

# Python + Docker + Mesos

Welcome

---

**Piotr Szwed**

*DevOps Evangelist, Freelancer*

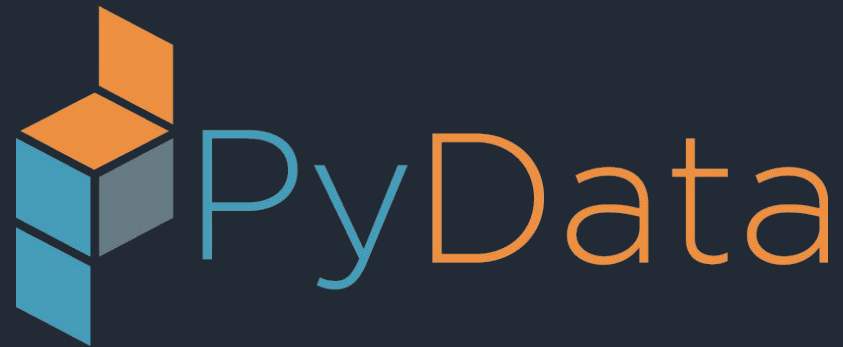
pszwed@gmail.com

**Jan Waś**

*Developer*

janek.jan@gmail.com



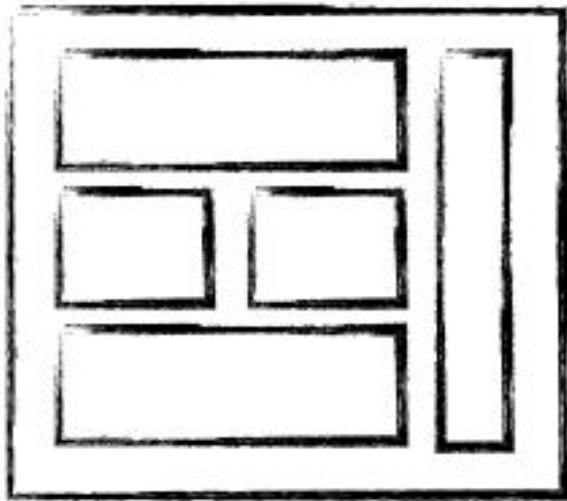


## Warsaw Mesos User Group

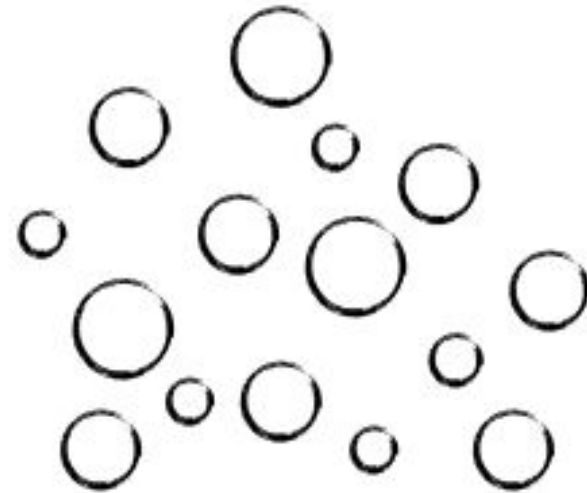




1. Theory
2. Live Demo: Hello World
3. Live Demo: Docker + Django + Middleware



MONOLITHIC/LAYERED



MICRO SERVICES

# Monolithic vs Microservices



Monolithic

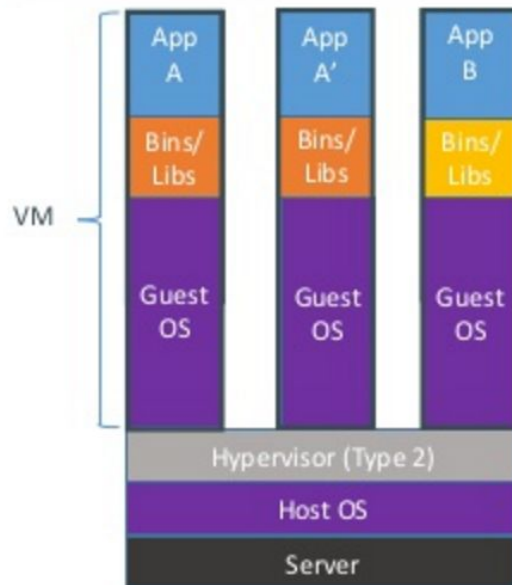


Microservices

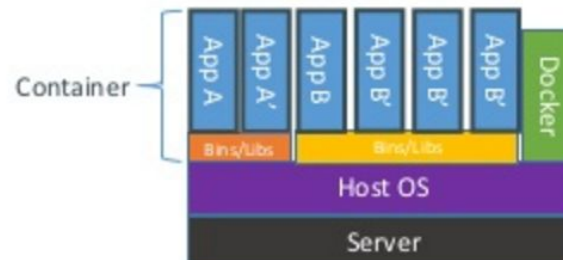




## Containers vs. VMs



Containers are isolated, but share OS and, where appropriate, bins/libraries







## What is Mesos?

### A distributed systems kernel

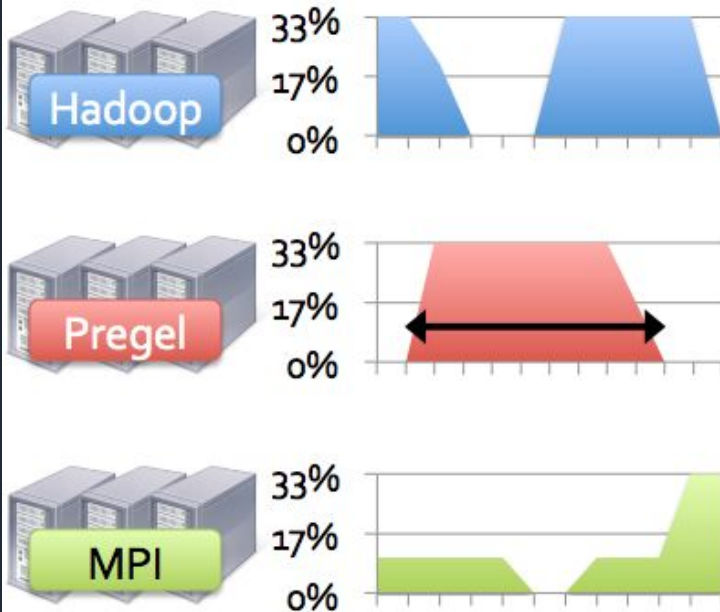
Mesos is built using the same principles as the Linux kernel, only at a different level of abstraction. The Mesos kernel runs on every machine and provides applications (e.g., Hadoop, Spark, Kafka, Elastic Search) with API's for resource management and scheduling across entire datacenter and cloud environments.

## Project Features

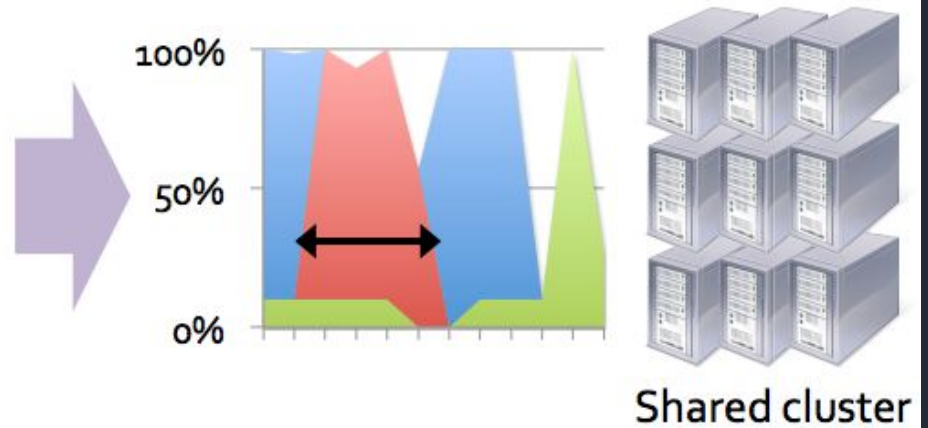
- Scalability to 10,000s of nodes
- Fault-tolerant replicated master and slaves using ZooKeeper
- Support for Docker containers
- Native isolation between tasks with Linux Containers
- Multi-resource scheduling (memory, CPU, disk, and ports)
- Java, Python and C++ APIs for developing new parallel applications
- Web UI for viewing cluster state



## Today: static partitioning

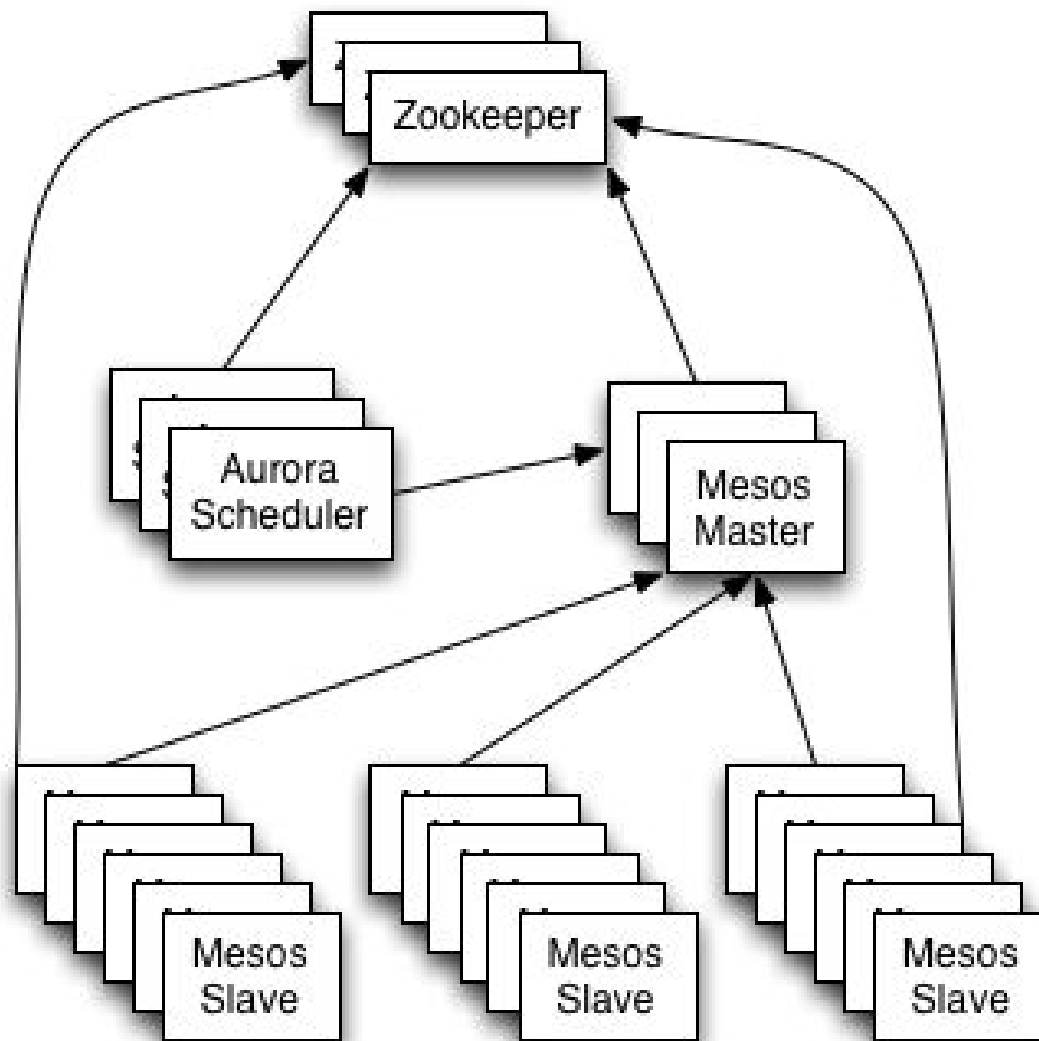


## Mesos: dynamic sharing

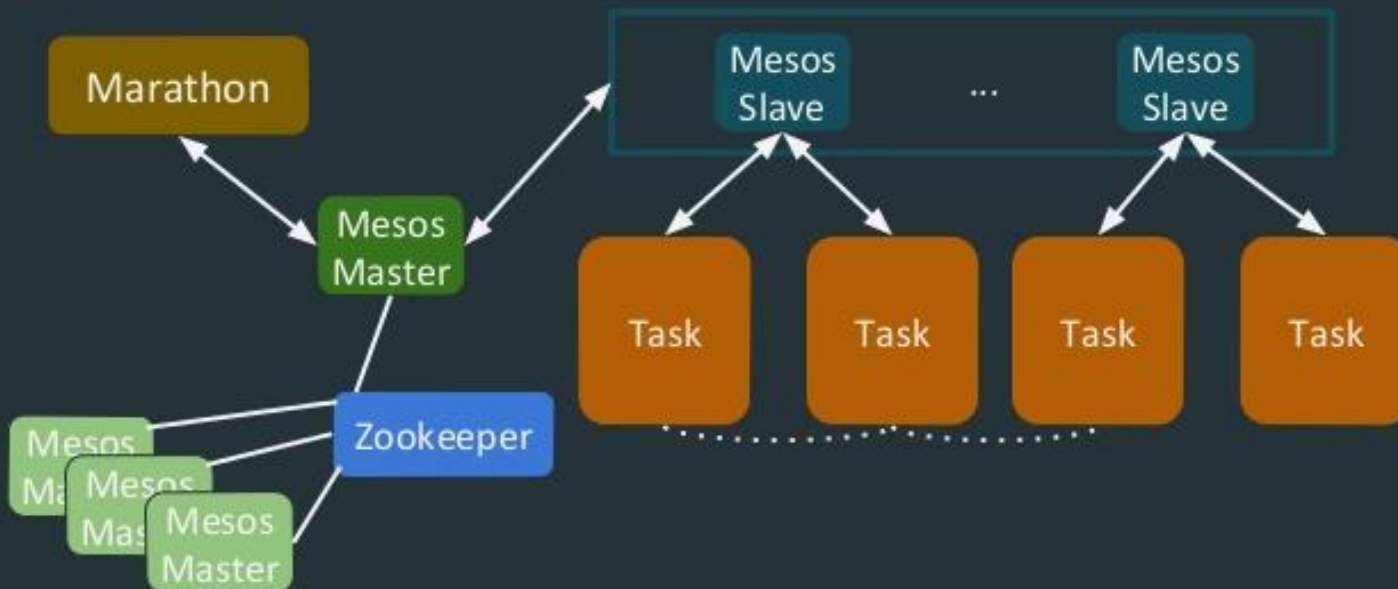


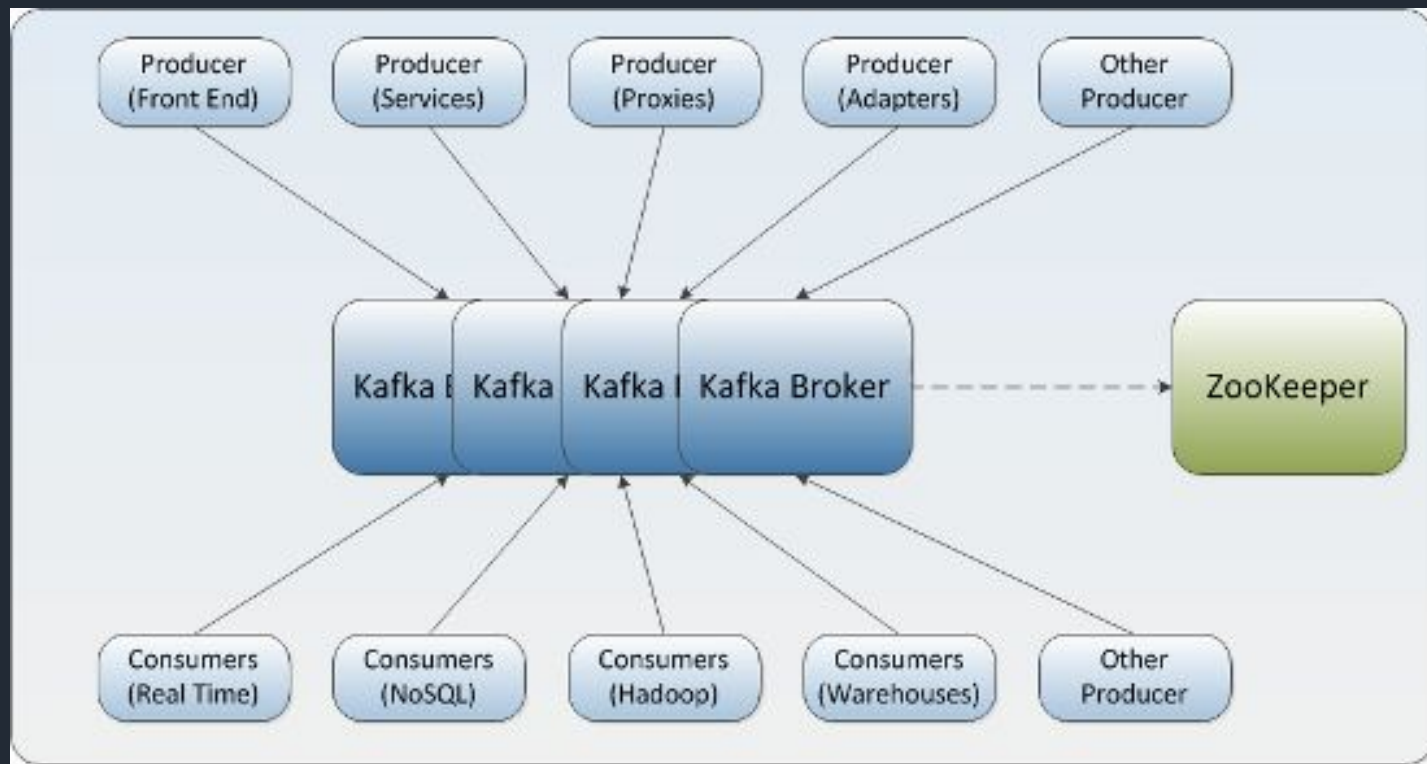
## Example Mesos Stack:

- Zookeeper (Yahoo)
- Exhibitor (Netflix)
- Chronos (Cron)
- Marathon (Mesos GUI)
- Kafka (message broker/LinkedIn)
- Mesos Master/Slave (Apache Foundation)
- Synapse/Nerve (ServiceDiscovery/Airbnb)
- Docker
- Datadog/NewRelic
- SaltStack (Ansible, Chef, Puppet)



## Mesos-on-Mesosphere





# Mesos - frameworks

## Long Running Services

- [Aurora](#) is a service scheduler that runs on top of Mesos, enabling you to run long-running services that take advantage of Mesos' scalability, fault-tolerance, and resource isolation.
- [Marathon](#) is a private PaaS built on Mesos. It automatically handles hardware or software failures and ensures that an app is “always on”.
- [Singularity](#) is a scheduler (HTTP API and web interface) for running Mesos tasks: long running processes, one-off tasks, and scheduled jobs.
- [SSSP](#) is a simple web application that provides a white-label “Megaupload” for storing and sharing files in S3.

## Big Data Processing

- [Cray Chapel](#) is a productive parallel programming language. The Chapel Mesos scheduler lets you run Chapel programs on Mesos.
- [Dpark](#) is a Python clone of Spark, a MapReduce-like framework written in Python, running on Mesos.
- [Exelixi](#) is a distributed framework for running genetic algorithms at scale.
- [Hadoop](#) Running Hadoop on Mesos distributes MapReduce jobs efficiently across an entire cluster.
- [Hama](#) is a distributed computing framework based on Bulk Synchronous Parallel computing techniques for massive scientific computations e.g., matrix, graph and network algorithms.
- [MPI](#) is a message-passing system designed to function on a wide variety of parallel computers.
- [Spark](#) is a fast and general-purpose cluster computing system which makes parallel jobs easy to write.
- [Storm](#) is a distributed realtime computation system. Storm makes it easy to reliably process unbounded streams of data, doing for realtime processing what Hadoop did for batch processing.



# Batch Scheduling

- [Chronos](#) is a distributed job scheduler that supports complex job topologies. It can be used as a more fault-tolerant replacement for Cron.
- [Jenkins](#) is a continuous integration server. The mesos-jenkins plugin allows it to dynamically launch workers on a Mesos cluster depending on the workload.
- [JobServer](#) is a distributed job scheduler and processor which allows developers to build custom batch processing Tasklets using point and click web UI.
- [GoDocker](#) is a batch computing job scheduler like SGE, Torque, etc. It schedules batch computing tasks via webui, API or CLI for system or LDAP users, mounting their home directory or other shared resources in a Docker container. It targets scientists, not developers, and provides plugin mechanisms to extend or modify the default behavior.

# Data Storage

- [Cassandra](#) is a performant and highly available distributed database. Linear scalability and proven fault-tolerance on commodity hardware or cloud infrastructure make it the perfect platform for mission-critical data.
- [ElasticSearch](#) is a distributed search engine. Mesos makes it easy to run and scale.
- [Hypertable](#) is a high performance, scalable, distributed storage and processing system for structured and unstructured data.
- [Tachyon](#) is a memory-centric distributed storage system enabling reliable data sharing at memory-speed across cluster frameworks.

**THANKS FOR WATCHING!**

