2000 nodes and beyond

how we scaled Kubernetes to 60,000-container clusters and where we're going next.

KubeCon Seattle
Nov 8th 2016
In the beginning...

During the cold winter of 2015, before v1.0, we knew we wanted to improve the scale of Kubernetes clusters, but we didn't even knew where we are and what it means...
In the beginning...

...we wrote simple tests to put some load on the cluster...

...those were Density and Load tests and results were bad.
First SLO

99th percentile of API call response times should be under 1s
Second SLO

99th percentile of e2e Pod startup latency should be under 5s
The story begins...
v1.0

100 Nodes
Life of an API request v1.0

API server

etcd

JSON

Go struct

JSON
Life of an API request

If you're interested in inner workings of the API server come to talk

Life of a Kubernetes API request

Today 3:40 pm

by Daniel Smith
Life of an API request v1.0
API object translation

v1 object \[\rightarrow\] reflection based conversion \[\leftarrow\] internal object
API object translation
v1.1

250 Nodes
Life of a watch request v1.0
Life of a watch request v1.1

API server

Cacher

etcd
v1.2

1000 Nodes
Testing at scale
Testing at scale
Testing at scale
Simulated clusters (Kubemark)

- Kubemark master
- HollowNodes
- External Kubernetes cluster
Life of an API request v1.1-

API server

etcd
v1.3

2000 Nodes
Life of an API request v1.2-

1. User sends a request to the API server.
2. The API server processes the request and retrieves data from etcd.
3. The data is converted from JSON format to Go struct format.
4. The Go struct is then converted back to JSON format for handling.

Google Cloud Platform
Life of an API request v1.3

API server

etcd

protobuf

Go struct

JSON
v1.4

2000 Nodes
Slacking off?

Preparation for changes coming in 1.5
v1.5

5000 Nodes (?)
etcd v2 -> v3
Why all this is really important?

Why do we care about API server that much?
Why all this is really important?

Create Deployment

API server

etcd

Deployment Controller

ReplicaSet Controller

Scheduler

Kubelet
Why all this is really important?

API server

Create Deployment

etcd

Deployment Controller
ReplicaSet Controller
Scheduler
Kubelet
Why all this is really important?
Why all this is really important?

- API server
- etcd
- Observed Deployment creation
- Deployment Controller
- ReplicaSet Controller
- Scheduler
- Kubelet
Why all this is really important?

API server

Deployment Controller

ReplicaSet Controller

Scheduler

Kubelet

etcd

Create ReplicaSet
Why all this is really important?

- **Deployment Controller**
- **ReplicaSet Controller**
- **Scheduler**
- **Kubelet**

API server - Create ReplicaSet - etcd
Why all this is really important?

etcd

Observed ReplicaSet creation in etcd

API server

Deployment Controller

ReplicaSet Controller

Scheduler

Kubelet
Why all this is really important?

- Deployment Controller
- Observed ReplicaSet creation
- API server
- Observed ReplicaSet creation
- Replication Controller
- Observed ReplicaSet creation
- Scheduler
- Kubelet
- etcd
Why all this is really important?

- API server
- etcd
- Create Pod
- Deployment Controller
- ReplicaSet Controller
- Scheduler
- Kubelet
Why all this is really important?

API server

Create Pod

etcd

Deployment Controller

ReplicaSet Controller

Scheduler

Kubelet
Why all this is really important?

Deployment Controller
ReplicaSet Controller
Scheduler
Kubelet

API server
Observed Pod creation in etcd

etcd

Google Cloud Platform
Why all this is really important?

API server

Deployment Controller
ReplicaSet Controller
Scheduler
Kubelet

etcd

Observed Pod creation
Observed Pod creation
Observed Pod creation

Why all this is really important?

- Google Cloud Platform
- Why all this is really important?
- API server
- etcd
- Deployment Controller
- ReplicaSet Controller
- Scheduler
- Kubelet
- Bind Pod to a Node
Why all this is really important?
Why all this is really important?

Observed PodSpec update in etcd

API server

etcd

Deployment Controller

ReplicaSet Controller

Scheduler

Kubelet
Why all this is really important?

- Observed PodSpec update
- Observed Pod binding

Deployment Controller
ReplicaSet Controller
Scheduler
Kubelet

API server
etcd
Why all this is really important?

- Google Cloud Platform
- Why all this is really important?
- API server
- etcd
- Start Pod and update PodStatus
- Deployment Controller
- ReplicaSet Controller
- Scheduler
- Kubelet

Google Cloud Platform
Why all this is really important?

API server

Update PodStatus

Deployment Controller
ReplicaSet Controller
Scheduler
Kubelet

etcd
Why all this is really important?

API server

etcd

Deployment Controller

ReplicaSet Controller

Scheduler

Kubelet

Observed PodStatus update in etcd
Why all this is really important?

- API server
- etcd
- Deployment Controller
- ReplicaSet Controller
- Scheduler
- Kubelet
- Observed PodStatus update
Second SLO

99th percentile of e2e Pod startup latency, from the time when the Pod is created to the time when user-space watch sees that the Pod is running, should be under 5s assuming that container image is present on the machine.
Most steps was already discussed. Except scheduling itself.
Most steps was already discussed. Except scheduling itself.
Simplified scheduler internals

Pod → Predicates → Nodes → Priorities → Scheduled Pods → Binding
Parallelization of Priorities

Nodes — Scheduled Pods — Prioritized Nodes
Scalability work was incremental:
• find a bottleneck,
• make it go away,
• repeat
Scalability work was incremental:

- find a bottleneck,
- make it go away,
- repeat

The further you go, the more complex fixes
Wrap up

We aim at 5000 Nodes for v1.5
Wrap up

We aim at 5000 Nodes for v1.5

99th percentile of API call response times should be under 1s

99th percentile of e2e Pod startup latency should be under 5s
SLOs today

Current SLOs are not enough.

We're working on it.
Life of an API request v1.2-
Life of an API request v1.3+

connection

handshake

API server

etcd
v1.6+

5000 Nodes
Give me ObjectMeta of Pod X
Here you go: Pod X
I need to project Pod onto ObjectMeta...
Life of an API request v1.6+

Give me ObjectMeta of Pod X
Life of an API request v1.6+

API server

Crunching… crunching...

etcd
Life of an API request v1.6+

I need to project Pod onto ObjectMeta...
Life of an API request v1.6+

Here you go: ObjectMeta for X