



马哥教育

IT人的高薪职业学院

# 消息队列与微服务 训练营(day3)



讲师：张士杰/杰哥

<http://www.magedu.com>



马哥教育

IT人的高薪职业学院

## 马哥教育近期开班

Linux直播班：2021年3月13日

Go直播班：2021年3月13日

Linux面授班：2021年3月22日

Python运维自动化班：2021年3月20日

DevOps面授班：2021年4月19日

Linux录播+Python录播随到随学



马哥教育

IT人的高薪职业学院

- 1.拥有 RHCA、OpenStack、EXIN DevOps Master 等专业证书。
- 2.多年大型互联网一线工作经验，曾在互联网金融和互联网电商等公司任职运维架构师、云平台架构师等职位。
- 3.曾维护企业数千台服务器的业务规模。
- 4.拥有十一年一线运维工作经验。
- 5.熟练虚拟化、Docker、kubernetes、公有云等业务环境规划、实施与运维。



- 第一天:
  - MQ简介及应用场景
  - RabbitMQ单机及使用
  - RabbitMQ集群及使用
- 第二天:
  - zookeeper单机、集群及实战案例
  - kafka应用基础
  - kafka实战案例
- 第三天:
  - 微服务简介
  - 生产者与消费者模型
  - 基于dubbo的生产者与消费者模型微服务实战案例



单体架构:

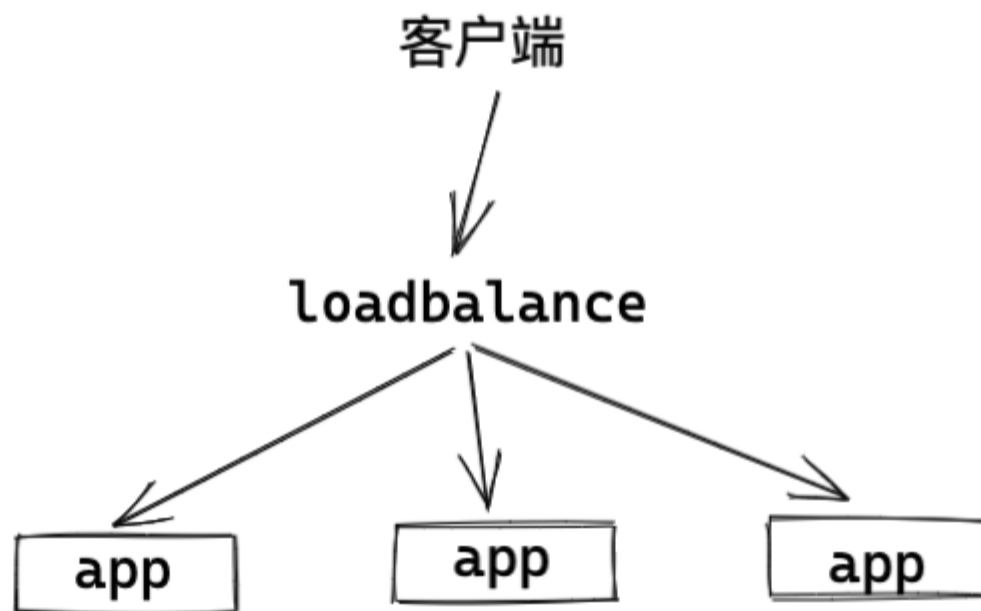


马哥教育

IT 人的高薪职业学院

单体架构:Monolithic Architecture，将应用的多个功能整体构建为一个部署包，服务运行后具备多个访问功能。

## 单体架构-Monolithic Architecture

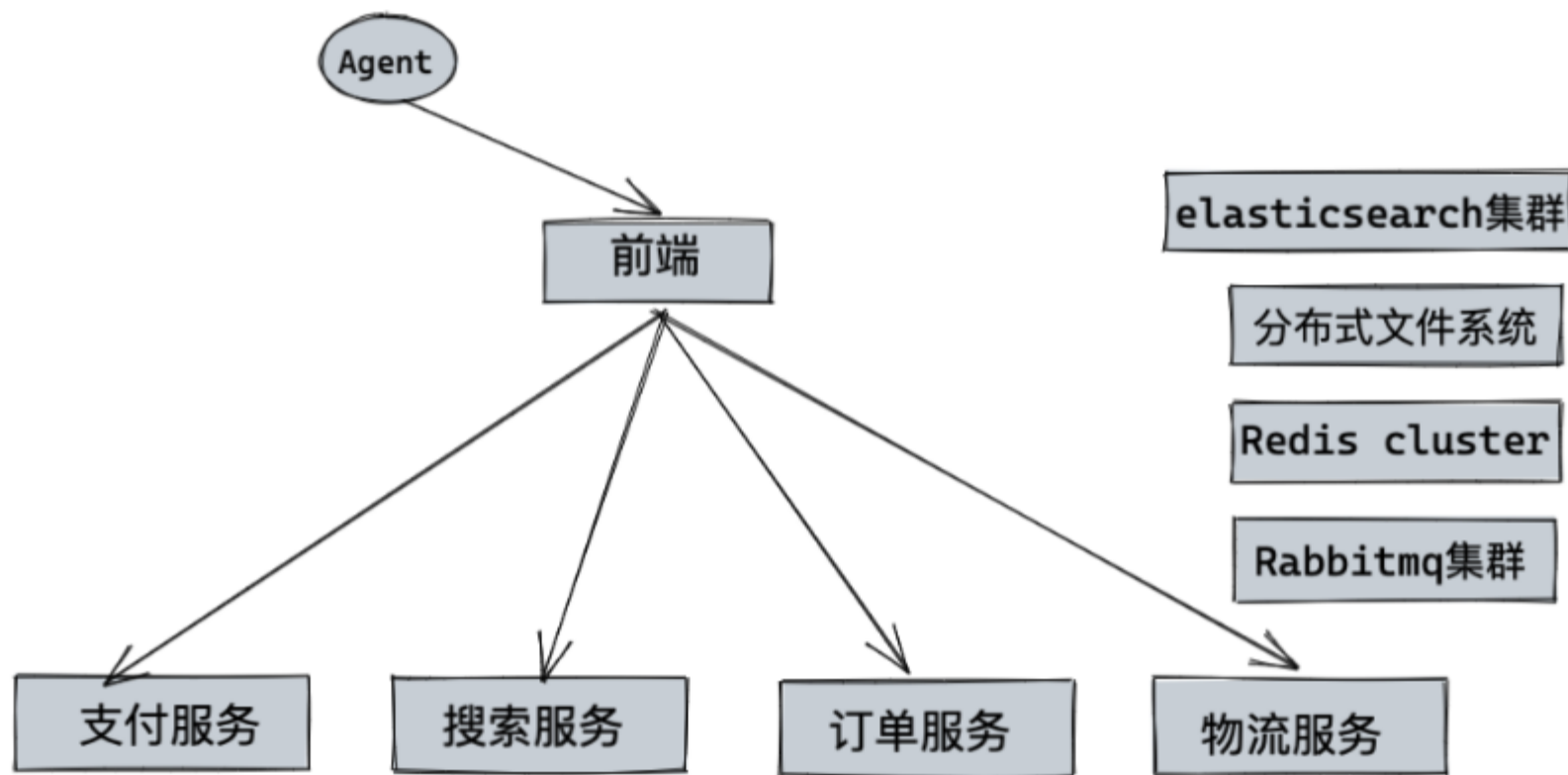




### 单体架构:

- 1.部署效率低: 一套代码几十兆上百兆, 需要较长的编译时间、部署时间和启动时间。
- 2.影响开发效率: 一个单体应用中包含多个功能, 但是每个功能可能是不同的开发人员或小组单独开发, 任何的功能变更都需要所有开发人员全部提交代码并完整构建。
- 3.影响业务稳定: 一个tomcat或jar包运行的代码过于臃肿, 某台服务器只需要提供一个功能但是不得不额外运行其他功能并消耗不必要的系统资源, 也可能导致出现内存溢出等问题。

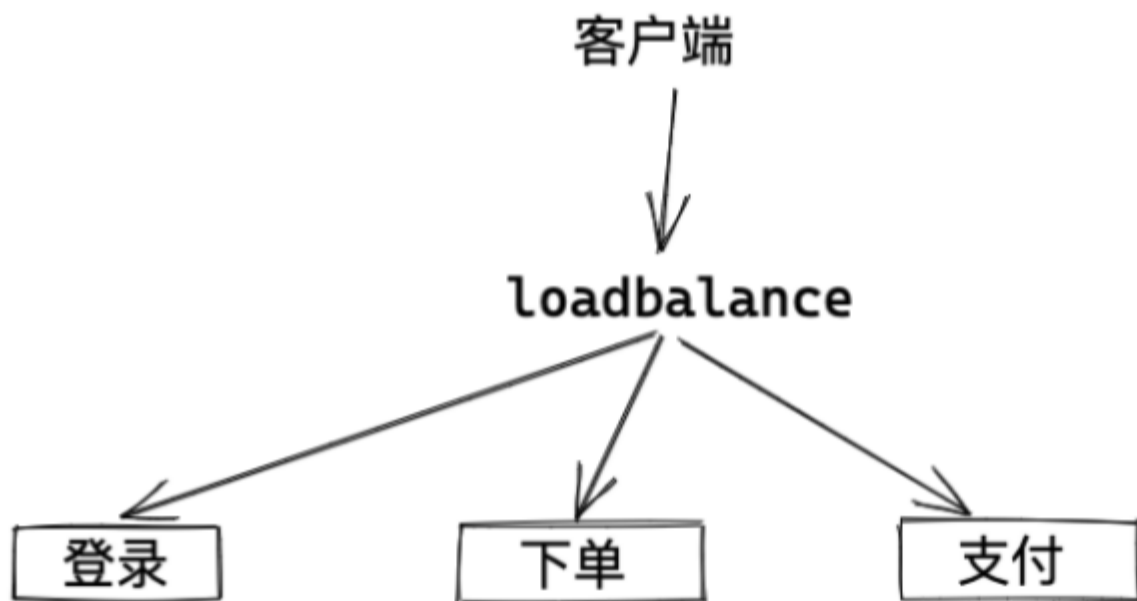
SOA(Service-Oriented Architecture): 面向服务的架构, SOA架构将应用程序按照不同功能进行拆分, 服务之间通过定义好的接口进行访问。





微服务: Micro Services, 2011年5月威尼斯的一个软件架构会议上讨论并提出的一個概念, 用于描述一些作为通用架构风格的设计原则, 2012年3月在波兰克拉科夫举行的33rd Degree Conference大会上, James Lewis(詹姆斯·里维)斯做了题为《Micro services - Java, the Unix Way》的演讲, 论了微服务的一些原则和特征, 例如单一服务职责、自动扩展等特征。

### Micro Services





独立部署：每个服务都是独立项目，可以独立放置，不依赖其他服务。

弹性伸缩：服务功能分离，可以在完全不影响其他服务的前提下，对任何一个服务进行横向扩容。

易于部署：部署包比单体体积更小，传输时间短、启动速度快，因此可以灵活部署。

缩短开发周期：单个服务功能减少，更容易对一个服务进行功能开发。

易于维护：单个运行环境需要的资源更小，方便后期运维。

支持多语言：服务之间通API访问，各服务可以使用不同的开发语言。

数据去中心化：每个微服务都可以按照需求有自己独立的、任意类型的数据库或采用不同的数据库技术（SQL、NoSQL等），进一步降低了微服务之间的耦合关联度，还可以减轻单体环境下数据库的负载。

# 如何实现微服务-落地要点:



马哥教育

IT人的高薪职业学院

微服务落地要点:

微服务如何落地(docker)

微服务之间如何发现对方(注册中心、服务发现)

微服务之间如何访问对方(服务访问-> restful API)

微服务如何快速扩容(服务编排) docker swarm

微服务如何监控(服务监控)

微服务如何升级与回滚(CI/CD)

微服务访问日志如何查看(ELK)

## 如何实现微服务-拆分方式:



马哥教育

IT人的高薪职业学院

### 实现微服务的服务拆分方式:

#### 横向拆分:

按照不同的业务进行拆分, 如支付、订单、登录、物流

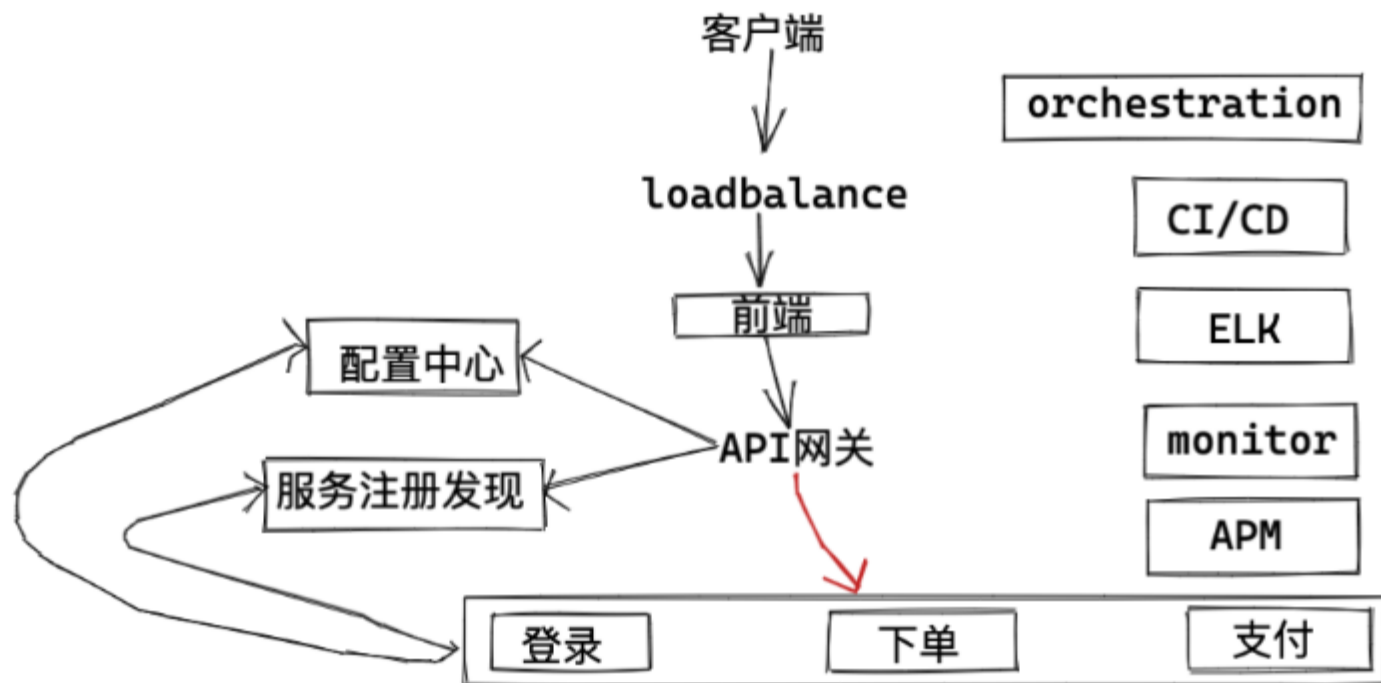
#### 纵向拆分:

把一个业务中的组件再细致拆分, 比如支付系统拆分为微信支付、银联支付、支付宝支付

微服务架构(Micro Service Architecture, MSA): 是SOA架构的升级, 拆分粒度更细, 微服务架构最早是由Fred George(弗雷德乔治)在2012年的一次技术大会上所提出, 他讲解了如何分拆服务以及如何利用MQ来进行服务间的解耦, 这就是最早的微服务架构雏形.



## Micro Service Architecture



## 如何实现微服务-业务模型:



马哥教育

IT人的高薪职业学院

### 生产者与消费者(Producer & consumer)模型:

消息生产者生产消息发送到消息服务器中,然后消息消费者从消息服务器中取出并且消费消息,消息被消费以后,消息服务器中不再有存储,所以其它消费者不会再消费到已经被消费的消息,消息服务器支持存在多个消费者同时连接,**但是同一个消息只会有一个消费者可以消费**,此模式也是目前使较多的业务处理模式。

### 发布者与订阅者(Publisher & Subscriber)模型:

消息发布者将消息发布到topic中,同时有多个消息消费者订阅消费该消息,和生产者与消费者方式不同,发布到topic的消息会被所有订阅者消费,用于和数据相关的业务场合,比如群聊天、订阅服务等场合。

## 如何实现微服务-开发环境:



马哥教育

IT人的高薪职业学院

开发环境:

Spring boot: 是一个快速开发框架, 内置Servlet运行环境。

Spring Cloud基于Spring Boot, 为微服务体系开发中的架构问题, 提供了一整套的解决方案——服务注册与发现、服务保护、分布式配置管理等。

Dubbo是阿里巴巴开源的分布式服务治理框架, 已经贡献给Apache基金会, 也具备服务注册与发现, 服务消费等功能, 出现比Spring Cloud早, 目前依然有很多互联网企业实现dubbo实现微服务。



高性能RPC调用：提供高性能的基于代理的远程调用能力，服务以接口为粒度，为开发者屏蔽远程调用底层细节。

智能负载均衡：内置多种负载均衡策略，智能感知下游节点健康状况，显著减少调用延迟，提高系统吞吐量。

服务自动注册与发现：支持多种注册中心服务，服务实例上下线实时感知。

高度可扩展能力：遵循微内核+插件的设计原则，所有核心能力如Protocol、Transport、Serialization(序列化)被设计为扩展点，平等对待内置实现和第三方实现。

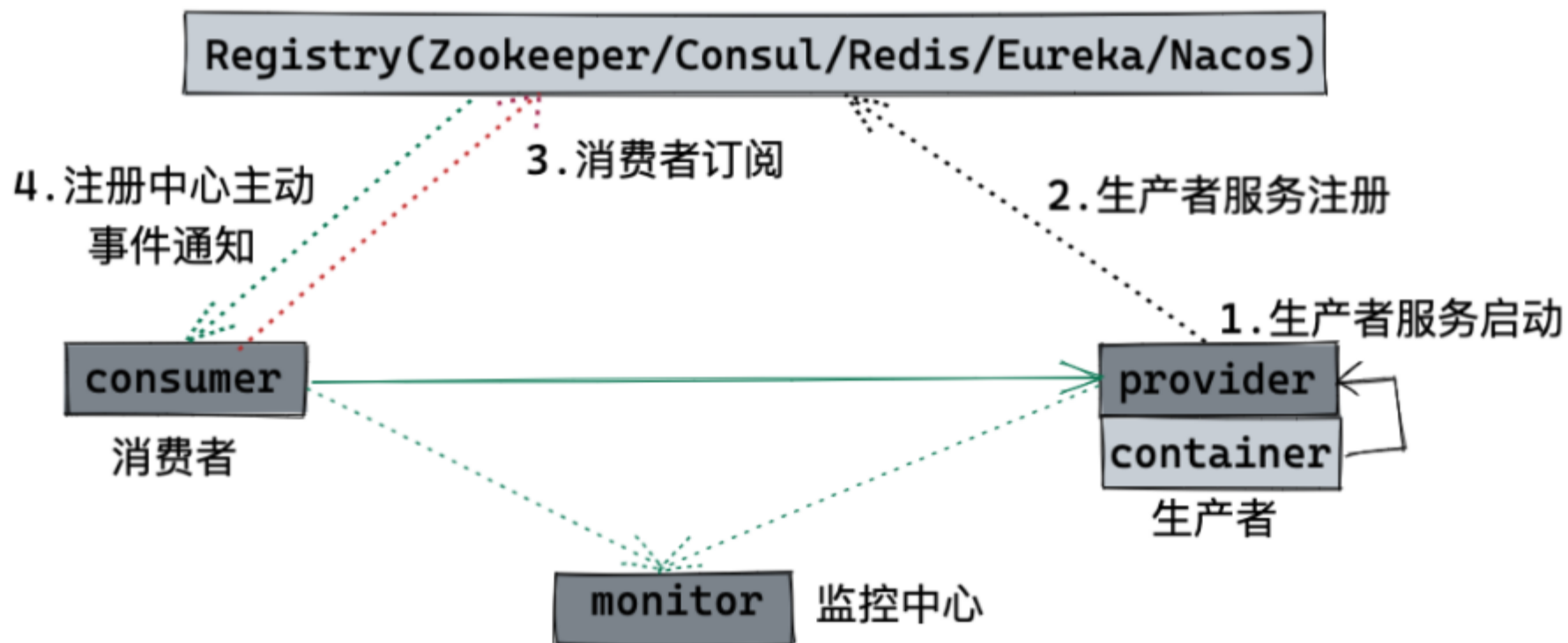
流量调度：内置条件、脚本等路由策略，通过配置不同的路由规则，轻松实现灰度发布，同机房优先等功能。

可视化的服务治理与运维：提供丰富服务治理、运维工具：随时查询服务元数据、服务健康状态及调用统计，实时下发路由策略、调整配置参数。





## Dubbo Architecture



## Dubbo微服务-生产者示例:



马哥教育

IT人的高薪职业学院

```
# bash jdk-install.sh
```

```
# tar xf dubbo-demo-provider-x.y.z-assembly.tar.gz
```

```
# cd dubbo-demo-provider-x.y.z
```

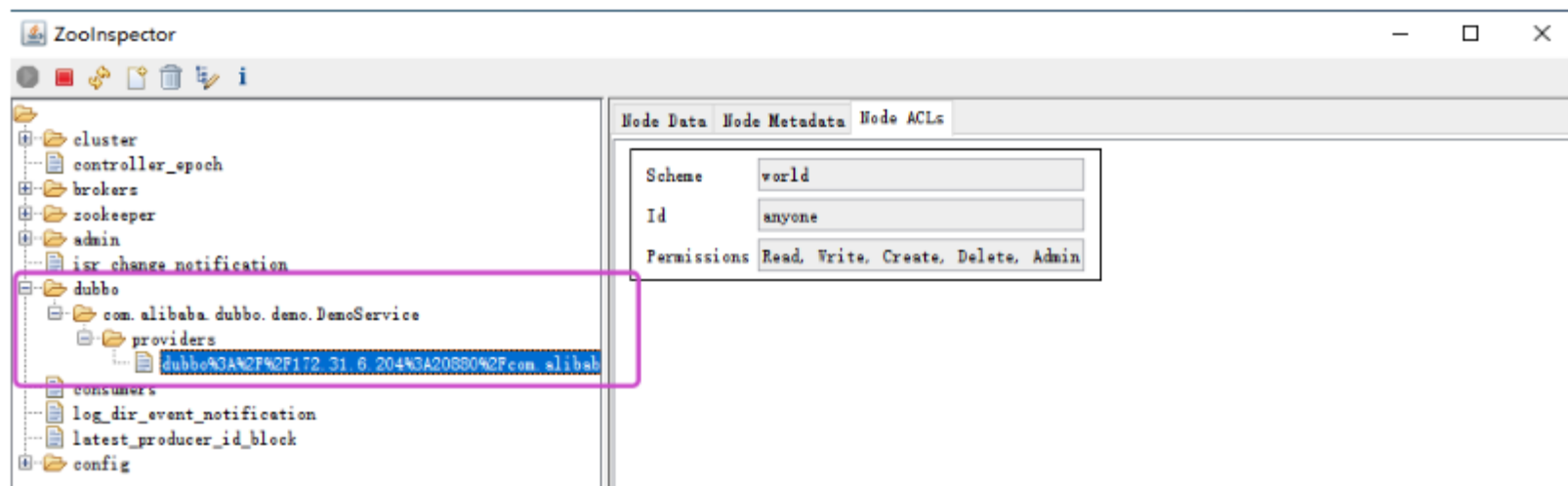
```
# vim conf/dubbo.properties
```

```
dubbo.registry.address=zookeeper://172.31.6.201:2181 | zookeeper://172.31.6.202:2181 | zookeeper://172.31.6.203:2181 #zk集群配置
```

```
#dubbo.registry.address=zookeeper://172.31.6.201:2181 #单zk节点配置
```

```
# ./bin/start.sh #启动provider
```

```
Starting the demo-provider ....
```



## Dubbo微服务-消费者示例:



马哥教育

IT人的高薪职业学院

```
# tar xvf dubbo-demo-consumer-x.y.z-assembly.tar.gz
```

```
# cd dubbo-demo-consumer-x.y.z/
```

```
# vim conf/dubbo.properties
```

```
dubbo.registry.address=zookeeper://172.31.6.201:2181 | zookeeper://172.31.6.202:2181 | zookeeper://172.31.6.203:2181
```

```
# ./bin/start.sh #启动consumer
```

Starting the demo-consumer ....OK!

PID: 25137

STDOUT: logs/stdout.log

The screenshot shows the ZooInspector web interface. On the left, a tree view displays the ZooKeeper hierarchy. The 'dubbo' node is expanded, showing 'com.alibaba.dubbo.monitor.MonitorService', 'com.alibaba.dubbo.demo.DemoService', 'consumers', and 'providers'. The 'consumers' node is further expanded, and a specific consumer node is highlighted with a pink box. On the right, the 'Node Data' tab is active, displaying various metadata fields for the selected consumer node.

Node Data	Node Metadata	Node ACLs
ACL Version	0	
Creation Time	1615020317170	
Children Version	0	
Creation ID	8589934645	
Data Length	0	
Ephemeral Owner	216172803975086081	
Last Modified Time	1615020317170	
Modified ID	8589934645	
Number of Children	0	
Node ID	8589934645	
Data Version	0	



```
# tail -f logs/stdout.log
Java HotSpot(TM) 64-Bit Server VM warning: ignoring option PermSize=128m; support was removed in 8.0
log4j:WARN No appenders could be found for logger (com.alibaba.dubbo.common.logger.LoggerFactory).
log4j:WARN Please initialize the log4j system properly.
log4j:WARN See http://logging.apache.org/log4j/1.2/faq.html#noconfig for more info.
[16:45:17] Hello world0, response form provider: 172.31.6.204:20880
[16:45:19] Hello world1, response form provider: 172.31.6.204:20880
[16:45:21] Hello world2, response form provider: 172.31.6.204:20880
[16:45:23] Hello world3, response form provider: 172.31.6.204:20880
[16:45:25] Hello world4, response form provider: 172.31.6.204:20880
```

## Dubbo微服务-monitor示例:



马哥教育

IT人的高薪职业学院

```
# bash jdk-install.sh
# tar xf apache-tomcat-x.y.z.tar.gz #上传并解压tomcat
# cd /apps/apache-tomcat-x.y.z/webapps
# unzip dubboadmin.war #上传dubbo-admin并解压

# vim ./dubboadmin/WEB-INF/dubbo.properties #编辑配置文件，修改zookeeper连接地址
dubbo.registry.address=zookeeper://172.31.6.201:2181 | zookeeper://172.31.6.202:2181 | zookeeper://172.31.6.203:2181
dubbo.admin.root.password=root
dubbo.admin.guest.password=guest

# /apps/apache-tomcat-x.y.z/bin/catalina.sh start #启动tomcat
```





[首页](#) [服务治理](#) [系统管理](#)

### sysinfo.statues

[首页](#) > [系统管理](#) > [sysinfo.statues](#)

[Dubbo版本](#) [系统快照](#) [系统状态](#) [系统日志](#) [系统环境](#)

#### 汇总

资源名称:

状态: 所有

信息:

负载	正常	load:0.01,cpu:2
内存	正常	max:439M,total:186M,used:116M,free:70M
注册中心	正常	172.31.6.203:2181(connected),172.31.6.201:2181(connected),172.31.6.202:2181(connected)
汇总	正常	



```
[INFO] dubbo-admin ..... SUCCESS [07:18 min]
[INFO] dubbo-admin-ui ..... SUCCESS [02:20 min]
[INFO] dubbo-admin-server ..... SUCCESS [05:08 min]
[INFO] dubbo-admin-distribution ..... SUCCESS [ 13.037 s]
[INFO] dubbo-admin-test ..... SUCCESS [01:50 min]
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 17:07 min
[INFO] Finished at: 2021-03-06T18:02:48+08:00
[INFO] -----
```



Dubbo Admin

服务查询

服务治理

服务测试

接口文档

服务Mock

服务统计

服务统计

服务关系

配置管理

服务查询

服务治理

服务测试

接口文档

服务Mock

服务统计

服务统计

服务关系

配置管理

基础信息

服务

com.alibaba.dubbo.demo.DemoService

应用

demo-consumer

组

版本

服务信息

提供者

消费者

IP地址

端口

超时(毫秒)

172.31.6.204

20880

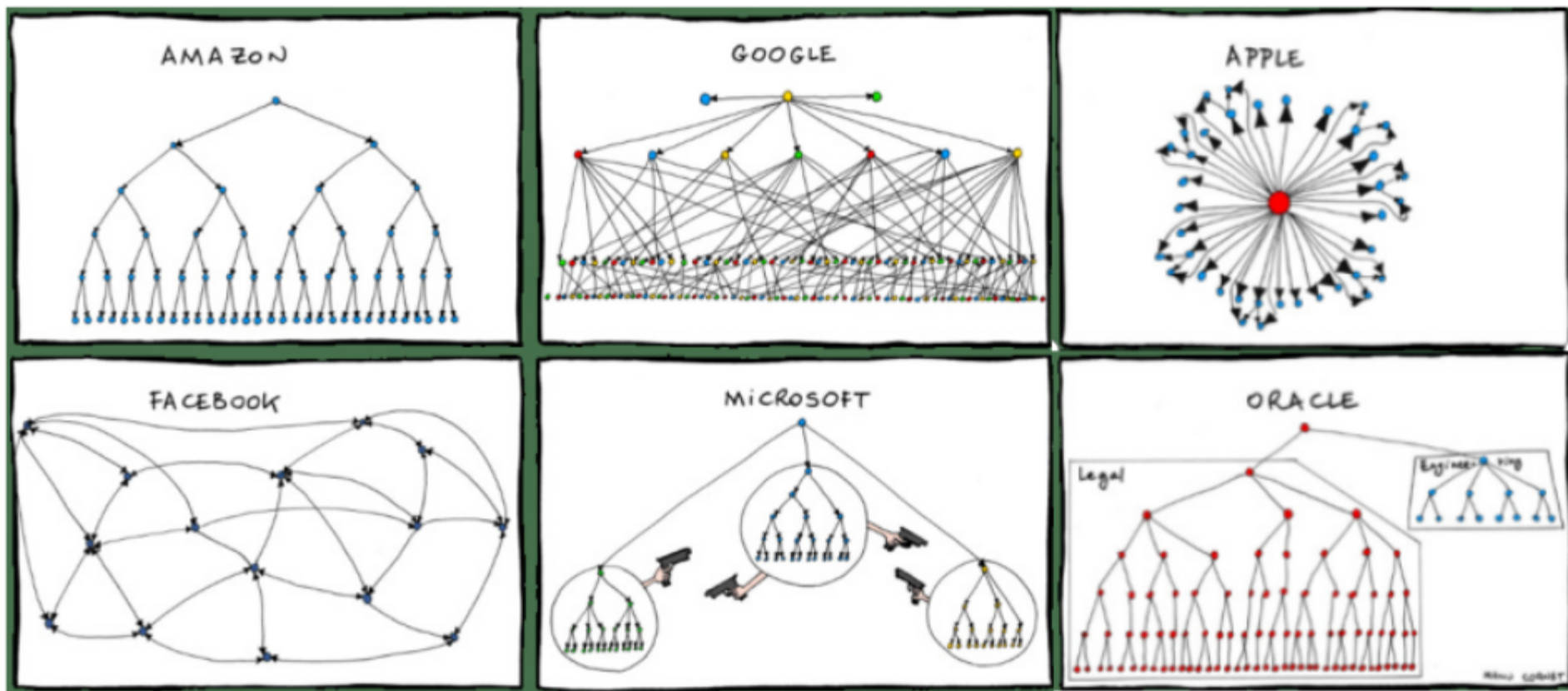
172.31.6.205





没有通用的架构，只有满足需求的架构！

没有最好，只有更好！





## —— 马哥教育近期开班 ——

Linux直播班：2021年3月13日

Go直播班：2021年3月13日

Linux面授班：2021年3月22日

Python运维自动化班：2021年3月20日

DevOps面授班：2021年4月19日

Linux录播+Python录播随到随学



马哥教育

IT人的高薪职业学院

# Thank You!



讲师：张士杰（杰哥）

<http://www.magedu.com>