Reaching Zen in Elasticsearch's
Cluster Coordination

Philipp Krenn

@xeraa

elastic

elasticsearch

elastic

elastic

Developer 🥑

Cluster Coordination?

elastic

Cluster State?

Cluster Metadata

Cluster Settings

Index Metadata

Lots more

`GET _cluster/state`

Only move forward

Do not lose data

```
{
  "cluster_name" : "docker-cluster",
  "cluster_uuid" : "nOHcm7Q3R5yMN5z1PoG6UQ",
  "version" : 29,
  "state_uuid" : "Of1zGOnoRaGgIfYw_w58MA",
  "master_node" : "P9UHiA-YSkesOfR7-G5O_Q",
  "blocks" : { },
  "nodes" : {
    "P9UHiA-YSkesOfR7-G5O_Q" : {
      "name" : "elasticsearch3",
      "ephemeral_id" : "MdWyvnTfRCuhzD9ftWtODw",
      "transport_address" : "172.21.0.3:9300",
      "attributes" : {
        ...
```

Main Components

Discovery

Master Election

Cluster State Publication

Zen

Zen to Zen2

Not pluggable

Why

https://www.elastic.co/guide/en/elasticsearch/resiliency/current/index.html

Repeated network partitions can cause cluster state updates to be lost (STATUS: DONE, v7.0.0)

How

https://github.com/elastic/elasticsearch-formal-models

TLA+ specification

TLC model checking

https://github.com/elastic/elasticsearch-formal-models/blob/master/cluster/isabelle/Preliminaries.thy

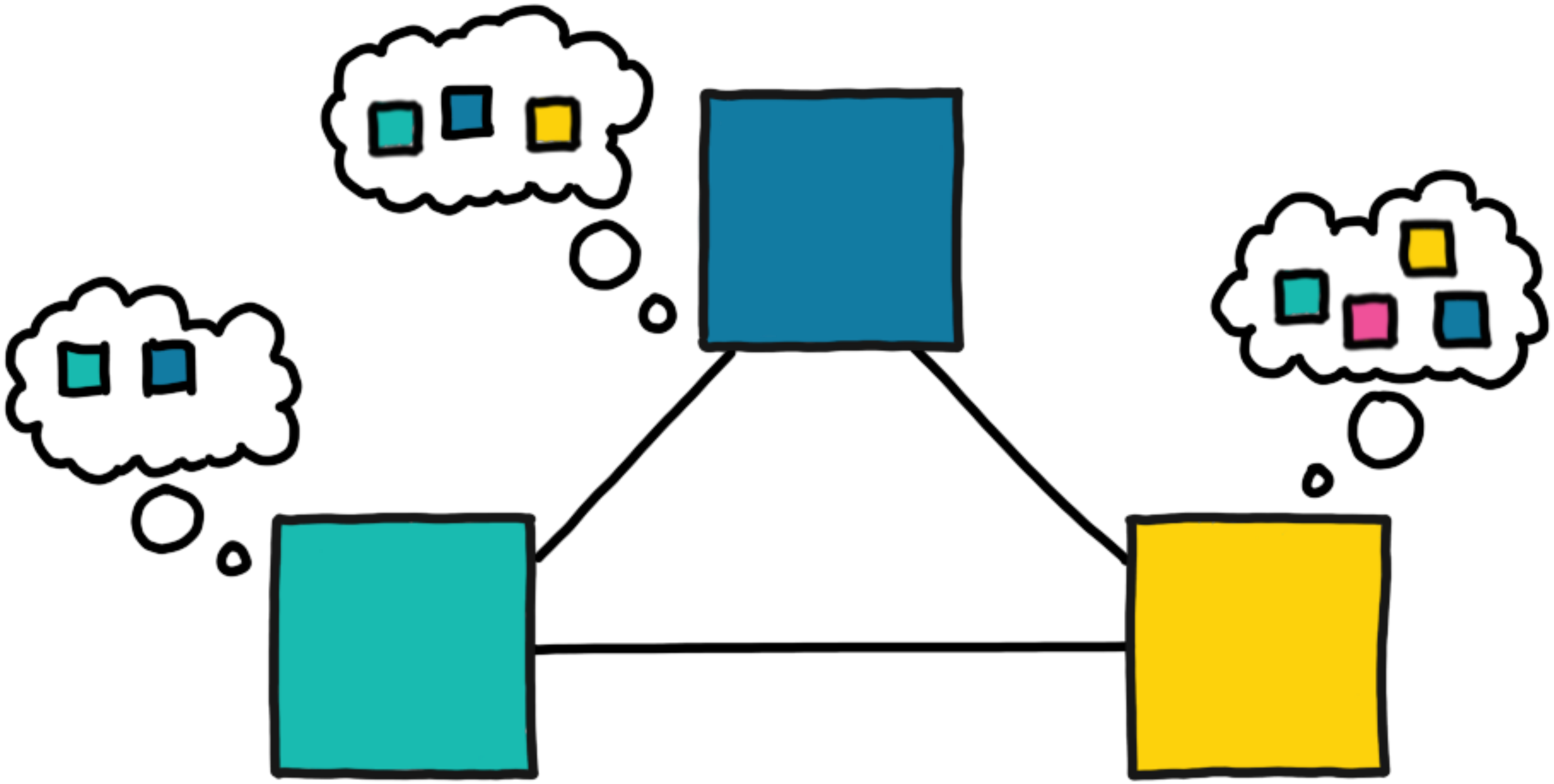
```
text \<open>It works correctly on finite and nonempty sets as follows:\<close>

theorem
  fixes S :: "Term set"
  assumes finite: "finite S"
  shows maxTerm_mem: "S \<noteq> {} \<Longrightarrow> maxTerm S \<in> S"
    and maxTerm_max: "\<And> t'. t' \<in> S \<Longrightarrow> t' \<le> maxTerm S"
proof -
  presume "S \<noteq> {}"
  with assms
  obtain t where t: "t \<in> S" "\<And> t'. t' \<in> S \<Longrightarrow> t' \<le> t"
  proof (induct arbitrary: thesis)
    case empty
    then show ?case by simp
    ...
```

elastic

Discovery

Where are master-eligible nodes?

Is there a master already?



Settings

`discovery.zen.ping.unicast.hosts` → `discovery.seed_hosts`

static

`discovery.zen.hosts_provider` → `discovery.seed_providers`

dynamic (file, EC2, GCE,...)

Master Election

Agree which node should be master

Form a cluster

elastic

FOLLOW THE LEADER

`discovery.zen.minimum_master_nodes`

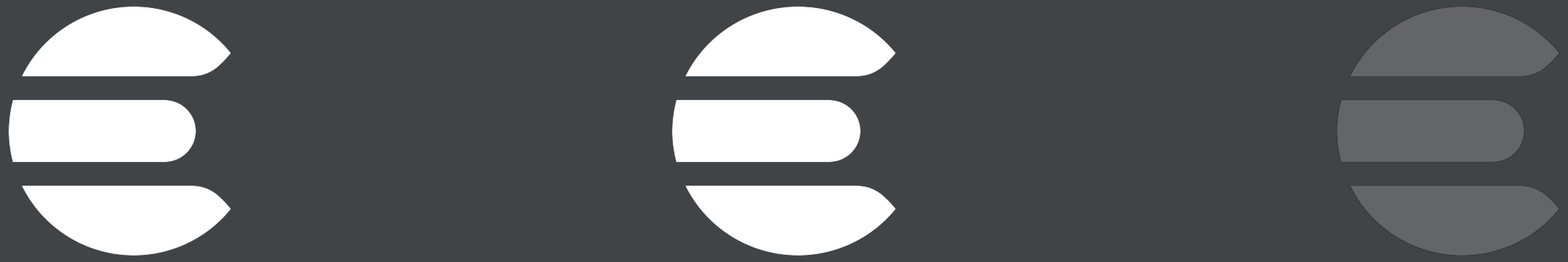
Trust users?

Scaling up or down?

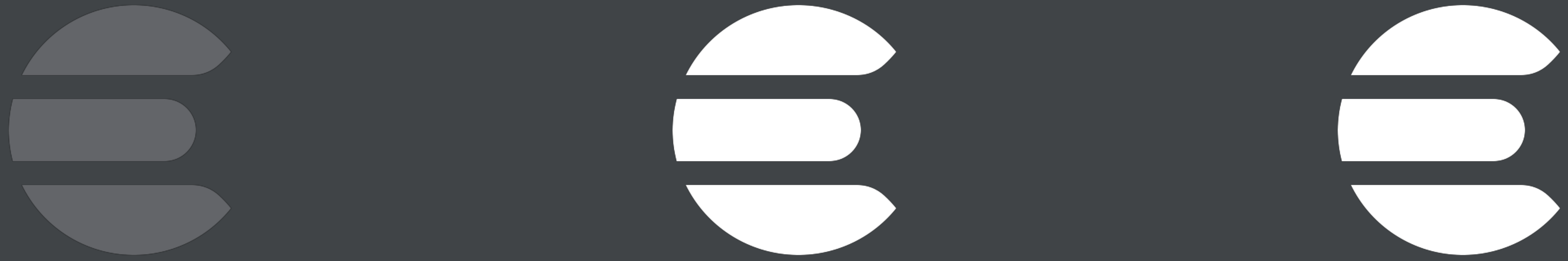
Three Node Cluster

discovery.zen.minimum_master_nodes: ~

discovery.zen.minimum_master_nodes: 2

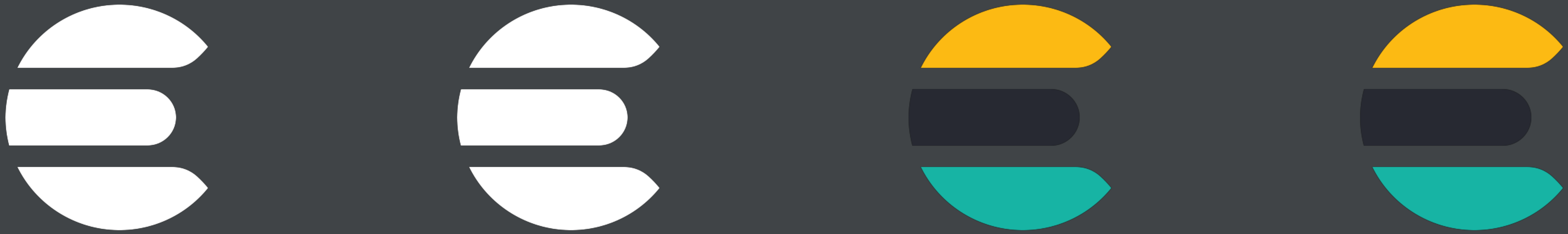


discovery.zen.minimum_master_nodes: 2



discovery.zen.minimum_master_nodes: 2

discovery.zen.minimum_master_nodes: 2



discovery.zen.minimum_master_nodes: 2

`cluster.initial_master_nodes`

List of node names for the very first election

OK

to set on multiple nodes as long as they are all consistent

Ignored

once node has joined a cluster even if restarted

Unnecessary

when joining new node to existing cluster

Upgrade 6 to 7

Full cluster restart: Set

`cluster.initial_master_nodes`

Rolling upgrade:

`cluster.initial_master_nodes` **not required**

Fresh Cluster

Empty `cluster.initial_master_nodes`

```
elasticsearch2    | {"type": "server",
                     "timestamp": "2019-05-24T14:02:51,173+0000",
                     "level": "WARN",
                     "component": "o.e.c.c.ClusterFormationFailureHelper",
                     "cluster.name": "docker-cluster",
                     "node.name": "elasticsearch2",
                     "message":
```

```
"master not discovered yet,
this node has not previously joined a bootstrapped (v7+) cluster,
and [cluster.initial_master_nodes] is empty on this node:
have discovered [
    {elasticsearch1}{pSUJ6OtSRWSrcWkRevLfyA}{_jIaabgyTQOHAOjcwUruIQ}
        {192.168.112.3}{192.168.112.3:9300}
        {ml.machine_memory=1073741824, ml.max_open_jobs=20, xpack.installed=true},
    {elasticsearch3}{ngaTCze8QHSHydCXsttXyw}{mbIad-A4SLOJvP7Ava5dEw}
        {192.168.112.4}{192.168.112.4:9300}
        {ml.machine_memory=1073741824, ml.max_open_jobs=20, xpack.installed=true}
];
```

```
discovery will continue using
        [192.168.112.3:9300, 192.168.112.4:9300] from hosts providers and [
    {elasticsearch2}{iANt64LESxqjJv8tHV5KKw}{KObYEuQ2TnamsiOefTUXgQ}
        {192.168.112.2}{192.168.112.2:9300}
        {ml.machine_memory=1073741824, xpack.installed=true, ml.max_open_jobs=20}
]
from last-known cluster state;
node term 0, last-accepted version 0 in term 0"
```

Dynamic Cluster Scaling

Master-ineligible: as before

Adding master-eligible: Just do it

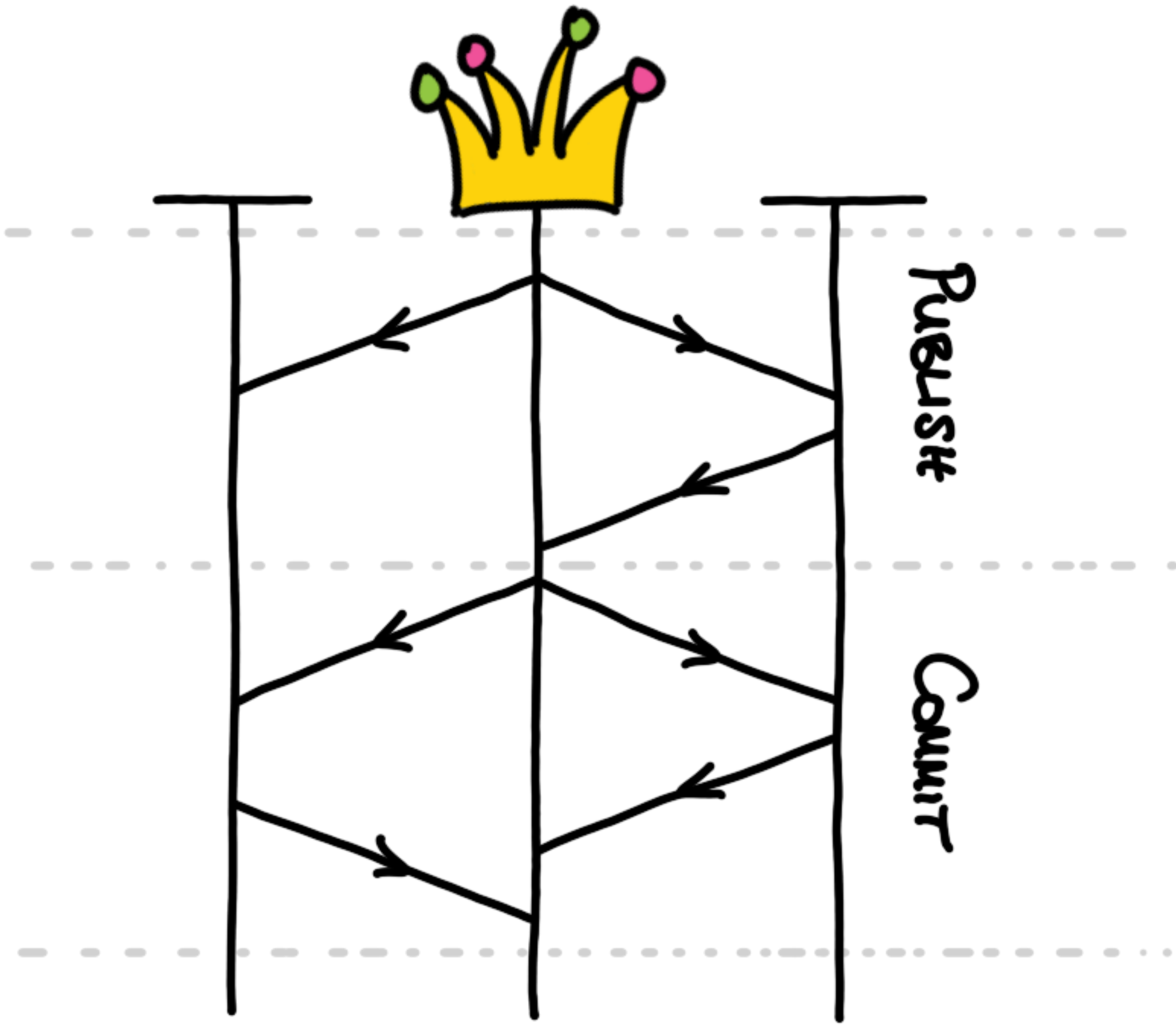
Removing master-eligible: Just do it

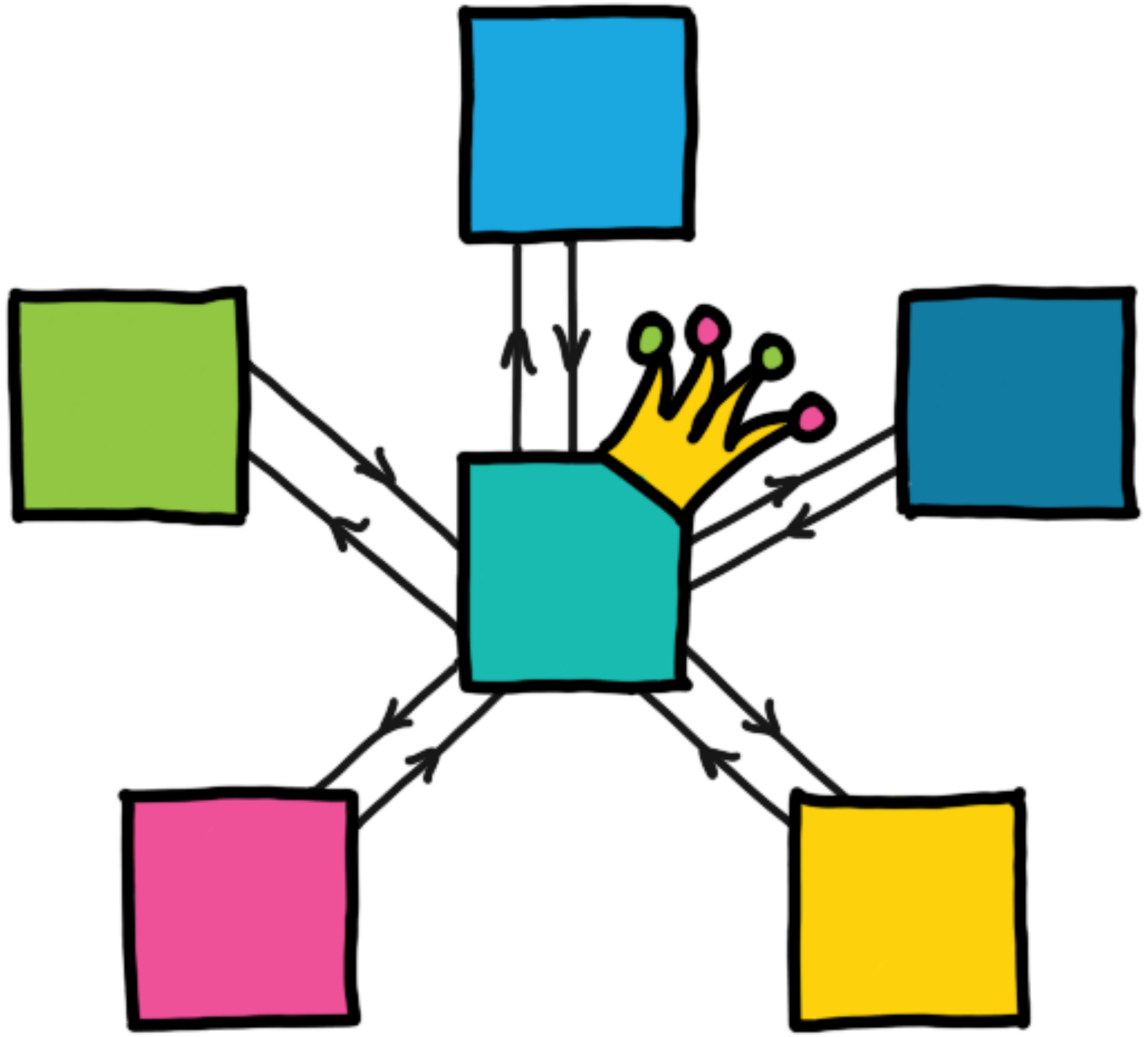
As long as you remove less than half of them at once

Cluster State
Publication

Agree on cluster state updates

Broadcast updates to all nodes


PUBLISH
COMMIT



Conclusion

Demo

https://github.com/xeraa/elastic-docker/tree/master/rolling_upgrade

```
elasticsearch1:
  image: docker.elastic.co/elasticsearch/elasticsearch:$ELASTIC_VERSION
  environment:
    - node.name=elasticsearch1
    - ES_JAVA_OPTS=-Xms512m -Xmx512m
    - discovery.zen.ping.unicast.hosts=elasticsearch2,elasticsearch3
    - discovery.zen.minimum_master_nodes=2
    #- discovery.seed_hosts=elasticsearch2,elasticsearch3
    #- cluster.initial_master_nodes=elasticsearch1,elasticsearch2,elasticsearch3
  volumes:
    - esdata_upgrade1:/usr/share/elasticsearch/data
  ports:
    - 9201:9200
  networks:
    - esnet
```

Zen to Zen2

Faster, safer, more debuggable

Tonight: Elasticsearch Meetup @Camunda

https://www.meetup.com/Elasticsearch-Berlin/

Reaching Zen in Elasticsearch's
Cluster Coordination

Philipp Krenn @xeraa

elastic