High Availability Kubernetes on Bare-Metal
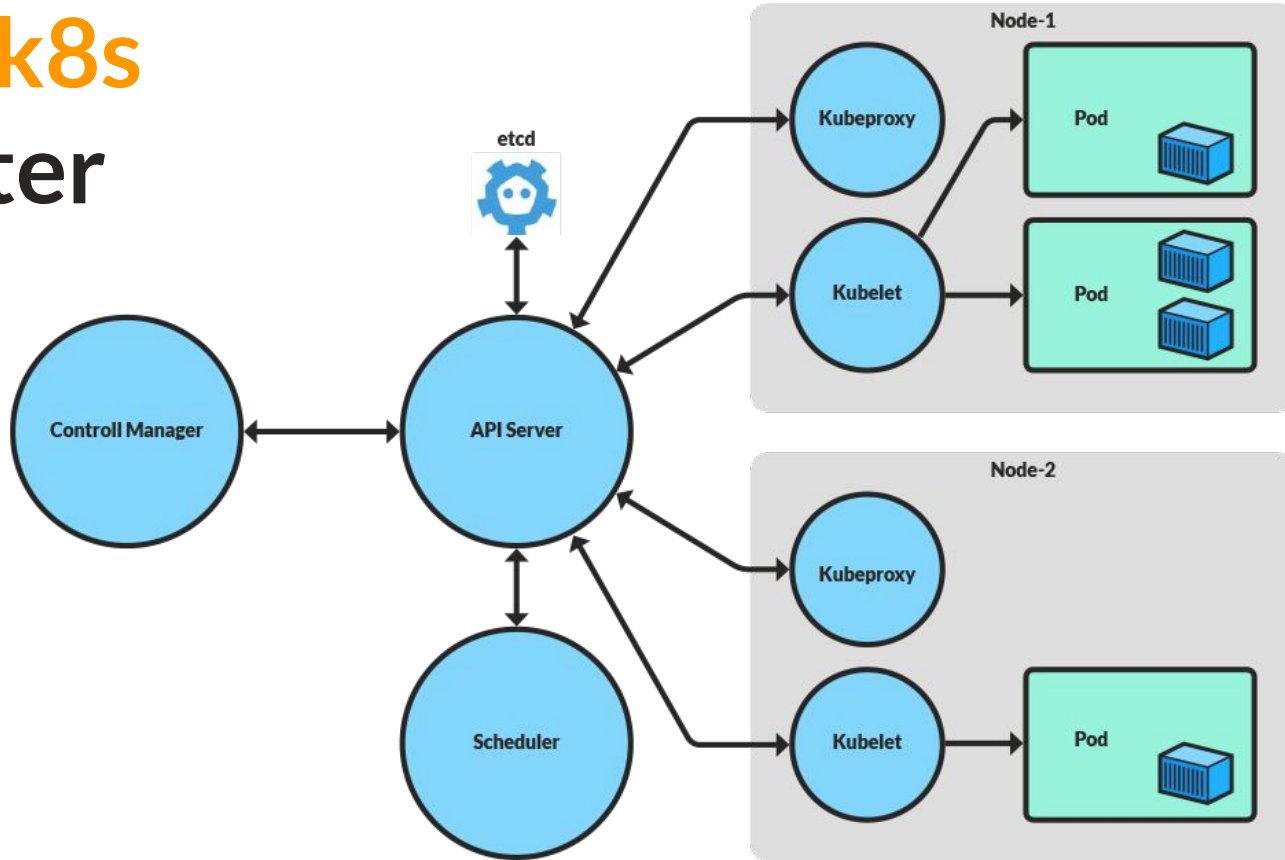
By Muhammad Kamran Azeem & Henrik Høegh

PRAQMA

The k8s cluster

High Availability Kubernetes on Bare-Metal
www.praqma.com

The k8s cluster

The big picture

PRAQMA

High Availability Kubernetes on Bare-Metal
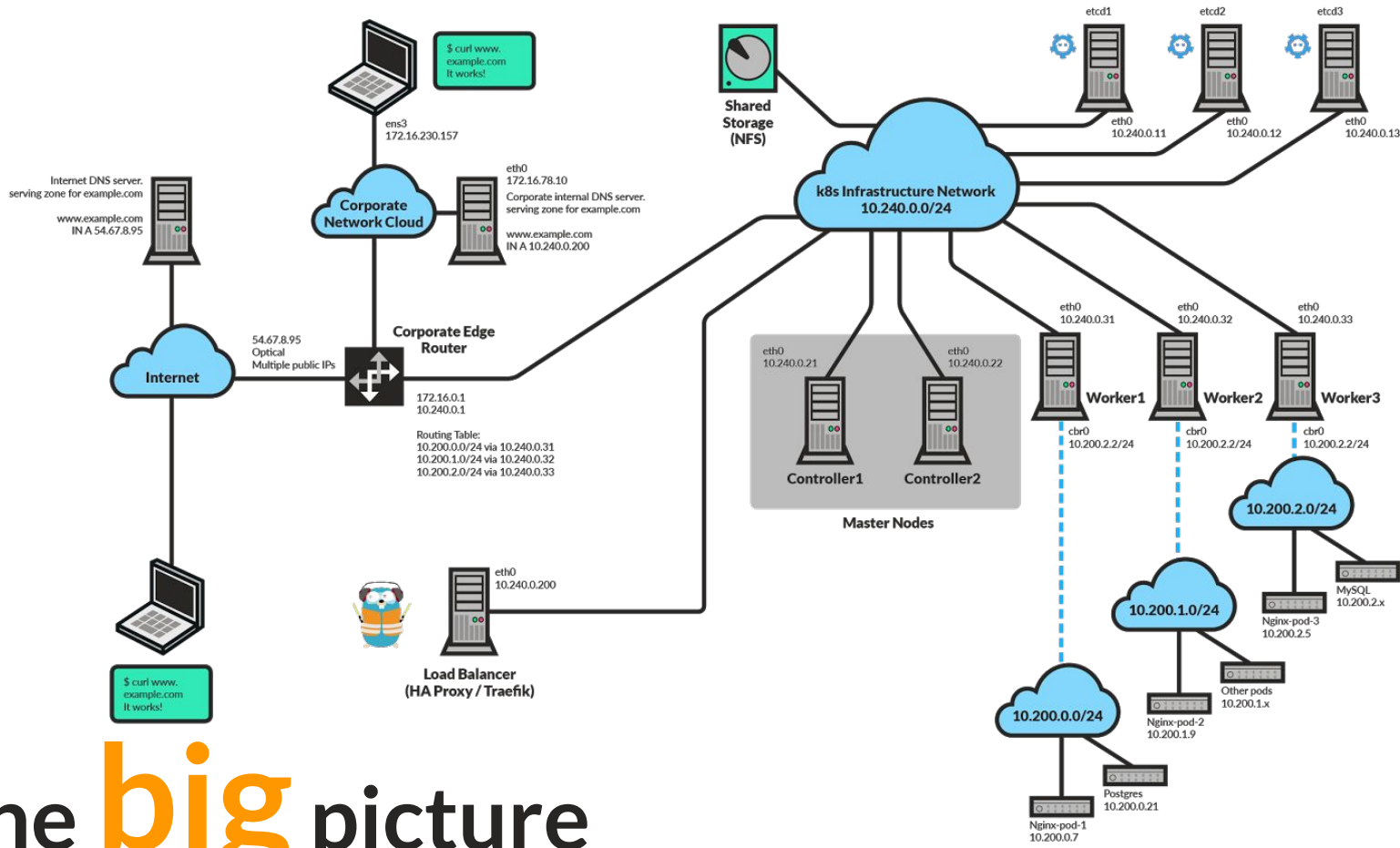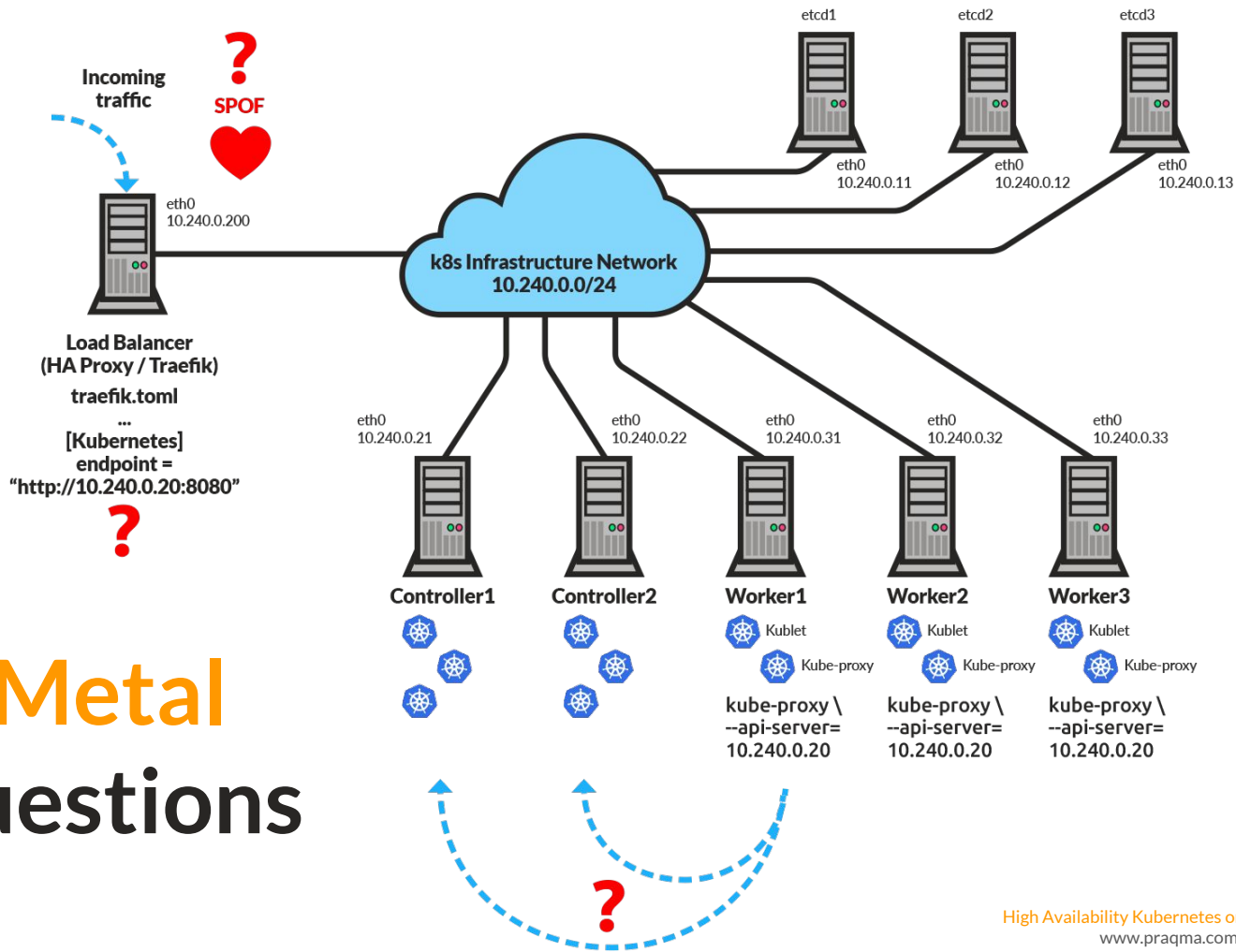www.praqma.com

In the cloud we can simply *use load balancers available from the cloud providers.*

Incoming traffic

etcd1 eth0 10.240.0.11
etcd2 eth0 10.240.0.12
etcd3 eth0 10.240.0.13

k8s Infrastructure Network
10.240.0.0/24

Cloud provided
Load Balancers
for each k8s
service

eth0 10.240.0.21
eth0 10.240.0.22
eth0 10.240.0.31
eth0 10.240.0.32
eth0 10.240.0.33

Controller1
Controller2
Worker1
Worker2
Worker3

Kublet
Kube-proxy

kube-proxy \
--api-server=
10.240.0.20

kube-proxy \
--api-server=
10.240.0.20

kube-proxy \
--api-server=
10.240.0.20

Cloud - ez pz!

10.240.0.20

Cloud provided
Load Balancer
for API-Server

PRAQMA

High Availability Kubernetes on Bare-Metal
www.praqma.com

So - What is the best way to *introduce HA to k8s cluster on bare-metal infrastructure?*

PRAQMA

Incoming
traffic

? SPOF ♥

etcd1   etcd2   etcd3

eth0   eth0   eth0
10.240.0.11   10.240.0.12   10.240.0.13

eth0
10.240.0.200

Load Balancer
(HA Proxy / Traefik)

traefik.toml
...
[Kubernetes]
endpoint =
"http://10.240.0.20:8080"

k8s Infrastructure Network
10.240.0.0/24

eth0   eth0   eth0   eth0   eth0
10.240.0.21   10.240.0.22   10.240.0.31   10.240.0.32   10.240.0.33

Controller1   Controller2   Worker1   Worker2   Worker3

                                      Kublet   Kublet   Kublet
                                      Kube-proxy   Kube-proxy   Kube-proxy

                                      kube-proxy \   kube-proxy \   kube-proxy \
                                      --api-server=   --api-server=   --api-server=
                                      10.240.0.20   10.240.0.20   10.240.0.20

Bare-metal

...not so *ez pz* :(

Proxy
10.240.0.20

SPOF

♥ ?

PRAQMA

High Availability Kubernetes on Bare-Metal
www.praqma.com

DNS?

- **TTL & Cache issues**
- **DNS becomes Single Point Of Failure**

... no way!

Meet Corosync & PaceMaker
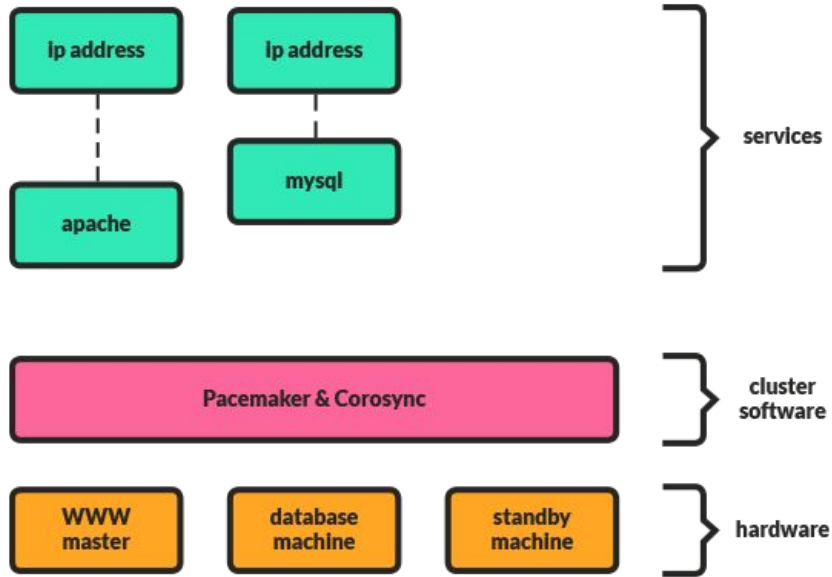

Corosync

The Corosync Cluster Engine

Pacemaker
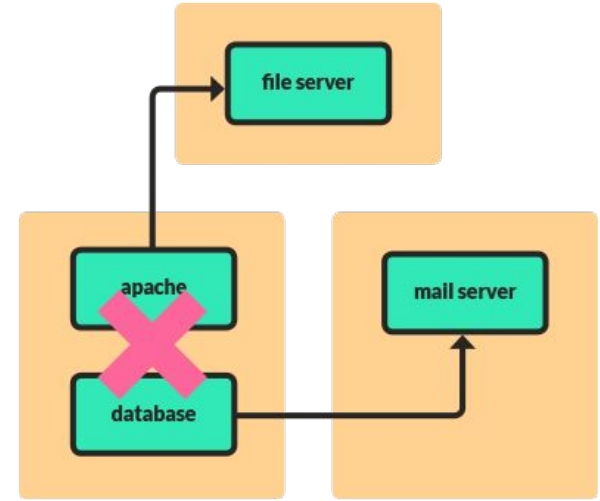
A scalable High Availability cluster resource manager.
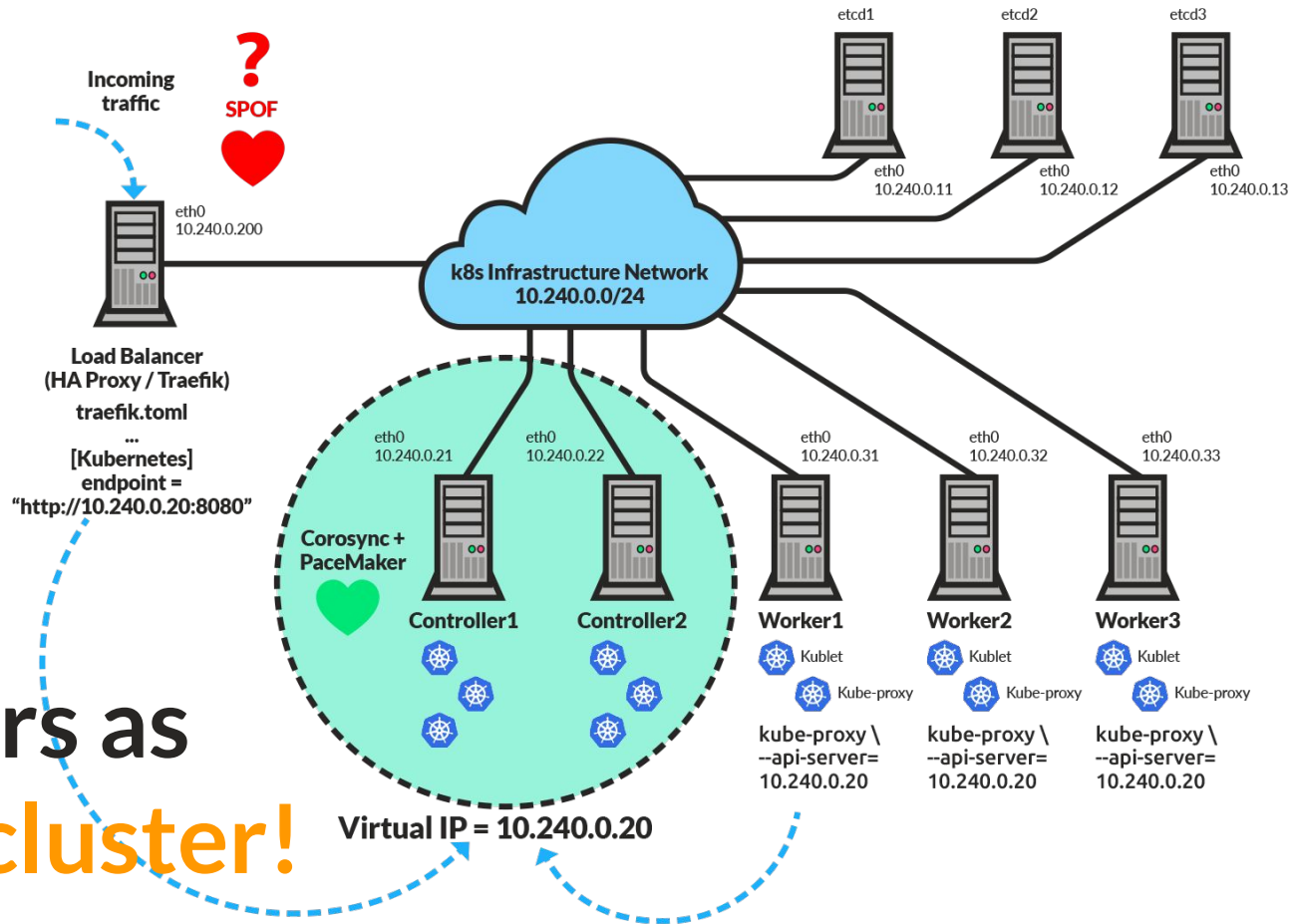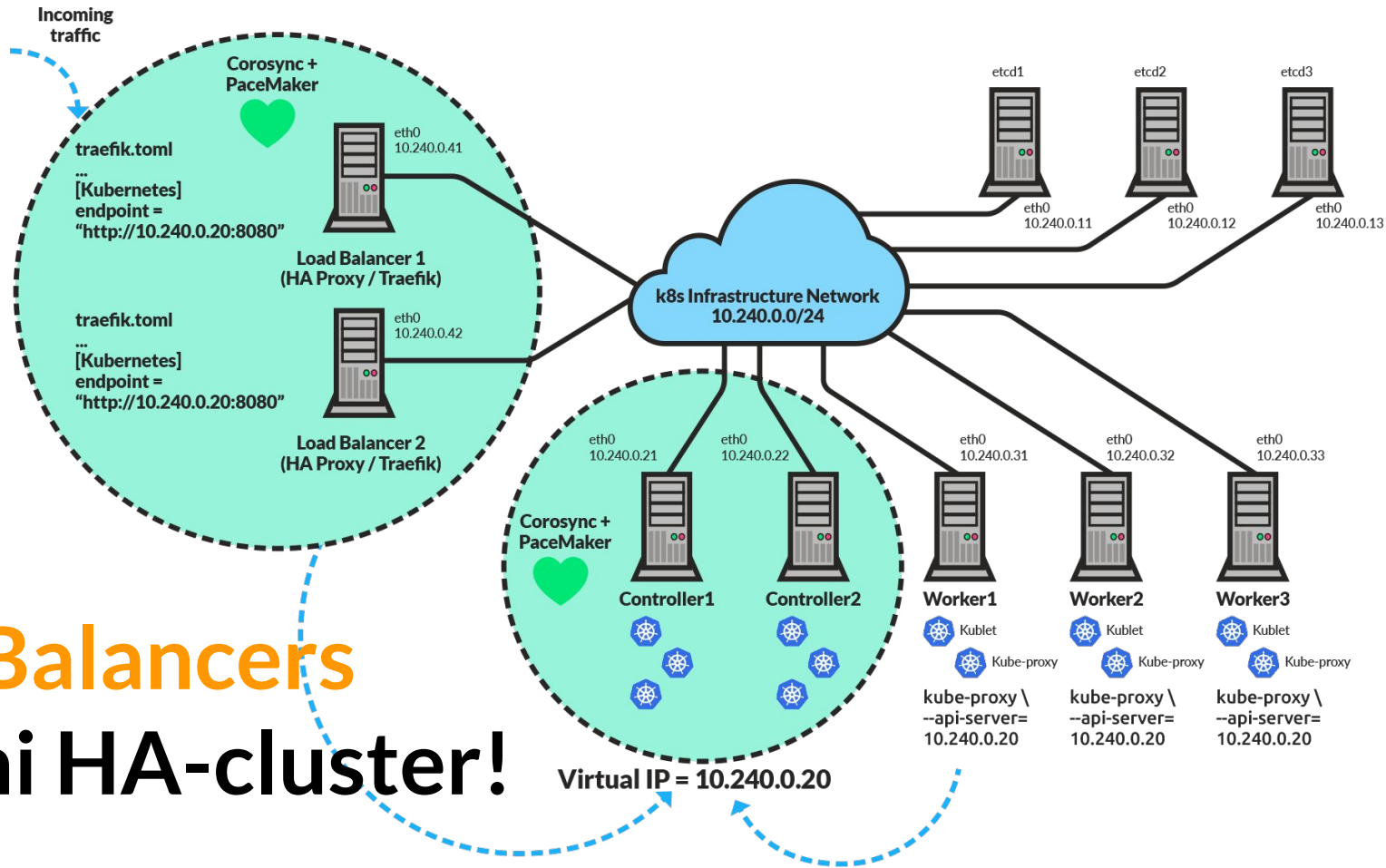
Pacemaker + Corosync - Visual Representation

Deploy

Monitor

Recover

PRAQMA

So ... Let's configure controllers as a mini-cluster! ... with heartbeat!

PRAQMA

Controllers as mini HA-cluster!

... and **Load balancers**
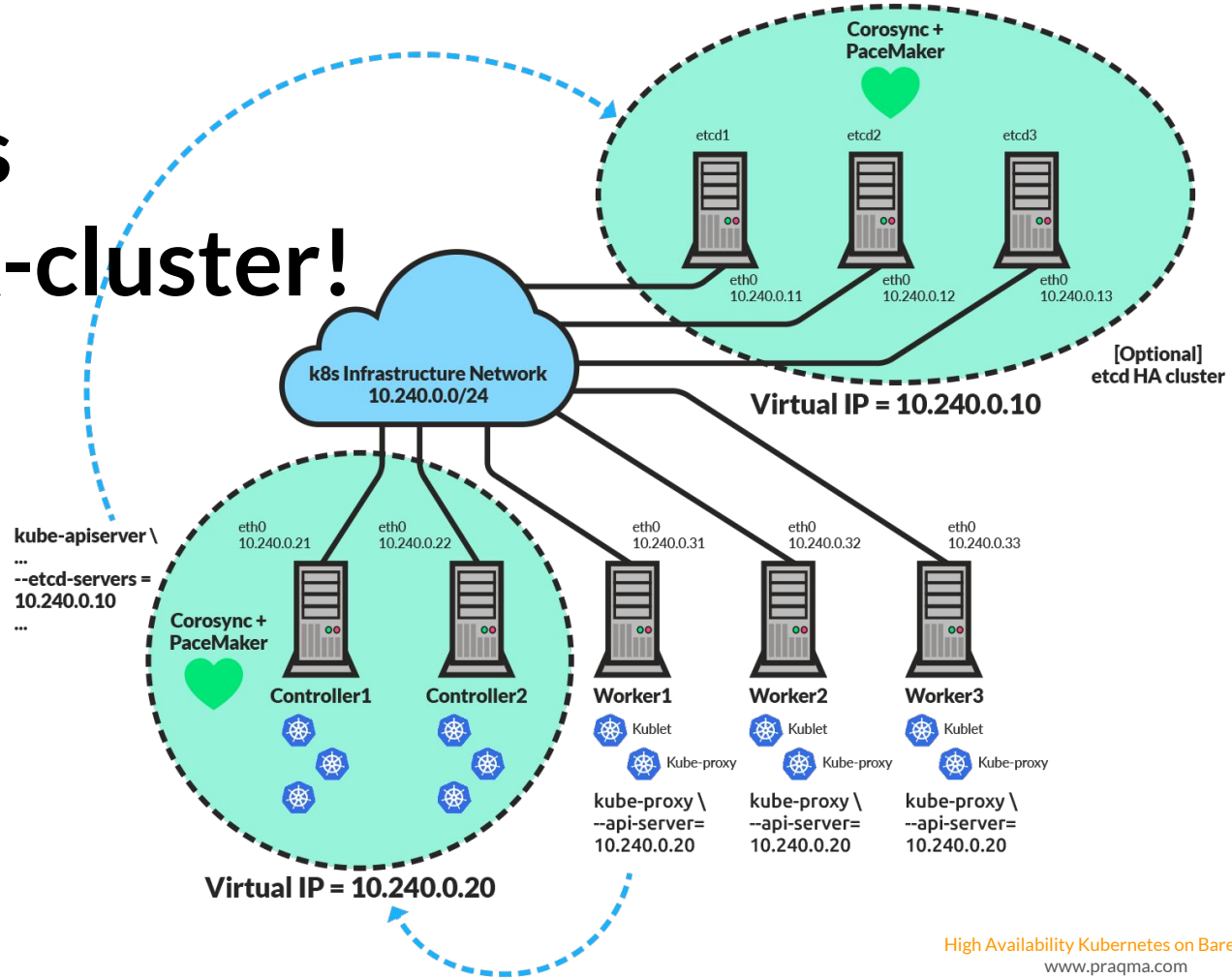
PRAQMA

Incoming traffic

Corosync + PaceMaker

traefik.toml
...
[Kubernetes]
endpoint = "http://10.240.0.20:8080"

eth0
10.240.0.41

Load Balancer 1
(HA Proxy / Traefik)

traefik.toml
...
[Kubernetes]
endpoint = "http://10.240.0.20:8080"

eth0
10.240.0.42

Load Balancer 2
(HA Proxy / Traefik)

etcd1

etcd2

etcd3

eth0
10.240.0.11

eth0
10.240.0.12

eth0
10.240.0.13

k8s Infrastructure Network
10.240.0.0/24

Corosync + PaceMaker

eth0
10.240.0.21

eth0
10.240.0.22

Controller1

Controller2

eth0
10.240.0.31

eth0
10.240.0.32

eth0
10.240.0.33

Worker1

Worker2

Worker3

Kublet

Kube-proxy

Kublet

Kube-proxy

Kublet

Kube-proxy

kube-proxy \
--api-server=
10.240.0.20

kube-proxy \
--api-server=
10.240.0.20

kube-proxy \
--api-server=
10.240.0.20

Virtual IP = 10.240.0.20

Load Balancers
as mini HA-cluster!

PRAQMA

High Availability Kubernetes on Bare-Metal
www.praqma.com

... and *etcd nodes!*

PRAQMA

Some explanation ...

PRAQMA

Corosync +
PaceMaker

k8s Infrastructure Network
10.240.0.0/24

PCS Resource
(Floating IP)
10.240.0.20

eth0
10.240.0.21

eth0
10.240.0.22

Corosync Totem Ring #1

Controller-1

Controller-2

Virtual IP = 10.240.0.20

PRAQMA

Node failure & Recovery

k8s Infrastructure Network
10.240.0.0/24

PCS Resource
(Floating IP)
10.240.0.20

eth0
10.240.0.21

eth0
10.240.0.22

Corosync Totem Ring #1

Controller-1

Controller-2
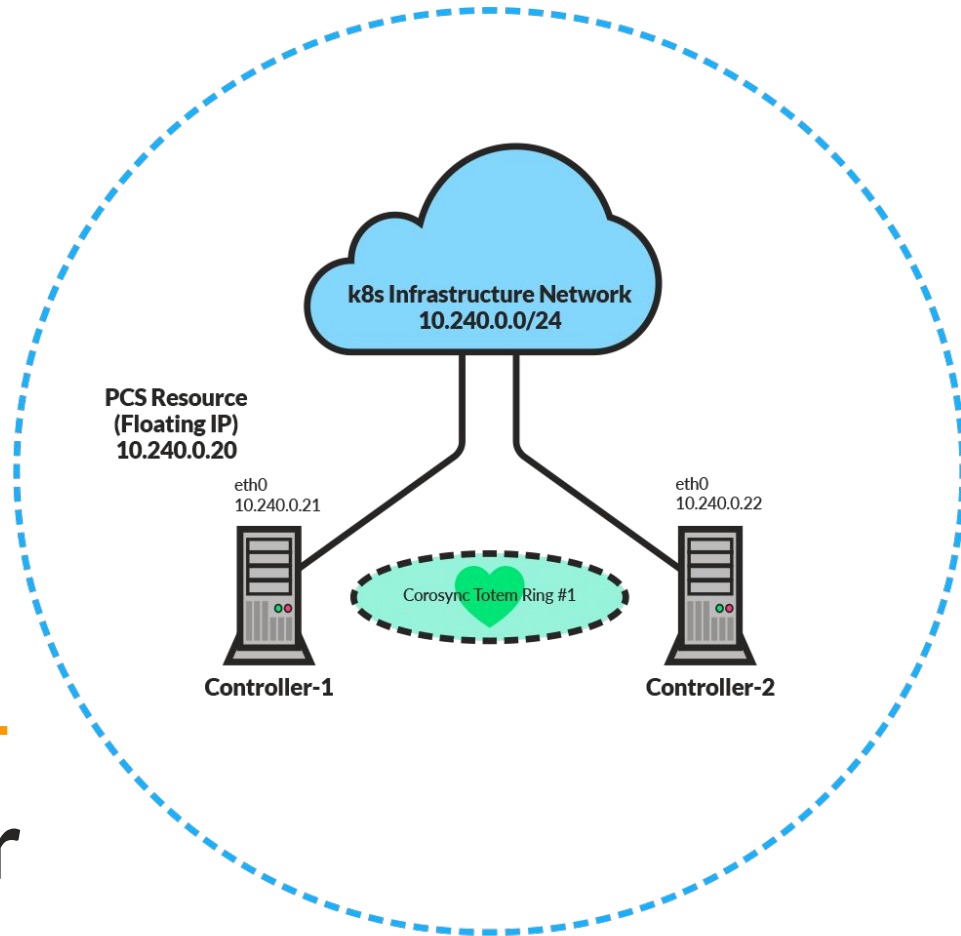
Corosync +
PaceMaker

Virtual IP = 10.240.0.20

PRAQMA

High Availability Kubernetes on Bare-Metal
www.praqma.com
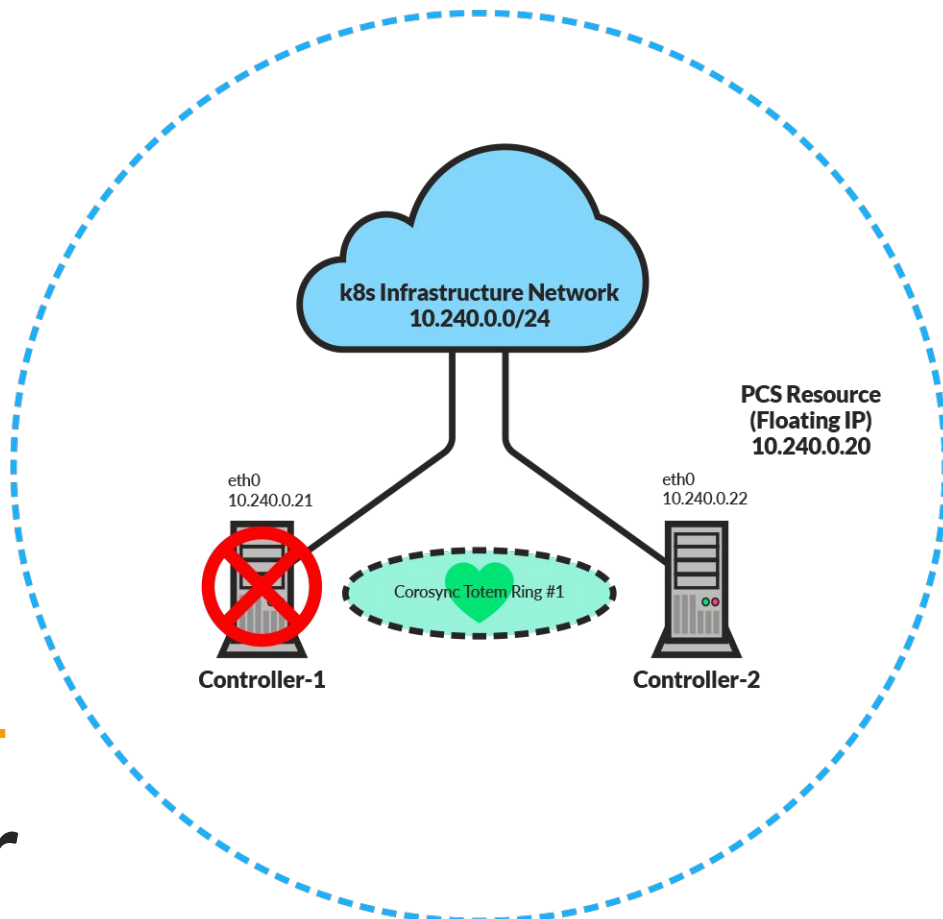
Multiple Communication Rings

k8s Infrastructure Network
10.240.0.0/24

Corosync Totem Ring #1

PCS Resource
(Floating IP)
10.240.0.20

eth0
10.240.0.41

eth1
192.168.101.41

eth2
192.168.102.41

Load balancer - 1

Redundant Network for
Second Totem Ring
192.168.101.0/24

Corosync Totem Ring #2

eth0
10.240.0.42

eth1
192.168.101.42

eth2
192.168.102.42

Load balancer - 2

Redundant Network for
Third Totem Ring
192.168.102.0/24

Corosync +
Pacemaker

Corosync Totem Ring #3

Corosync +
PaceMaker

PRAQMA

Virtual IP = 10.240.0.20

High Availability Kubernetes on Bare-Metal
www.praqma.com

Setup

Note: Requires Broadcast and Multicast

https://aws.**amazon**.com/vpc/faqs/

gateway.
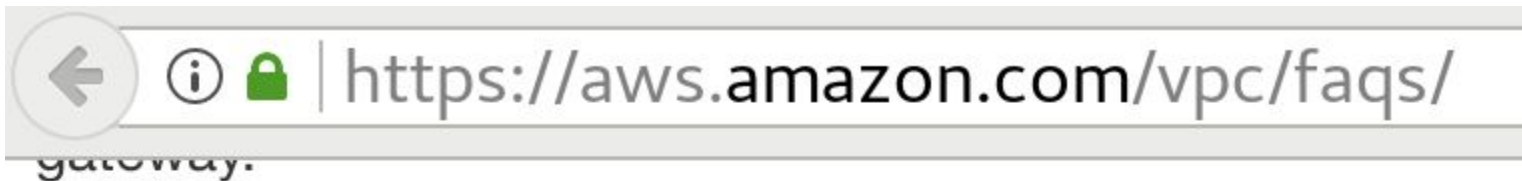
Q. Does Amazon VPC support multicast or broadcast?

No.

So …

Use VMs on local hypervisor

or

Use bare-metal!

On all (pacemaker) cluster nodes:

yum install corosync pacemaker pcs

```
pcs cluster setup --name ControllerHA \
  controller-1 controller-2 \
  --force

pcs cluster start --all

pcs cluster enable --all

pcs resource create ControllerVIP \
  ocf:heartbeat:IPaddr2 ip=10.240.0.20 \
  cidr_netmask=32 op monitor interval=30s
```

PRAQMA

```
[root@controller1 ~]# pcs status
Cluster name: ControllerHA
Stack: corosync
Current DC: controller2.praqma.local - partition with quorum
Last updated: 11:18:22 2017  Last change: 18:48:25 2017 by hacluster via crmd on
controller1.praqma.local

2 nodes and 1 resource configured

Online: [ controller1.praqma.local controller2.praqma.local ]

Full list of resources:

 ControllerVIP    (ocf::heartbeat:IPaddr2):    Started controller1.praqma.local

Daemon Status:
  corosync: active/enabled
  pacemaker: active/enabled
  pcsd: active/enabled
[root@controller1 ~]#
```
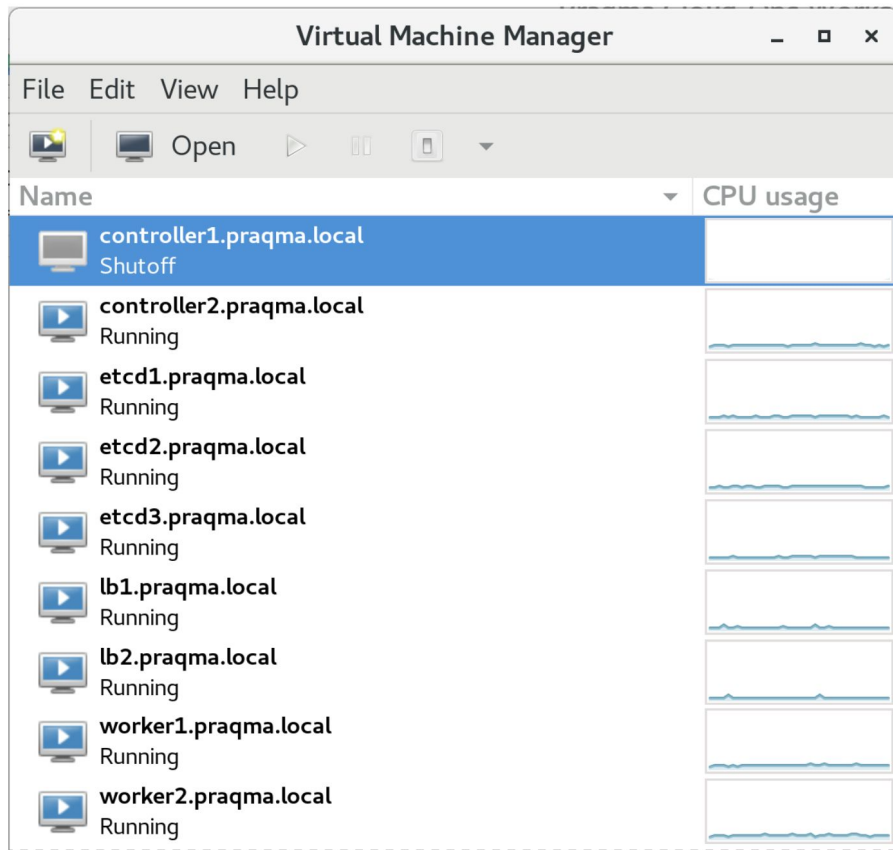
PRAQMA

Demo !!!

- Link to Kubernetes book (WIP)

Thanks!

Questions and comments?

kaz@praqma.net , heh@praqma.net

PRAQMA