火龙果•整理
uml.org.cn

Docker & Docker Hub

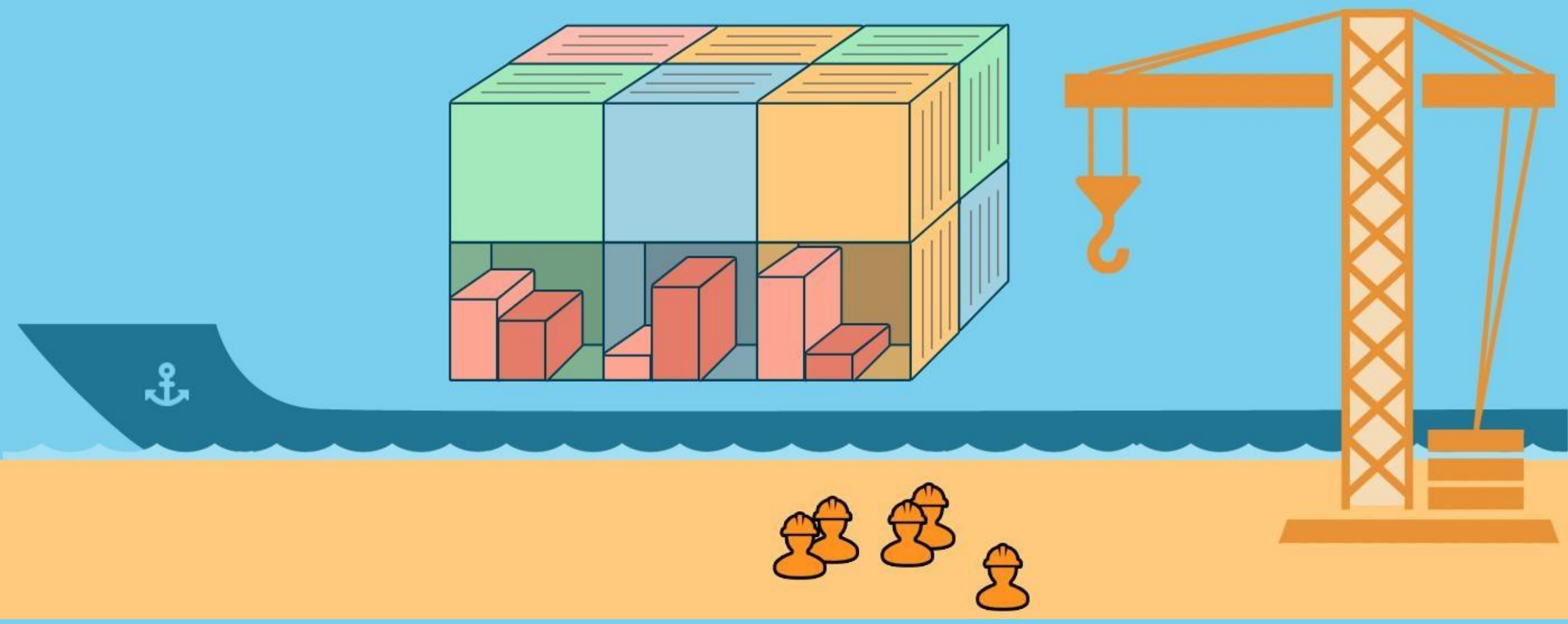
火龙果•整理
uml.org.cn

Agenda

1. Docker Chinese community
2. Docker & Docker Governance
3. Namespaces & Cgoups
4. Storage Drive Kernel -> AUFS
5. Docker Hub & Docker Registry
6. Docker Registry API
7. Community Version Docker Hub
8. Docker Remote API
9. Dockerfile
10. OpenSource Projects Base Docker

火龙果•整理
uml.org.cn

About Docker Community in China
dockboard.org



https://github.com/dockboard

火龙果•整理
uml.org.cn

QINIU
七牛云存储

腾讯云

MT 美团云
Meituan Open Services

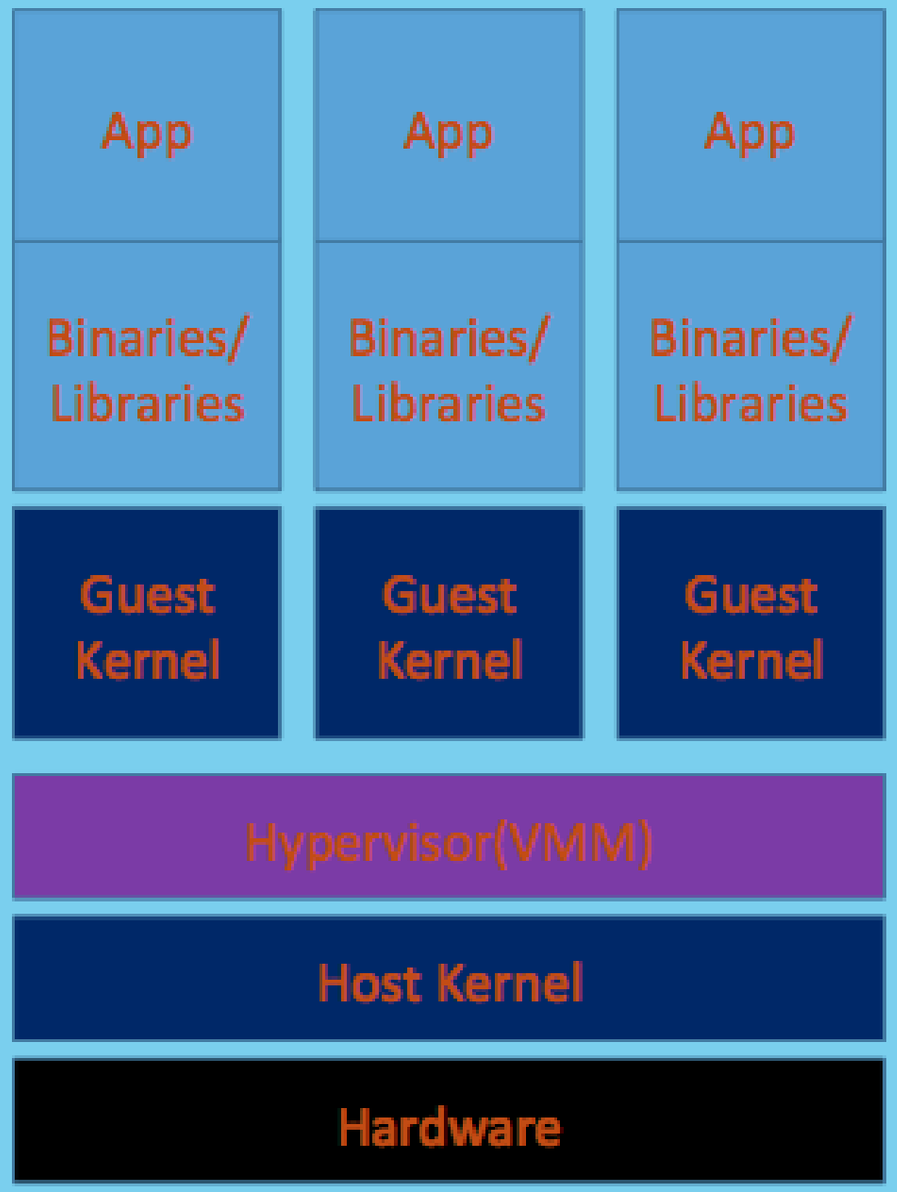
UCLOUD

CODE

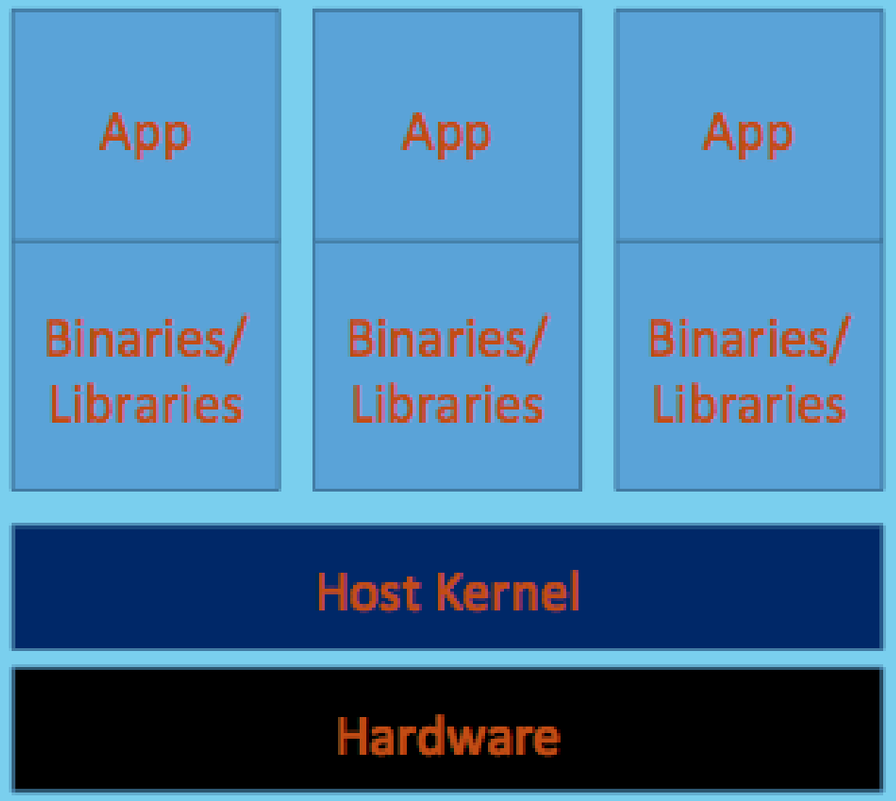
HUAWEI | 华为云服务 Beta

火龙果•整理
uml.org.cn

System Virtualization VS Container Virtualization



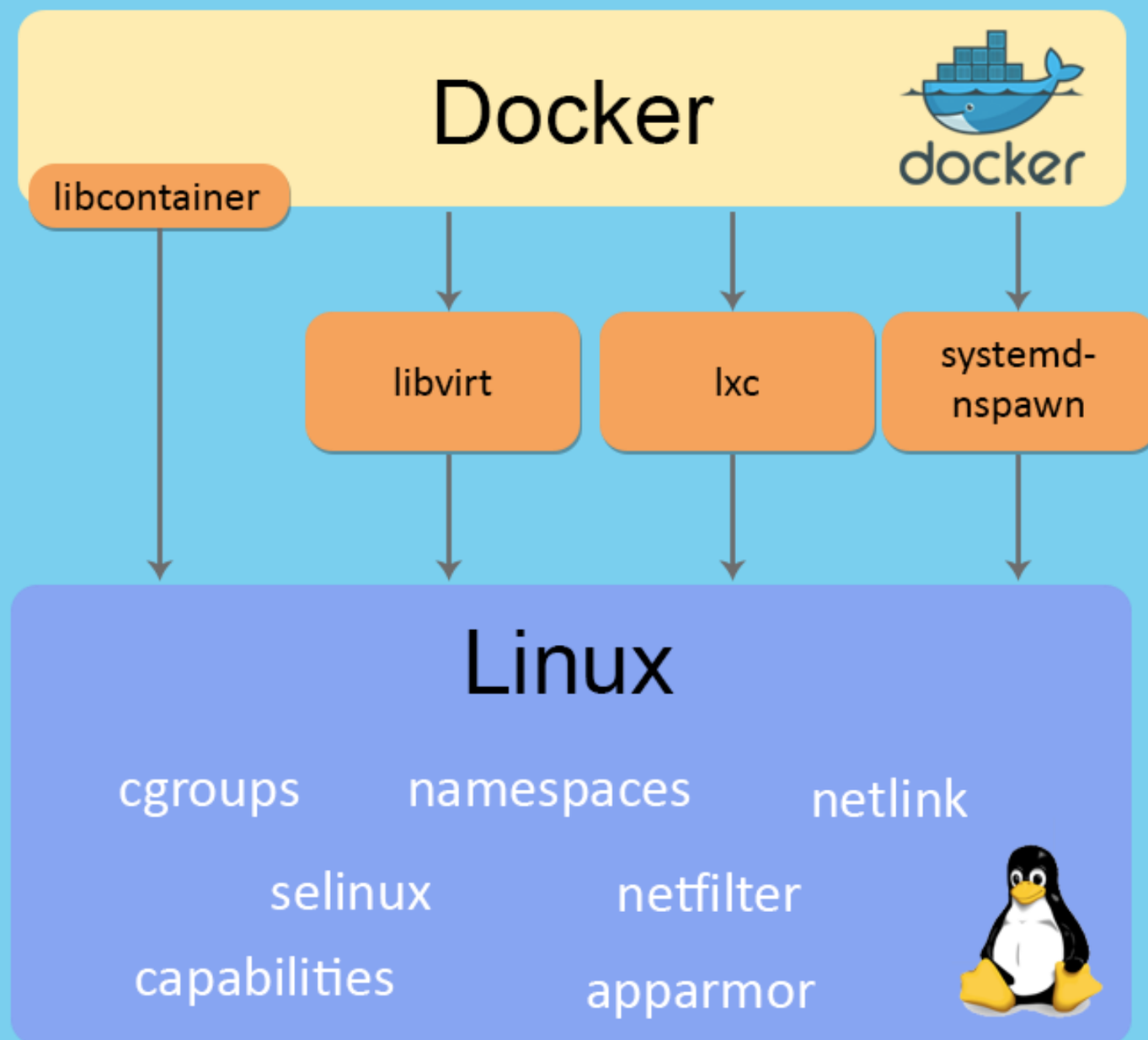
System Virtualization



Container Virtualization

火龙果•整理
uml.org.cn

What's Docker ?



Docker Hub +

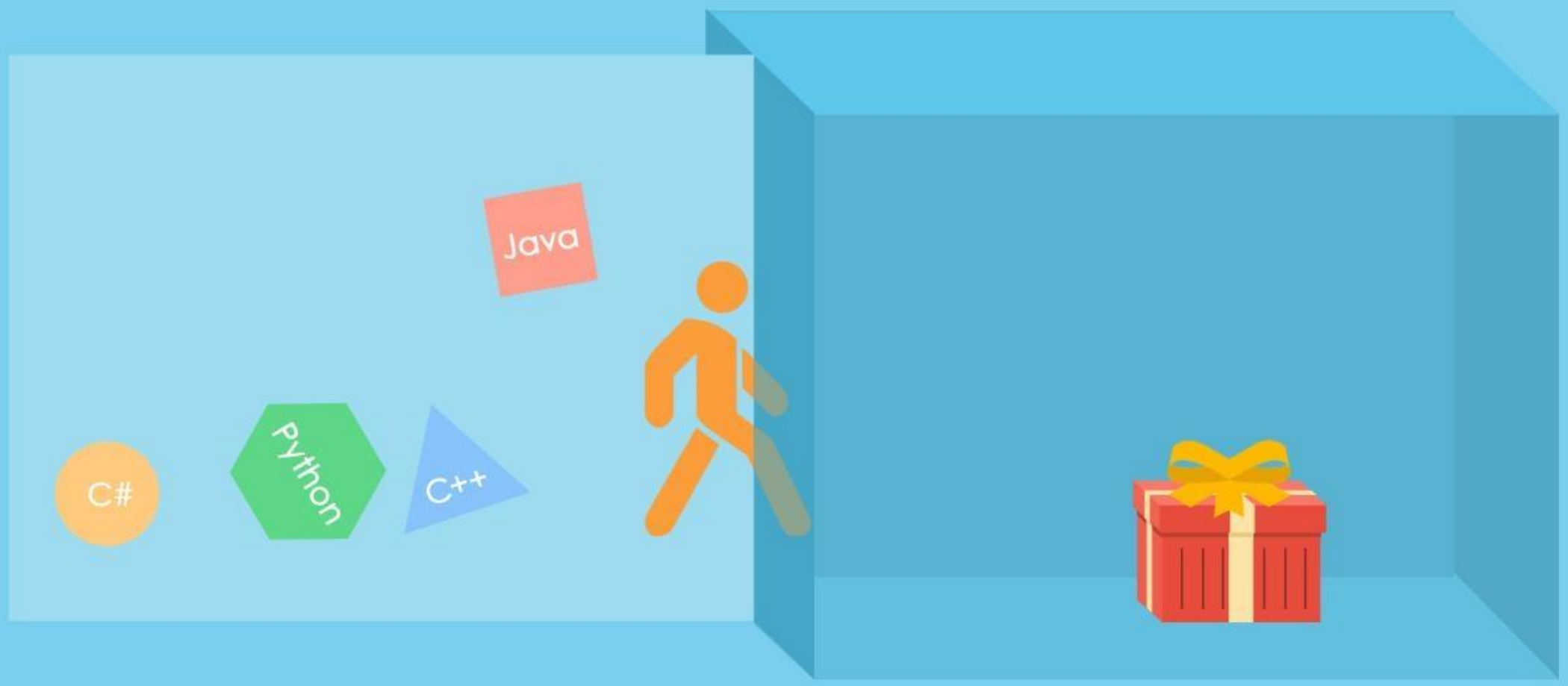
火龙果•整理
uml.org.cn

DGAB -> Docker Governance Advisory Board

1.DGAB or Fond?
2.Seats for commit, no fee or sponsorship.
3.Membership for 12 months, 24 months max.
4.DGAB is not charge.

火龙果•整理
uml.org.cn

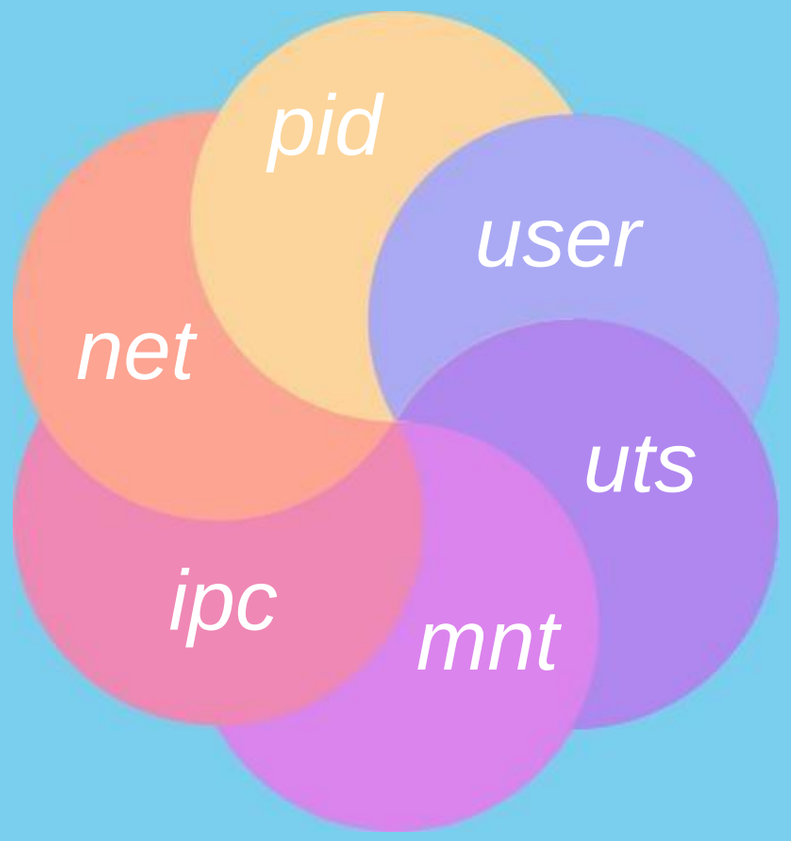
How to use Docker?



火龙果•整理
uml.org.cn

What’s namespaces?

1. pid namespace
2. net namespace
3. mnt namespace
4. uts namespace
5. ipc namespace



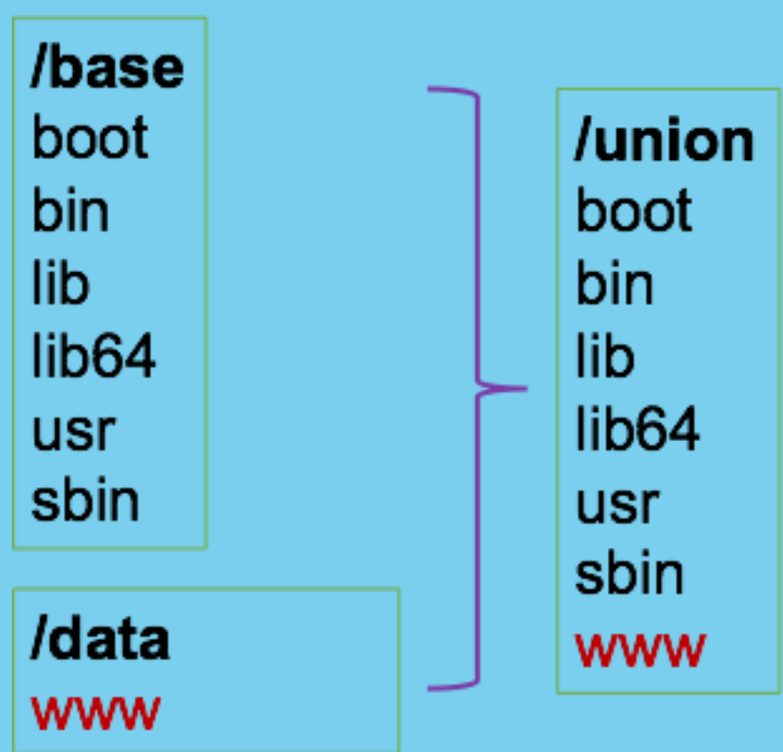
火龙果•整理
uml.org.cn

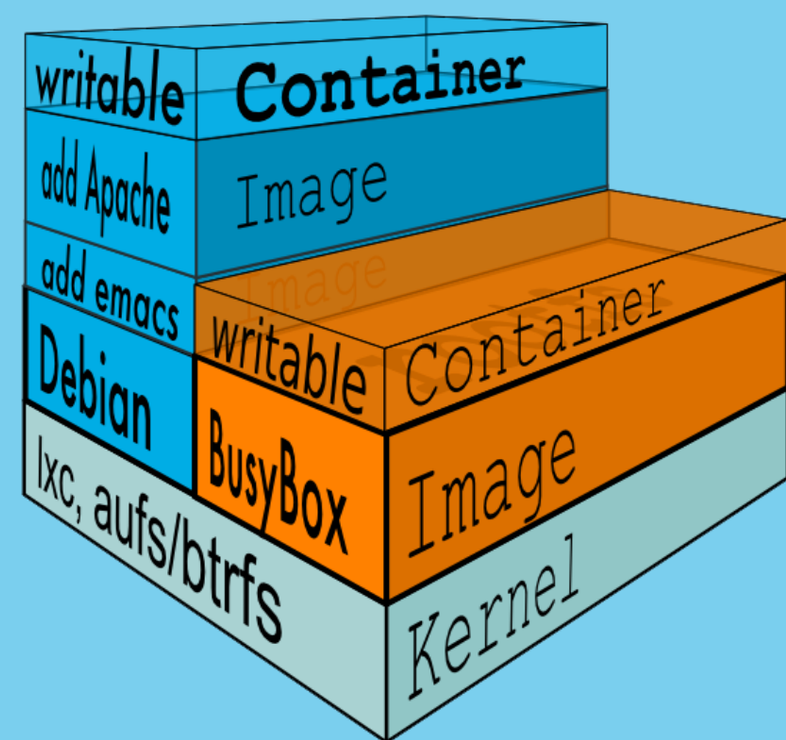
What’s cgroups

1. CPU
2. Memory
3. Blkio
4. Net_cls
5. Net_prio
6. Device

火龙果•整理
uml.org.cn

What's AUFS?





mount -t aufs -o br=/base=ro:/data=rw none /union

火龙果•整理
uml.org.cn

Docker Local Storage Path -> /var/lib/docker

diff
842b5a724d2d
f106b5d7508a
1e8abad02296
511136ea3c5a
...

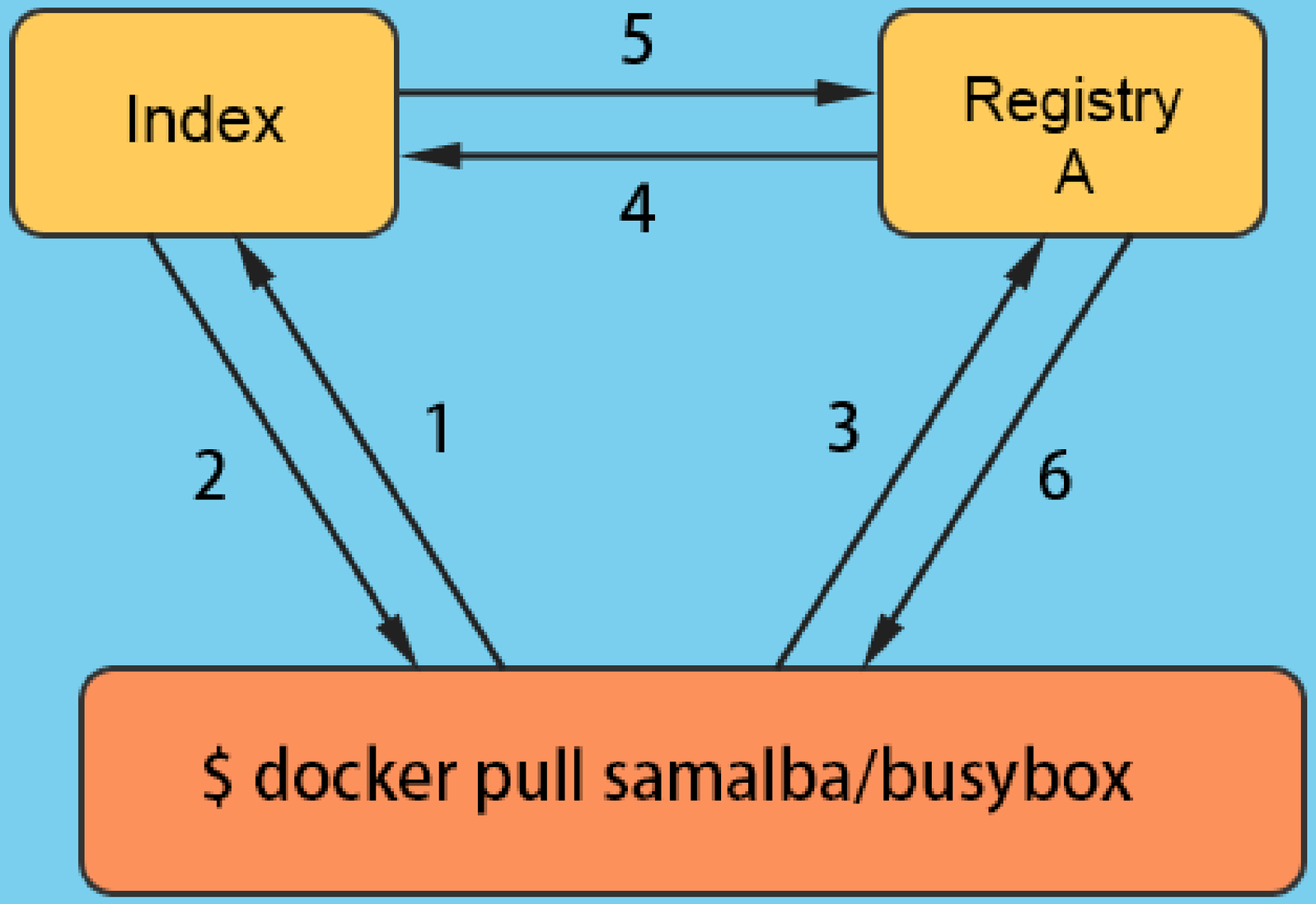
layer
842b5a724d2d
f106b5d7508a
1e8abad02296
511136ea3c5a
...

mnt
842b5a724d2d
f106b5d7508a
1e8abad02296
511136ea3c5a
...

mount -t aufs –o br=
diff/511136ea3c5a=ro:
diff/1e8abad02296=ro:
diff/f106b5d7508a=ro:
diff/842b5a724d2d=rw
none
mnt/842b5a724d2d

火龙果•整理
uml.org.cn

Docker Hub & Registry Workflow



火龙果•整理
uml.org.cn

How To Deploy Docker Registry?

https://github.com/dotcloud/docker-registry

docker run -p 5000 samalba/docker-registry

火龙果•整理
uml.org.cn

Docker Registry API -> Push

1. docker 向 HUB 服务器注册 repository：PUT /v1/repositories/<username>/<repository>
2. 参数是 JSON 格式的 <repository> 所有 image 的 id 列表，按照 image 的构建顺序排列。
3. 根据 <repository> 的 <tags> 进行循环：
   3.1 获取 <image> 的 JSON 文件：GET /v1/images/<image_id>/json
   3.2 如果没有此文件或内容返回 404 。
   3.3 docker push 认为服务器没有 image 对应的文件，向服务器上传 image 相关文件。
      3.3.1 写入 <image> 的 JSON 文件：PUT /v1/images/<image_id>/json
      3.3.2 写入 <image> 的 layer 文件：PUT /v1/images/<image_id>/layer
      3.3.3 写入 <image> 的 checksum 信息：PUT /v1/images/<image_id>/checksum
   3.4 上传完此 tag 的所有 image 后，向服务器写入 tag 信息：
      PUT /v1/repositories/(namespace)/(repository)/tags/(tag)
4. 所有 tags 的 image 上传完成后，向服务器发送所有 images 的校验信息：
   PUT /v1/repositories/(namespace)/(repo_name)/images

火龙果•整理
uml.org.cn

Docker Registry API -> Pull

1. docker 访问 Hub 服务器 repository 的 images 信息：
   GET /v1/repositories/<username>/<repository>/images
2. docker 访问 registry 服务器 repository 的 tags 信息：
   GET /v1/repositoies/<username>/<repository>/tags
3. 根据 <repository> 的 <tags> 中 image 信息进行循环：
   3.1 获取 <image> 的 Ancestry 信息：GET /v1/images/<image_id>/ancestry
   3.2 获取 <image> 的 JSON 数据：GET /v1/images/<image_id>/json
   3.3 获取 <image> 的 Layer 文件：GET /v1/images/<image_id/layer

火龙果•整理
uml.org.cn

Golang Version Docker Registry

```
beego.Router("/_ping", &controllers.PingController{}, "get:GetPing")

//Static File
beego.Router("/favicon.ico", &controllers.StaticController{}, "get:GetFavicon")

ns := beego.NewNamespace("/v1",

        beego.NSRouter("/_ping", &controllers.PingController{}, "get:GetPing"),

        beego.NSRouter("/_status", &controllers.StatusController{}),

        beego.NSRouter("/users", &controllers.UsersController{}, "get:GetUsers"),
        beego.NSRouter("/users", &controllers.UsersController{}, "post:PostUsers"),

        beego.NSNamespace("/repositories",
                beego.NSRouter("/:namespace/:repo_name/tags/:tag", &controllers.RepositoryController{}, "put:PutTag"),
                beego.NSRouter("/:namespace/:repo_name/images", &controllers.RepositoryController{}, "put:PutRepositoryImages"),
                beego.NSRouter("/:namespace/:repo_name/images", &controllers.RepositoryController{}, "get:GetRepositoryImages"),
                beego.NSRouter("/:namespace/:repo_name/tags", &controllers.RepositoryController{}, "get:GetRepositoryTags"),
                beego.NSRouter("/:namespace/:repo_name", &controllers.RepositoryController{}, "put:PutRepository"),
        ),

        beego.NSNamespace("/images",
                beego.NSRouter("/:image_id/ancestry", &controllers.ImageController{}, "get:GetImageAncestry"),
                beego.NSRouter("/:image_id/json", &controllers.ImageController{}, "get:GetImageJSON"),
                beego.NSRouter("/:image_id/layer", &controllers.ImageController{}, "get:GetImageLayer"),
                beego.NSRouter("/:image_id/json", &controllers.ImageController{}, "put:PutImageJson"),
                beego.NSRouter("/:image_id/layer", &controllers.ImageController{}, "put:PutImageLayer"),
                beego.NSRouter("/:image_id/checksum", &controllers.ImageController{}, "put:PutChecksum"),
        ),
)
```

火龙果•整理
uml.org.cn

Docker Remote API -> Image & Container

1. POST /images/create
2. GET /images/(name)/history
3. POST /images/(name)/insert
4. GET /images/(name)/json
5. GET /images/json
6. POST /images/(name)/push
7. DELETE /images/(name)
8. GET /images/search
9. POST /images/(name)/tag

1. POST /containers/(id)/attach
2. GET /containers/(id)/changes
3. POST /containers/(id)/copy
4. POST /containers/create
5. GET /containers/(id)/export
6. GET /containers/(id)/json
7. POST /containers/(id)/kill
8. GET /containers/json
9. GET /containers/(id)/logs
10. POST /containers/(id)/pause
11. DELETE /containers/(id)
12. POST /containers/(id)/restart
13. POST /containers/(id)/start
14. POST /containers/(id)/stop
15. GET /containers/(id)/top
16. POST /containers/(id)/unpause
17. POST /containers/(id)/wait

火龙果•整理
uml.org.cn

What’s Dockerfile ?

```
# VERSION 0.1
# DOCKER-VERSION  0.7.3
# AUTHOR:         Sam Alba <sam@docker.com>
# DESCRIPTION:    Image with docker-registry project and dependecies
# TO_BUILD:       docker build -rm -t registry .
# TO_RUN:         docker run -p 5000:5000 registry

# Latest Ubuntu LTS
from    ubuntu:14.04
# Update
run apt-get update
run apt-get -y upgrade
# Install pip
run apt-get -y install python-pip
# Install deps for backports.lzma (python2 requires it)
run apt-get -y install python-dev liblzma-dev libevent1-dev
add . /docker-registry
add ./config/boto.cfg /etc/boto.cfg
# Install core
run pip install /docker-registry/depends/docker-registry-core
# Install registry
run pip install file:///docker-registry#egg=docker-registry[bugsnag]
env DOCKER_REGISTRY_CONFIG /docker-registry/config/config_sample.yml
env SETTINGS_FLAVOR dev
expose 5000
cmd exec docker-registry
```

火龙果•整理
uml.org.cn

OpenSource Projects Base Docker

1.PaaS : Tsuru/Octohost/Deis/StackDock/Flynn/tusum/Dokku/Spin-Docker/Peas/Tutum
2.Private Registry: quay.io/orchard/docker.cn
3.OS: CoreOS/Atomic
4.CI: Drone/Shippable
5.Book: dockerbook.com
6.Kubernetes

火龙果•整理
uml.org.cn

End & Thanks