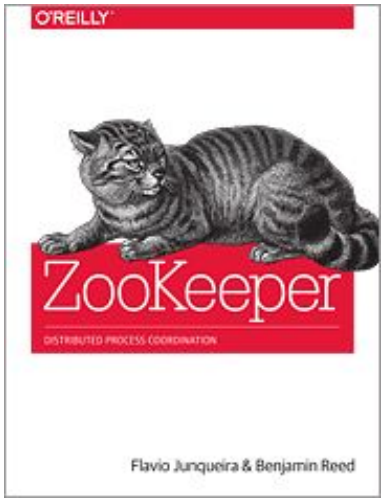
- Found a bug or issue? Let us know
- PRs are appreciated!
- Your PR is auto built by Travis CI

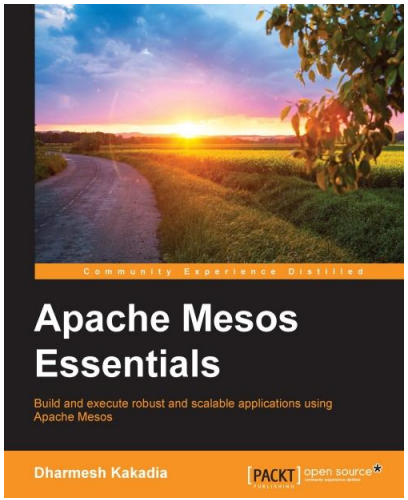
Blog post by Viktor Sadovnikov @ CS blog

http://container-solutions.com/moved-ci-jenkins-travis

Further reading







MesosCon Amsterdam

1 August - 1 September 2016

Hilton Amsterdam

Organized by the Linux Foundation

http://events.linuxfoundation.org/events/mesoscon-europe

We like to know what you think

Fill in the minimesos survey http://bit.ly/1rEdJz0

Let’s stay in touch

https://github.com/ContainerSolutions/minimesos

http://minimesos.org

http://www.mesosframeworks.com

http://www.container-solutions.com

@containersoluti @minimesos @Frank_Scholten @sadovnikov

Thank you!