

Brought by **Geekbang** **InfoQ**  
极客邦科技



Connect Container Community

# 全球容器技术大会

剖析容器企业实践 关注容器生态圈开源项目



# Private PAAS in Dianping based on Docker

Simon Sheng

# Agenda

1

- Design goals

2

- Component & Architecture

3

- Docker Customization

4

- Problems and Solutions

5

- Current Status

6

- Summary

# Design goals



Seamless integration with devops

Standardized application runtime

Replace KVM with the transparency to devops

High density deployment

Fast scale in/out

# The main user cases

## Dev

- Deploy War package
- Login into docker VM (RO)

## Operation

- Slale in/out App
- Clean up App
- Login into docker VM (RW)

# We are on the top of ...

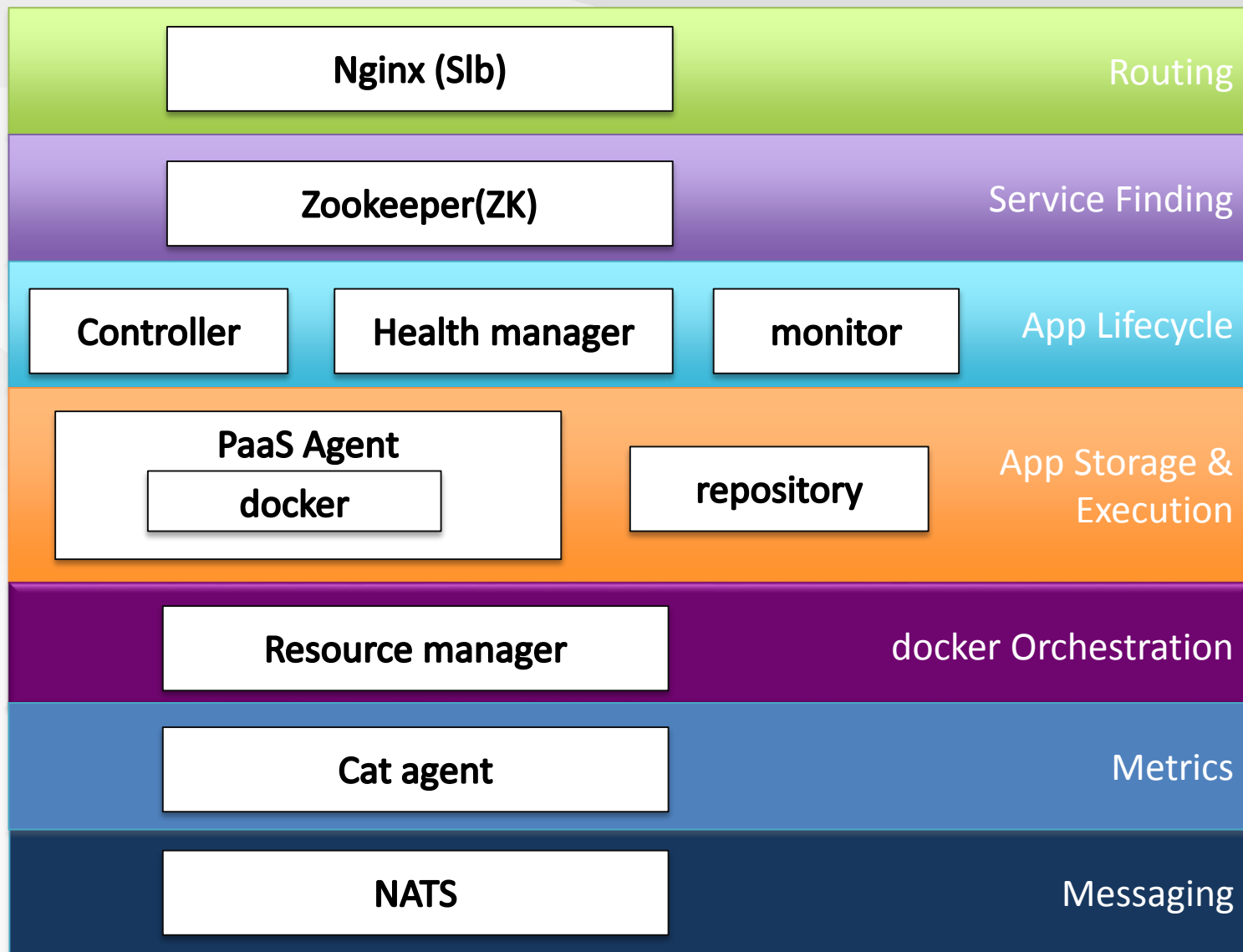


# Why Docker



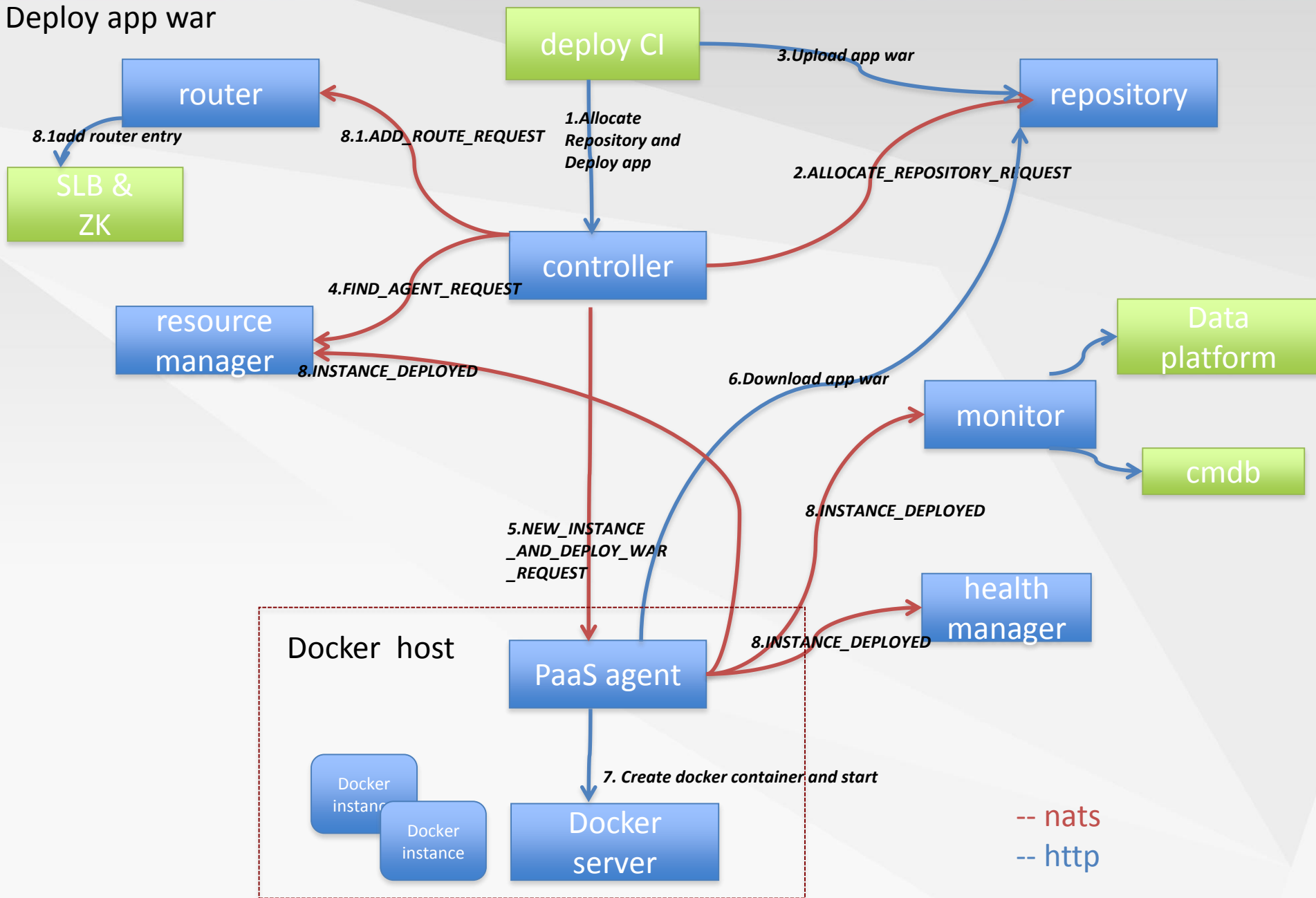
- Boots up in seconds
- Lightweight Linux container
- Layered files provides reusability of the runtime (But do not touch the old files ...)
- API is easy for use (docker-java)
- And ecosystem ...

# Components

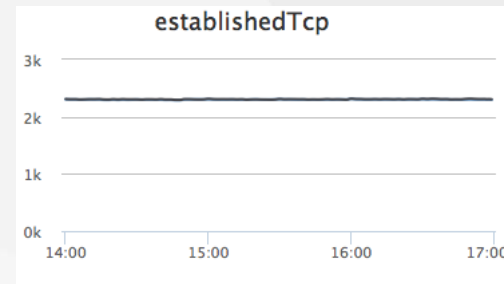
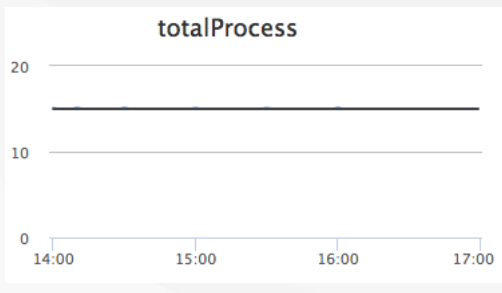
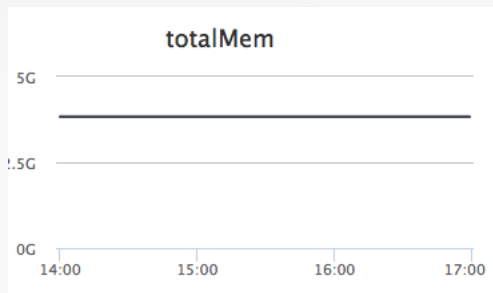
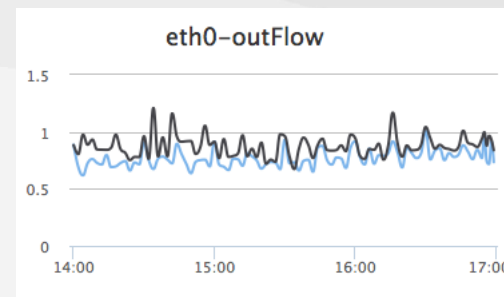
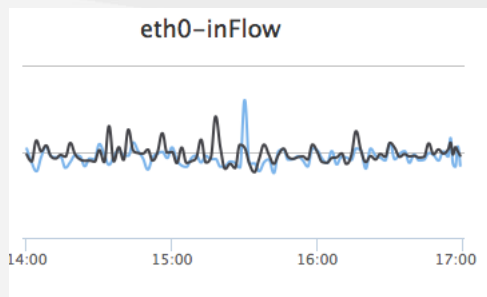
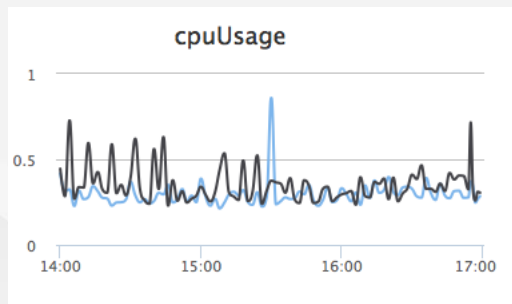




## Deploy app war



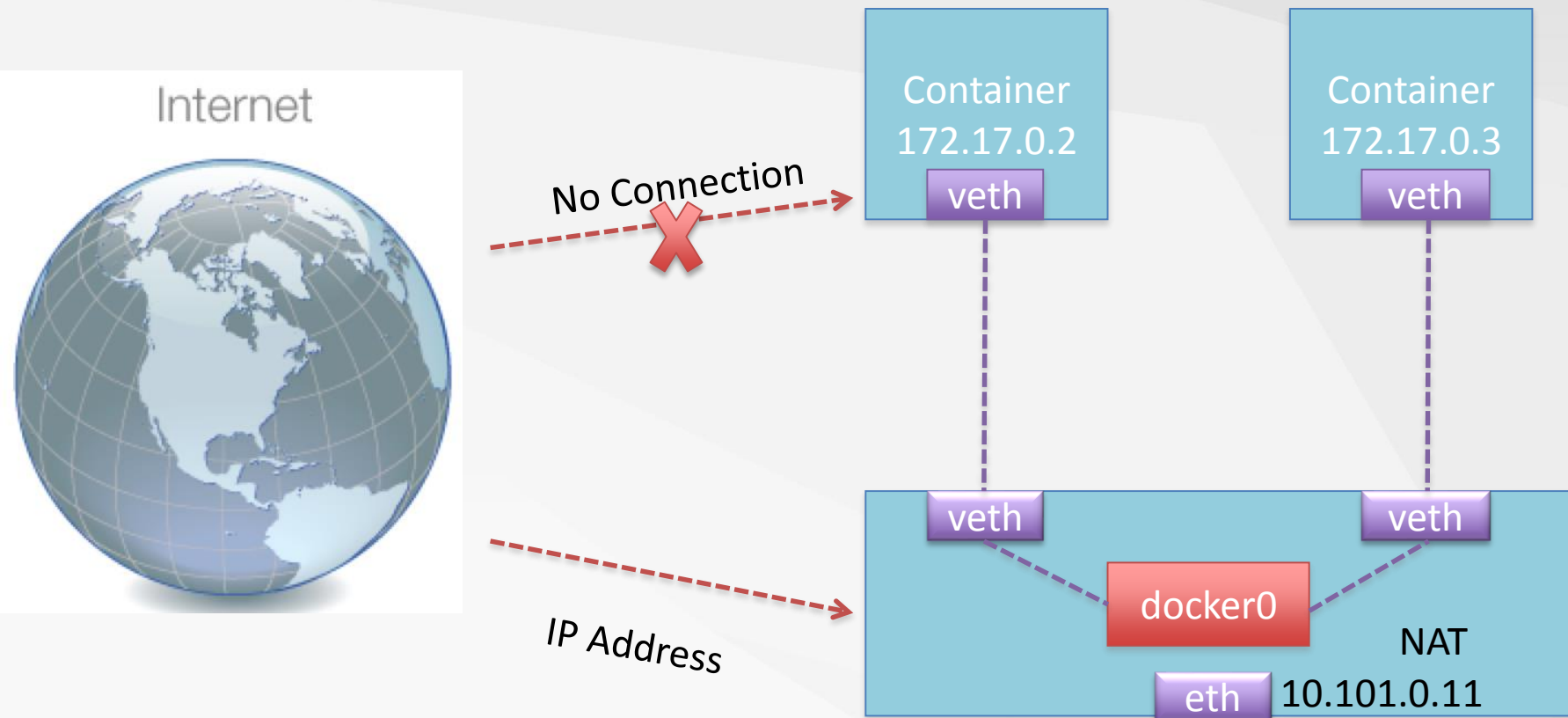
# Collect metrics to CAT(Central Application Tracking)



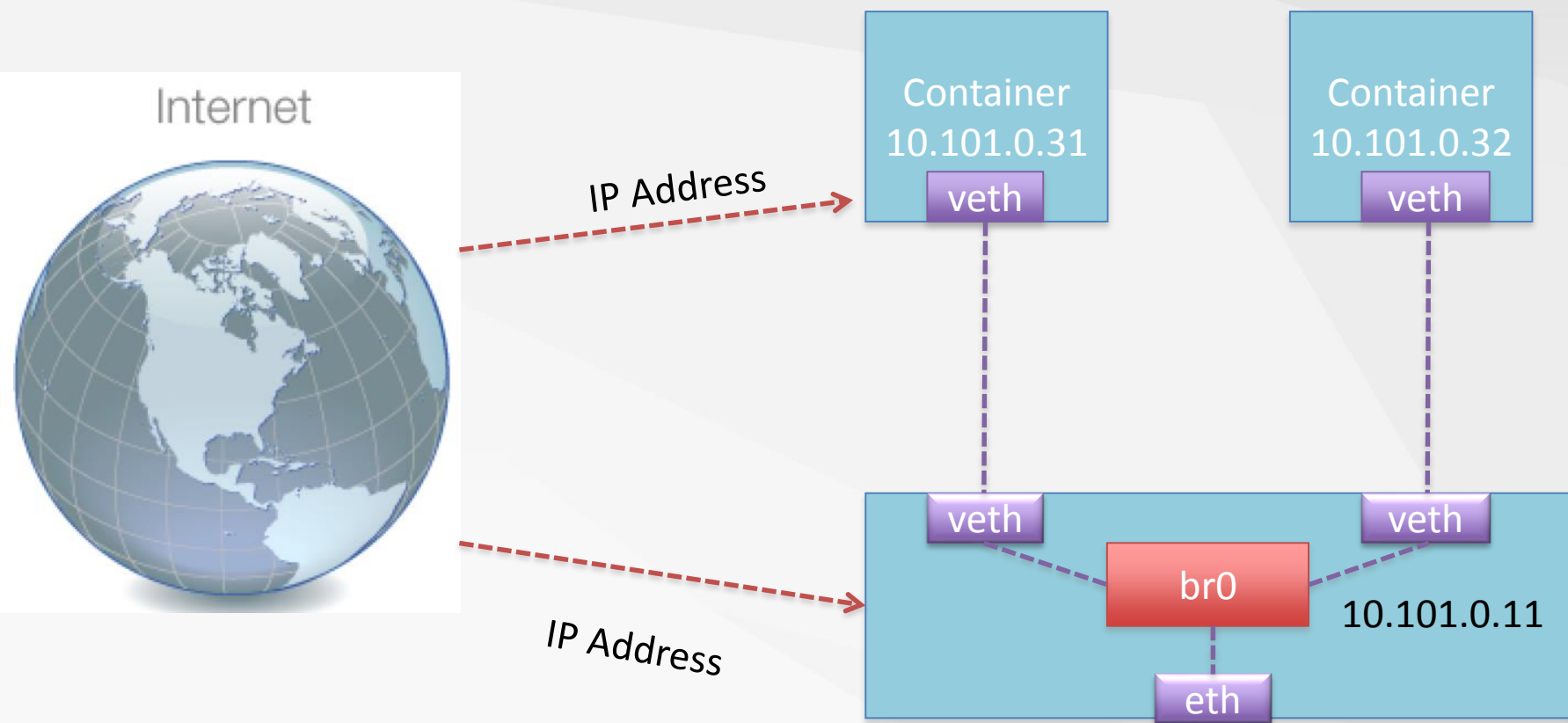
# Container metrics data

- metrics from cgroup
  - Memory
  - CPU
- metrics from container
  - Network
  - Process
  - Disk

# Docker default NAT network



# Customized public network



# Image

- Setup private registry
- Build different image (runtime) for different kind of app (web, mq, search etc.).
- Pre-fetch image from registry before creating container to accelerate creation speed.

# Problem

- Host machine crashed randomly
- Vmcore-dmesg: dm\_thin operation causes the problem

# Solution

- Make the blk discard false

```
docker -d --storage-driver=devicemapper --  
storage-opt dm.mountopt=nodiscard --  
storage-opt dm.blkdiscard=false
```



# Problem

- Puppet can not update file in container
- Container rootfs comes from
  - Layered image produced by Dockerfile
  - Volumes produced in host machine by PaaS agent (dynamically generated at create time)
- Puppet update is to delete the old file and replace with a new one.
- Volumes can not be deleted in container (Note: file under volume dir can be deleted )

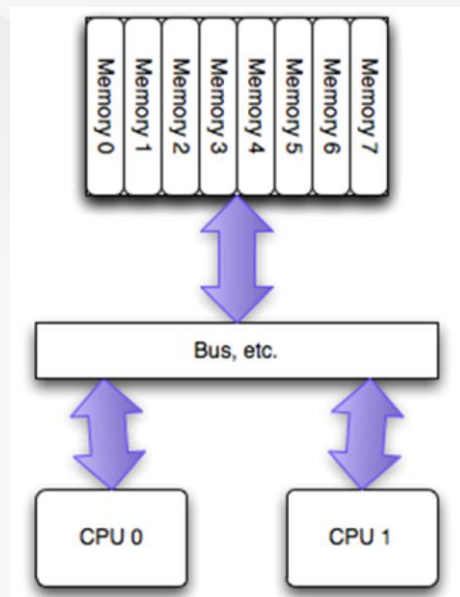
# Solution

- Move the volumes files into read-write layer of the container

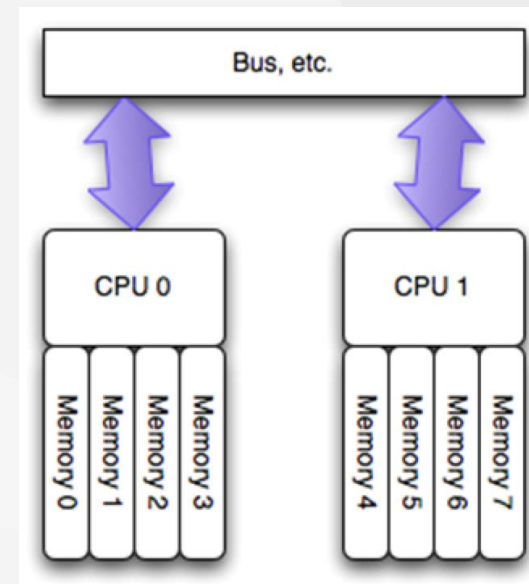
# Problem

- Swap frequency while physical memory is still redundant

UMA



NUMA



# Solution

```
[root@PaaS-10-3-3-11 ~]# numactl -show
policy: default
preferred node: current
physcpubind: 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
cpubind: 0 1
nodebind: 0 1
membind: 0 1
```

```
[root@PaaS-10-3-3-11 ~]# numactl --hardware
available: 2 nodes (0-1)
node 0 cpus: 0 1 2 3 4 5 12 13 14 15 16 17
node 0 size: 32739 MB
node 0 free: 15488 MB
node 1 cpus: 6 7 8 9 10 11 18 19 20 21 22 23
node 1 size: 32768 MB
node 1 free: 19356 MB
node distances:
node  0  1
 0:  10  20
 1:  20  10
```

- Allocate docker CPU group to use “local” memory
  - cpuset 0,1,2,3,4,5
  - cpuset 6,7,8,9,10,11
- Adjust system vm.swapiness

# Problem

- Some application in docker can not detect stale connection quickly.
- Too small opened file/processes limit.

# Solution

- Adjust tcp.keepalive configuration in host .
  - `echo 300 >/proc/sys/net/ipv4/tcp_keepalive_time`
- Adjust ulimit configuration in docker init.
  - `/etc/init.d/docker`

# Problem

- Fix docker problem need to restart docker daemon which will stop the applications.

# Solution

- We had to migrate the service capacity of the host out, then shut down the daemon.
- Hope the community come up with good solution.



# Problem

- Health check of the docker container
- Candidate may be
  - Port detection
  - File detection
  - ...

# Solution

- Make convention that each application expose a health check url.
- Thus check the `http://ip:port/url`

# Current Status

- 600+



- 2800+



- 200+



- 2



# Summary

- We have improved the efficiency by seamless integration with devops
- Docker provided the standard runtime by the layered image (Dockerfile)
- Docker can be used as “VM” by combining with public network.
- Docker is good at high density deployment and fast scale in/out.

# THANKS

全球容器技术大会

剖析容器企业实践 关注容器生态圈开源项目