



flight





flight

**Joe Smith**

Site Reliability Engineer | @Yasumoto





# Twitter's Production Scale

Aurora and Mesos Operations



# Agenda

- History of Aurora and Mesos at Twitter
- Cluster Management
- Tradeoffs





Apache Aurora



Apache Mesos



# Key Principles

- **Empower** Service Owners
- Accountability for services and infrastructure
- **Reliability** is the foundation of infrastructure





# Twitter 2012





Puppet





# Roadblocks



# Capacity

- Statically Allocated Hosts
- Requesting new hardware
- Expanding existing services



# Experimentation

- New canary instances
- Prove out a new service



# Configuration Changes

- Puppet changes required Ops





# Joe's First Week



\$\$\$





# Adoption

Vote of Confidence



# Refined Primitives

- Health Checks for Services
- Improved Deployments for Thousands of instances
- Environments to test and canary new builds and configs



# Better Abstraction

- Engineers **do not worry** about which host they are running on
  - Kernel upgrades
  - Hardware failure detection
  - Host Repair and Replacement



# Aurora Mesos Cluster Growth







# Cluster Maintenance

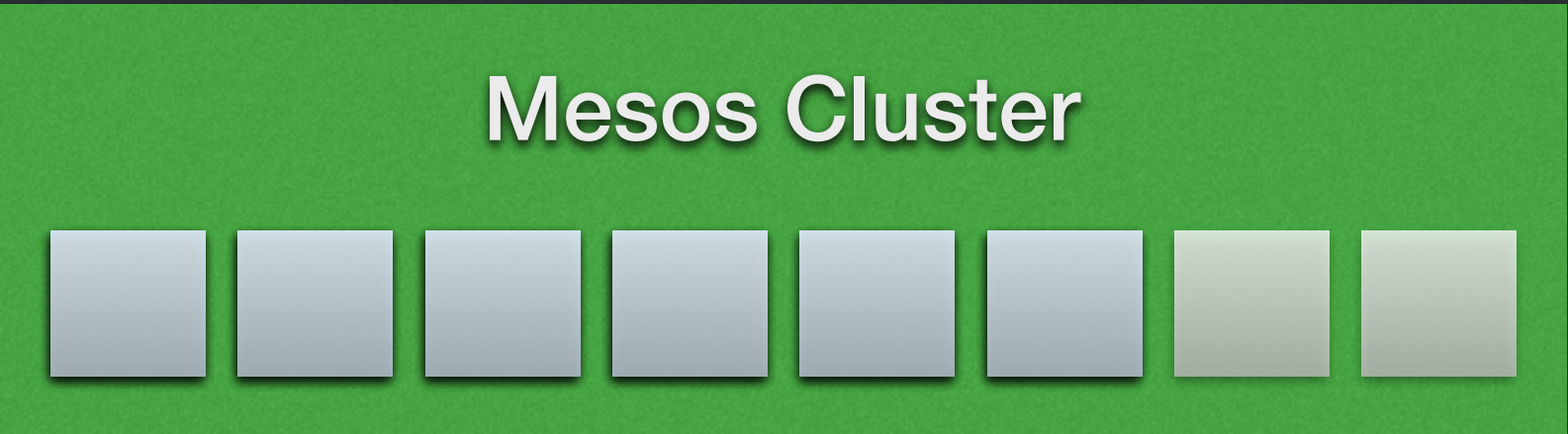
More Machines, Fewer Operators



aurora\_admin.pex



Aurora Schedulers



```
aurora_admin start_maintenance -filename=hosts.txt
```

```
aurora_admin drain_hosts -filename=hosts.txt
```

```
aurora_admin end_maintenance -filename=hosts.txt
```



# Use One Tool For Maintenance

- Leverage one set of rock-solid APIs
  - Great documentation



```
vector = self._client.sla_get_safe_domain_vector(self.SLA_MIN_JOB_INSTANCE_COUNT, hostnames)
host_groups = vector.probe_hosts(
    percentage,
    duration.as_(Time.SECONDS),
    grouping_function)
results, unsafe_hostnames = format_sla_results(host_groups, unsafe_only=True)
if results:
    print_results(results)
    return unsafe_hostnames
return unsafe_hostnames
```

[https://github.com/apache/aurora/blob/master/src/main/python/apache/aurora/admin/host\\_maintenance.py](https://github.com/apache/aurora/blob/master/src/main/python/apache/aurora/admin/host_maintenance.py)



# No Snowflakes

- Treat every machine the same
- Resist the urge to allow special-cases
  - Limiting especially in the short-term





# Tradeoffs

Not all glory



# Centralized Management

- One place to **enforce policy and view cluster state**
- Only one single point of failure to lose control of the cluster
  - Plan for Unplanned Outages
- Logic to the Edges: Let the cluster self-manage itself



# Efficiency

- Drive higher utilization by co-locating tasks
- Interference between tasks using shared resources



# Opinionated Service Design

- Application Engineers **focus on business-specific logic**
- **Difficult to customize** workflow and diverge





40%

Twitter's Datacenter has  
migrated to Aurora and Mesos





Thank You

@Yasumoto