

The background features a vibrant, abstract color gradient transitioning from blue and purple on the left to orange and yellow on the right. A diagonal band of darker purple and red runs across the middle.

AWS
re:Invent

OPN211

How Zalando runs Kubernetes clusters at scale on AWS

Henning Jacobs
Senior Principal
Zalando SE

DAMEN

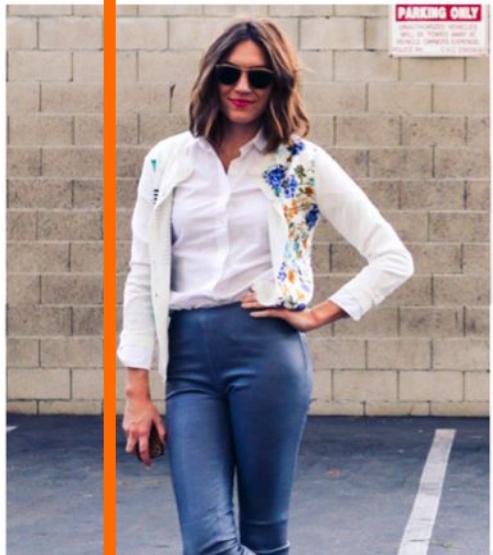
HERREN

KINDER

 zalando

Inspiration

Neu Bekleidung Schuhe Sport Accessoires Wäsche Premium Marken Sale



THE EUROPEAN ONLINE PLATFORM FOR FASHION

UNSER MUST-HAVE

COLOR FLASH

TRENDZOOM

BUSINESS AS USUAL



ZALANDO AT A GLANCE

~ **5.4** billion EUR

revenue 2018

~ **14,000**

employees in
Europe

> **80%**

of visits via
mobile devices

> **300**
million

visits
per
month

> **28**

million
active customers

> **400,000**

product choices

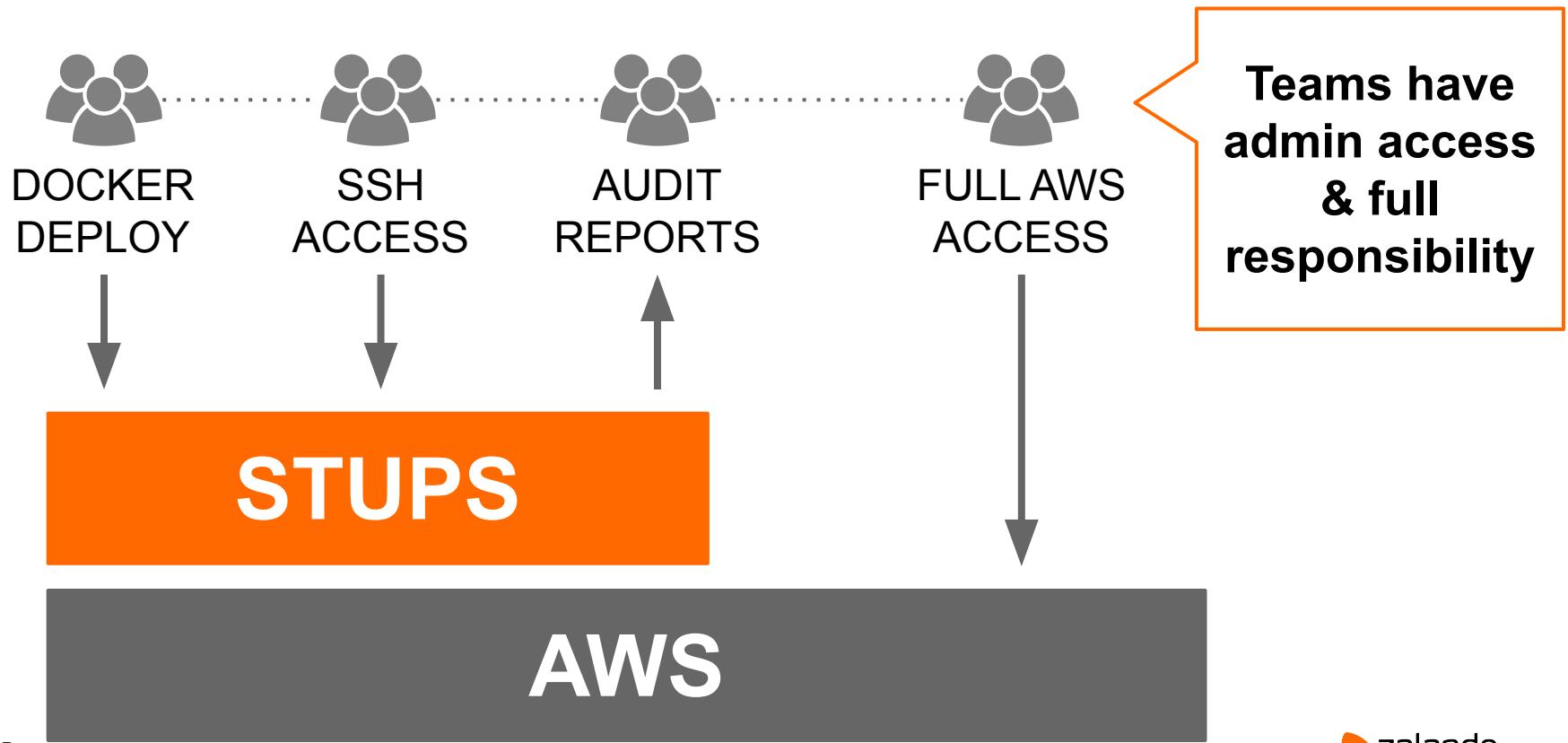
> **2,000**

brands

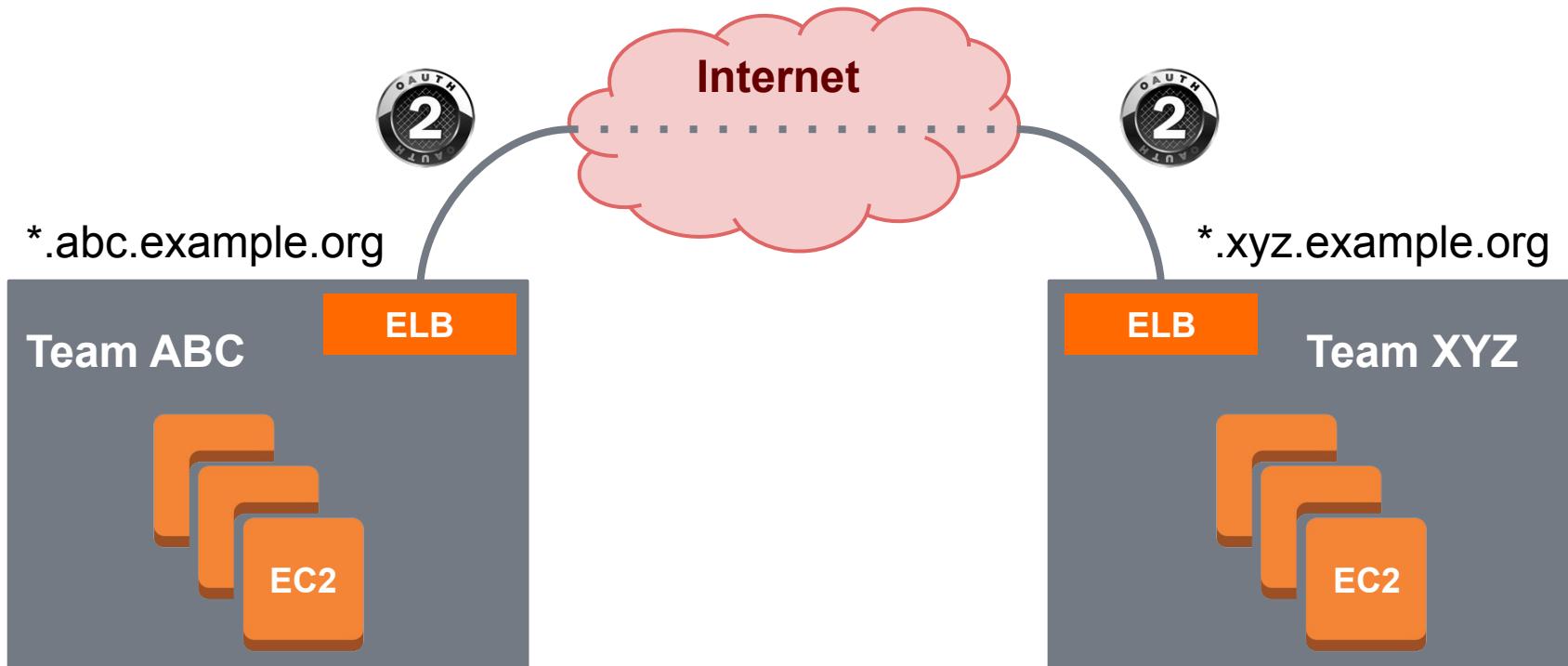
17

countries

2015: JOURNEY INTO THE CLOUD



2015: ISOLATED AWS ACCOUNTS



INFRASTRUCTURE @ ZALANDO



STUPS (toolset around AWS)

AWS accounts per team.

All instances must run the same AMI.

PowerUser access to Production.

You build it, you run EVERYTHING.



Kubernetes

Clusters per product (multiple teams).

Instances are not managed by teams.

Hands off approach.

A lot of stuff out of the box.

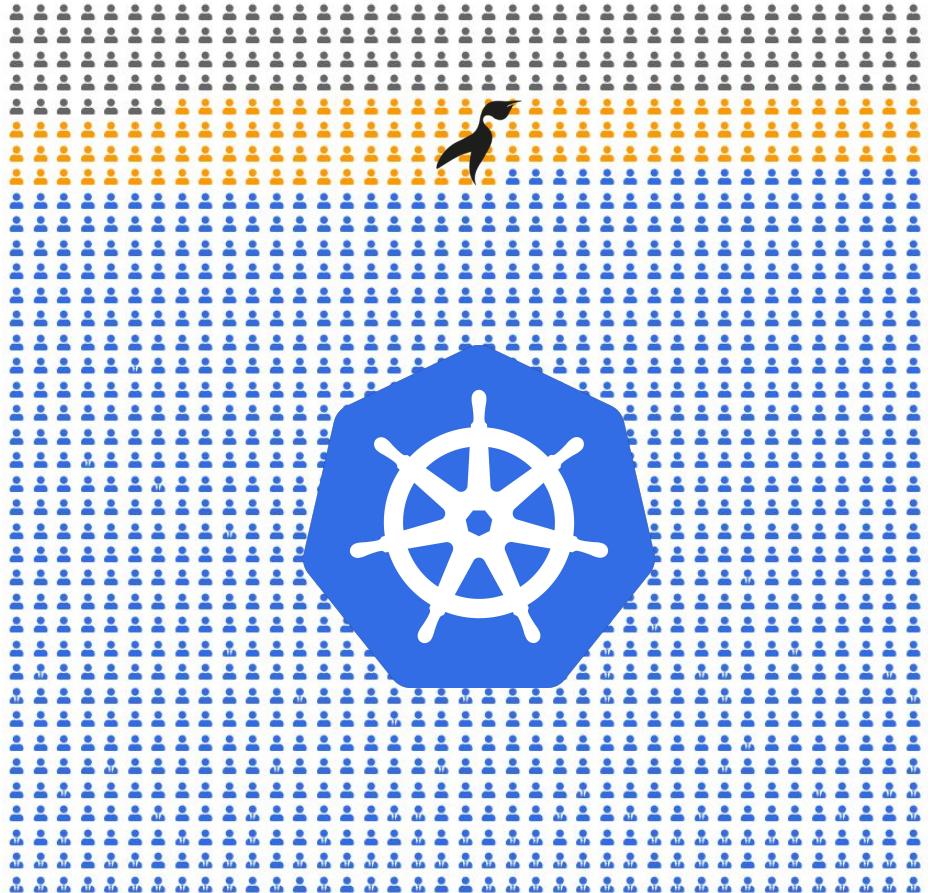
2019: SCALE

396 Accounts



140 Clusters

2019: DEVELOPERS USING KUBERNETES



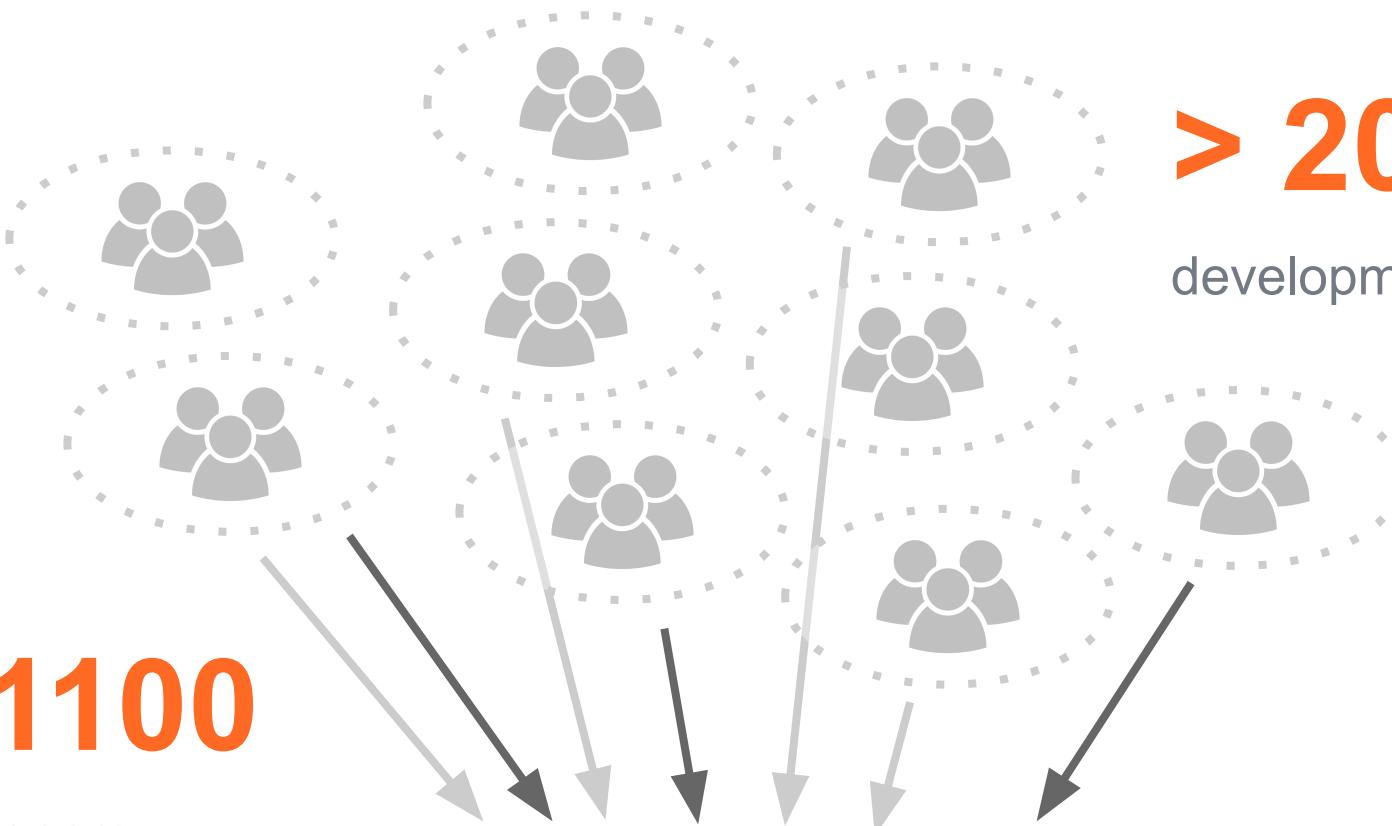
> 1100

developers

Platform

> 200

development teams



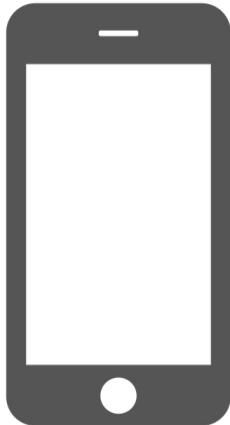
YOU BUILD IT, YOU RUN IT

The traditional model is that you take your software to the wall that separates development and operations, and throw it over and then forget about it. Not at Amazon.

You build it, you run it. This brings developers into contact with the day-to-day operation of their software. It also brings them into day-to-day contact with the customer.

- A Conversation with Werner Vogels, ACM Queue, 2006

ON-CALL: YOU OWN IT, YOU RUN IT

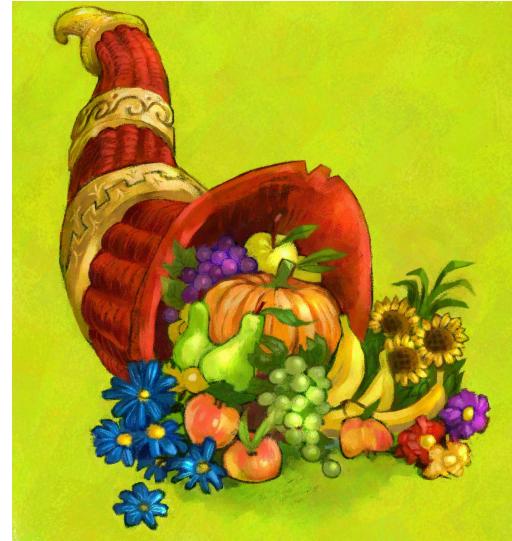


*When things are broken,
we want people with the best
context trying to fix things.*

- [Blake Scrivener, Netflix SRE Manager](#)

GOALS

- No manual operations
- No pet clusters
- Reliability
- Autoscaling
- Latest Kubernetes
- Cost efficient



ARCHITECTURE

Pairs of clusters, each cluster in isolated account

AWS Acc. foobar-test



Cluster
foobar-test

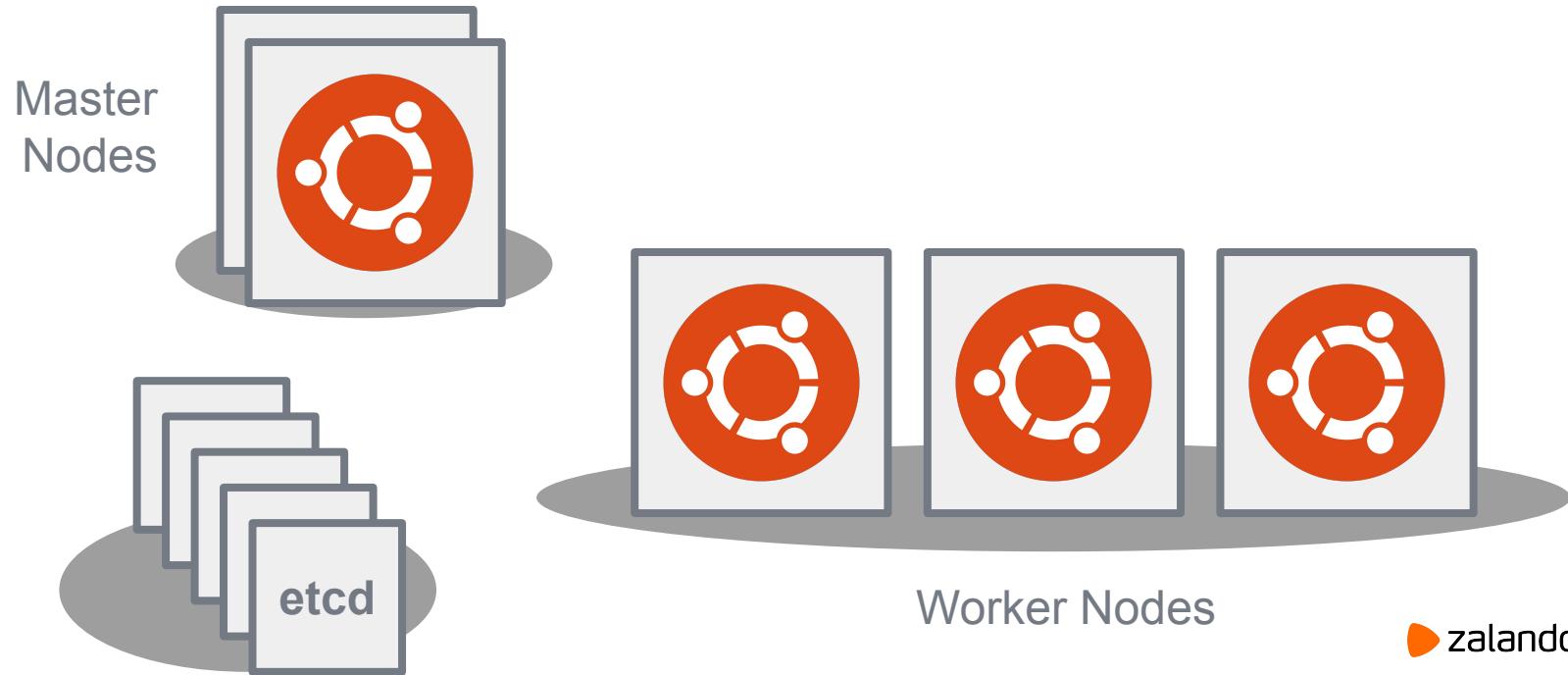
AWS Acc. foobar



Cluster
foobar

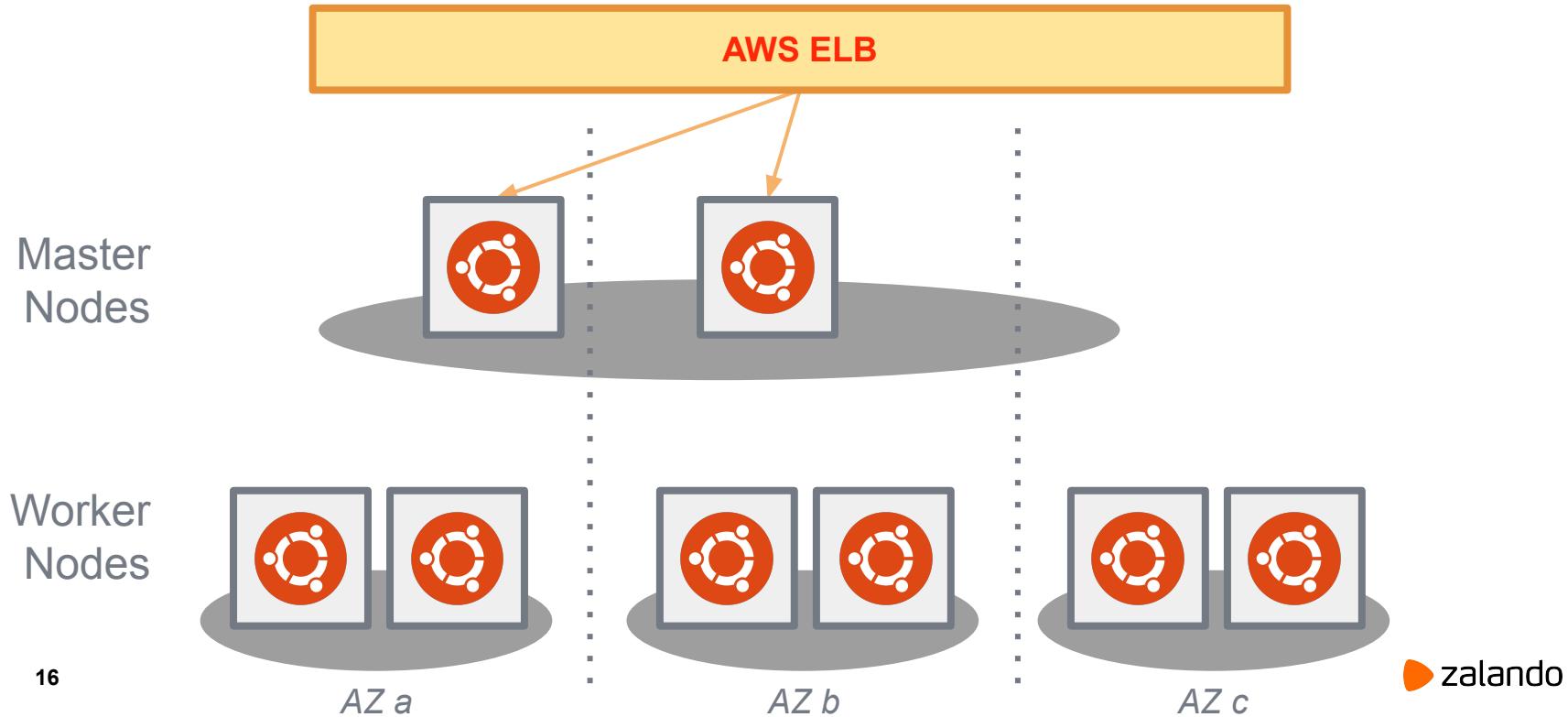
ARCHITECTURE

CloudFormation stacks, node pools w/ self-baked Ubuntu AMI



ARCHITECTURE

<https://cluster-id.example.org>



CLUSTER METADATA (CLUSTER-REGISTRY)

```
clusters:  
- id: "cluster-id"  
  api_server_url: "https://cluster-id.example.org"  
  config_items:  
    Key: "value"  
  environment: "test"  
  region: "eu-central-1"  
  lifecycle_status: "ready"  
  node_pools:  
    - name: "worker-pool"  
      instance_type: "m5.large"  
      min_size: 3  
      max_size: 20
```

CLUSTER CONFIGURATION

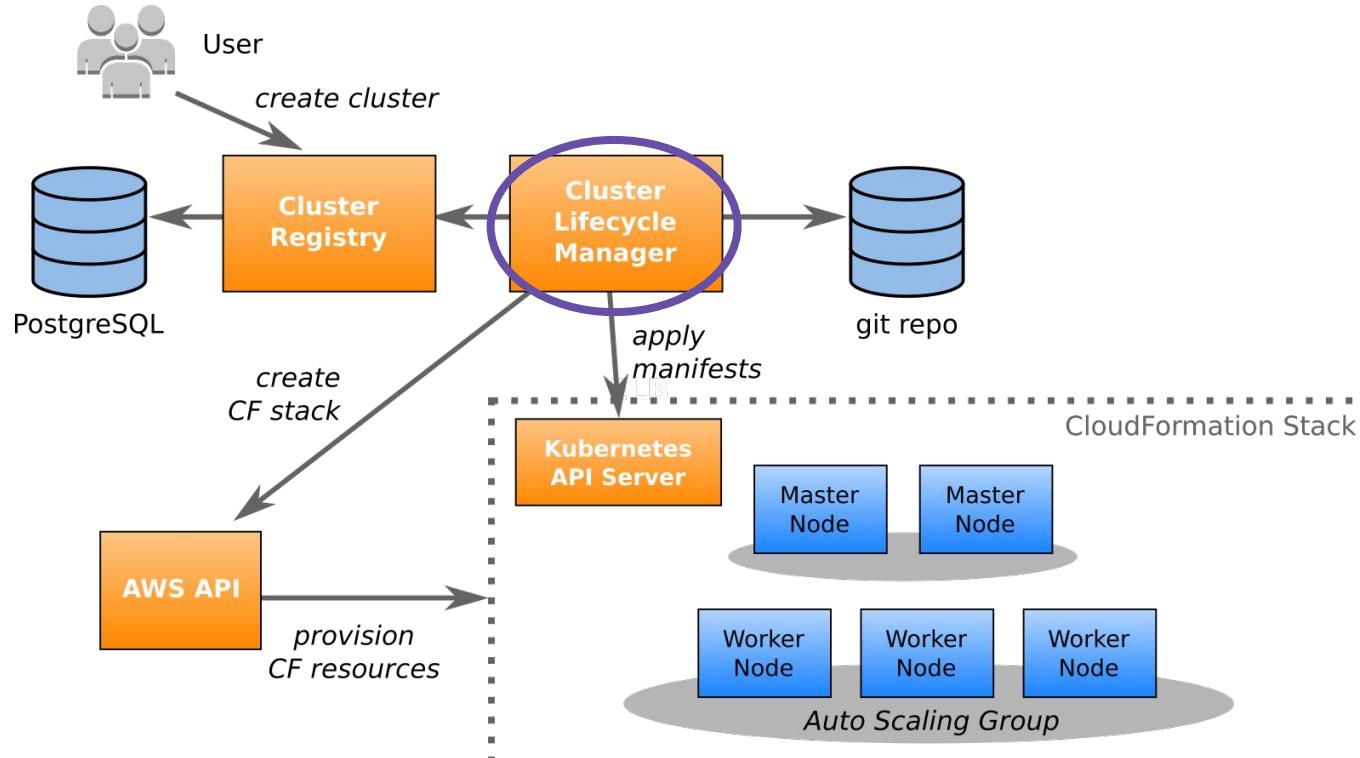
```
cluster
└── cluster.yaml      # Kubernetes cluster stack
└── etcd-cluster.yaml # etcd cluster stack
└── manifests
    └── ...
└── node-pools        # master/worker nodes
    └── ...
```

github.com/zalando-incubator/kubernetes-on-aws

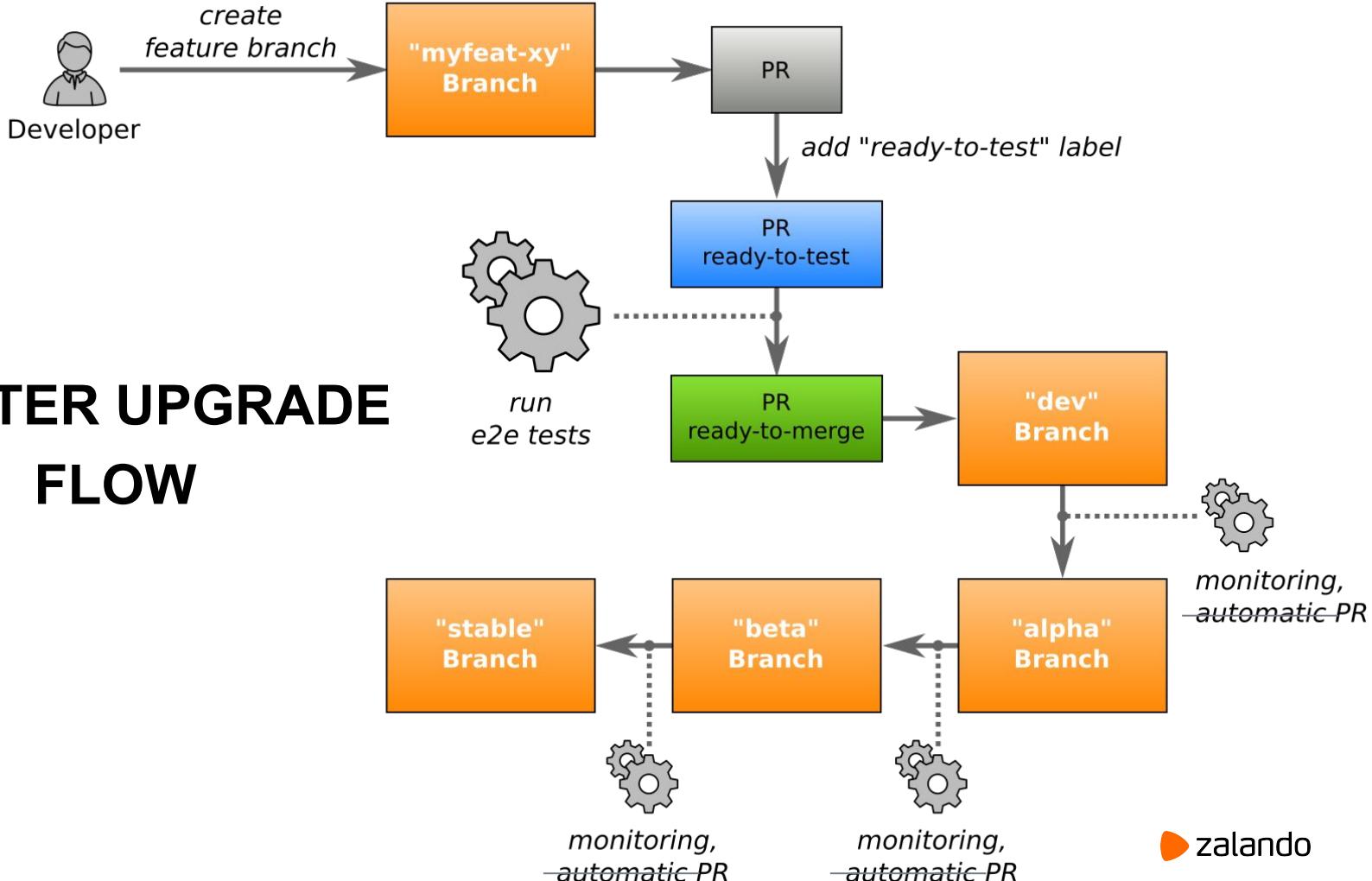
KUBERNETES CLUSTER MANIFESTS

Branch: dev ▾ kubernetes-on-aws / cluster / manifests /		Create new file	Upload files	Find file	History
 mikkeloscar	Merge pull request #2642 from zalando-incubator/feature/enable-local-...	...	✓ Latest commit 0928b6e	3 days ago	
..					
	01-platformcredentialsset	PCS schema: define the schema for tokens		last month	
	01-vertical-pod-autoscaler	Updated CRD and images to v0.6.1		2 months ago	
	01-visibility	Drop quota from visibility namespace		5 days ago	
	02-kube-aws-iam-controller	Change field name		2 months ago	
	admission-control	Update admission controller		7 days ago	
	audittrail-adapter	Add RBAC for audittrail-adapter		2 months ago	
	cadvisor	New directories for docker/kubelet		last month	
	cluster-lifecycle-controller	CA/CLC: fix race condition during eviction		25 days ago	
	coredns-local	Zappr was not working, so docker image didn't pass		24 days ago	
	cron	add cron namespace to all cluster, such that we can introduce best pr...		2 years ago	
	dashboard	Fix Kubernetes dashboard rbac role binding		last month	
	default-limits	Use the correct feature flag in default-limits		9 months ago	
	efs-provisioner	EFS provisioner: limits/requests & priority		last month	
	emergency-access-service	RBAC: list/create can't be used with resourceNames		2 months ago	

CLUSTER LIFECYCLE MANAGER (CLM)



CLUSTER UPGRADE FLOW



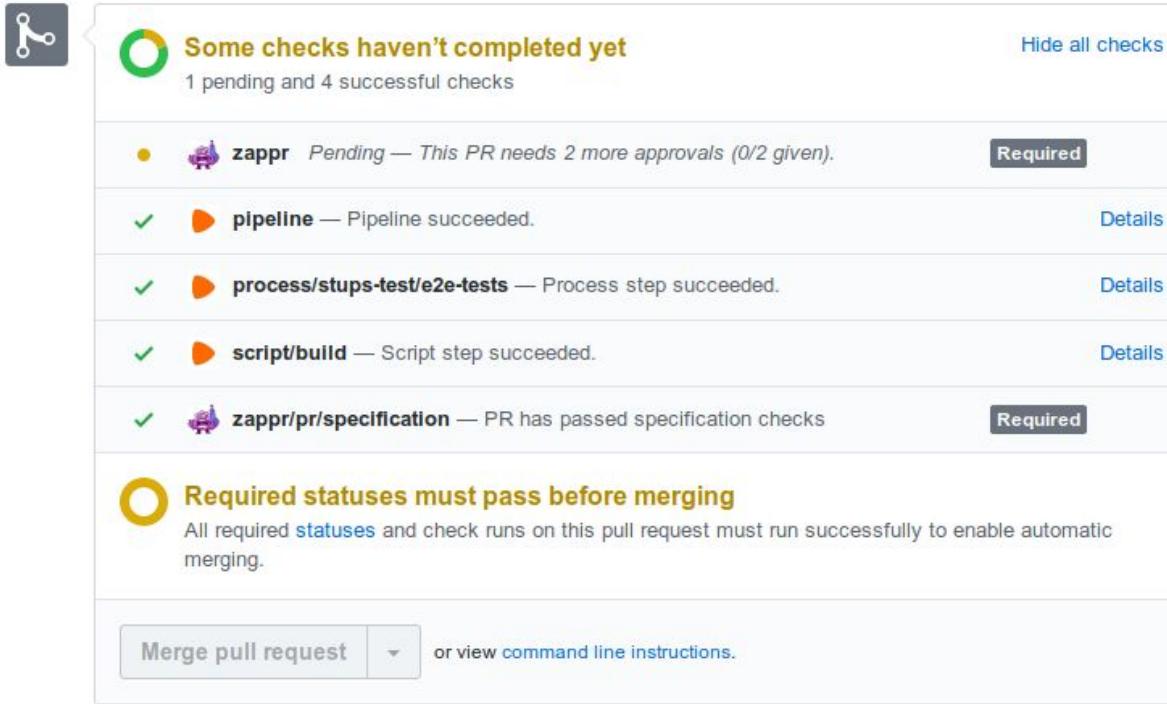
CLUSTER CHANNELS

Channel	Description	Clusters
dev	Development and playground clusters	3
alpha	Main infrastructure cluster (important to us)	1
beta	Non-prod clusters for the rest of the org	65+
stable	Production clusters.	65+

github.com/zalando-incubator/kubernetes-on-aws

E2E TESTS ON EVERY PR

Add more commits by pushing to the **skipper-average-value** branch on **zalando-incubator/kubernetes-on-aws**.



The screenshot shows a GitHub pull request interface with the following details:

- Status Summary:** Some checks haven't completed yet (1 pending and 4 successful checks).
- Pending Approval:** zappr Pending — This PR needs 2 more approvals (0/2 given). (Required)
- Successful Checks:**
 - pipeline — Pipeline succeeded. (Details)
 - process/stups-test/e2e-tests — Process step succeeded. (Details)
 - script/build — Script step succeeded. (Details)
 - zappr/pr/specification — PR has passed specification checks (Required)
- Required Statuses:** Required statuses must pass before merging. All required statuses and check runs on this pull request must run successfully to enable automatic merging.
- Action Buttons:** Merge pull request or view command line instructions.

E2E TESTS



Conformance Tests

Upstream Kubernetes e2e conformance tests

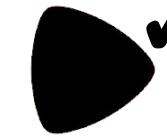
159



StatefulSet Tests

Rolling update of stateful sets including volume mounting

2



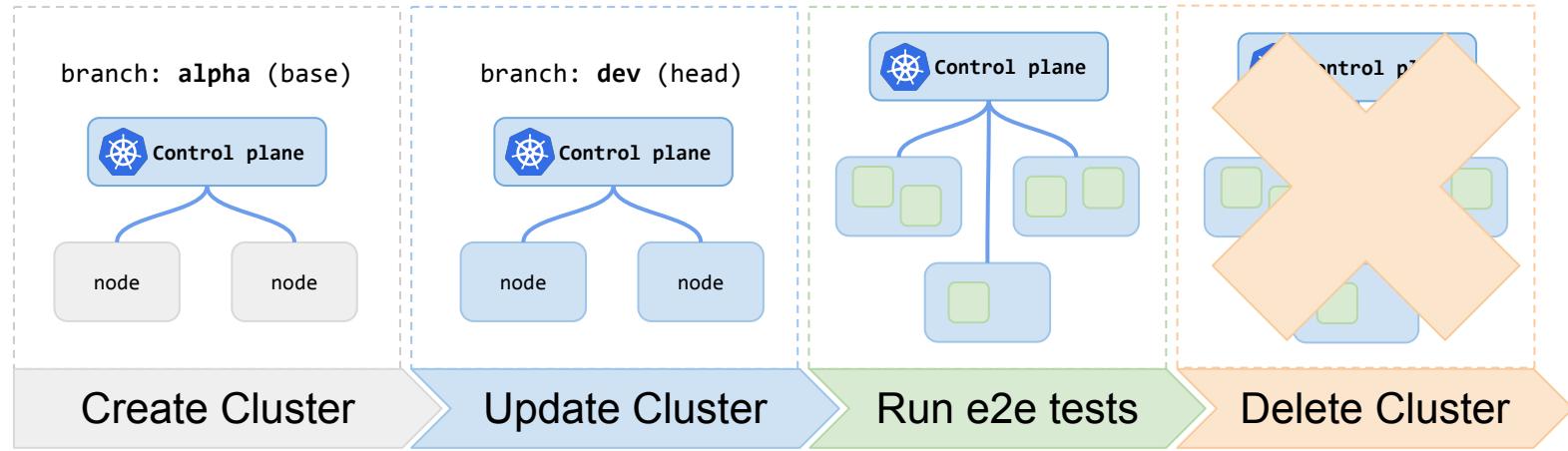
Zalando Tests (custom)

Custom tests for ingress, external-dns, PSP etc.

17

RUNNING E2E TESTS

Testing dev to alpha upgrade





UPGRADING NODES

NAÏVE NODE UPGRADE STRATEGY

Auto Scaling Group

Min:	3
Max:	9
Current:	5
Desired:	5

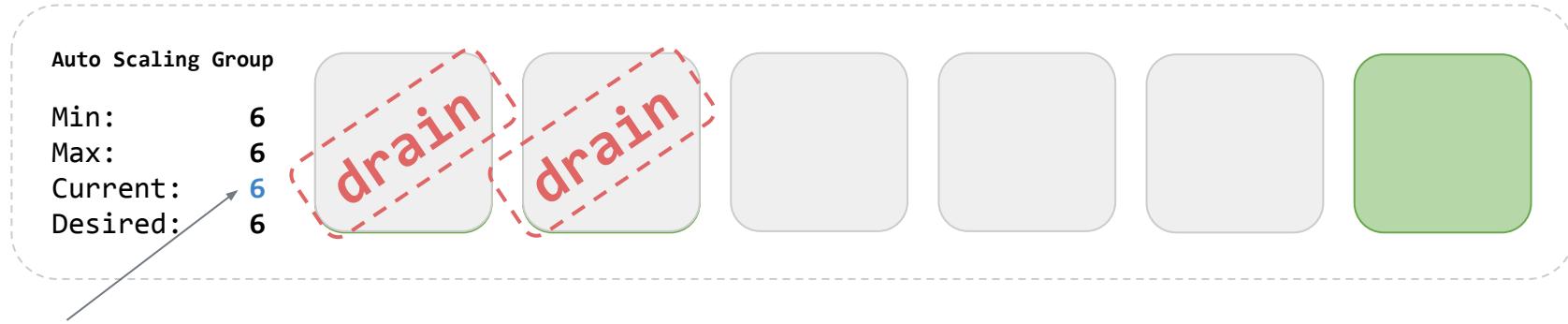


NAÏVE NODE UPGRADE STRATEGY



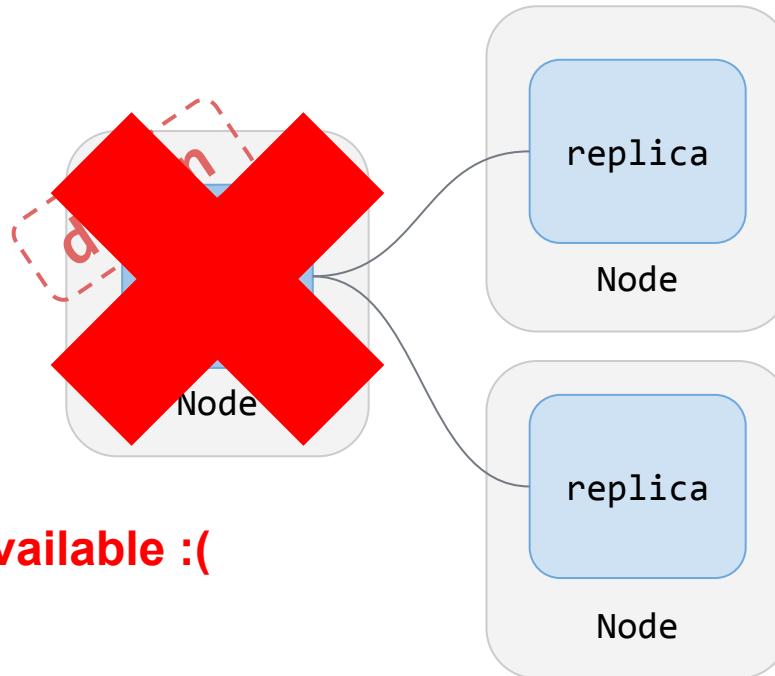
Set ASG size to current + 1

NAÏVE NODE UPGRADE STRATEGY



PROBLEMS WITH THE NAÏVE STRATEGY

What about stateful applications like Postgres?



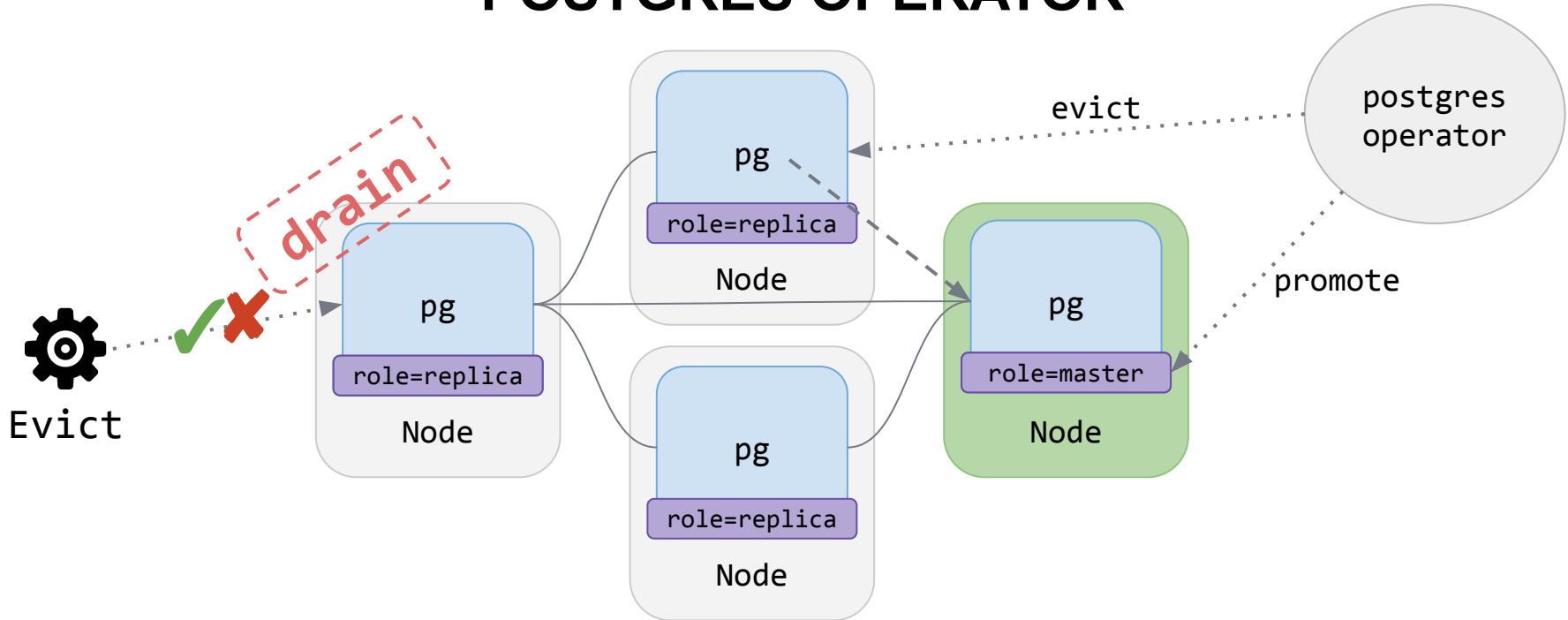
Postgres cluster unavailable :(





STATEFUL WORKLOADS (POSTGRES)

POSTGRES OPERATOR



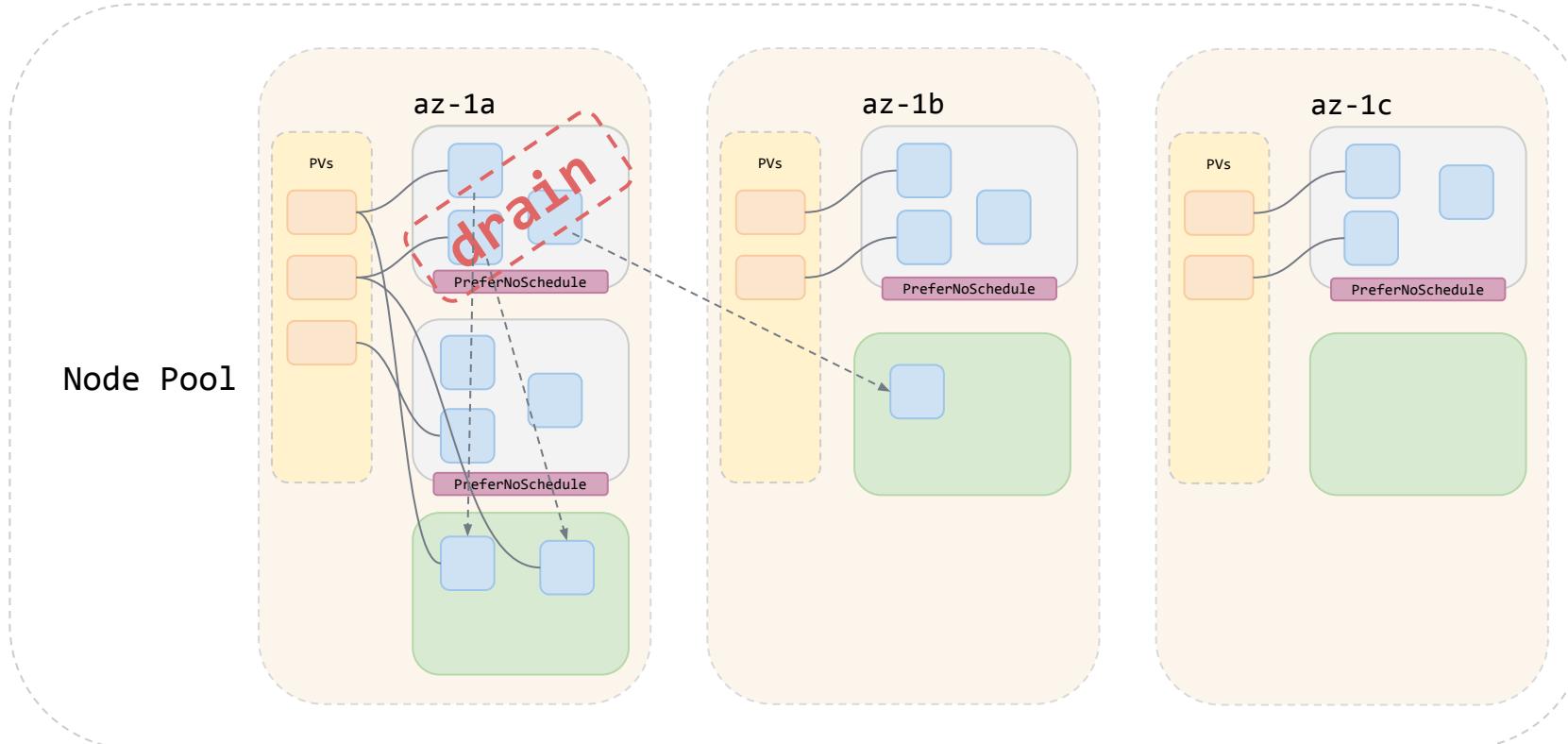
github.com/zalando-incubator/postgres-operator

POSTGRES OPERATOR

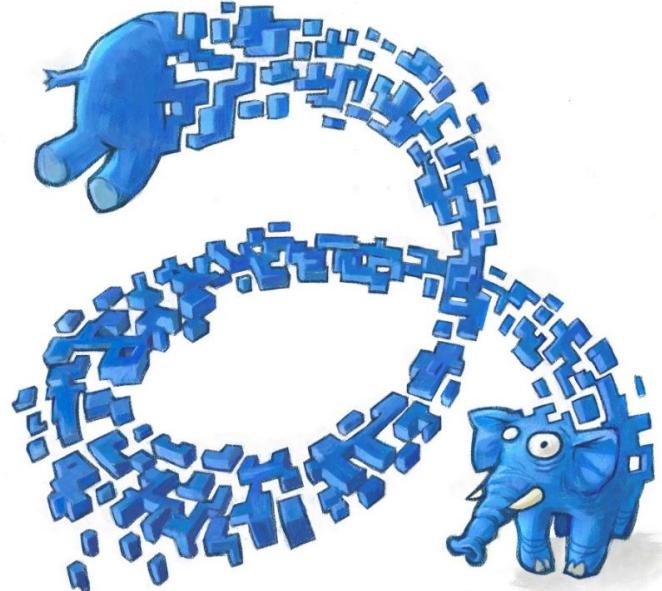
```
apiVersion: policy/v1beta1
kind: PodDisruptionBudget
metadata:
  name: "postgres-cluster"
spec:
  minAvailable: 1
  selector:
    matchLabels:
      application: "postgres-cluster"
      role: "master"
```

github.com/zalando-incubator/postgres-operator

ROLLING UPGRADE OF NODES



POSTGRES OPERATOR



Application to manage
PostgreSQL clusters on
Kubernetes

>500

clusters running
on Kubernetes



Elasticsearch
2.500 vCPUs
1 TB RAM

Elasticsearch in Kubernetes

github.com/zalando-incubator/es-operator/

SLAS FOR CLUSTER UPDATES

- Respect **PodDisruptionBudgets**
- **Force-terminate Pods after 3 days** (or 8h on test)
- Cluster updates can be blocked anytime!

```
zkubectl cluster-update block [+ REASON]
```



DEPLOY & USER INTERFACE

APP DEPLOYMENT CONFIGURATION

```
└── deploy/apply
    ├── deployment.yaml
    ├── credentials.yaml # Zalando IAM
    ├── ingress.yaml
    └── service.yaml
    delivery.yaml          # Zalando CI/CD
```



APP INGRESS.YAML

```
kind: Ingress
metadata:
  name: "..."
spec:
  rules:
    # DNS name your application should be exposed on
    - host: "myapp.foo.example.org"
      http:
        paths:
          - backend:
              serviceName: "myapp"
              servicePort: 80
```



CONTINUOUS DELIVERY PLATFORM

DEPLOYMENT UNITS	RENDERING-ENGINE	Exclude PRs: <input type="checkbox"/>					
Pipeline	Started	Action	Pipeline Runs				
pr-1785-2	3h ago	i	 BUILD	 PERF	 DEPLOY-BRANCH	 TEST-BRANCH	 DEPLOY-DOCS
pr-1785-1	3h ago	i	 BUILD	 PERF	 DEPLOY-BRANCH	 TEST-BRANCH	 DEPLOY-DOCS
Remove `any` usage from our code pr-1784-1	6h ago	i	 BUILD	 PERF	 DEPLOY-BRANCH	 TEST-BRANCH	 DEPLOY-DOCS
master-1188 master-1188	1d ago	i	 BUILD	 PERF	 DEPLOY-BRANCH	 TEST-BRANCH	 DEPLOY-DOCS
Do not bundle node_modules in re package pr-1783-1	1d ago	i	 BUILD	 PERF	 DEPLOY-BRANCH	 TEST-BRANCH	 DEPLOY-DOCS
master-1187 master-1187	1d ago	i	 BUILD	 PERF	 DEPLOY-BRANCH	 TEST-BRANCH	 DEPLOY-DOCS
Document the renderer contribution workflow pr-1775-2	2d ago	i	 BUILD	 PERF	 DEPLOY-BRANCH	 TEST-BRANCH	 DEPLOY-DOCS
master-1186 master-1186	2d ago	i	 BUILD	 PERF	 DEPLOY-BRANCH	 TEST-BRANCH	 DEPLOY-DOCS
pr-1778-1	2d ago	i	 BUILD	 PERF	 DEPLOY-BRANCH	 TEST-BRANCH	 DEPLOY-DOCS
Document the renderer contribution workflow pr-1775-1	6d ago	i	 BUILD	 PERF	 DEPLOY-BRANCH	 TEST-BRANCH	 DEPLOY-DOCS

CDP: DEPLOY

The screenshot shows a deployment interface with the following components:

- Top Navigation:** DEPLOYMENT UNITS, TRACKING-DEPLOY, and MASTER-91.
- Environment Headers:** TRACKING-DEPLOY, TEST, STAGING, PROD.
- Deployment Status:** Ran for 1 m, 16 s.
- Deployment Details:** Deployment merchant-parcels, ReplicaSet merchant-parcels-bd76cbc9b, Ingress merchant-parcels, PlatformCredentialsSet [REDACTED], postgresql [REDACTED]-db, Secret [REDACTED]-credentials, Service merchant-parcels.
- Logs:** scalyr logs for each Pod.
- Feedback Form:** Please give us your feedback, with radio buttons for Yes and No, and a text area for What could be improved upon? with a Submit button.

"glorified kubectl apply"

EMERGENCY ACCESS SERVICE

Emergency access by referencing Incident

```
zkubectl cluster-access request \
    --emergency -i INC REASON
```



Privileged production access via 4-eyes

```
zkubectl cluster-access request REASON
zkubectl cluster-access approve USERNAME
```



KUBERNETES WEB VIEW

Clusters default ▾

Search Kubernetes objects. 

CLUSTER RESOURCES

- Namespaces
- Nodes
- PersistentVolumes

CONTROLLERS

- StackSets
- Stacks
- Deployments
- CronJobs
- Jobs
- StatefulSets

POD MANAGEMENT

- Ingresses
- Services
- Pods
- ConfigMaps
- CRDS
- PlatformCredentialsSets
- postgresqls

META

- Resource Types
- Events

/ default / pods,stacks,deployments,services

Pods   

Name	Application	Component	Ready	Status	Restarts	Age	IP	Node	Nominated Node	Readiness Gates	CPU Usage	Memory Usage	Created
even-master-33-5db9d68c8d-5srrh	even		1/1	Running	0	5d18h			<none>	<none>	3m	313 MiB	2019-08-29 16:35:03
even-master-33-5db9d68c8d-72px8	even		1/1	Running	0	5d18h			<none>	<none>	3m	297 MiB	2019-08-29 17:05:16
even-master-33-5db9d68c8d-rhh9m	even		1/1	Running	0	5d18h			<none>	<none>	2m	312 MiB	2019-08-29 17:20:13

Stacks   

Name	Desired	Current	Up-to-date	Available	Traffic	No-Traffic-Since	Age	Created
even-master-27	3	0	0	0	0	131d	210d	2019-02-05 20:11:56
even-master-29	3	0	0	0	0	131d	140d	2019-04-17 08:30:22
even-master-30	3	0	0	0	0	131d	131d	2019-04-25 12:19:11
even-master-31	3	0	0	0	0	50d	131d	2019-04-25 12:30:11
even-master-32	3	0	0	0	0	64d	64d	2019-07-01 15:19:45
even-master-33	3	3	3	3	100		50d	2019-07-15 12:18:58

Deployments   

Name	Ready	Up-to-date	Available	Age	Containers	Images	Selector	Created
------	-------	------------	-----------	-----	------------	--------	----------	---------

kubectl get pods,stacks,deployments,...

SEARCHING ACROSS 140+ CLUSTERS

Search

Search Text Search!

Resource Types CronJob DaemonSet Deployment Ingress Namespace Node PlatformCredentialsSet Pod ReplicaSet Service StackSet StatefulSet unselect all

etcd-operator (Deployment)
/cluster: namespaces/default/deployments/etcd-operator
Created: 2018-10-18 13:23:39 source.zalan.co **/etcd-operator:v0.9.2-master-2**
name: etcd-operator

etcd-operator (Deployment)
/cluster: namespaces/wpi/deployments/etcd-operator
Created: 2019-08-12 12:30:07 e.stups.zalan.co **/etcd-operator:v0.9.3**
application: deployment-id: d-e8yt17ub9hxt513sr27w66ea environment: staging pipeline-id: l-7bic5kvki6khadtadtqzq5hy3q version: master-7

etcd-operator (Deployment)
/cluster: namespaces/default/deployments/etcd-operator
Created: 2018-10-19 14:13:50 source.zalan.co **/etcd-operator:v0.9.2-master-3**
name: etcd-operator

etcd-operator (Deployment)
/clusters namespaces/default/deployments/etcd-operator
Created: 2018-05-04 11:01:36 tups.zalan.co **/etcd-operator:v0.6.1-2**
app: etcd component: operator

etcd-operator (Deployment)
/clusters namespaces/incentives/deployments/etcd-operator
Created: 2018-07-03 08:12:51 .zalan.co **/etcd-operator:v0.9.3**
application: deployment-id: d-so5ukevu2piyw5bdigzxc4gx3 environment: staging version: master-26

CLUSTER RESOURCES

/ all / pods

Pods



Namespaces

Nodes

PersistentVolumes

CONTROLLERS

StackSets

Stacks

Deployments

CronJobs

Jobs

StatefulSets

POD MANAGEMENT

Ingresses

Services

Pods

ConfigMaps

CRDS

PlatformCredentialsSets

postgresqls

META

Resource Types

Events

Namespace	Name	Application	Component	Ready	Status	Restarts	Age	IP	Node	Nominated Node	CPU Usage	Memory Usage
kube-system	audittrail-adapter-lt2kb	audittrail-adapter		1/1	Running	151	6d9h	172.31.0.70	ip-172-31-0-70.eu-central-1.compute.internal	<none>	1m	20 MiB
kube-system	audittrail-adapter-s6tvf	audittrail-adapter		1/1	Running	23	6d10h	172.31.18.89	ip-172-31-18-89.eu-central-1.compute.internal	<none>	6m	23 MiB
default	devcon-cluster-manager-74746ff998-gg4m9	devcon-cluster-manager		1/1	Running	13	12d	10.2.211.12	ip-172-31-12-12.eu-central-1.compute.internal	<none>	0m	83 MiB
default	devcon-cluster-manager-74746ff998-4zzwn	devcon-cluster-manager		1/1	Running	13	12d	10.2.3.13	ip-172-31-4-240.eu-central-1.compute.internal	<none>	0m	63 MiB
visibility	zmon-sentry-agent-7867b8d48b-rddbg	zmon-sentry-agent		1/1	Running	11	10d	10.2.211.124	ip-172-31-12-12.eu-central-1.compute.internal	<none>	0m	17 MiB
default	[REDACTED]	main		1/1	Running	10	67m	10.2.178.190	ip-172-31-16-111.eu-central-1.compute.internal	<none>	236m	212 MiB
default	cert-manager-ff4d7884d-s6jnm	cert-manager		1/1	Running	8	12d	10.2.220.9	ip-172-31-20-103.eu-central-1.compute.internal	<none>	5m	51 MiB

UPGRADE TO KUBERNETES 1.14

"Found 1223 rows for 1 resource type in 148 clusters in 3.301 seconds."

all / nodes

Nodes

Label Columns Labels to show as columns (comma separated) or "*" to show all labels

Label Selector Label selector (label=value)

Filter Roles=worker, Version=v1.14.6

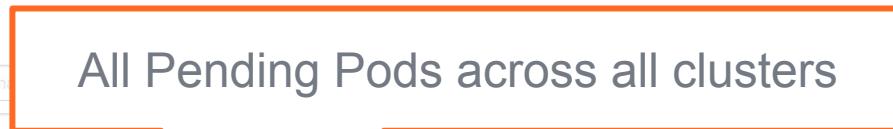
Show CPU/Memory Usage

Cluster	Name	Status	Roles	Age	Version	Internal-IP	External-IP	OS-Image	Kernel-Version	Container-Runtime	Created
[REDACTED]	[REDACTED]	Ready	worker	4h33m	v1.14.6	[REDACTED]	[REDACTED]	Ubuntu 18.04.3 LTS	4.15.0-1045-aws	docker://18.9.7	2019-08-27 12:27:12
[REDACTED]	[REDACTED]	Ready	worker	17m	v1.14.6	[REDACTED]	[REDACTED]	Ubuntu 18.04.3 LTS	4.15.0-1045-aws	docker://18.9.7	2019-08-27 16:44:04
[REDACTED]	[REDACTED]	Ready	worker	152m	v1.14.6	[REDACTED]	[REDACTED]	Ubuntu 18.04.3 LTS	4.15.0-1045-aws	docker://18.9.7	2019-08-27 14:29:10

CLUSTER RESOURCES

all / all / pods

Pods



Label Columns Labels to show as columns (comma-separated)

Label Selector Label selector (label=value)

Filter Status=Pending

Submit

All Pending Pods across all clusters

Namespaces
Nodes
PersistentVolumes
CONTROLLERS
Deployments
CronJobs
Jobs
DaemonSets
StatefulSets

POD MANAGEMENT

Ingresses

Services

Pods

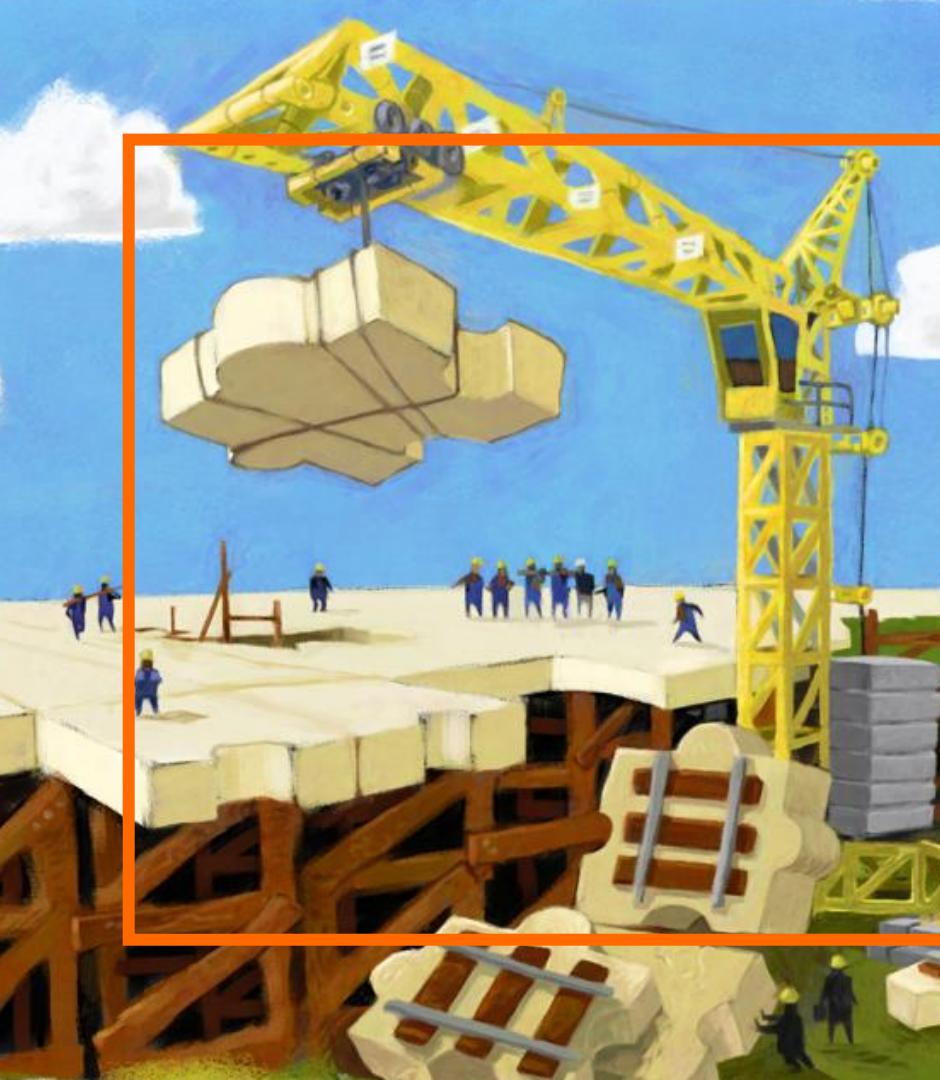
ConfigMaps

META

Resource Types

Events

Cluster	Namespace	Name	Ready	Status	Restarts	Age	IP	Node	Nominated Node	Readiness Gates	Created
	kube-system	3d6b56-h8bml	0/1	Pending	0	77s	<none>	<none>	<none>	<none>	2019-08-07 17:30:04
	kube-system	5798-lv6c5	0/1	Pending	0	17m	<none>	<none>	<none>	<none>	2019-08-07 17:13:39
	default	b7-8w66g	0/1	Pending	0	144m	<none>	<none>	<none>	<none>	2019-08-07 15:06:34
		676f-4x8g9	0/1	Pending	0	4h46m	<none>	<none>	<none>	<none>	2019-08-07 12:45:02
		676f-8jdvk	0/1	Pending	0	4h46m	<none>	<none>	<none>	<none>	2019-08-07 12:45:02
		676f-dmjg4	0/1	Pending	0	4h46m	<none>	<none>	<none>	<none>	2019-08-07 12:45:02
		676f-qj94v	0/1	Pending	0	4h46m	<none>	<none>	<none>	<none>	2019-08-07 12:45:02
		676f-rt4md	0/1	Pending	0	4h46m	<none>	<none>	<none>	<none>	2019-08-07 12:45:02



AVOIDING CONFIGURATION DRIFT

CLUSTER CONFIGURATION

Clusters look mostly the same, except:

- secrets, e.g. credentials for external logging provider
- node pools and their instance sizes

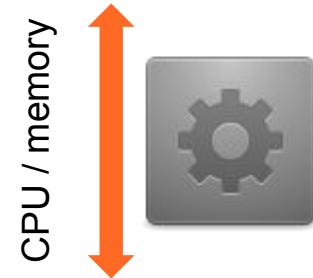
Cluster-specific config items are stored in Cluster Registry

CLUSTER AUTOSCALER



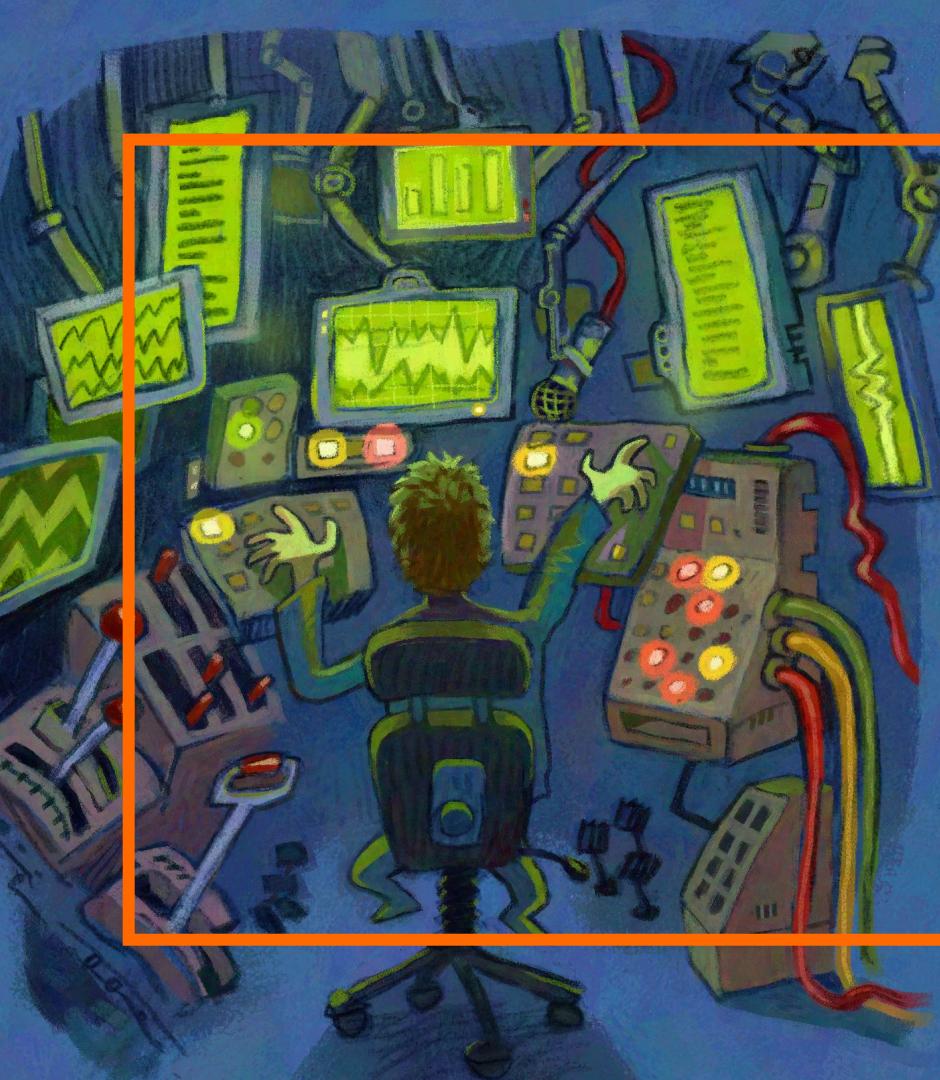
VERTICAL POD AUTOSCALER

- Prometheus
- External DNS
- Heapster / Metrics Server
- our ALB Ingress Controller



VERTICAL POD AUTOSCALER





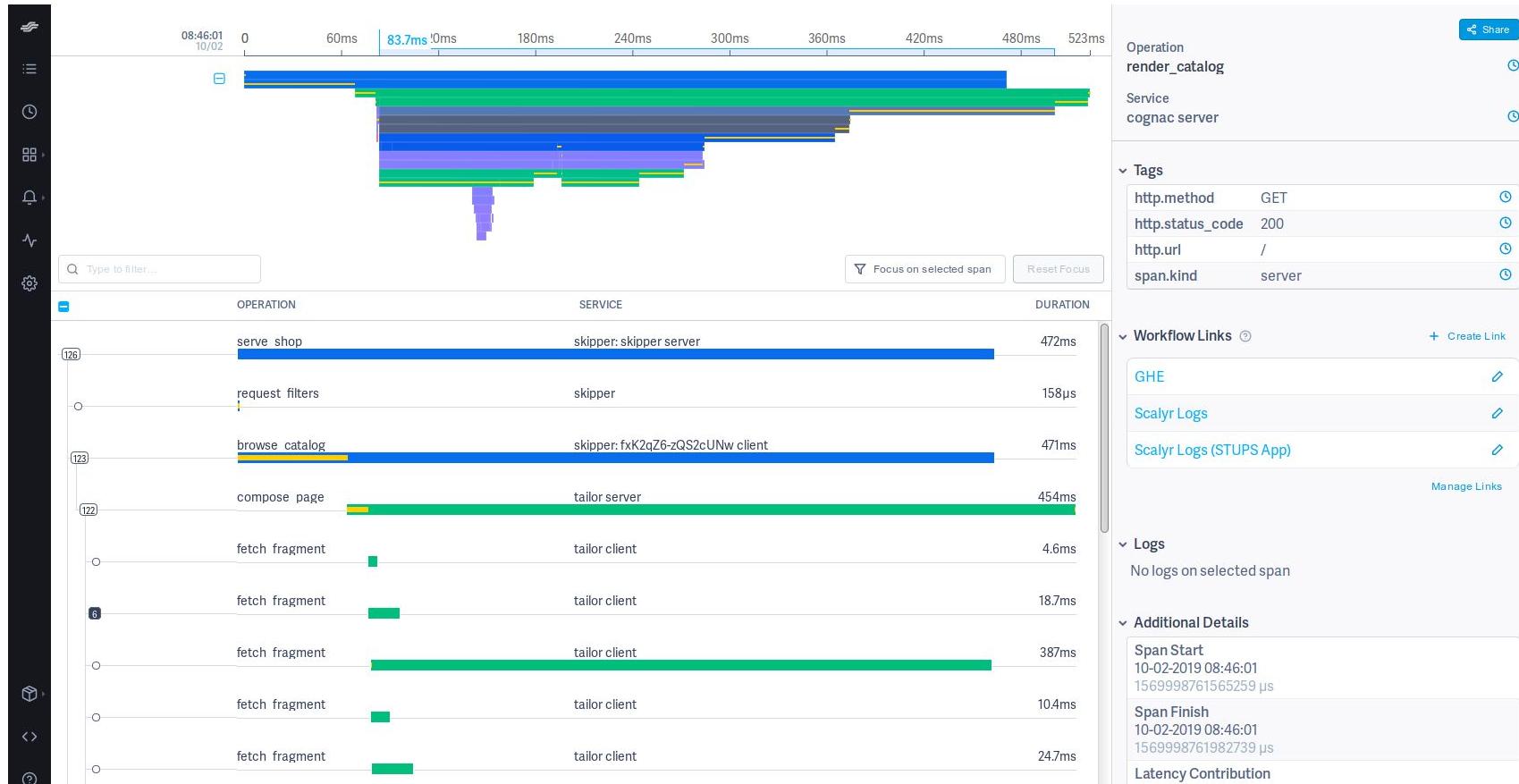
MONITORING & COST EFFICIENCY

MONITORING SYSTEM - ZMON

- Dynamic entity registration
(clusters, pods, ..)
- Generic checks on entity attributes,
e.g. for all production clusters
"Less than 60% of worker nodes are ready"
- OpsGenie alerts



OPENTRACING



KUBERNETES RESOURCE REPORT

Overview Clusters Ingresses Teams Applications Pods

Cluster [REDACTED]

[https://\[REDACTED\]](https://[REDACTED])

MASTER NODES

WORKER NODES

PODS

CPU REQUESTS / ALLOCATABLE

MEMORY REQUESTS / ALLOCATABLE

MONTHLY COST

2

15

325

55.7 / 60.6

183.1 GiB / 241.5 GiB

1,687.16 USD

You can potentially save [REDACTED] every month by optimizing resource requests and reducing slack.

Price per requested vCPU is [REDACTED] per hour and per requested GiB memory is [REDACTED] per hour.

Nodes

Name	Role	Instance Type	S?	Version	CC	MC	CPU	Memory (GiB)	Cost
[REDACTED]	worker	m4.xlarge	✗	v1.10.5	4	15.7 GiB	<div><div style="width: 0.7%;">0.7</div></div> <div><div style="width: 3.7%;">3.7</div></div> <div><div style="width: 3.8%;">3.8</div></div> <div><div style="width: 3.7%;">3.7</div></div> <div><div style="width: 10.8%;">10.8</div></div> <div><div style="width: 15.1%;">15.1</div></div>	70.13	
[REDACTED]	worker	m4.xlarge	✗	v1.10.5	4	15.7 GiB	<div><div style="width: 1.0%;">1.0</div></div> <div><div style="width: 3.4%;">3.4</div></div> <div><div style="width: 3.8%;">3.8</div></div> <div><div style="width: 7.2%;">7.2</div></div> <div><div style="width: 14.1%;">14.1</div></div> <div><div style="width: 15.1%;">15.1</div></div>	70.13	
[REDACTED]	worker	m4.xlarge	✗	v1.10.5	4	15.7 GiB	<div><div style="width: 0.2%;">0.2</div></div> <div><div style="width: 3.8%;">3.8</div></div> <div><div style="width: 3.8%;">3.8</div></div> <div><div style="width: 3.4%;">3.4</div></div> <div><div style="width: 8.0%;">8.0</div></div> <div><div style="width: 15.1%;">15.1</div></div>	70.13	
[REDACTED]	master	m4.large		v1.10.5	2	7.8 GiB	<div><div style="width: 0.3%;">0.3</div></div> <div><div style="width: 1.0%;">1.0</div></div> <div><div style="width: 1.8%;">1.8</div></div> <div><div style="width: 2.8%;">2.8</div></div> <div><div style="width: 1.3%;">1.3</div></div> <div><div style="width: 7.3%;">7.3</div></div>	87.66	
[REDACTED]	worker	m4.xlarge	✗	v1.10.5	4	15.7 GiB	<div><div style="width: 0.2%;">0.2</div></div> <div><div style="width: 2.7%;">2.7</div></div> <div><div style="width: 3.8%;">3.8</div></div> <div><div style="width: 3.3%;">3.3</div></div> <div><div style="width: 9.4%;">9.4</div></div> <div><div style="width: 15.1%;">15.1</div></div>	70.13	

RESOURCE REPORT: TEAMS

ID	C	A	P	CR	MR	CPU	Memory (MiB)	Cost	Slack Cost			
	3	14	114	457.9	1.7 TiB	69.01	457.9	1,074,024	1,780,420	27,558.50	13,327.47	<input type="button" value=" "/>
	1	9	251	428.95	426.2 GiB	137.99	428.95	164,613	436,384	19,406.82	13,111.24	<input type="button" value=" "/>

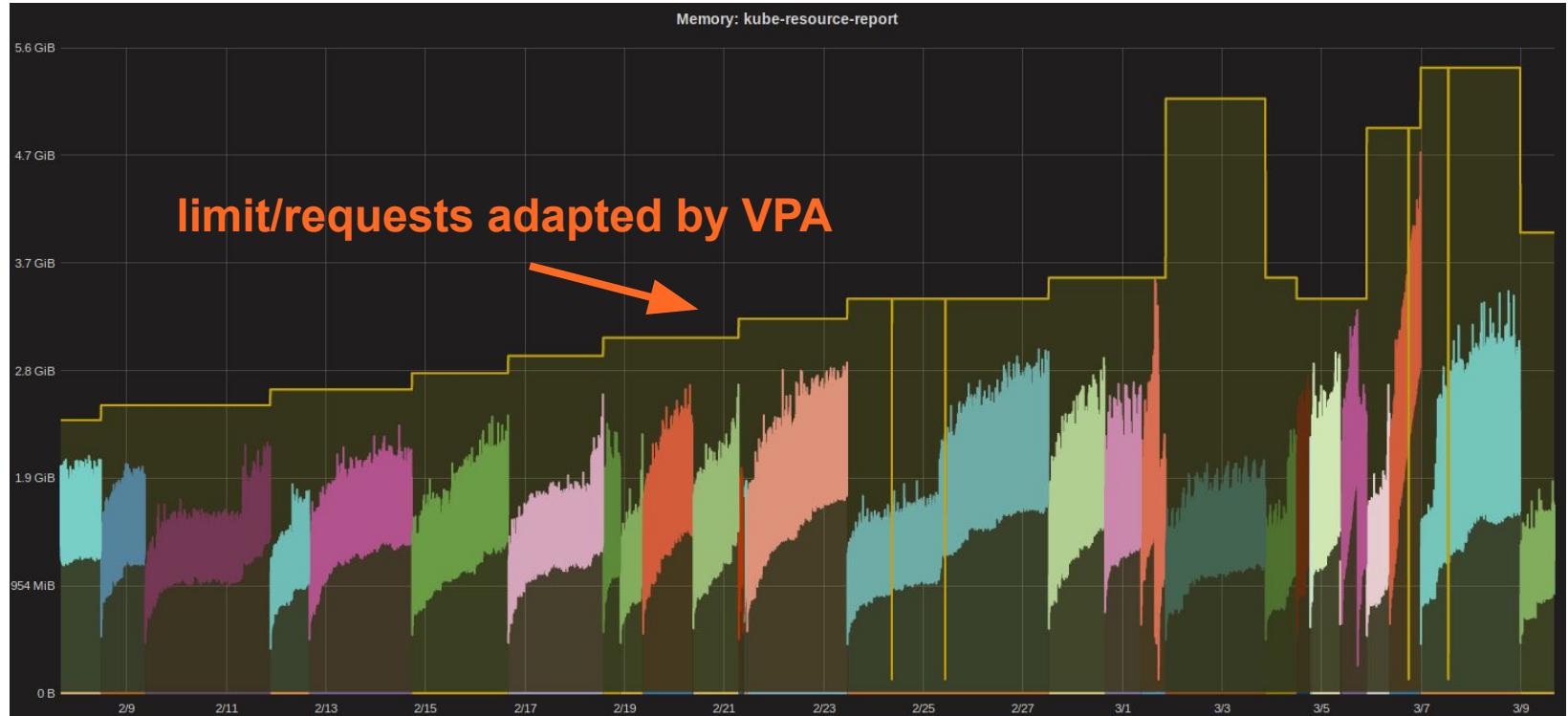


Sorting teams by
Slack Costs

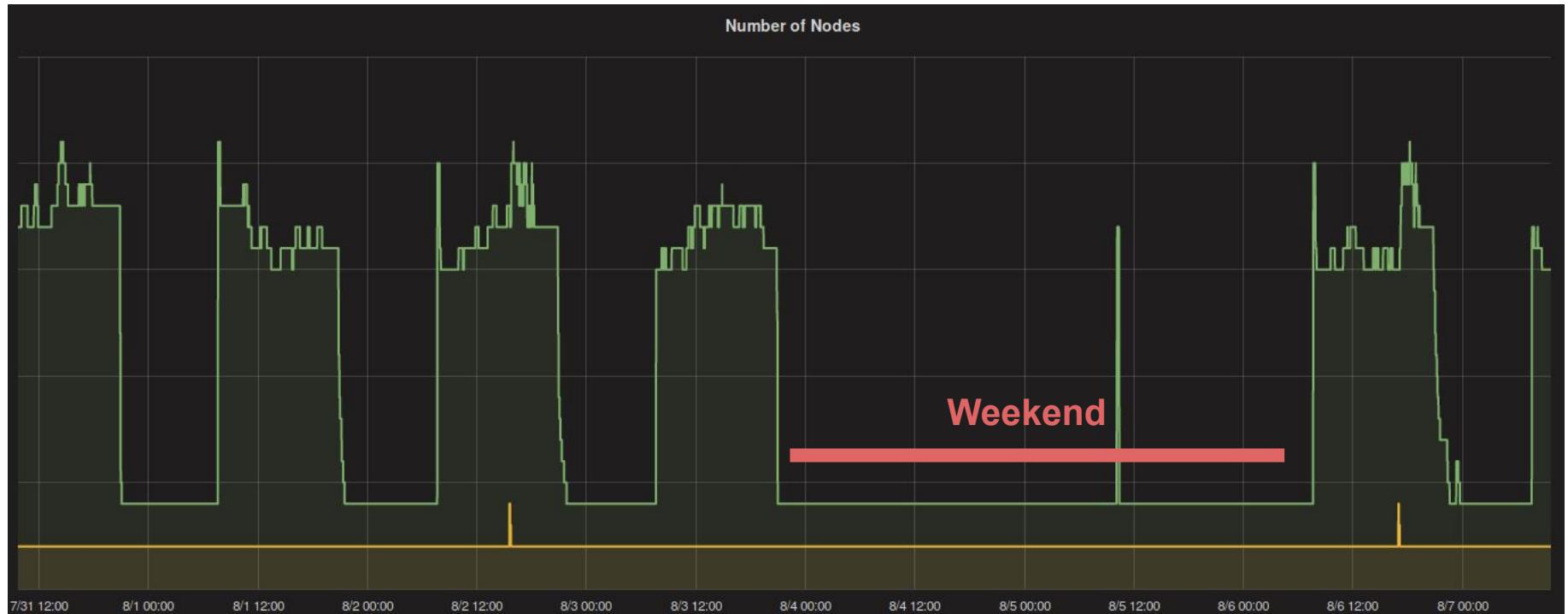
KUBERNETES APPLICATION DASHBOARD



VERTICAL POD AUTOSCALER



DOWNSCALING DURING OFF-HOURS



KUBERNETES JANITOR

hjacobs / kube-janitor

Code Issues 10 Pull requests 3 Security Insights

Clean up (delete) Kubernetes resources after a configured TTL (time to live)

kubernetes kubernetes-operator cleanup resource-management ttl garbage-collector

- **TTL** and **expiry date** annotations, e.g.
 - set time-to-live for your test deployment
- **Custom rules**, e.g.
 - delete everything without "app" label after 7 days

EC2 SPOT NODES

Role	Instance Type	S?	Version	CC	MC	CPU	Memory (GiB)	Cost
worker	m4.2xlarge		v1.12.5-custom.master-1	8	31.4 GiB	<div style="width: 1.6%;">1.6</div> <div style="width: 6.9%;">6.9</div> <div style="width: 7.8%;">7.8</div>	<div style="width: 8.8%;">8.8</div> <div style="width: 14.3%;">14.3</div> <div style="width: 30.9%;">30.9</div>	350.64
worker	m4.4xlarge	✖	v1.12.5-custom.master-1	16	62.9 GiB	<div style="width: 4.1%;">4.1</div> <div style="width: 9.0%;">9.0</div> <div style="width: 15.8%;">15.8</div>	<div style="width: 51.2%;">51.2</div> <div style="width: 62.3%;">62.3</div> <div style="width: 62.4%;">62.4</div>	193.73

72% savings

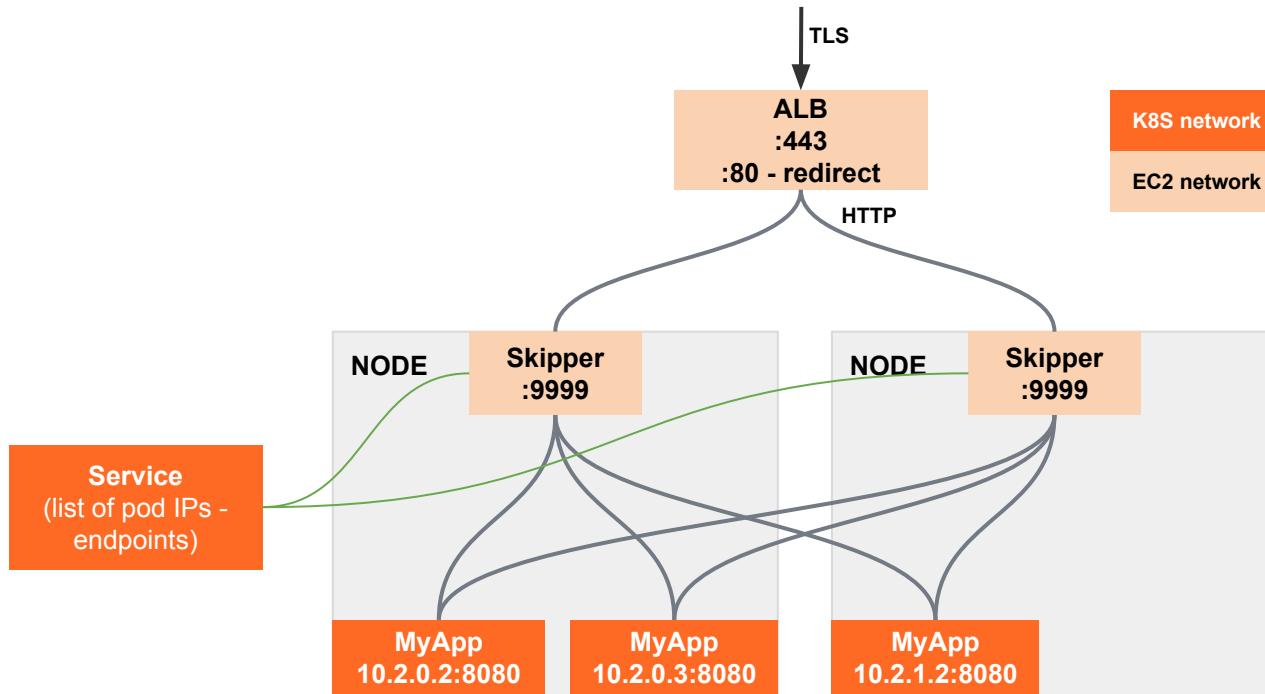


OUR SETUP VS VANILLA KUBERNETES

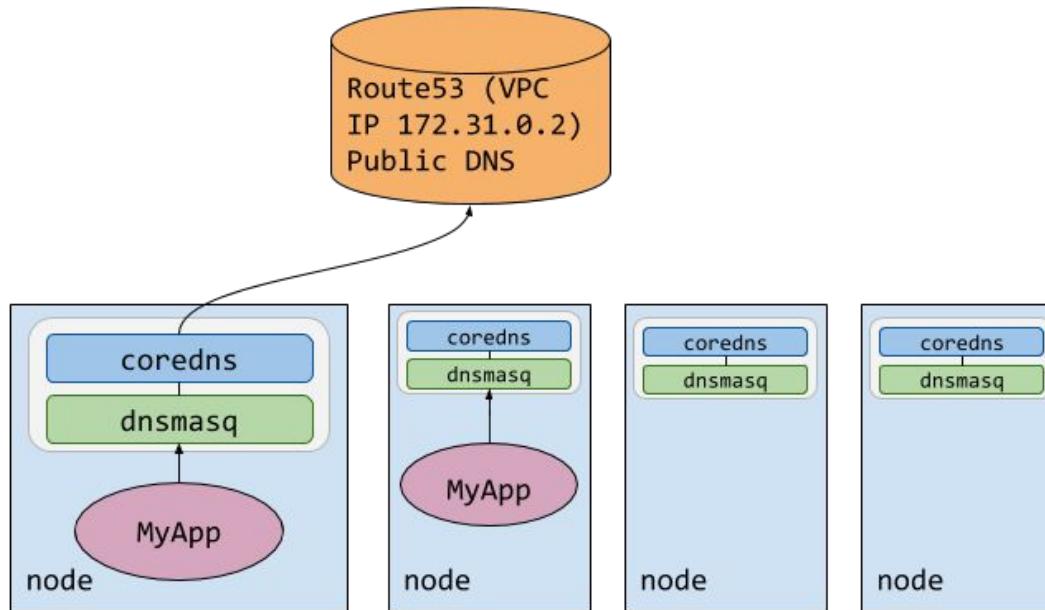
HOW MUCH DO WE DIVERGE?

- API access via Zalando OAuth
- CPU throttling disabled via Kubelet flag
- No memory overcommit (requests == limits)
- Ingress: External DNS, Skipper, AWS ALB
- Custom CRDs: Zalando OAuth, Postgres, StackSet
- Kubernetes Downscaler
- DNS setup (CoreDNS DaemonSet, ndots: 2)

INGRESS: ALB + SKIPPER



DNS: COREDNS AS DAEMONSET



NON-PROD VS PROD

- Non-production similar to plain hosted Kubernetes
- Production:
 - No write access (only via CI/CD)
 - Compliance webhooks
 - Require production-ready Docker images



COMPLIANCE FOR PRODUCTION

- Pods require **application** label pointing to application registry
⇒ establishes link to owning team
- Docker images must be built from master via CDP



NOTE: teams can freely choose their namespace(s)

**HOWTOS**

Home
Plan ▾
Setup ▾
Design ▾
Code ▾
Build ▾
Test ▾
Deploy ^
Autoscale Your Application
Configure deployments from PRs
[Setting Resource Requests and Limits](#)

Expose Application to Internet
Give Your Application Access to AWS
Give Your Application Access to Kubernetes
Migrate a STUPS Application

Run Applications in Production
Run Stateful Applications
Store Passwords in Deployment Configurations
Troubleshoot a Failing Deployment

Setting Resource Requests and Limits

Table of contents

[Correctly Configure Resource Requests and Limits](#)
[Existing Application](#)
[New Application](#)

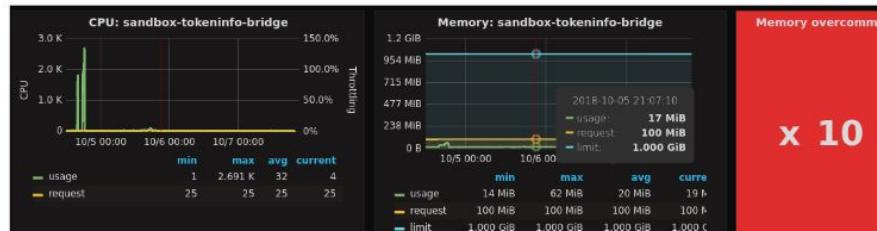
Resource requests are specified per container in the [Pod](#) spec of your [Kubernetes deployment](#), for example in the file `deployment.yaml`. Resource requests are used by the Kubernetes scheduler to decide if a [Pod](#) can fit on a node or not. It's important to choose the right amount of resources as it will provide better resource utilization in the clusters and prevent running with too much slack capacity which translates to wasted resources.

Correctly Configure Resource Requests and Limits

Existing Application

If you have an existing application running you can check that it's configured with sensible resource requests and limits by looking at the [Application Dashboard](#). This dashboard will show the current CPU and Memory usage along with the specified requests and limits. Additionally it will indicate if the application is over committing on memory meaning `limits > requests`.

In this example the application has the `limits` set 10 times higher than the `requests`.



MONTHLY DEVELOPER NEWSLETTER

KUBERNETES

- **Priority support for node pools:** For Cyberweek we added support for having fallback node pools with different instance types and priority in order to provide capacity even when AWS runs out of the desired instance types.
We will keep this setup going forward allowing better instance availability especially for scarce instance types. You can request custom fallback pools! See the [docs](#).
- **zkubectl tunnel** allows you to access non-public resources by setting up a series of port forwards through your Kubernetes cluster. For instance, you can use it to access protected ElastiCache and RDS databases from your local machine, e.g. to dump and restore their data. See [the docs](#) for other use cases and how to setup a tunnel.
- **Kube-Web-View:** You can now reach Scalyr, Lightstep, GitHub, CDP, Pier One, Postgres, the API Portal and [more](#) from [Kube Web View](#):

[playground](#) / [api-infrastructure](#) / [pods](#) / [api-monitoring-controller-88665d575-x5fjr](#)

api-monitoring-controller-88665d575-x5fjr

pod	api-monitoring-controller-88665d575-x5fjr	
application	api-monitoring-controller	
version	master-17e2aa93bf84aae93eac2e53f000737f1af0fc70-489	

SUMMARY

- Seamless updates
- Avoid pet clusters
- Small disruptions are normal
- Automated cluster e2e tests
- Documentation & communication



FUTURE

- API version updates (1.16+)
- Improved Autoscaling
- Improved StackSet, Gradual Rollout
- Migrations
- Cost efficiency
- Looking at VPC CNI, AWS IAM, EKS, ...



KUBERNETES FAILURE STORIES



- Zalando's Failure Stories - KubeCon EU 2019
- Build Errors of Continuous Delivery Platform
- Total DNS outage in Kubernetes cluster

COMMON PITFALLS

- Insufficient e2e tests
- Readiness & Liveness Probes
- Resource Requests & Limits
- DNS



OPEN SOURCE & MORE

Cluster Config

github.com/zalando-incubator/kubernetes-on-aws

Skipper HTTP Router & Ingress controller

github.com/zalando/skipper

Ingress Controller for AWS

github.com/zalando-incubator/kube-ingress-aws-controller

Kubernetes Web View

codeberg.org/hjacobs/kube-web-view

More Zalando Tech Talks

github.com/zalando/public-presentations



Thank you!

Henning Jacobs
@try_except_