

OPENSTACK DAYS
CHINA

Topic: OpenStack容器相关项目介绍

Speaker: 陆鸿斌 & 乔立勇



议程

- 容器与虚拟化技术的对比
- OpenStack上和容器相关的项目有哪些？
Kolla, Solum, Kuryr, Magnum, Murano, Nova-docker, Nova-lxd, libvirt-lxc, Heat docker plugin, Higgins (Zun)
- 其他与OpenStack 和容器相关的项目，Clear Container, Ciao
- 容器与虚拟化融合的发展趋势

容器 VS 虚拟化 (1)

	容器	虚拟化
原理	Linux Cgroup, Linux 命名空间等隔离技术	模拟硬件, 完整的软件栈
资源使用情况	高效共享资源, 紧耦合	占用更多资源 更重, 有资源损耗 (5%)
灵活性&安全性	不可迁移, 隔离性, 网络性能有待提高	可在线热迁移, 隔离性好



容器 VS 虚拟化 (2)

容器	虚拟化
适合运行临时性任务 1日志分析 2大数据计算	持久性任务 NFV
web服务器，微服务	数据库
IoT/软件打包部署	异构性OS，虚拟桌面，其他独占的服务器



OpenStack中和容器有关的项目

Kolla, Solum

Magnum, Murano

Kuryr, Nova-docker, Nova-lxd, libvirt-lxc,

Heat docker plugin

Zun

Kolla(1)

- 14年9月成立
- 目标：使用Docker和ansible提供生产级的OpenStack各服务的打包和部署的功能
 - docker 提供镜像的build
 - ansible 提供部署，升级
 - 使用kubernet/mesos 部署镜像
- 受益：简化部署，简化运维，提升devops，可重现，可依赖，更快（对比devstack 14mins ~ 9 mins）

Kolla(2)功能

容器化了以下组件

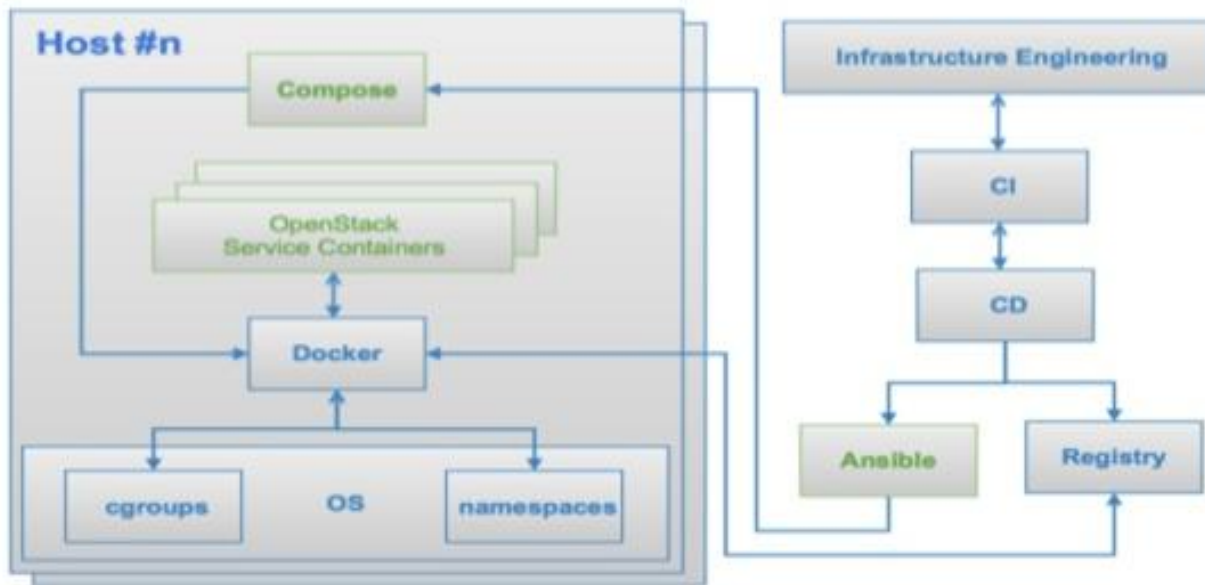
```
aodh          designate    heat        keystone    mariadb     nova        swift
base          dind        heka        kibana      memcached   openstack-base  tempest
ceilometer   elasticsearch  horizon    kolla-toolbox  mistral     openvswitch  tgtd
ceph         glance      ironic      kuryr      mongoddb    rabbitmq     trove
cinder       gnocchi     iscsid      magnum     murano      rally        zaqar
cron         haproxy     keepalived  manila     neutron     sahara
```

提供Ansible 部署(playbooks)

```
ceilometer   cinder     elasticsearch  heat    iscsi    magnum    memcached  murano  prechecks
ceph         cleanup   glance        horizon  keystone  manila    mistral    neutron  rabbitmq
certificates  common    haproxy       ironic  kibana   mariadb   mongoddb   nova    swift
```



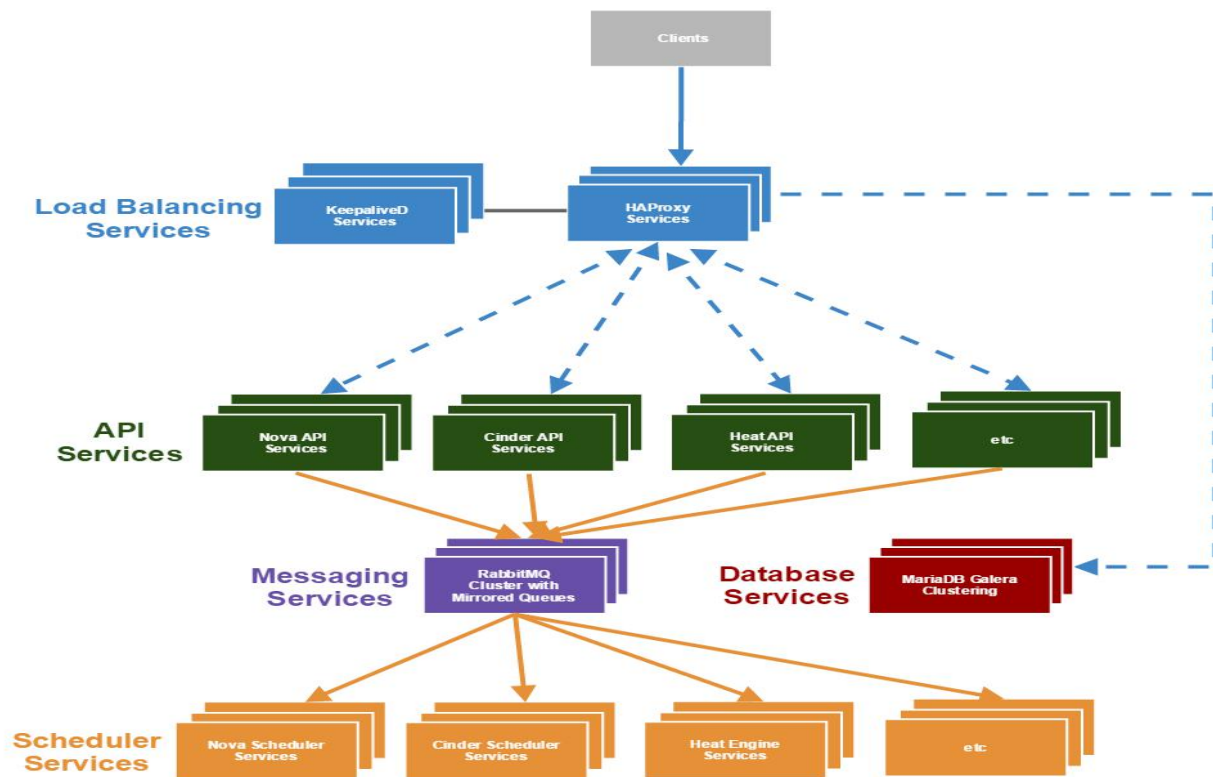
Kolla架构



0
1
2
3
4
5
6
7
8
9
A
B
C
D
E
F
G
H
I
J
K
L
M
N
O
P
Q
R
S
T
U
V
W
X
Y
Z
[
\
]
^
_
`
a
b
c
d
e
f
g
h
i
j
k
l
m
n
o
p
q
r
s
t
u
v
w
x
y
z
{
|
}
~
0
1
2
3
4
5
6
7
8
9
A
B
C
D
E
F
G
H
I
J
K
L
M
N
O
P
Q
R
S
T
U
V
W
X
Y
Z
[
\
]
^
_
`
a
b
c
d
e
f
g
h
i
j
k
l
m
n
o
p
q
r
s
t
u
v
w
x
y
z
{
|
}
~



kolla(HA)



Kolla 实践

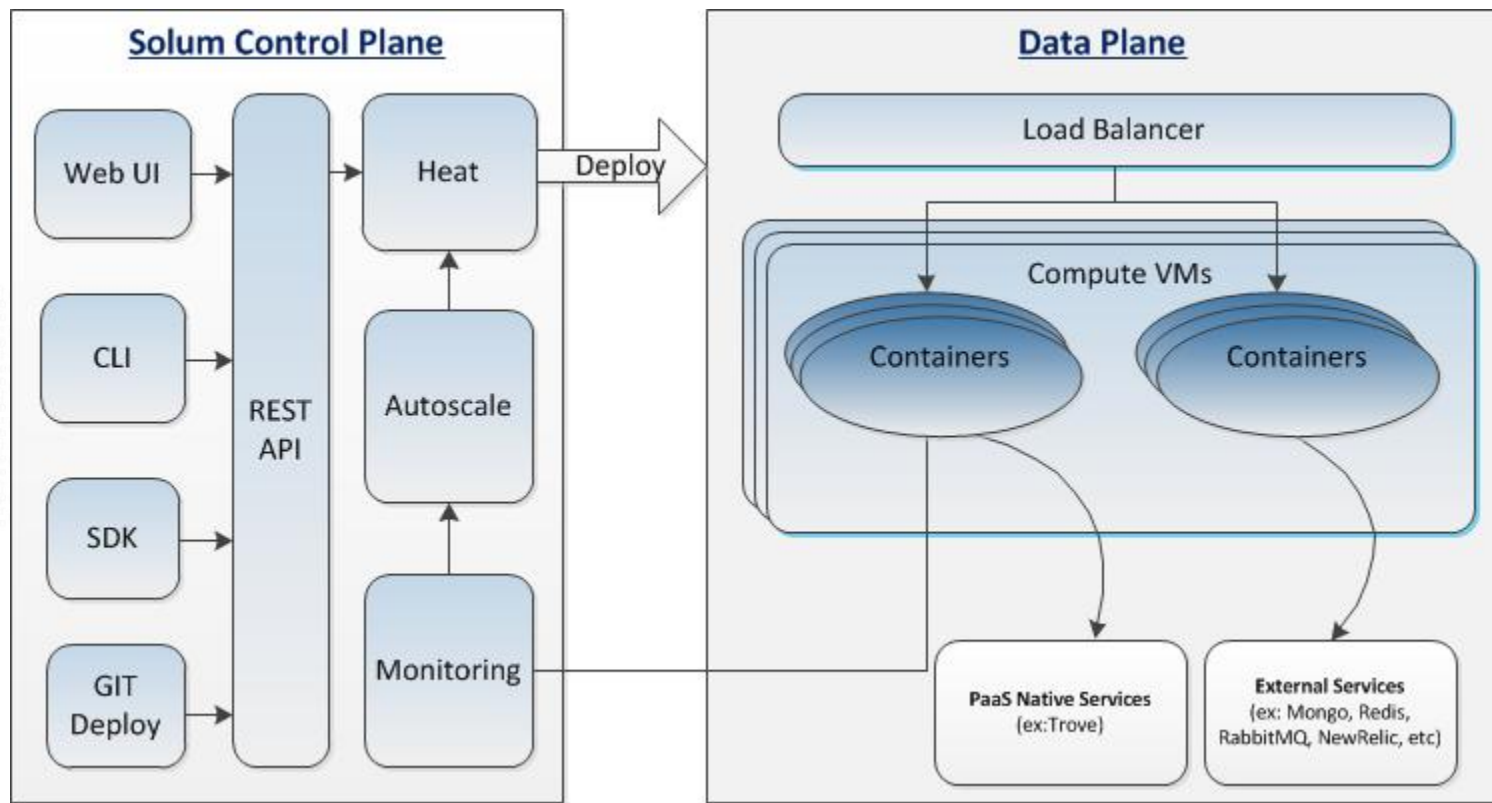
- kolla-genpwd
- * kolla-build --base [ubuntu|centos|fedora|oraclelinux] --type [binary|source]
- kolla-ansible prechecks -i <path/to/multinode/inventory/file>
- kolla-ansible pull -i <path/to/multinode/inventory/file>
- kolla-ansible deploy -i <path/to/multinode/inventory/file>
- kolla-ansible post-deploy
- kolla/tools/init-runonce



Solum

目标: 提供OpenStack中持续集成/开发方案
OpenStack的原生方案, 利用OpenStack中的各种服务, 面向开发者提供编程语言透明的持续集成方案。

Solum 系统架构



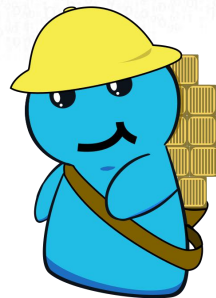
Solum 实践

- `solum languagepack create <NAME> <GIT_REPO>`
- `solum app create --app-file <app_file> [--param-file param_file]`
- languagepack -> docker base image
- app -> heat and docker container

kuryr(1)

一个让Docker能够使用Neutron服务的插件

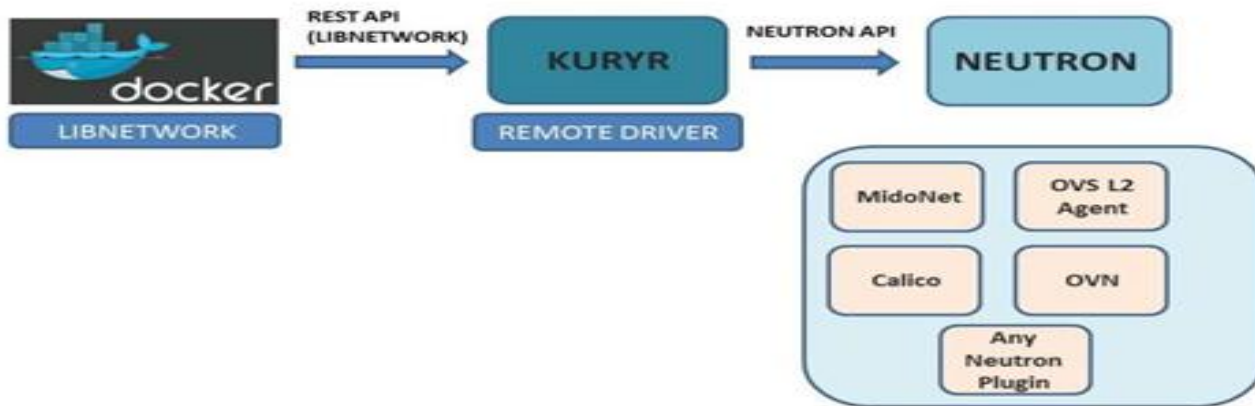
- Neutron Docker CNM
- 目标是支持各种容器编排引擎 eg:
Kubernetes, Mesos, Docker Swarm
- 与 Neutron, Magnum, Kolla 协作工作



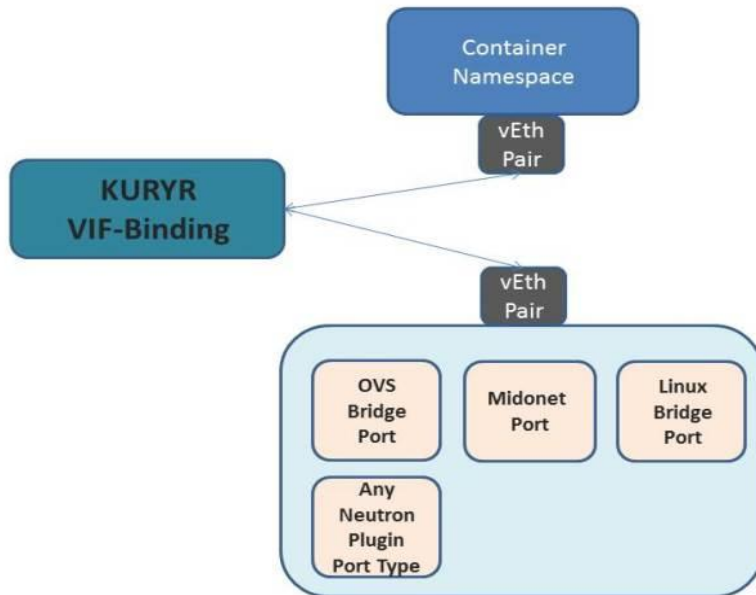
kuryr(2)

```
$ sudo docker network create --driver=kuryr \ --ipam-driver=kuryr \ --subnet 10.0.0.0/16 \ --gateway 10.0.0.1 \ --ip-range 10.0.0.0/24 foo
```

```
$ sudo docker run --net=foo -itd --name=container1 busybox
```



kuryr(3)



Magnum(1)

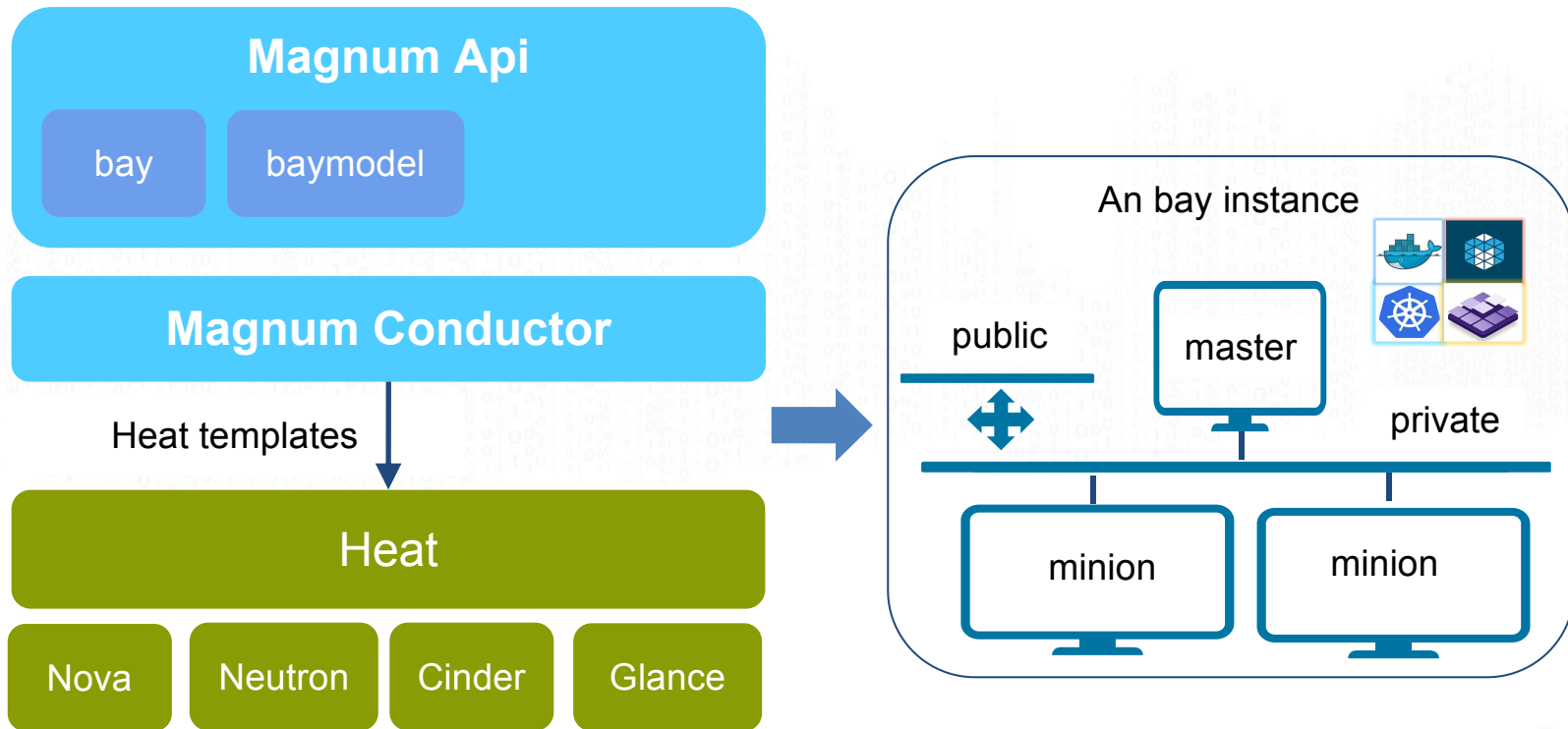
14年年底成立

集成容器编排引擎（COE），最大限度地利用OpenStack中的各种服务提供在OpenStack上的容器编排引擎管理服务：

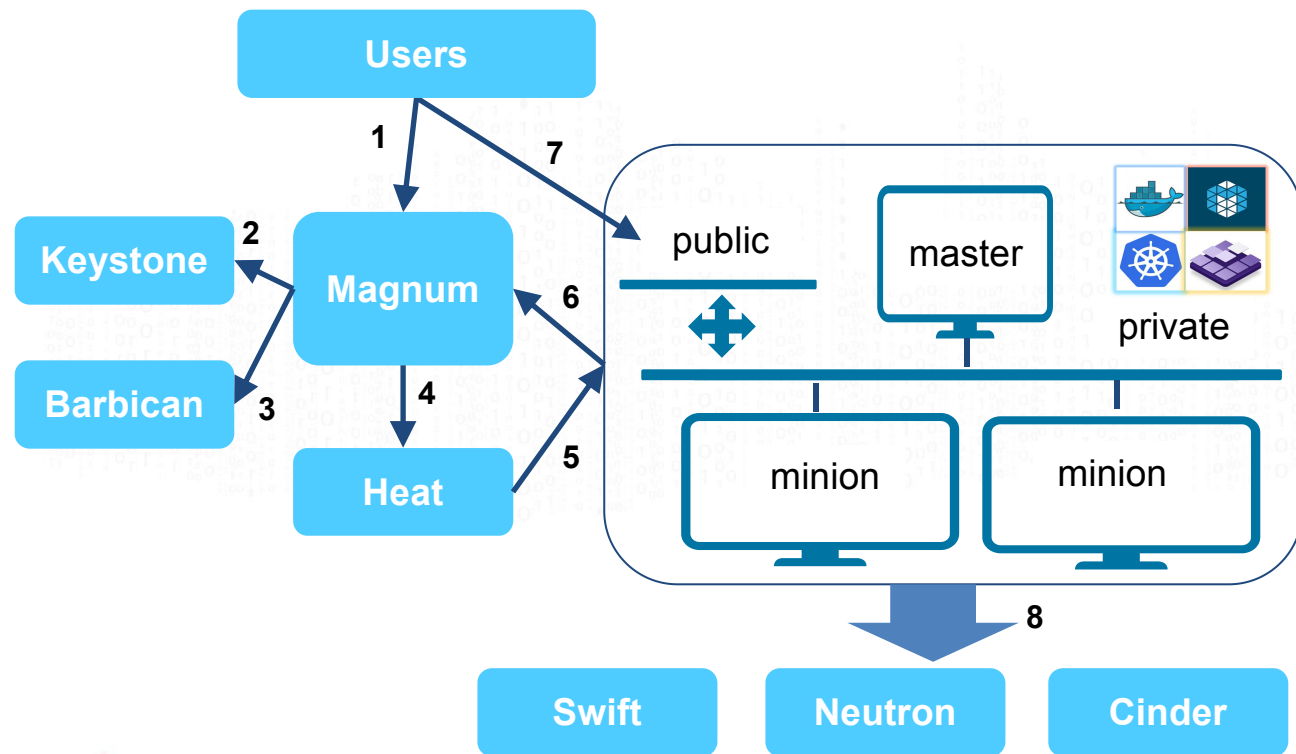
- 虚拟机里跑容器
- 物理机里跑容器
- COE: Swarm kubernete mesos DC/OS



Magnum架构



Magnum 控制流



1. 用户请求创建bay
2. 生成bay专用的账号 (Keystone trust & trustee)
3. 生成根密匙和根证书, 储存在Barbican
4. 用Heat模板创建bay
5. Heat创建stack
6. 各个节点请求Magnum签署证书
7. 用户请求Magnum签署证书, 然后用密匙访问bay的API
8. 根据具体的请求, bay用Keystone trust访问OpenStack其他的服务



Magnum UI

Baymodels - OpenSt... x

localhost/dashboard/project/baymodels

openstack demo admin

Create Baymodel

Info ⚠

Node Spec ⚠

Network ⚠

Labels

Baymodel Name

Name of the baymodel to create.

Container Orchestration Engine *

Choose a Container Orchestration Engine

Public

Enable Registry

Disable TLS

Baymodel Name
An arbitrary human-readable name

Container Orchestration Engine
Specify the Container Orchestration Engine to use.

Public
Make baymodel public. Default: False

Enable Registry
Enable docker registry in the Bay. Default: False

Disable TLS
Disable TLS in the Bay. Default: False

Cancel Next > Create



Murano

目标: 提供OpenStack中的应用目录服务

提供各种应用和服务的发布和生命周期管理，并提供
UI和API.

把任何事情都定义为应用
YAQL语言模板定义应用

Murano特性

应用目录:

- 浏览, 过滤, 依赖

应用目录管理:

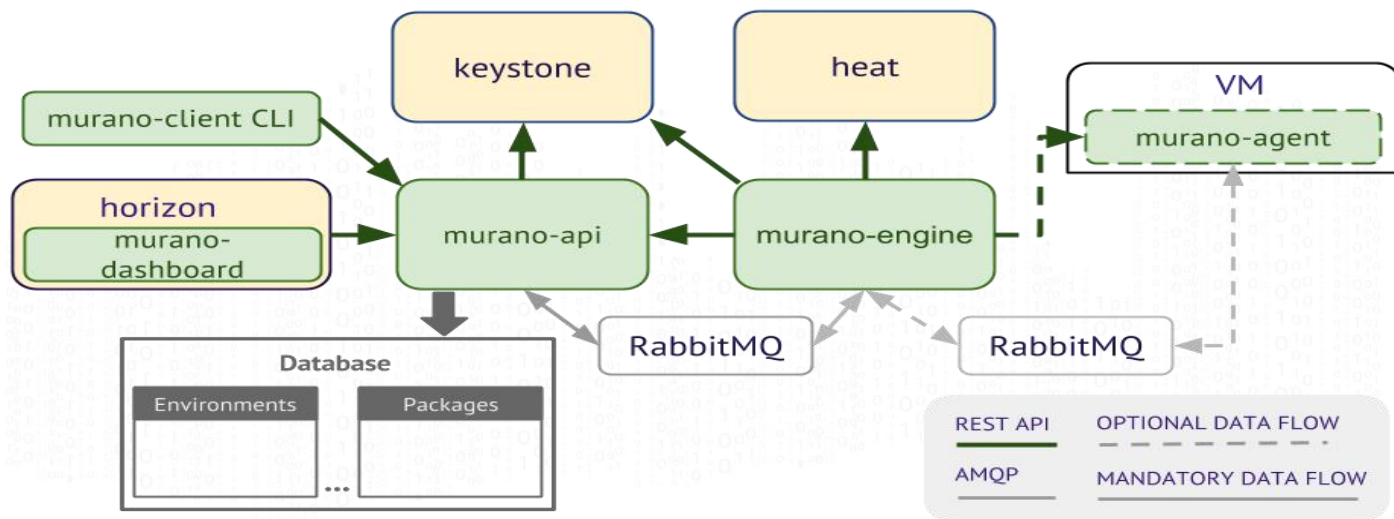
- 从zip/URL/应用仓库加载, 修改更新, 日志跟踪

应用生命周期管理:

- 配置集成, HA和自动扩容, 隔离



Murano架构



Murano UI

The screenshot displays the Murano UI interface. At the top left is the OpenStack logo and the text "openstack". To its right is a "demo" dropdown menu and a user profile icon labeled "admin". A left-hand navigation sidebar contains several menu items: "Project", "Admin", "Identity", "Murano", "Application Catalog", "Environments", "Applications", and "Manage". The main content area is titled "Applications" and "Recent Activity". Below the "Recent Activity" section, there is a light blue banner stating "No recent activity to report at this time." Below this banner are two filter dropdowns: "App Category: All" and "Environment: k8s-cluster". To the right of these filters are search and filter buttons. The main area contains six application cards, each with an icon, a title, a brief description, a "Details" link, and two action buttons: "Add to Env" and "Quick Deploy".

Applications

Recent Activity

No recent activity to report at this time.

App Category: All Environment: k8s-cluster Filter Filter

Apache HTTP Server
The Apache HTTP Server Project is an effort to develop and maintain an open-source HTTP server for modern operating systems inc...
[Details »](#)
Add to Env Quick Deploy

Docker Container
Application that can run any arbitrary docker container to deploy one of the 13,000+ apps available on Docker Hub at https://r...
[Details »](#)
Add to Env Quick Deploy

Docker Redis
Redis is an open-source, networked, in-memory, key-value data store with optional durability. It is written in ANSI C. The deve...
[Details »](#)
Add to Env Quick Deploy

Docker Standalone Host
Standalone docker implementation, employs single VM running docker service. Implements DockerHost interface (along with Kube...
[Details »](#)
Add to Env Quick Deploy

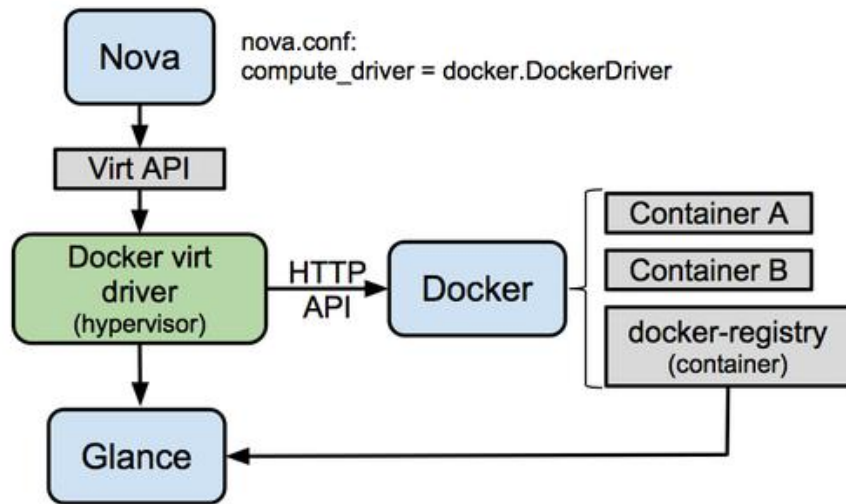
Kubernetes Cluster
Kubernetes is an open source system for managing containerized applications across multiple hosts, providing basic mechanisms f...
[Details »](#)
Add to Env Quick Deploy

Kubernetes Pod
Kubernes Pod - A collection of containers which will be scheduled onto the same node, which share and an IP and port space, an...
[Details »](#)
Add to Env Quick Deploy



Nova-docker/lxc/lxd

- 用户通过Nova的API使用容器
- Nova调用drivers与容器交互
- 容器镜像储存在Glance
- 优点：充分利用Nova的功能
- 缺点：不能使用原生的容器API，因此使用容器时受到限制



Heat-docker

- 提供一个Heat的资源创建Docker容器
- 需要提供Docker daemon的endpoint, 作为参数
- 优点: 实现Docker容器与OpenStack其他资源的混合编排
- 缺点: 只提供一小部分Docker的功能

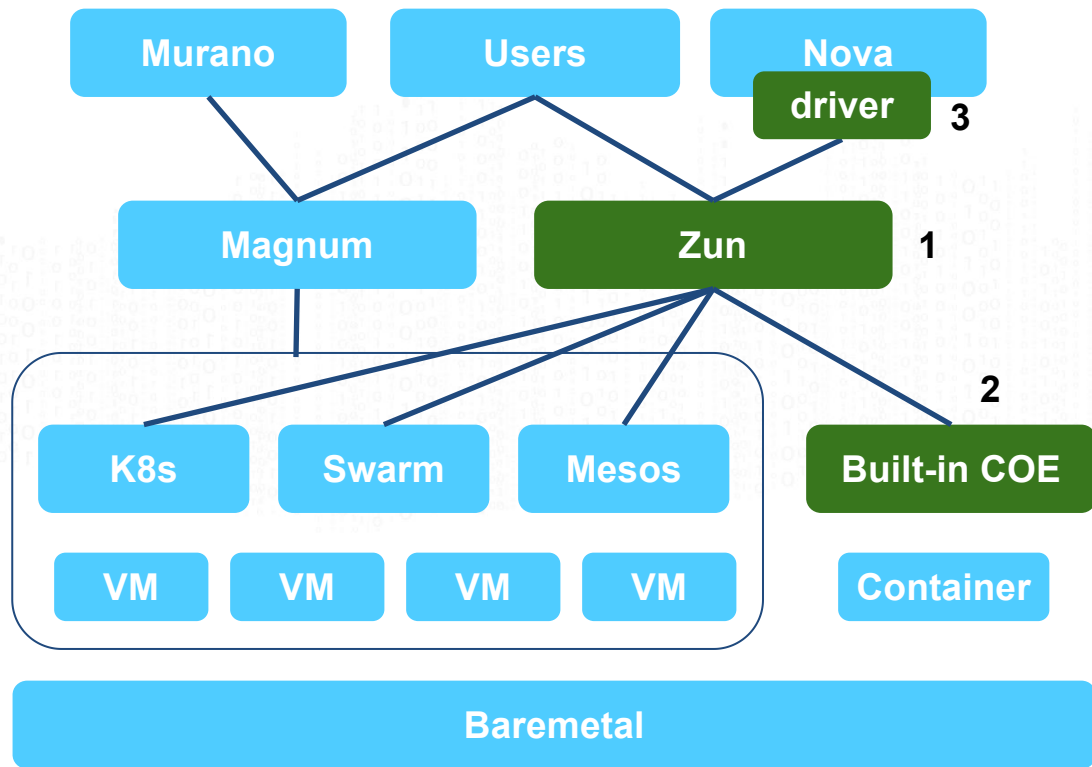
```
heat_template_version: 2013-05-23
description: Test template
resources:
  apache:
    type: DockerInc::Docker::Container
    properties:
      image: marouen/apache
      port_specs:
        - 80
      docker_endpoint: http://host:2375
```



Zun (1)

- Zun (之前叫Higgins), 提供OpenStack的容器服务
- 提供OpenStack原生的API管理容器, 计划支持多种容器技术
 - Container runtimes: Docker, Rkt, Clear Conainer, etc.
 - COEs: Kubernetes, Docker Swarm, etc.

Zun (2)



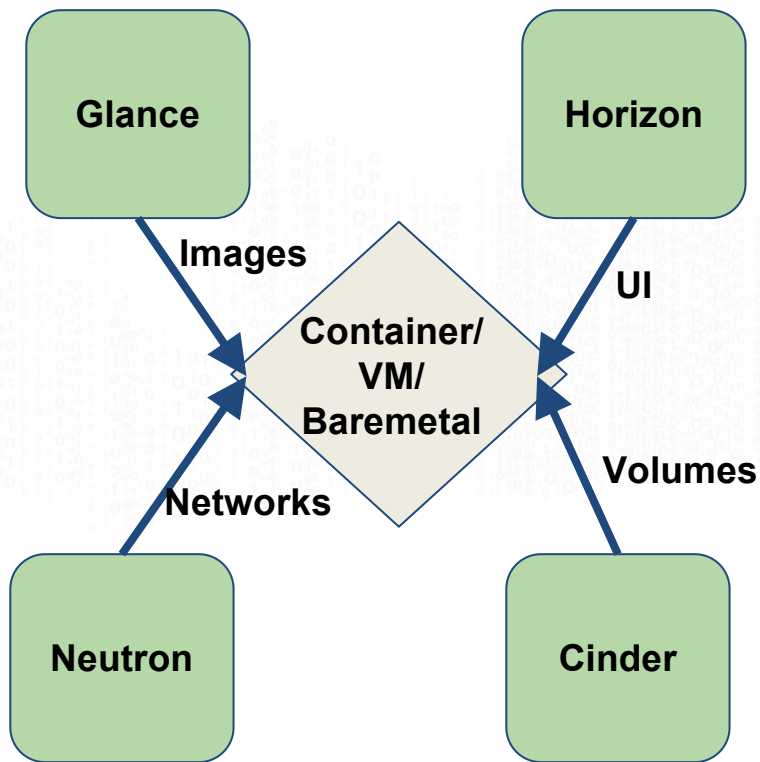
1. 为各种容器编排引擎提供API入口
2. 提供原生的COE作为reference implementation
3. 提供nova的driver，代替nova-docker将容器接入Nova



Zun (3)

在一个的平台上管理各种计算资源：容器，虚拟机，物理机

- 统一的网络：Neutron
- 统一的储存池：Cinder
- 统一的镜像仓库：Glance
- 统一的UI：Horizon
- ...

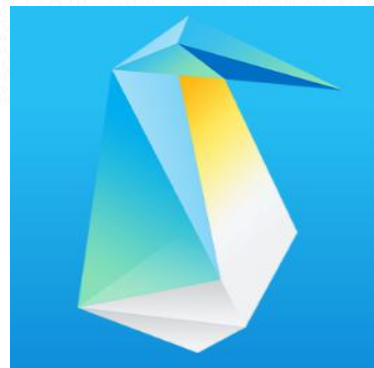


Clear container & Ciao

Clear container

使用了Intel VT特性，基于KVM技术的容器技术。通过优化现有的代码，移除冗余组件，实现在KVM虚拟机里运行容器的技术。

- 启动速度小于150毫秒
- 内存开销大约18M-20M

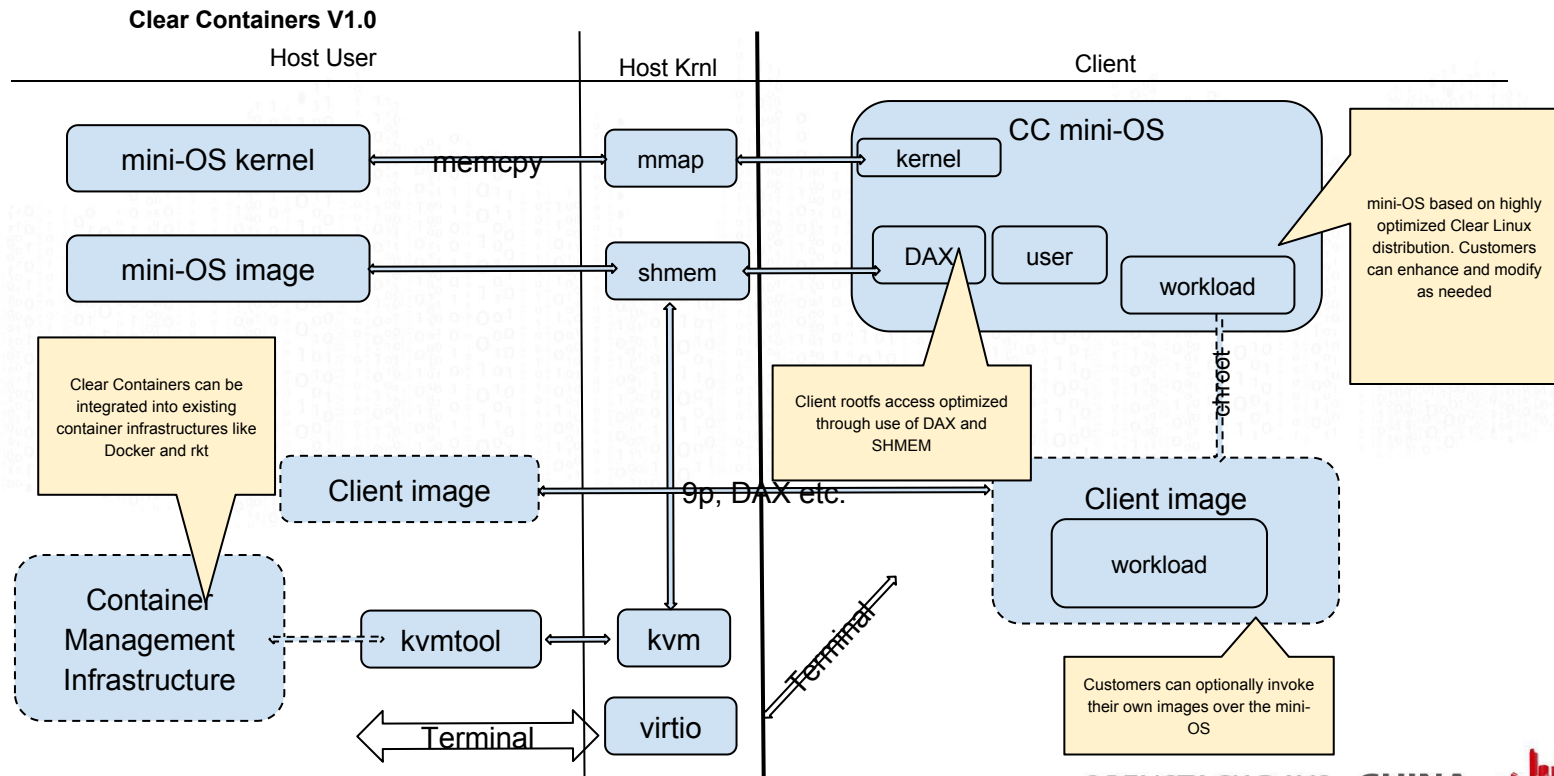


Clear container的关键优化

- 快速的轻量级 hypervisor kvm-tool
- 优化的kernel
- 优化的systemd
- 使用kernel 4.0 的DAX技术实现文件系统到VM的快速访问（0拷贝），降低容器内存使用情况。
- KSM技术允许VM/容器安全地共享内存页面



Clear container 架构 v1.0



CIAO

"Cloud Integrated Advanced Orchestrator".

目标是提供快速，简单，安全，适合大规模部署的云计算管理系统。

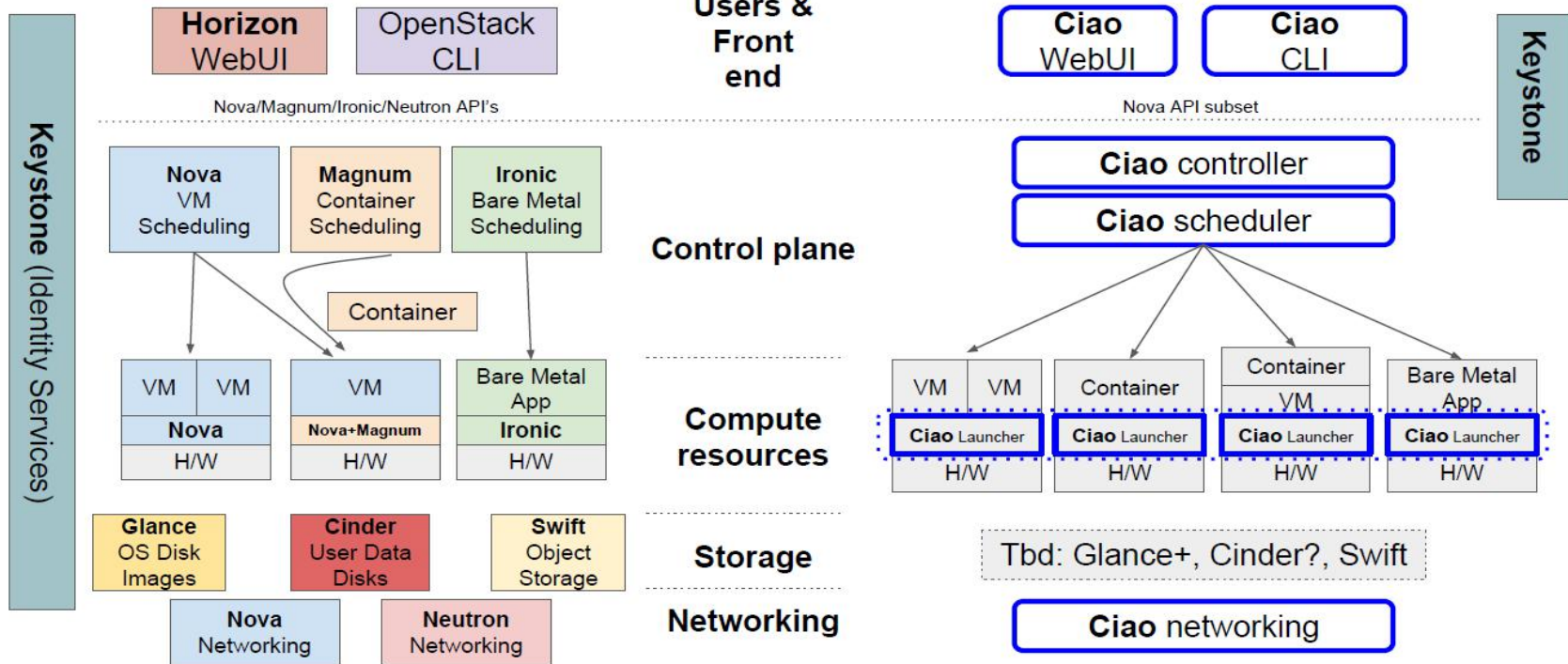
- Go
- SSNTP（简单安全节点传输协议）
- 支持VM，容器，裸机（workload）
- 集成OpenStack的服务



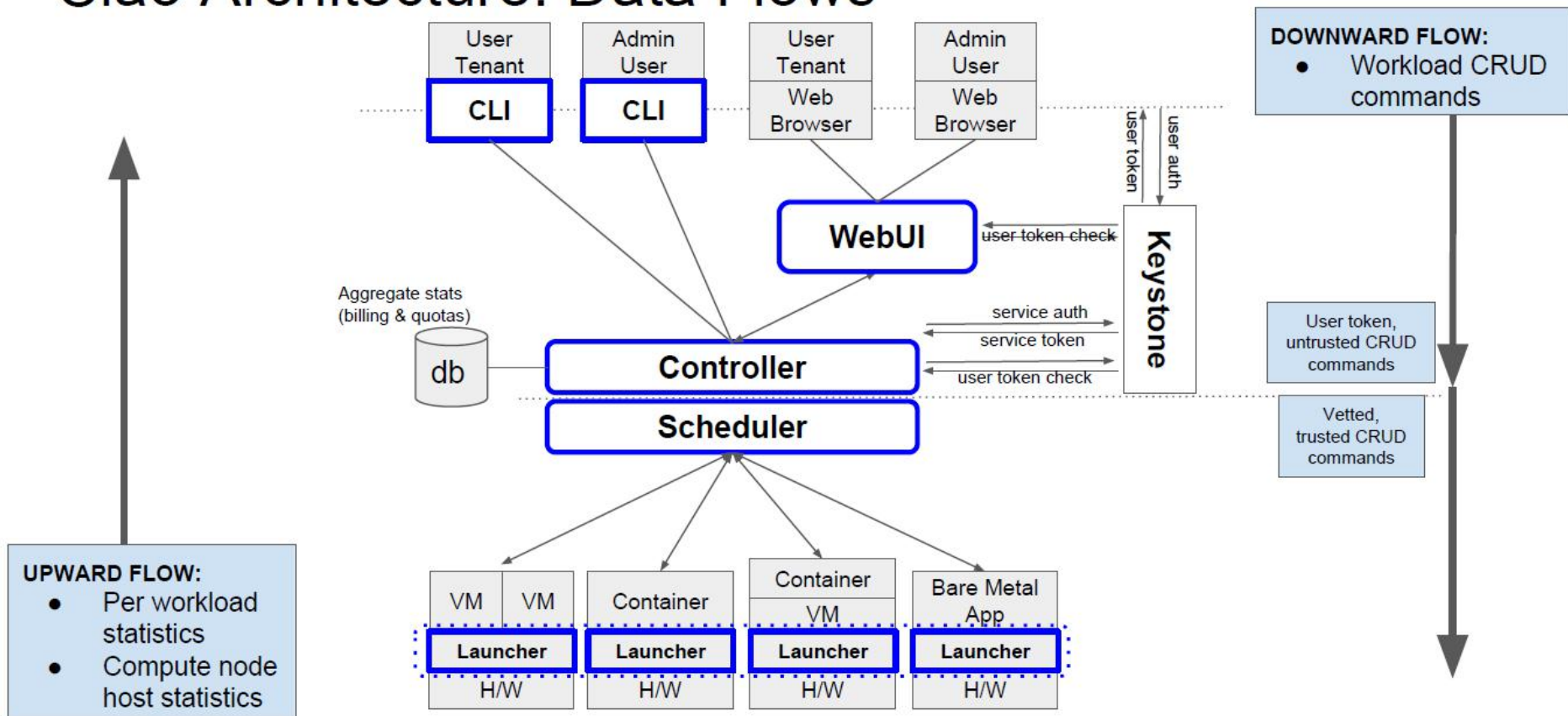
OpenStack VS Ciao

OpenStack

Ciao



Ciao Architecture: Data Flows



容器与OpenStack的结合总结



容器与OpenStack的结合（1）

❖ OpenStack管理的虚拟机/物理及上跑容器(Magnum, Murano, Solum)

- 优点：
 - 资源有效共享统一由OpenStack Nova管理
- 缺点：
 - VM跑容器的性能损失
 - 容器要解决网络的性能
 - 容器的存储



容器与OpenStack的结合（2）

- ❖ 各种COEs管理的主机上用容器跑OpenStack服务（k8s-kolla, mesos-kolla）
 - 优点：
 - 容器性能保障
 - 更高效地利用资源
 - 缺点：
 - 容器隔离性
 - VM的网络，存储
 - OpenStack 服务的复杂性

容器与OpenStack的结合（3）

- ❖ 同时管理容器和虚拟机 (Nova-docker, heat-docker, Nova-lxc/lxd, Nova+Zun)
 - 优点：
 - 简单
 - 整合了现有的资源，如网络，存储
 - 缺点：
 - 不利于计算资源共享

Q&A
Thanks!

