



云端基于Docker微服务应用的 架构实践



闫长海 阿里云技术专家

自我介绍

- 闫长海 花名：易嘉
- 阿里云计算专家，负责容器服务研发
- 曾任职IBM公有云Bluemix研发
- E-mail: changhai.ych@alibaba-inc.com
- 微信：chaghaiych



大纲

- 什么是微服务
- 容器管理微服务的挑战
- 容器微服务管理
- 微服务的持续交付

加速应用交付，缩短梦想距离

传统IT系统

今天



大兵团作战

战术小队

Dev vs. Ops

DevOps

缓慢变化

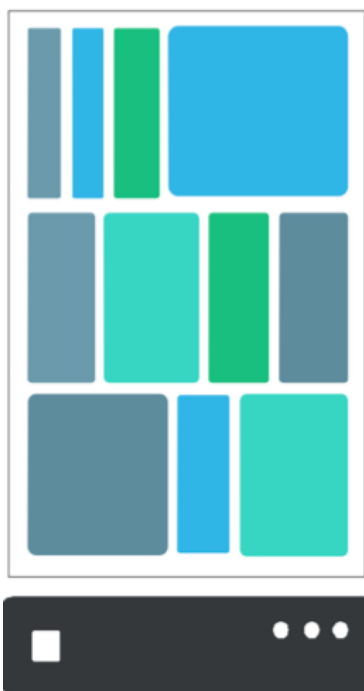
快速迭代

整体式应用
(monolithic app)

微服务

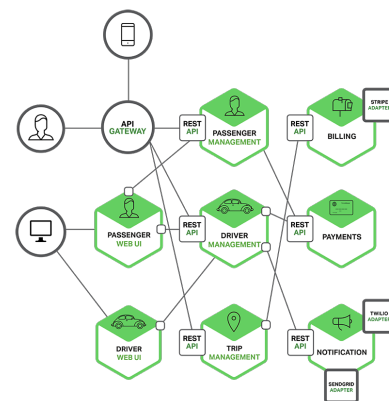
物理服务器、
存储、网络

云计算



微服务

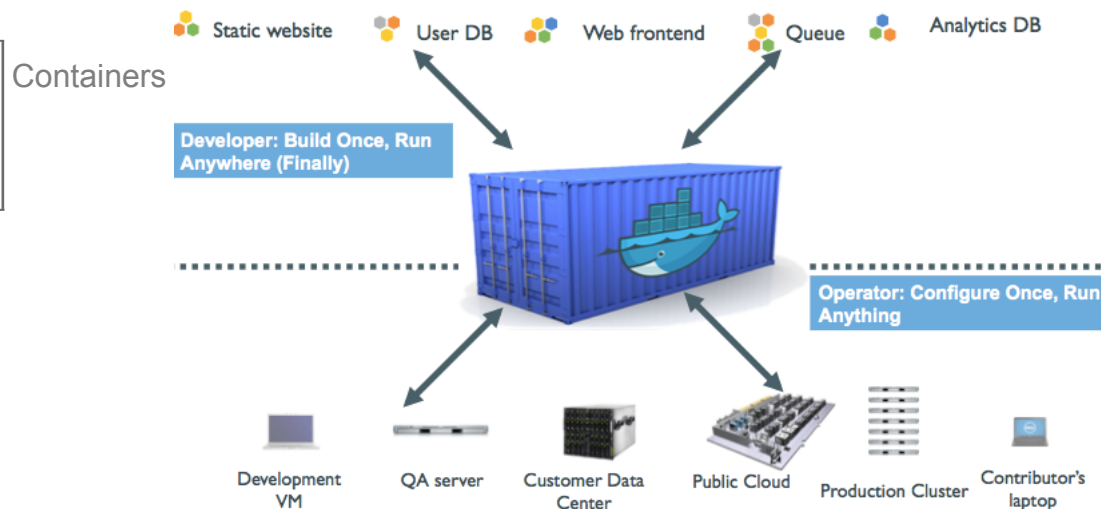
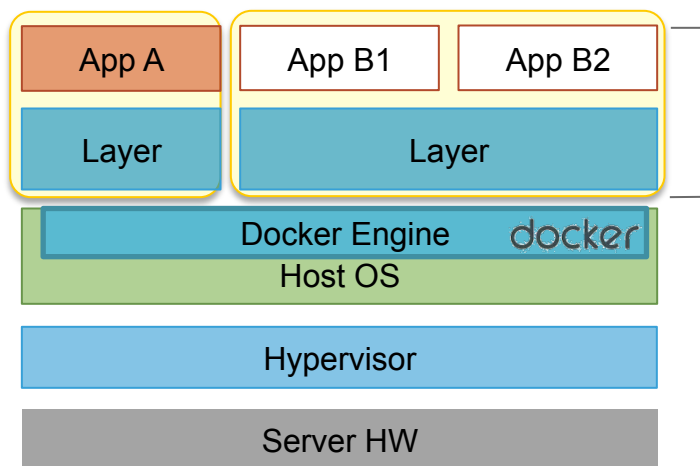
- 小 按照业务单一职责设计
- 独 立进程，独立部署
- 轻 量级的通信机制
- 松 耦合



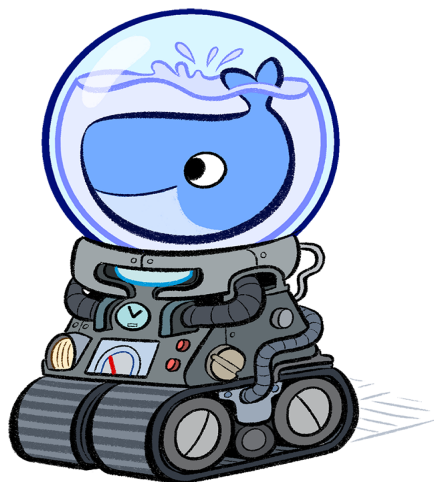
用Docker部署微服务

- Docker是一种轻量级的操作系统虚拟化方案
 - 细粒度资源隔离 - 敏捷，降低企业成本
 - 定义了环境无关的标准的交付、部署规范 - 良好可移植性，提高运维效率
- 当想把Docker容器当做轻量级虚拟机时，请三思！
 - 正确姿势
 - 每次代码提交重新构建镜像
 - 禁止修改运行中的镜像
 - 利用volume保存持久化数据

Docker容器方式



云端漫步：开始 Docker 之旅



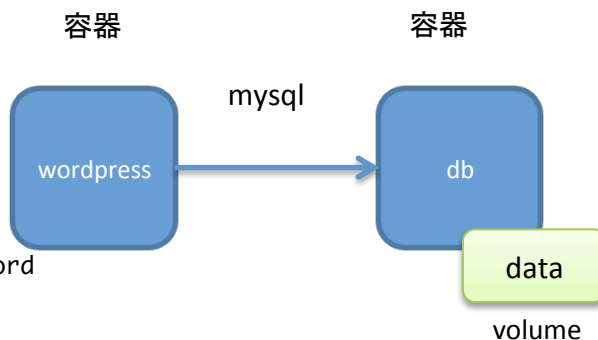
Docker Machine

- 配置安装
 - 安装[Docker Toolkit](#)
 - 安装云驱动
 - [ECS driver for Docker Machine](#)
 - AWS, GCE, [等等](#).
- 在阿里云创建Docker运行环境

```
export ECS_ACCESS_KEY_ID=xxxxxx
export ECS_ACCESS_KEY_SECRET=xxxxxx
docker-machine create --driver aliyunecs mytest
eval "$(docker-machine env mytest)"
docker run -d nginx
```

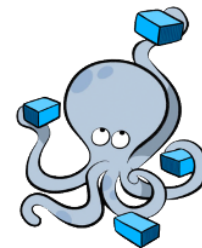
容器编排 - Docker Compose

```
version: '2'
services:
  wordpress:
    image: wordpress:4
    ports:
      - 80
    restart: always
    links:
      - db:mysql
    network_mode: bridge
  db:
    image: mysql:5.7
    environment:
      MYSQL_ROOT_PASSWORD: password
    restart: always
    volumes:
      - data:/var/log/mysql
    network_mode: bridge
volumes:
  data:
    driver: local
```



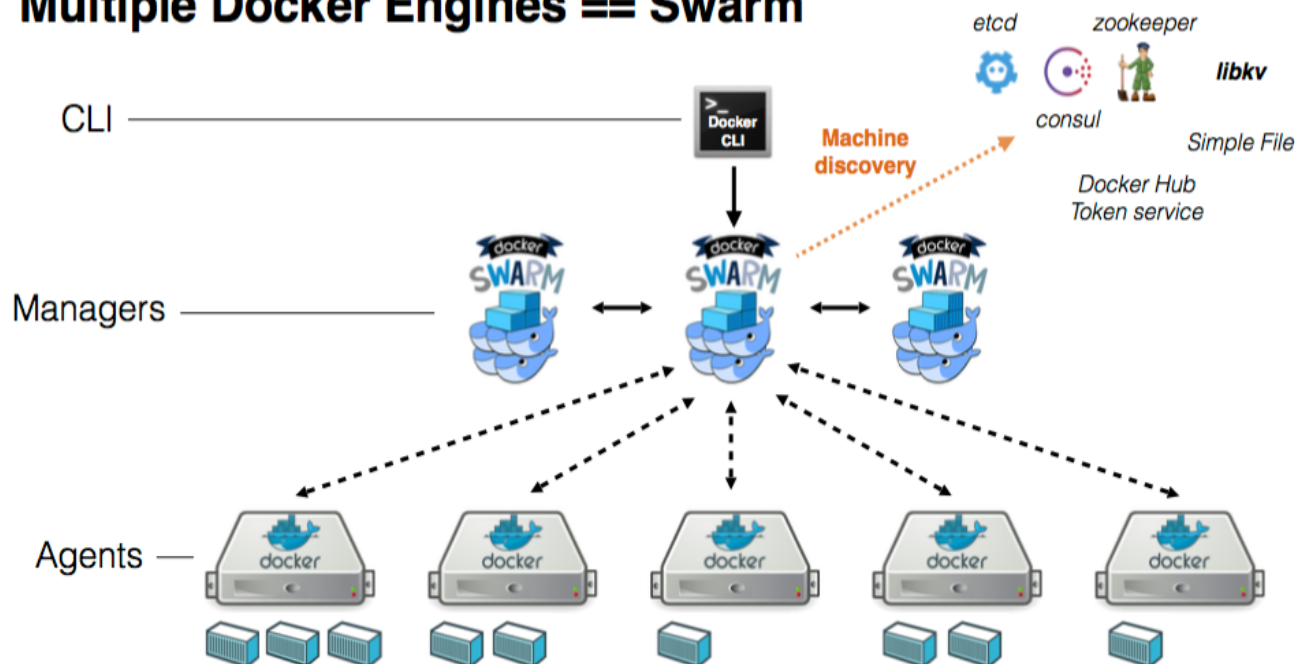
一键部署: `docker-compose up`
手动伸缩: `docker-compose scale wordpress=3`

- 优点
 - 简单好用, 便于开发
 - 编排容器、存储和网络
- 不足
 - 面向开发和部署, 不支持自动化运维



容器集群管理 - Docker Swarm

Multiple Docker Engines == Swarm



优点

- 支持标准的 Docker API
- 灵活、可插拔的容器调度

不足

- 面向容器、缺少微服务支持

容器即服务 Container as a Service



等

Docker Cloud
(tutum.co)

Amazon EC2
Container Service

Google
Container Engine

阿里云
容器服务

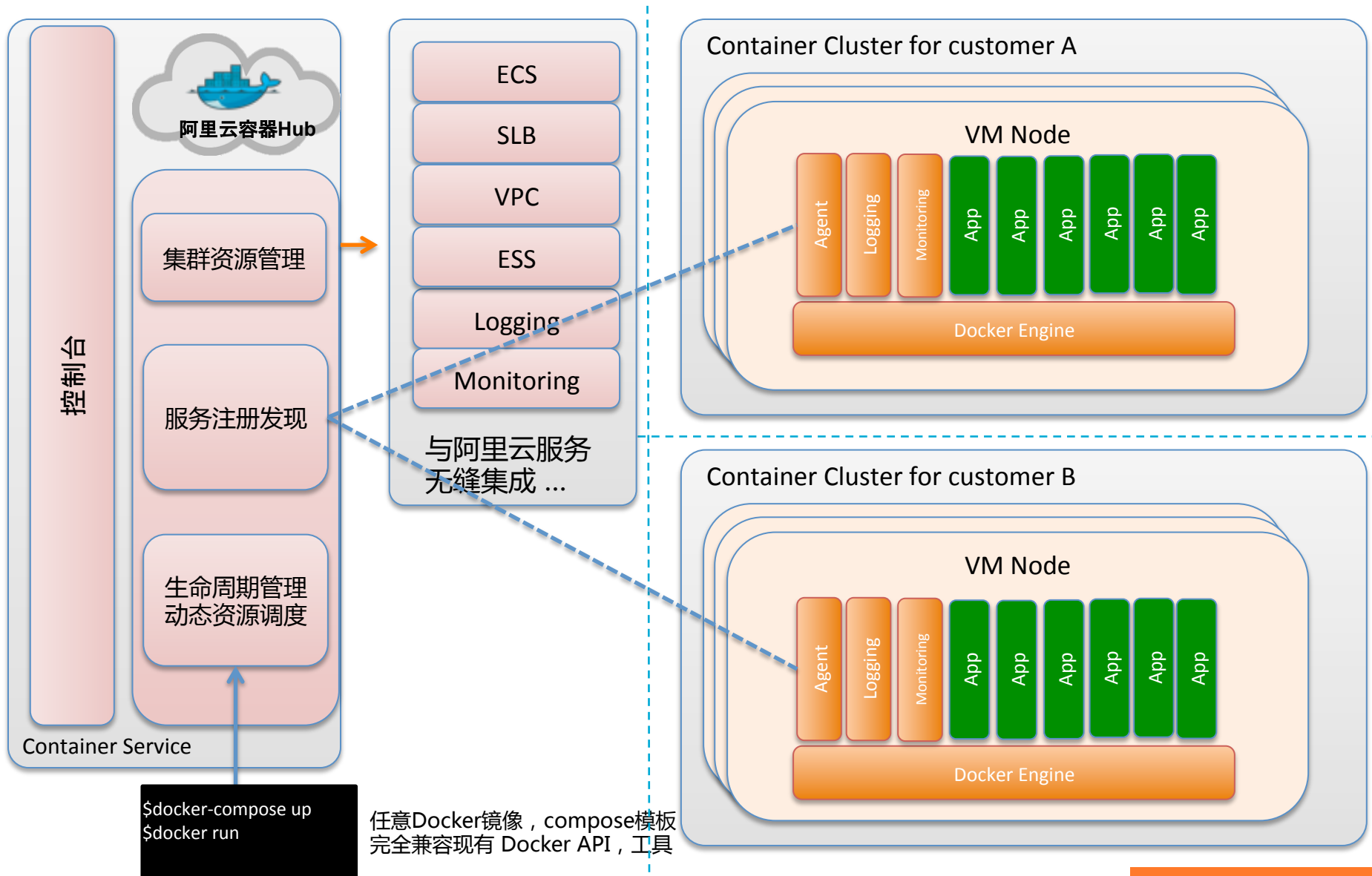
Docker Swarm API
Compose template

ECS API
Compose template/
Task definition

Kubernetes API
Pod/Service

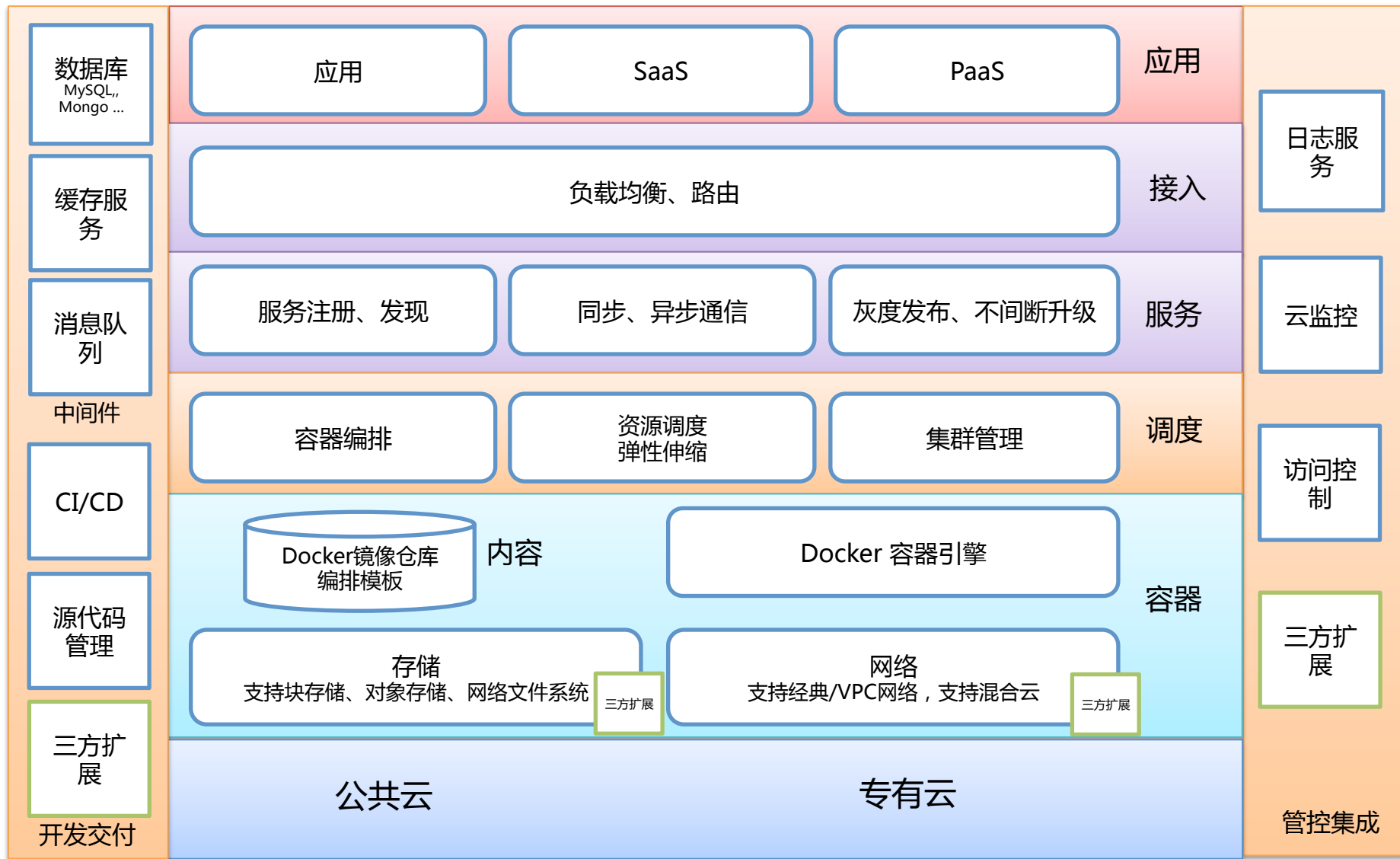
Docker Swarm API
Compose template

阿里云容器服务



Dedicated and isolated customer domains

阿里云容器服务



服务编排部署 – 增强生命周期管理

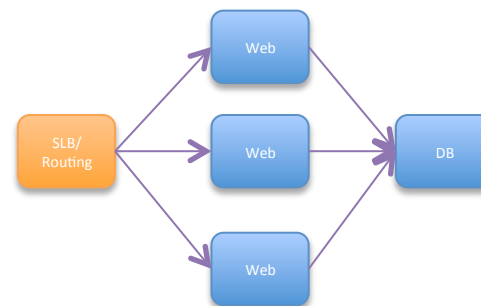
- 健康监测
- 路由注册
- 容器监控
- 日志收集
- 集群部署
- 云服务引用 (RDS, KVStore ...)

Wordpress

```
web:
  image: 10.32.161.160:5000/wordpress:4
  ports:
    - "80"
  environment:
    - WORDPRESS_AUTH_KEY=changeme
  restart: always
  links:
    - db:mysql
  labels:
    aliyun.logs: /var/log
    aliyun.probe.url: http://container/test
    aliyun.probe.initial_delay_seconds: 10
    aliyun.routing.port_80: http://wordpress
    aliyun.scale: 3

db:
  image: 10.32.161.160:5000/mysql
  environment:
    MYSQL_ROOT_PASSWORD: password
  restart: always
  labels:
    aliyun.logs: /var/log/mysql
```

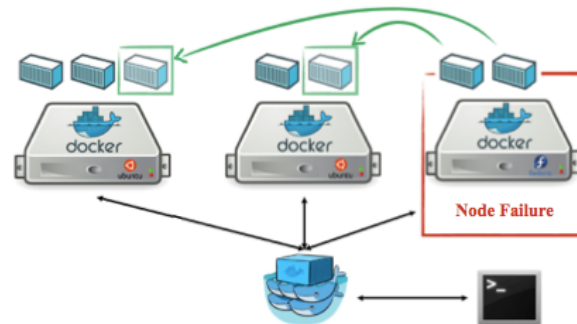
日志收集
健康监测
路由注册
集群部署
自定义监控



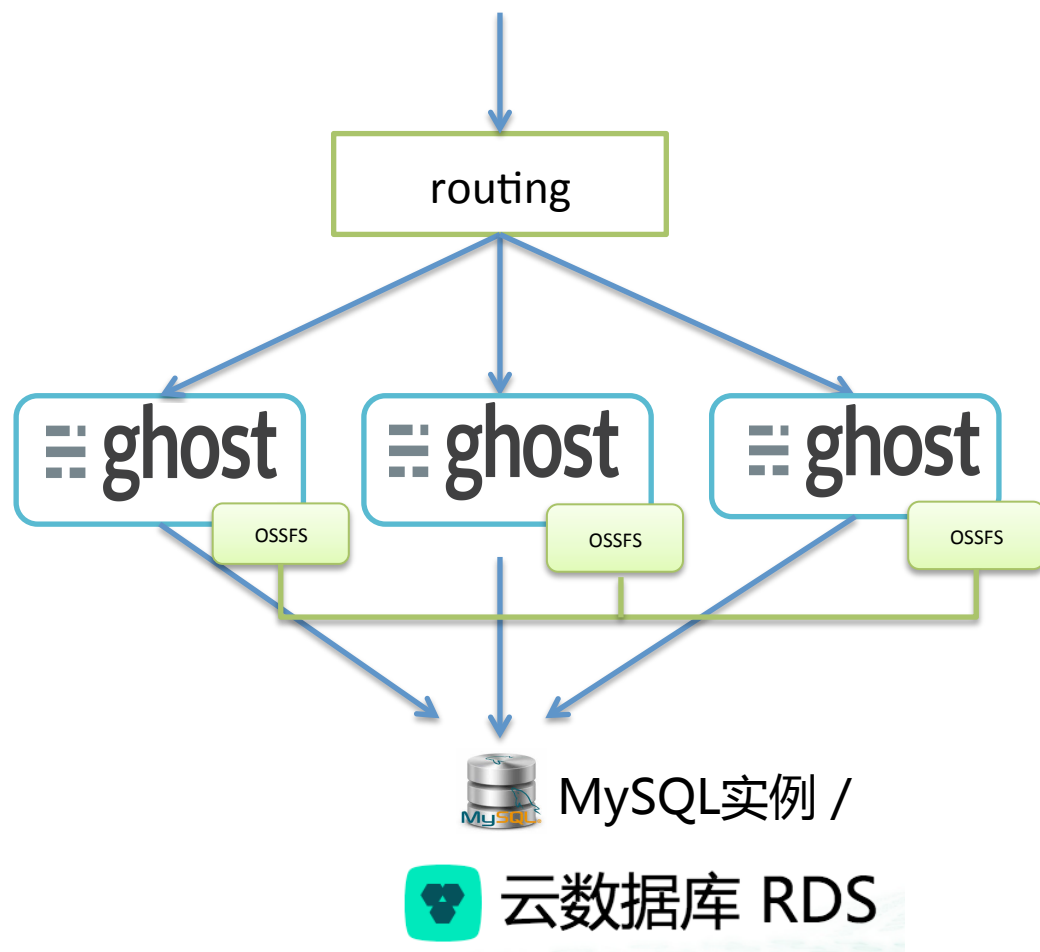
服务调度扩展

- 服务密切关系
 - E.g. MySQL主从部署 (anti-affinity)
- 全局部署
 - 每个节点一个实例
 - E.g. 安装监控日志Agent到每个节点
- 细粒度资源限制
 - CPU share 等
- 容器重新调度
 - reschedule:on-node-failure
 - 网络, 存储
- 跨可用区高可用支持
 - availability:az==2

```
master:
  image: tutum/mysql:5.6
  environment:
    - MYSQL_USER=user
    - MYSQL_PASS=test
    - REPLICATION_MASTER=true
    - REPLICATION_USER=rep1
    - REPLICATION_PASS=rep1
  ports:
    - "3306"
slave:
  image: tutum/mysql:5.6
  environment:
    - MYSQL_USER=user
    - MYSQL_PASS=test
    - REPLICATION_SLAVE=true
    - affinity:service!=master
  ports:
    - "3306"
  links:
    - master:mysql
```



存储管理



- 利用Docker Volume Plugin支持不同的存储类型
 - 块存储 云盘
 - 对象存储 OSSFS
 - 网络文件系统 NFS
- 支持第三方存储插件

ghost:
image: registry.aliyuncs.com/acs-sample/
ghost:0.7
links:
- db:mysql
ports:
- "2368"
environment:
- GHOST_URL=http://my-ghost-blog.com
labels:
aliyun.scale: '3'
restart: always
db:
external:

服务注册

Spring Cloud

```
@Configuration
@ComponentScan
@EnableAutoConfiguration
@EnableEurekaClient
@RestController
public class Application {

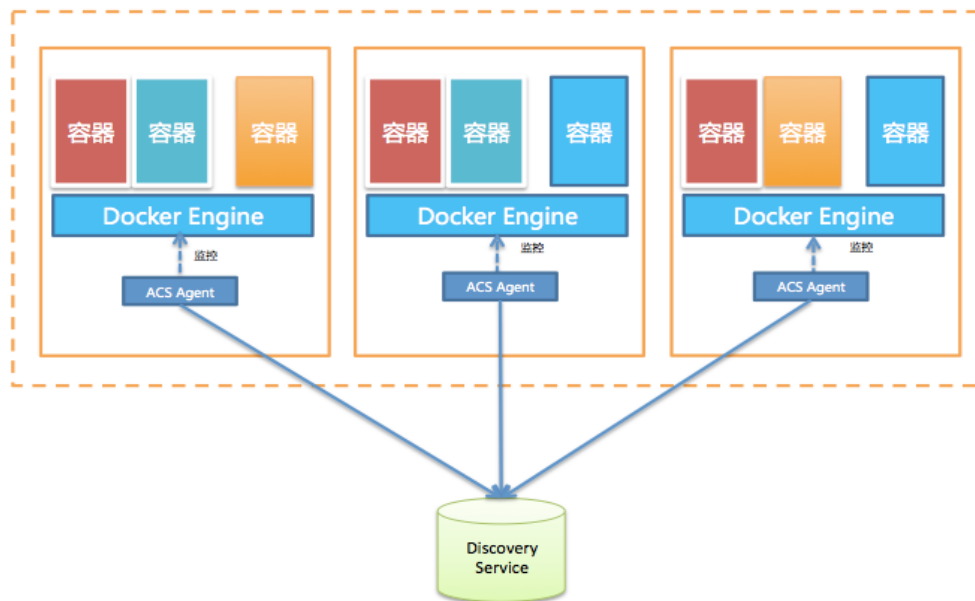
    public static void main(String[] args) {
        new SpringApplicationBuilder(Application.class).
            web(true).run(args);
    }
}
```

application.yml:

```
eureka:
  client:
    serviceUrl:
      defaultZone: http://localhost:8761/eureka/
```

容器服务

- 自动服务注册
- 自动服务发现
- 编排支持



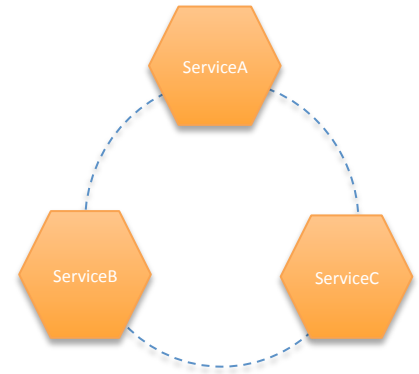
容器服务发现

linking with DNS
alias

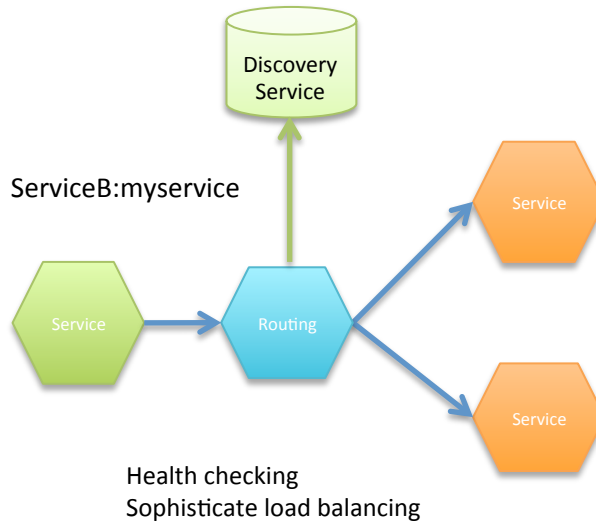
ServiceB:myservice



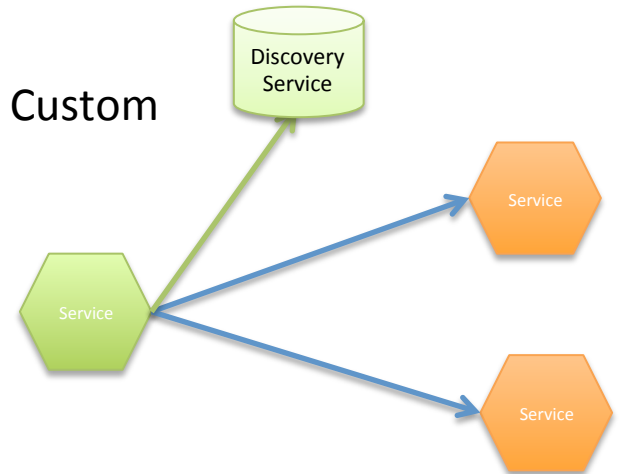
DNS
alias



Linking with load
balancing



Custom



容器服务弹性路由

http://*.company.com

负载均衡SLB

labels:

aliyun.routing.port_80: http://shopping.company.com

集群 (VPC内)

路由服务

http://shopping.company.com

http://payment.company.com

容器

容器

容器

Docker Engine

监控

容器服务Agent

虚拟机节点

容器

容器

容器

Docker Engine

监控

容器服务Agent

虚拟机节点

容器

容器

容器

Docker Engine

监控

容器服务Agent

虚拟机节点

Discovery Service

服务监控报警

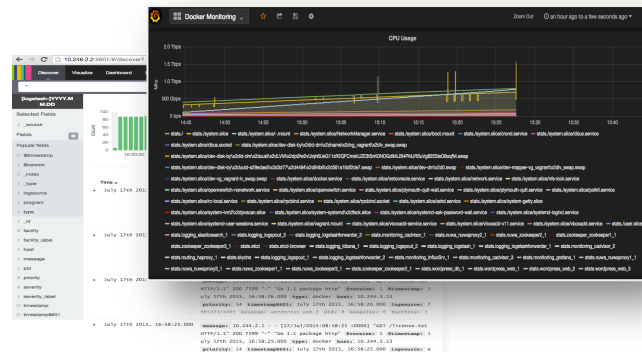
• 集成的容器监控

- Aliyun 扩展会根据标签来收集和聚合监控数据及日志数据

- `aliyun.monitoring.script: /opt/mp.sh`
 - `aliyun.logs:/var/log/mysql/`

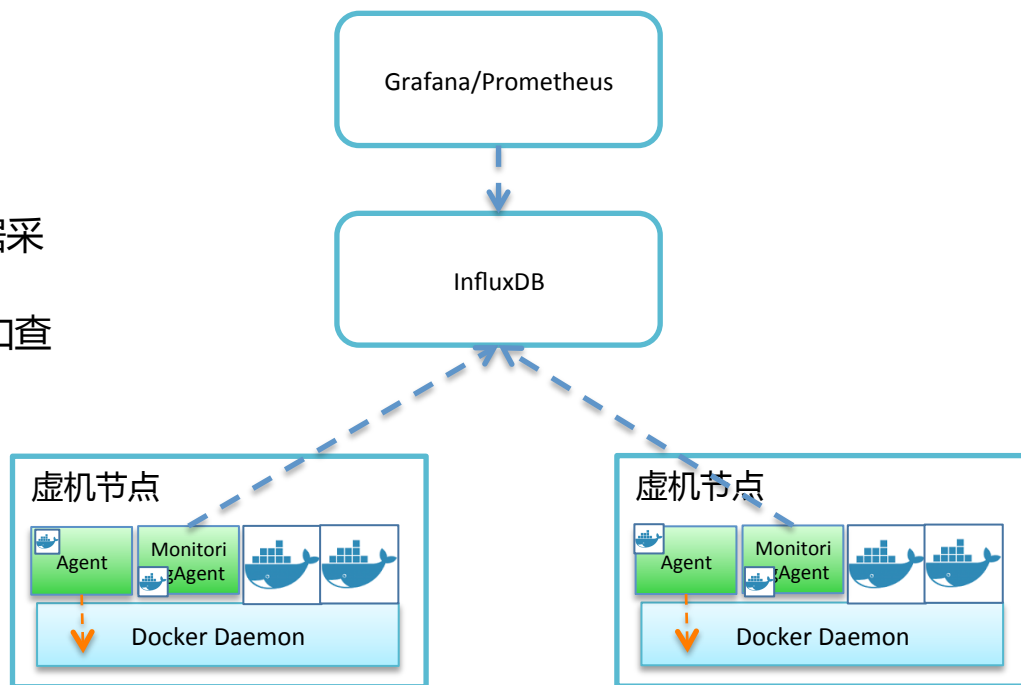
- 插件机制支持采集及集成扩展
- 内置阿里云云监控集成
- 支持三方扩展

- `aliyun.monitoring.addone.influxdb :`

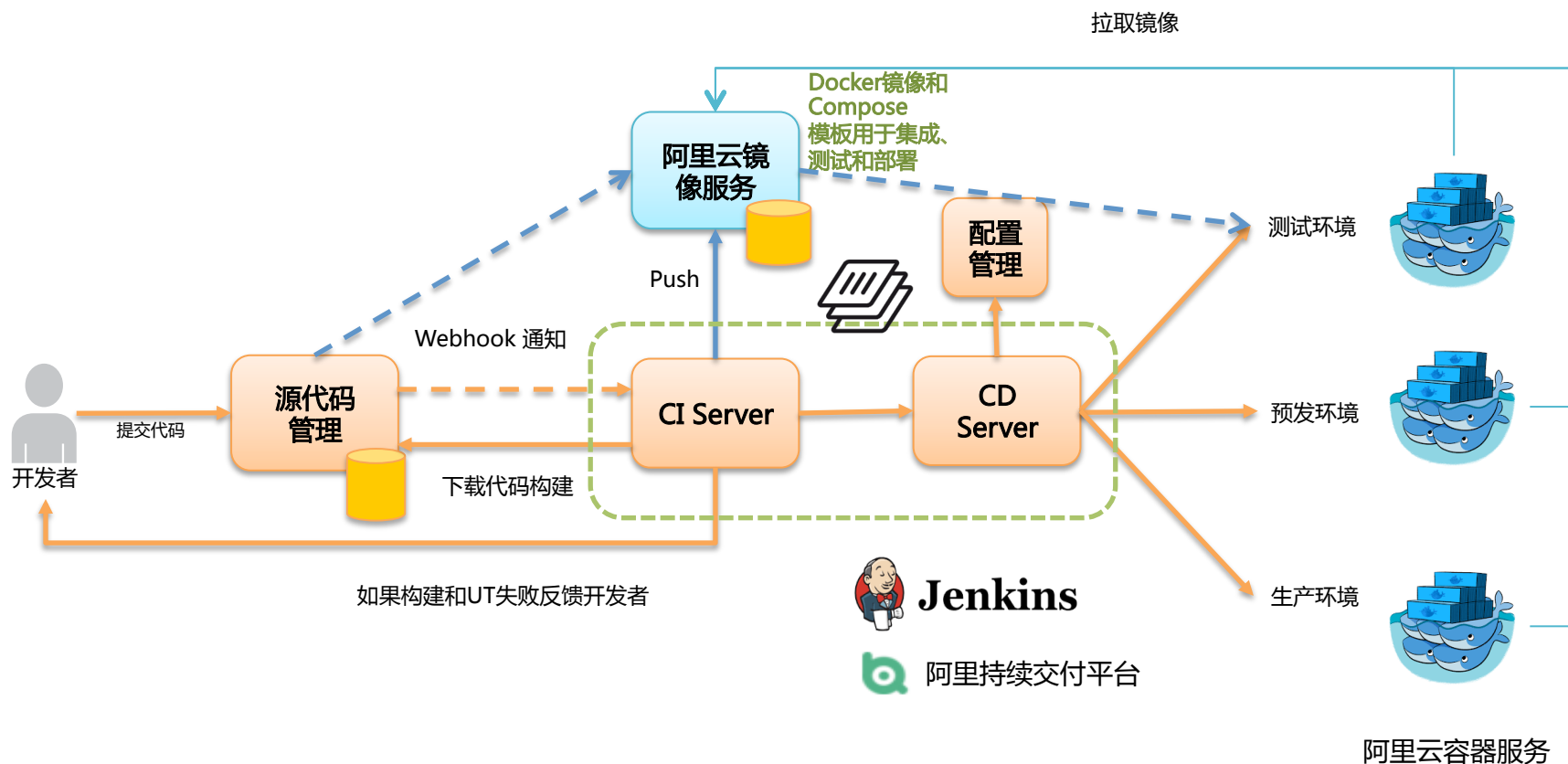


• 开源监控方案

- cAdvisor在每个节点上进行监控指标数据采集
- InfluxDB 负责收集汇总监控指标数据，和查询
- Grafana/Prometheus 来展现监控指标



完整的容器化持续交付流程



容器即服务-Container-as-a-Service



谢谢



云栖社区Docker团队博客

<https://yq.aliyun.com/teams/11>

阿里云容器服务

<https://www.aliyun.com/product/container-service/>