

# 微服务架构 ( Micro Service ) 在 Cloud Foundry中的应用

吴疆

Pivotal Web Service 资深经理

# Pivotal



CLOUDFOUNDRY Pivotal CF®



Pivotal





**Microservices**

IN



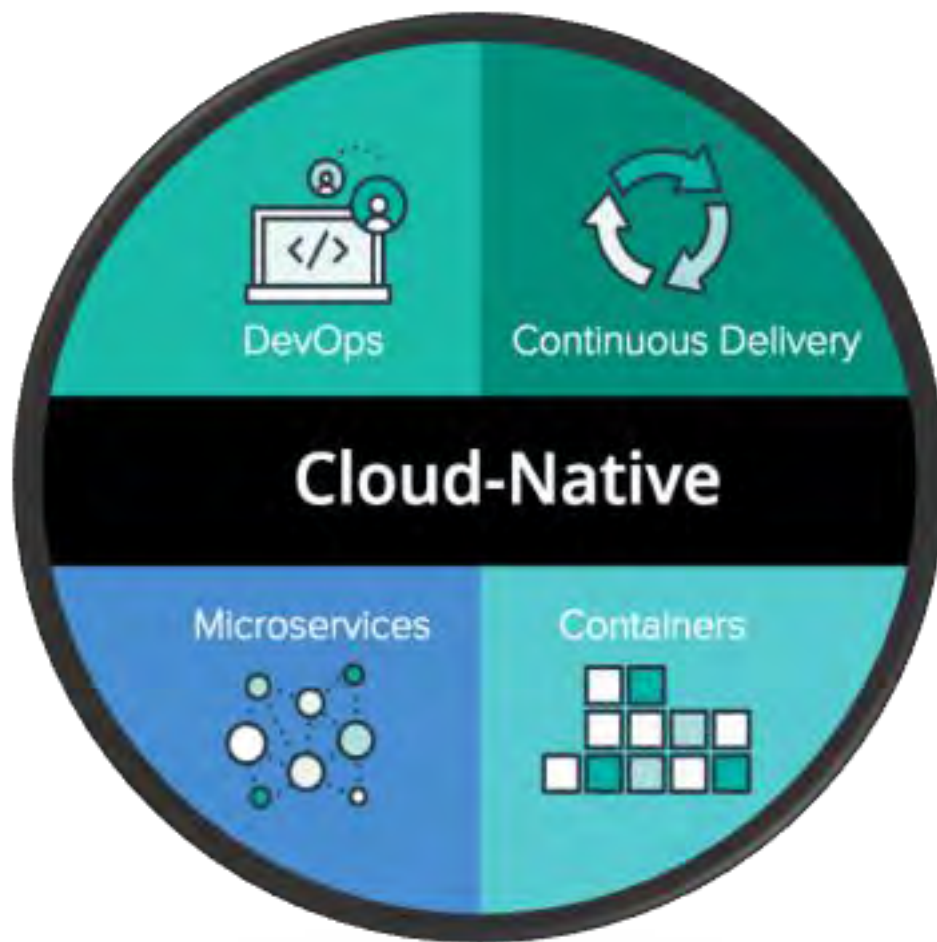
**CLOUD FOUNDRY**

- Cloud Foundry是什么？
  - 开源的
  - Platform as a Service ( PaaS ) 平台

- PaaS平台是什么？
  - 一种云计算的架构模式
  - 为云原生应用提供运行时的应用平台



CLOUD FOUNDRY



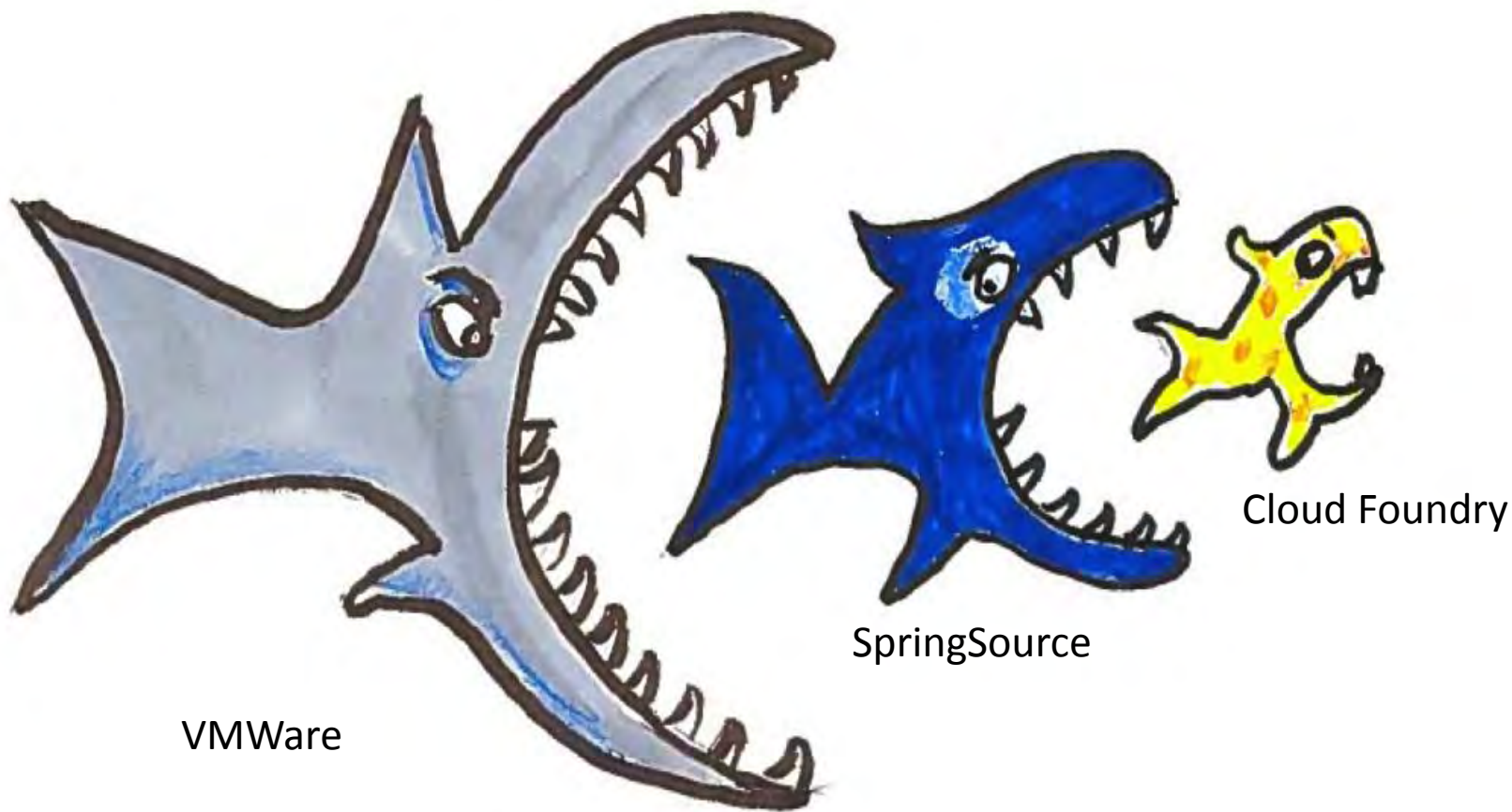
- Pivotal CF ( PCF ) 是什么？
  - Cloud Foundry的Pivotal的发行版
  - Cloud Foundry



- Pivotal Web Service ( PWS ) 是什么？
  - Pivotal维护
  - 部署在AWS上的Cloud Foundry
  - 提供公有云的服务
  - <https://run.pivotal.io>



Pivotal Web Services



# Cloud Foundry的历史

2017.thegiac.com



Google



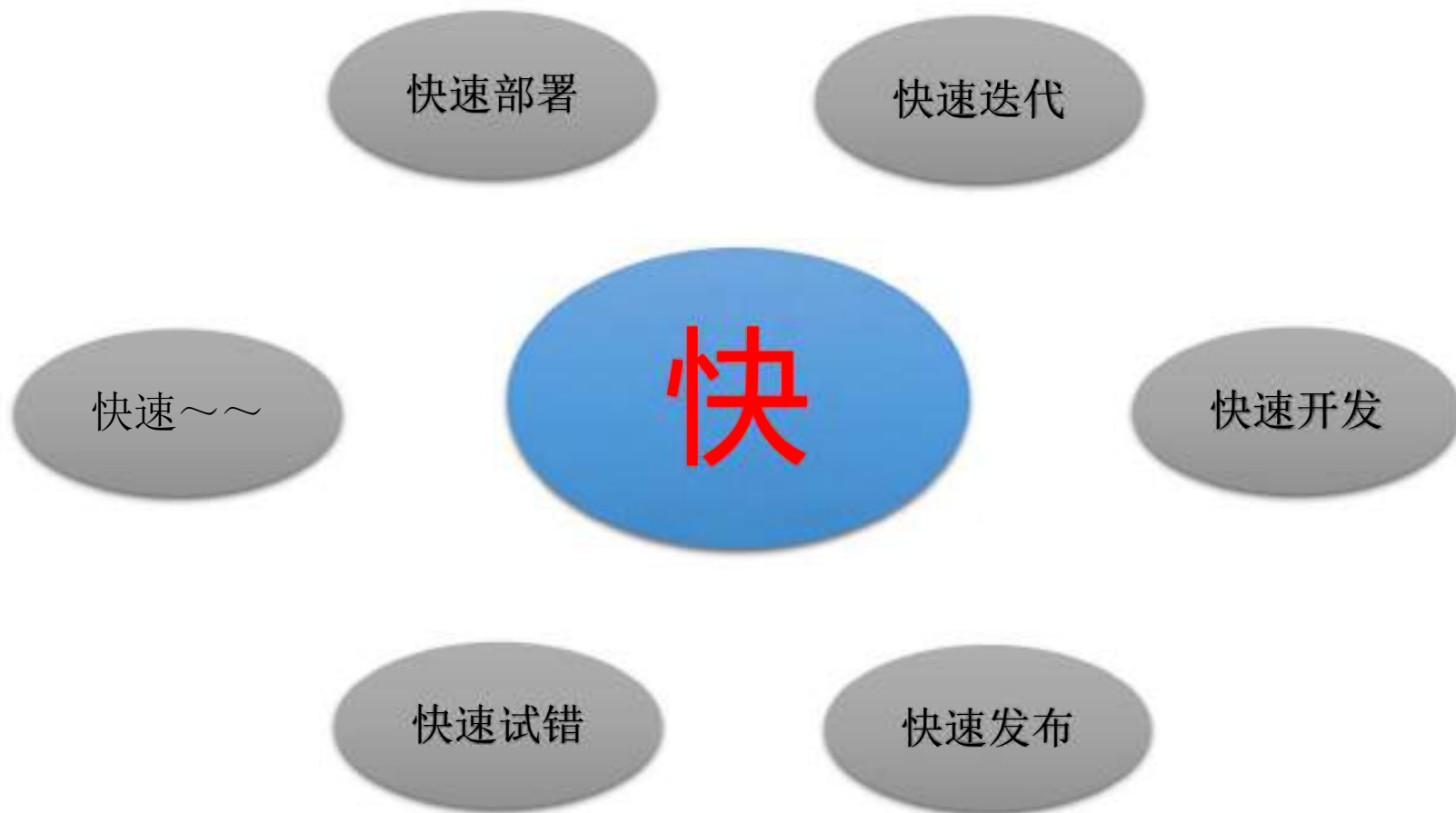
Microsoft

IBM

IT行业进入了互联网时代

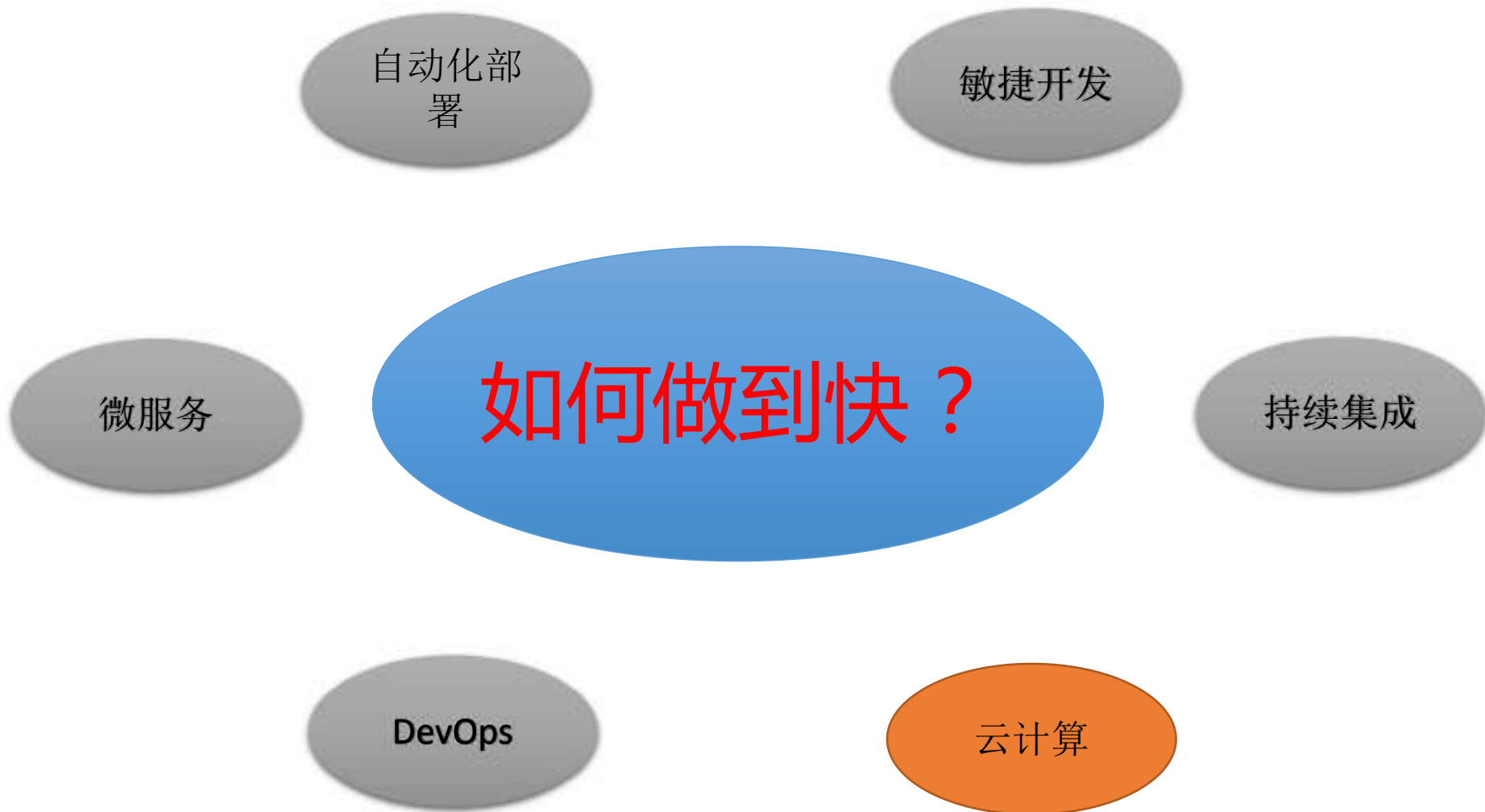
2017.thegiac.com

# 互联网应用的特点





无坚不破，唯快不破  
...speed defines the winner.



# IaaS云平台



我需要一台服务器



服务器准备好了



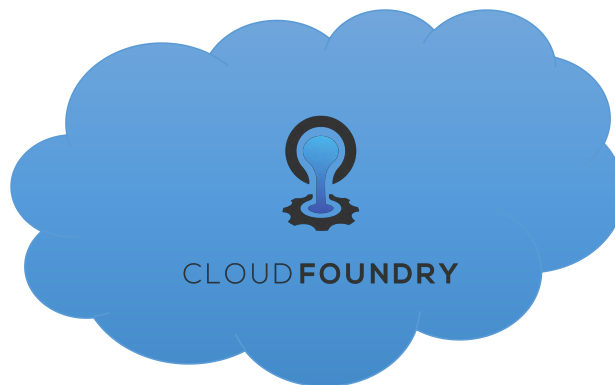
太好了，我可以开始了：

1. 安装数据库
2. 编译打包
3. 安装依赖的库
4. 部署

# PaaS云平台

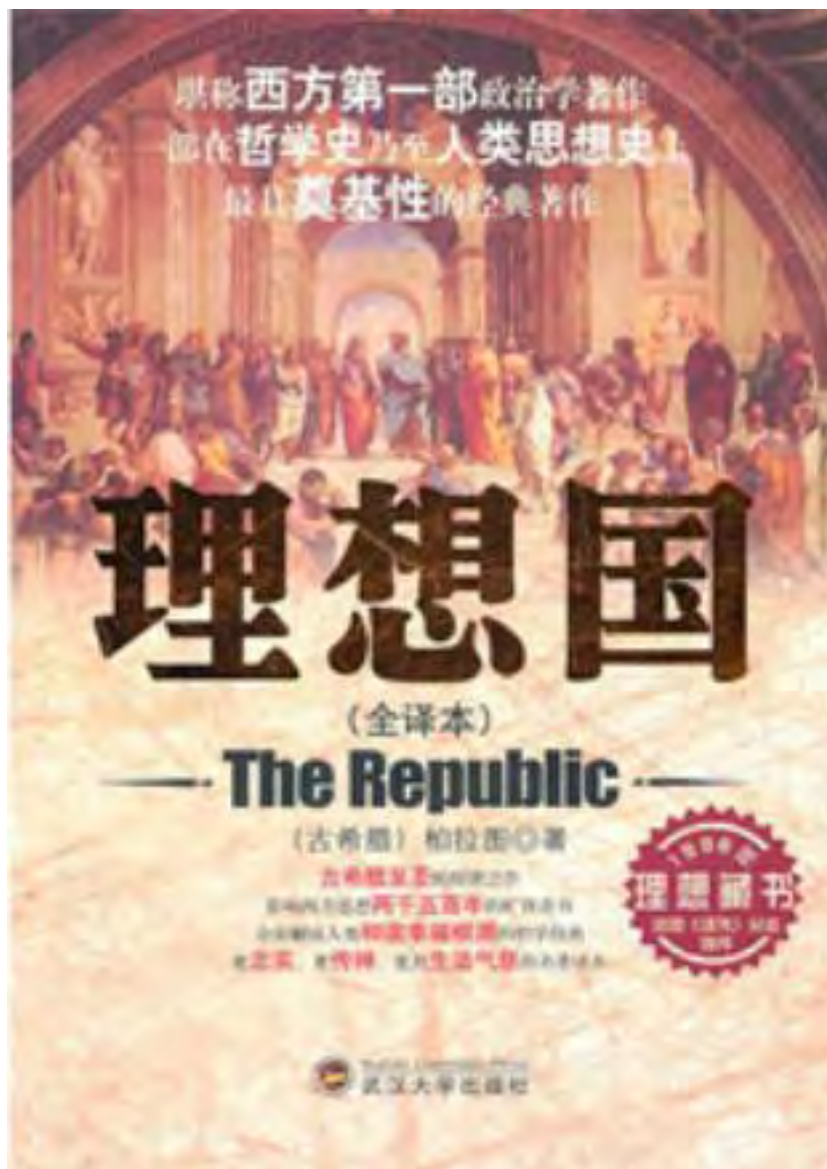


这是我的代码，请帮我运行



好了



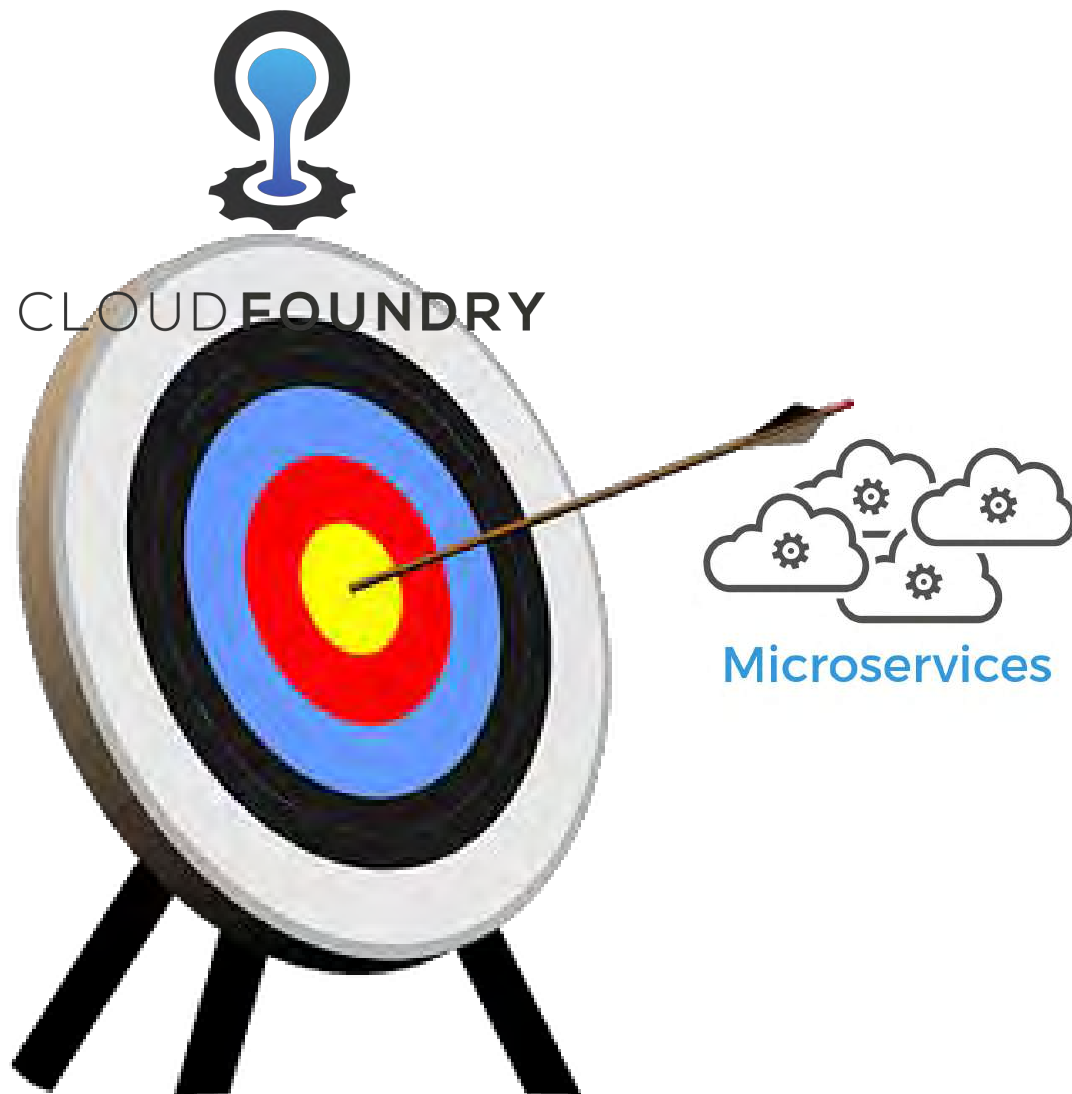


# Cloud Foundry的设计目标

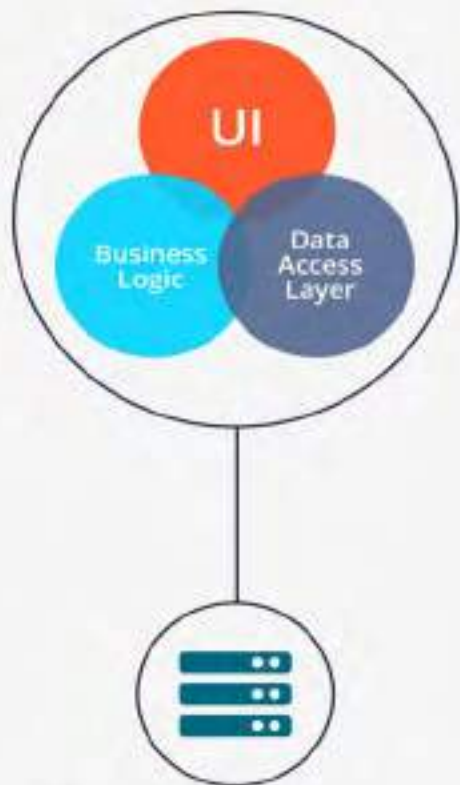
- 云原生应用的云原生平台
  - 云原生
  - 支持多种云平台—Multi Cloud
    - AWS, GCP, Azure
    - OpenStack, vSphere
  - 易于扩展
  - 快速发布，快速升级



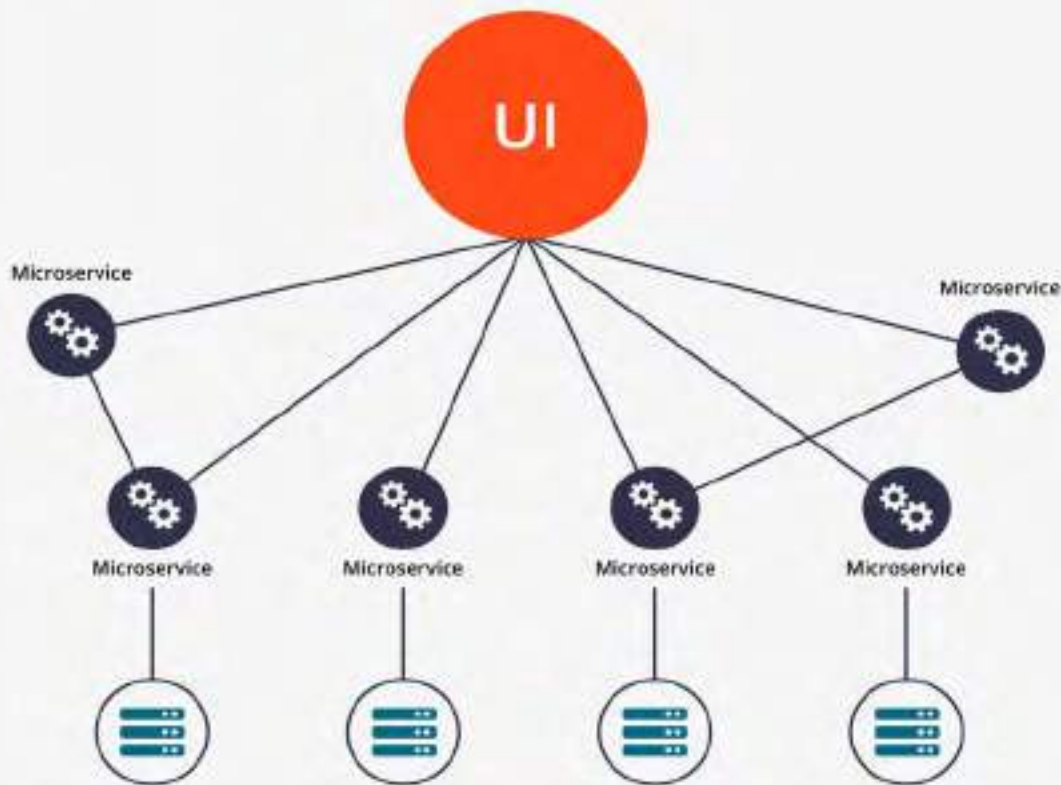




微服务架构是把复杂功能分解为若干独立但相互协作的微服务的架构模式



Monolithic Architecture



Microservice Architecture

# 微服务架构的优势

快速发布

快速试错



Microservices

独立扩展

快速应用  
新技术

# 微服务架构的优势

快速试错

快速发布

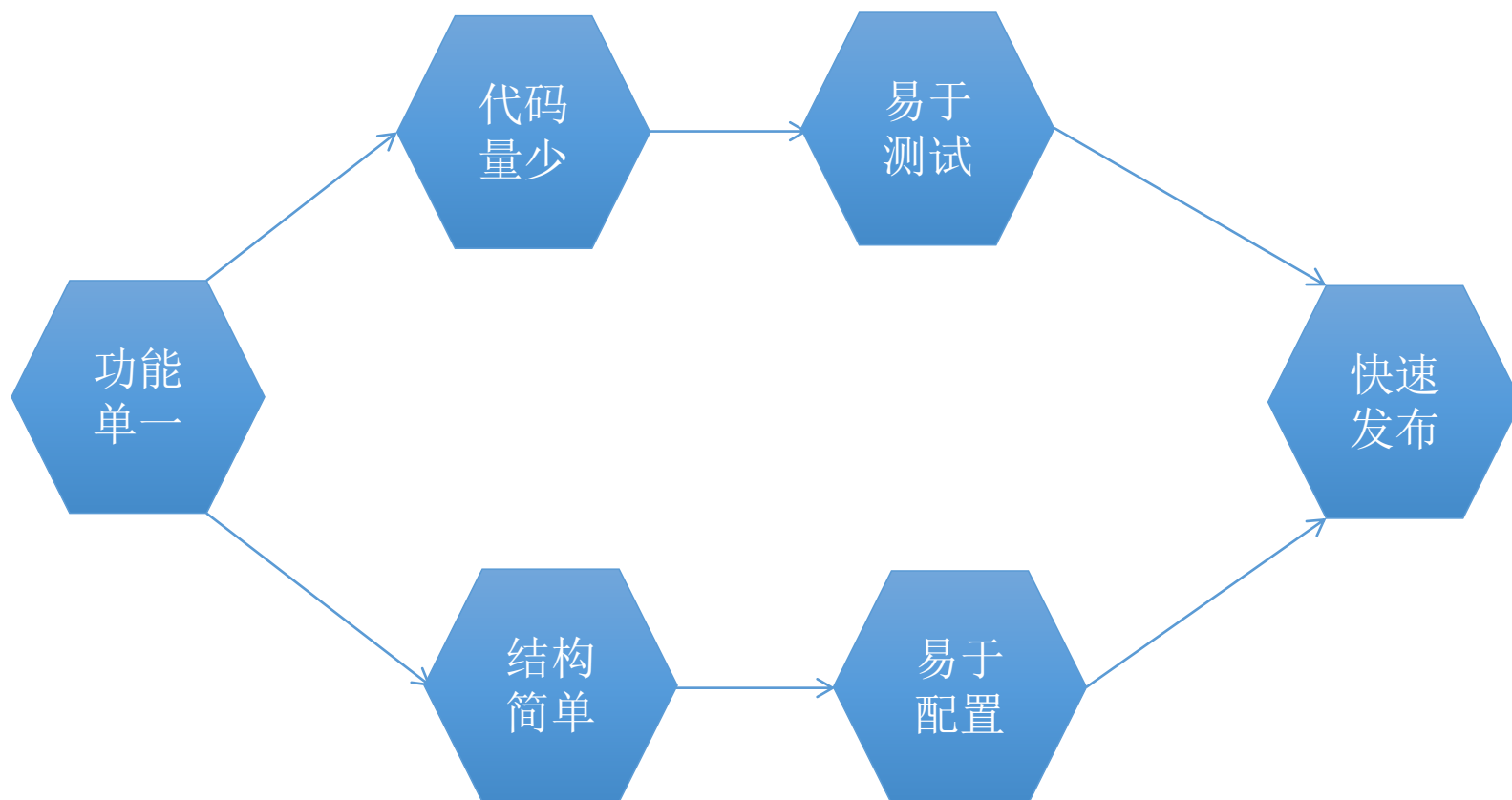


Microservices

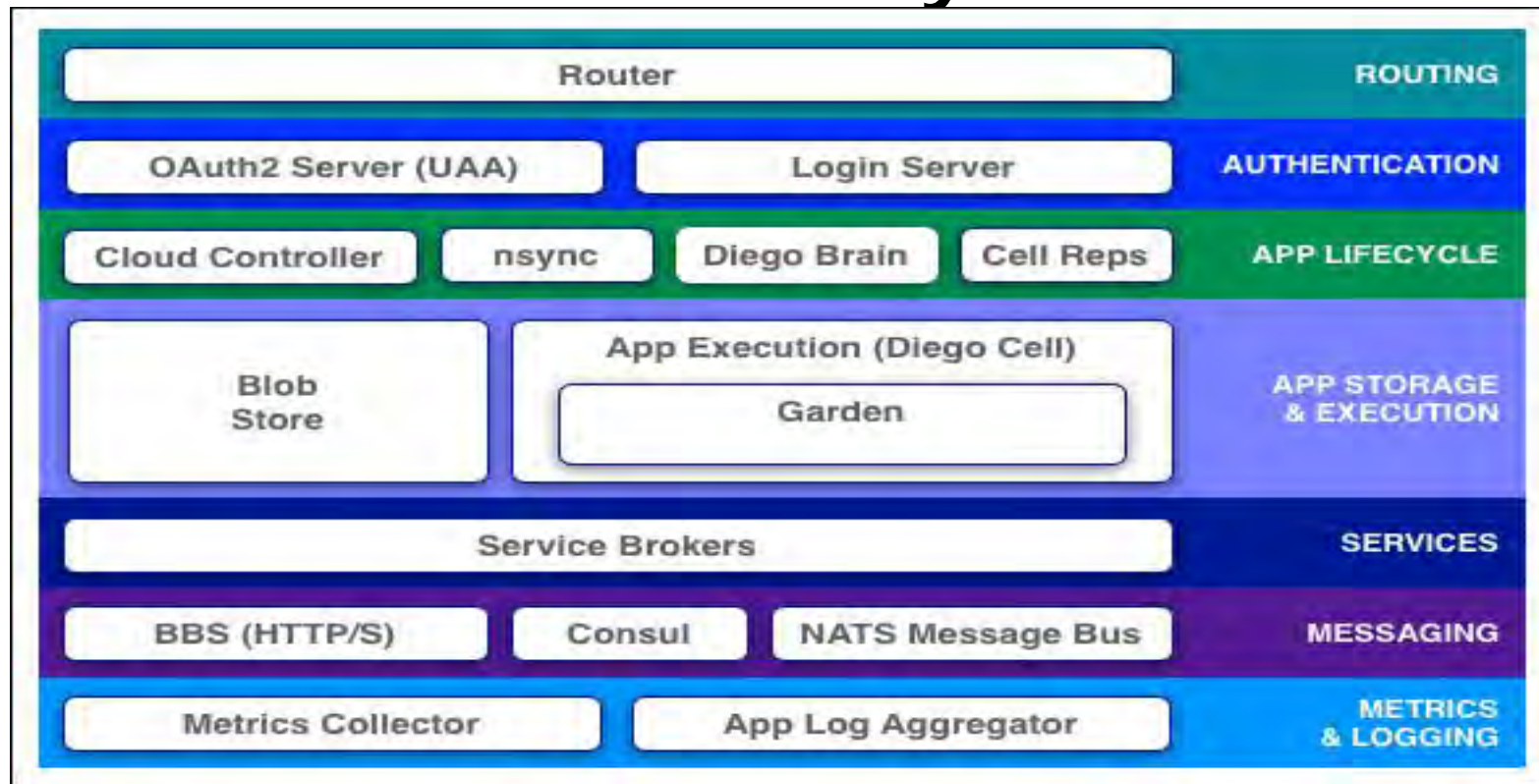
独立扩展

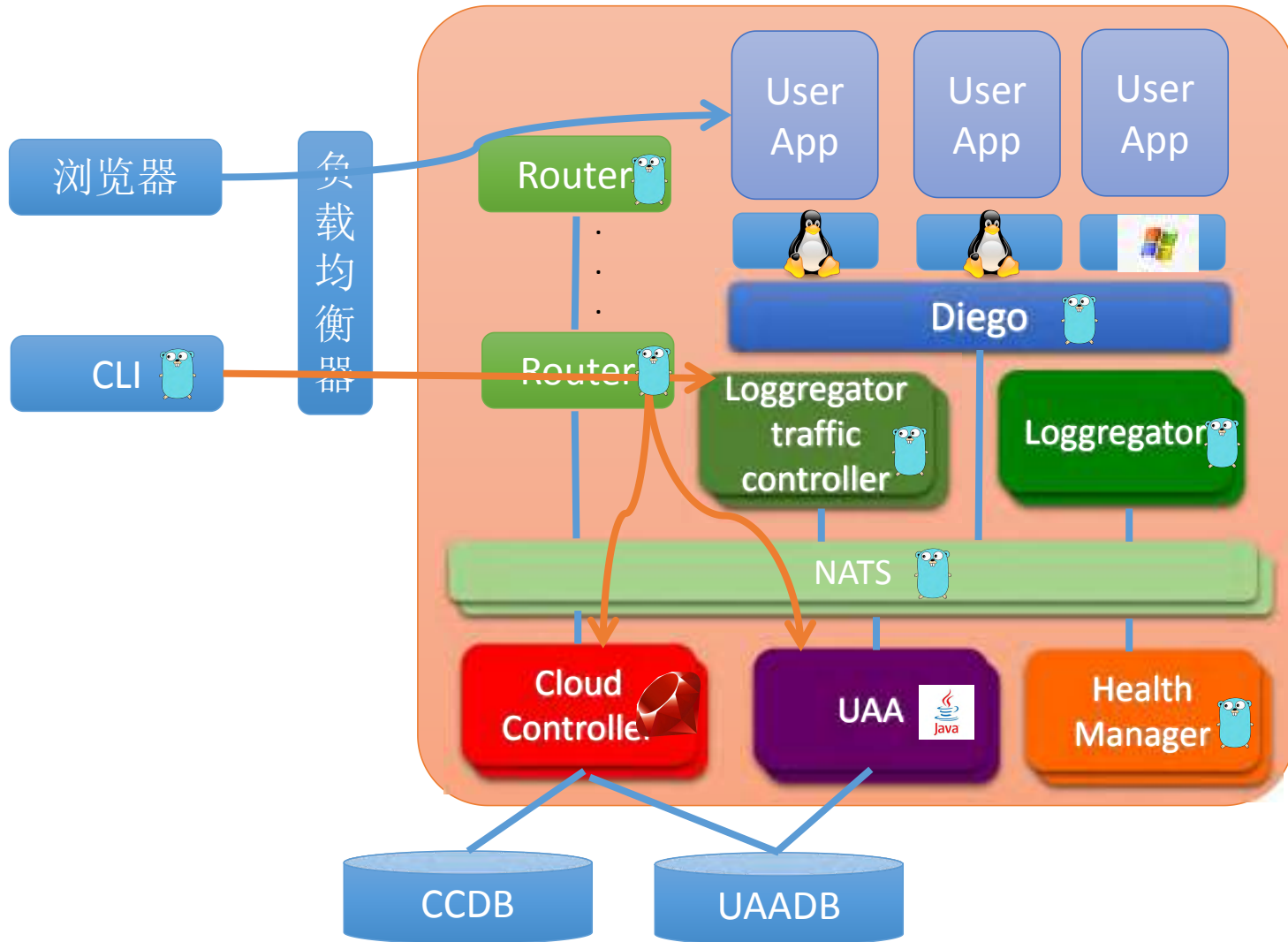
快速应用  
新技术

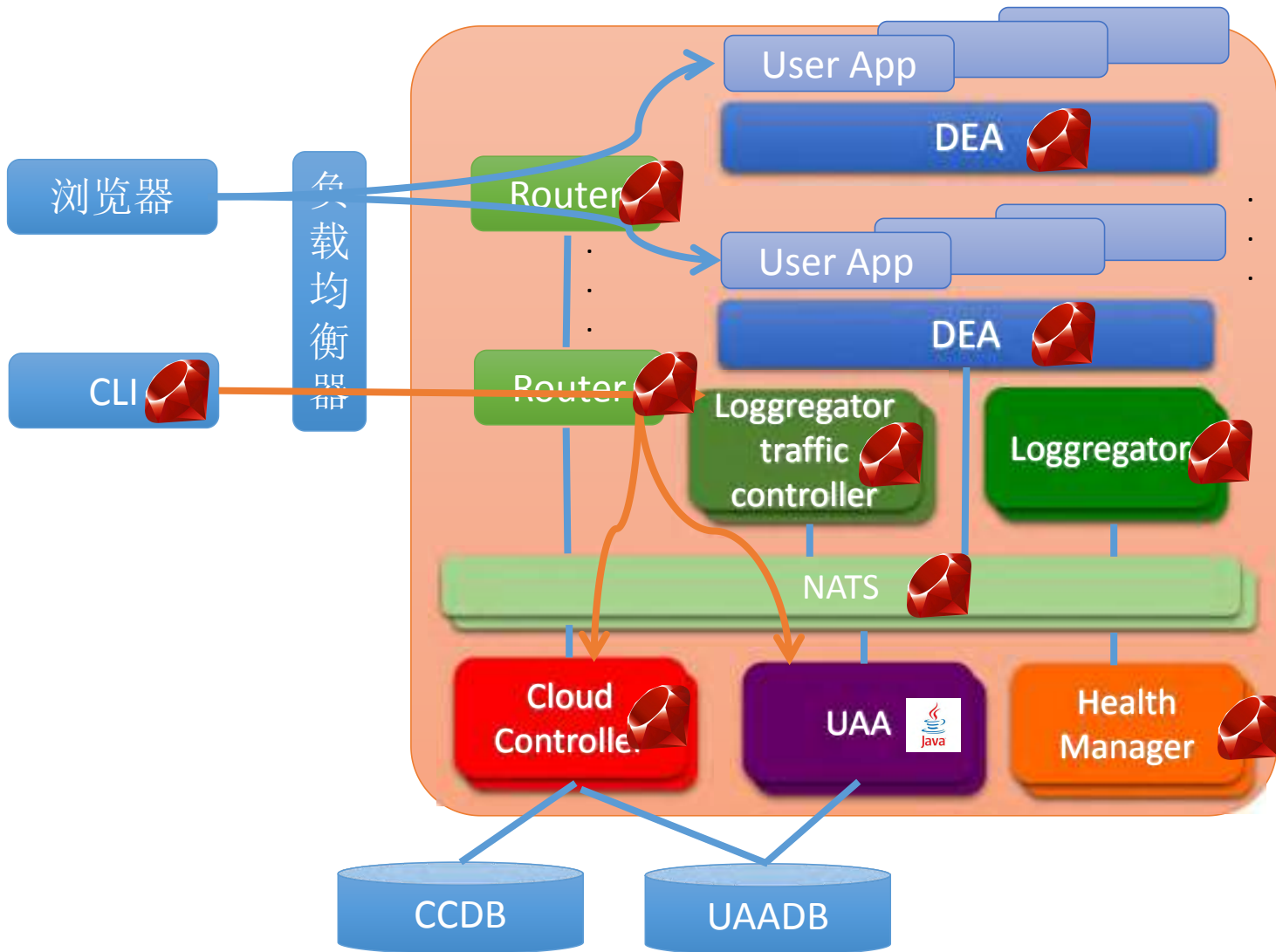
# 微服务架构的优势——快速发布



# Cloud Foundry架构图

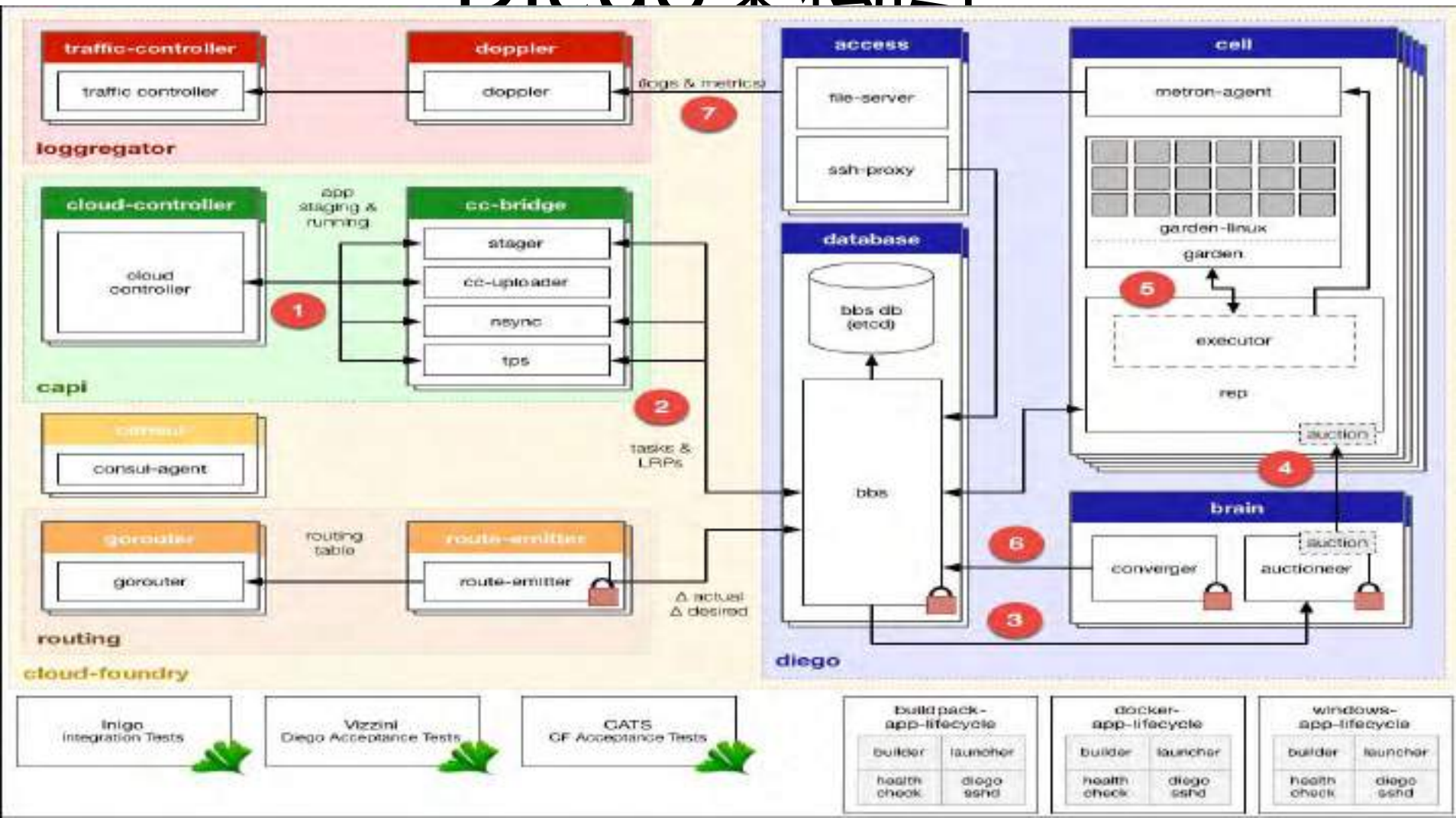




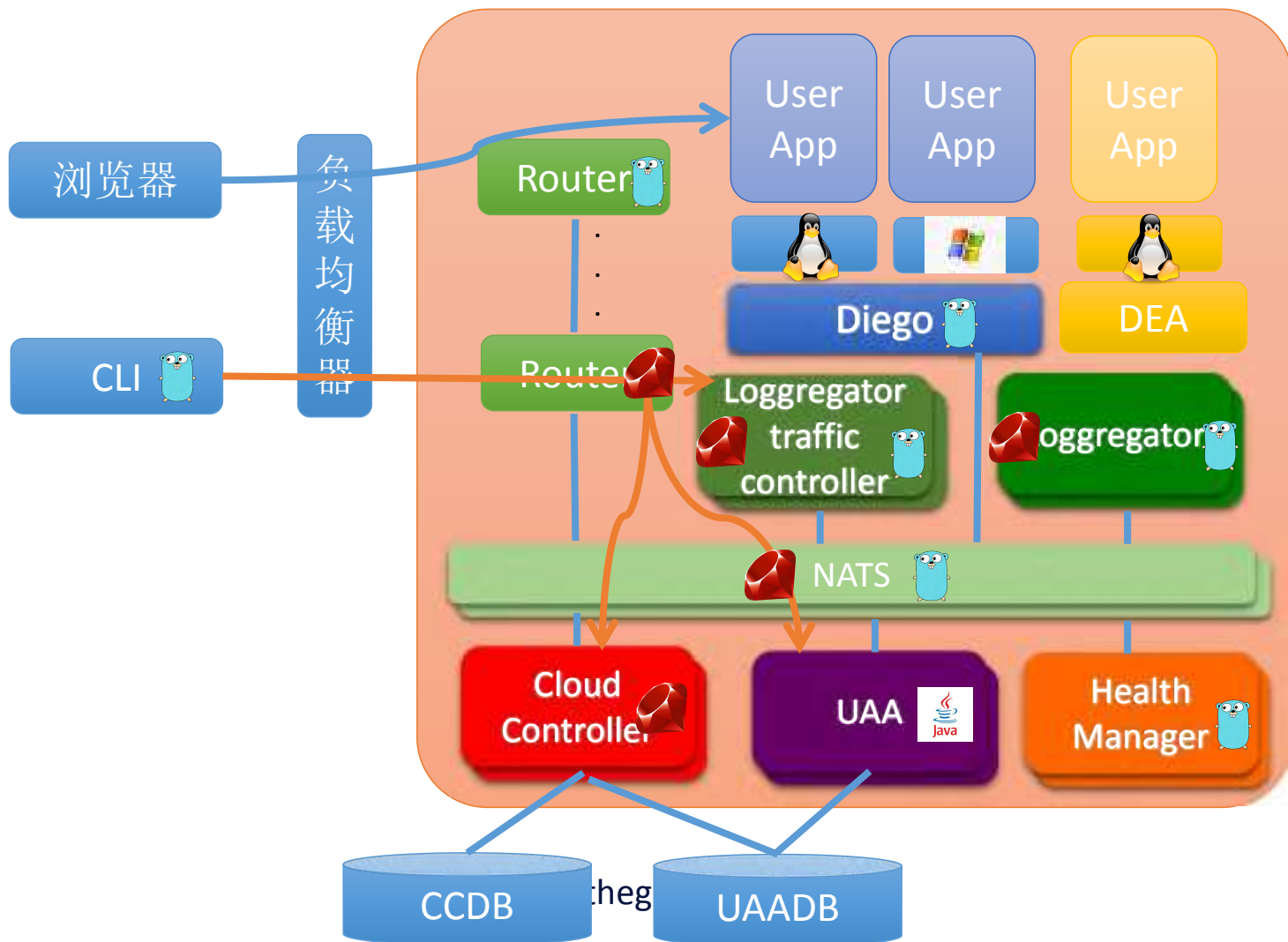




## Diego架构图



# Cloud Foundry快速发布和滚动升级



# 微服务架构的优势

快速发布

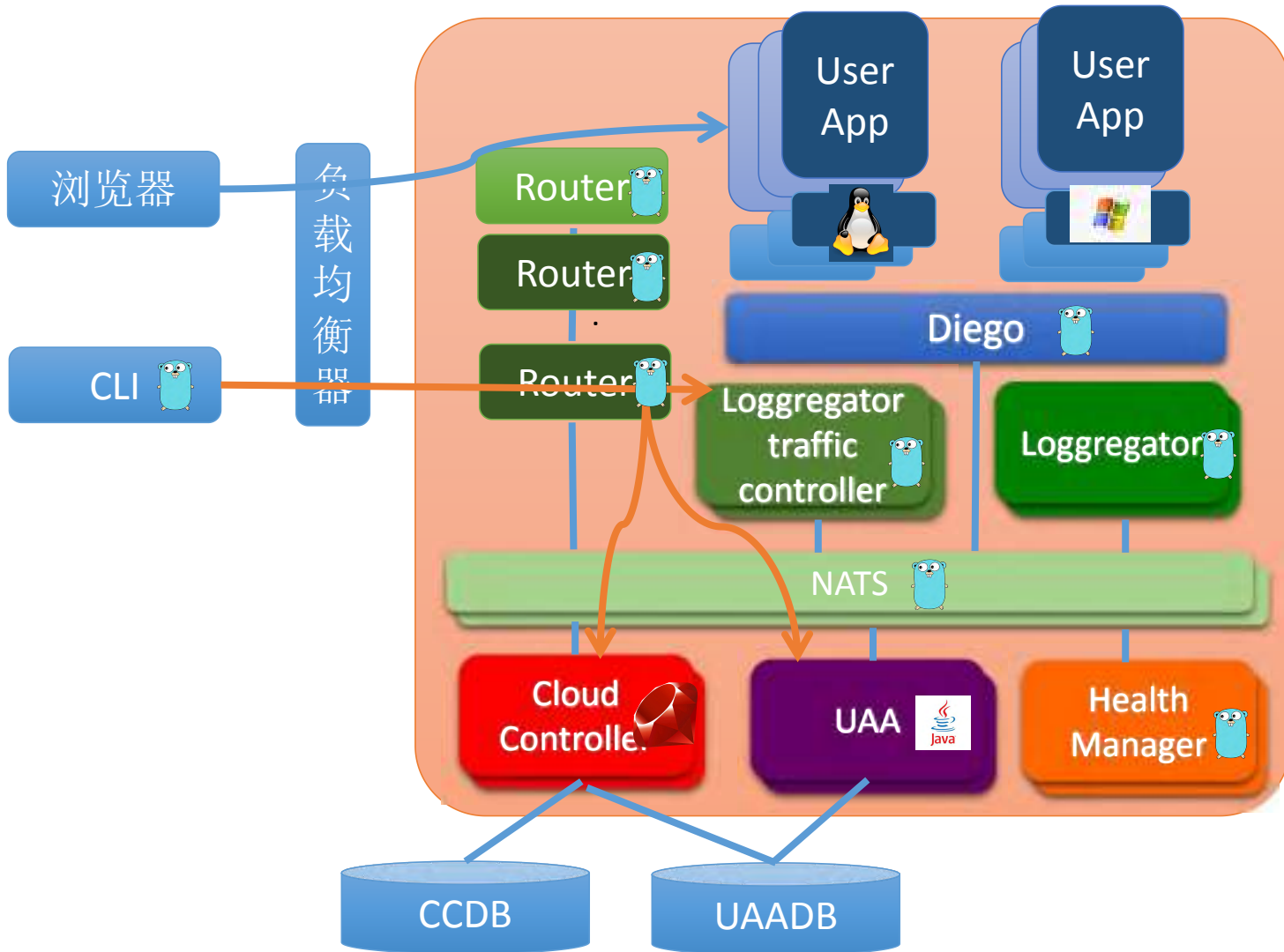
快速试错



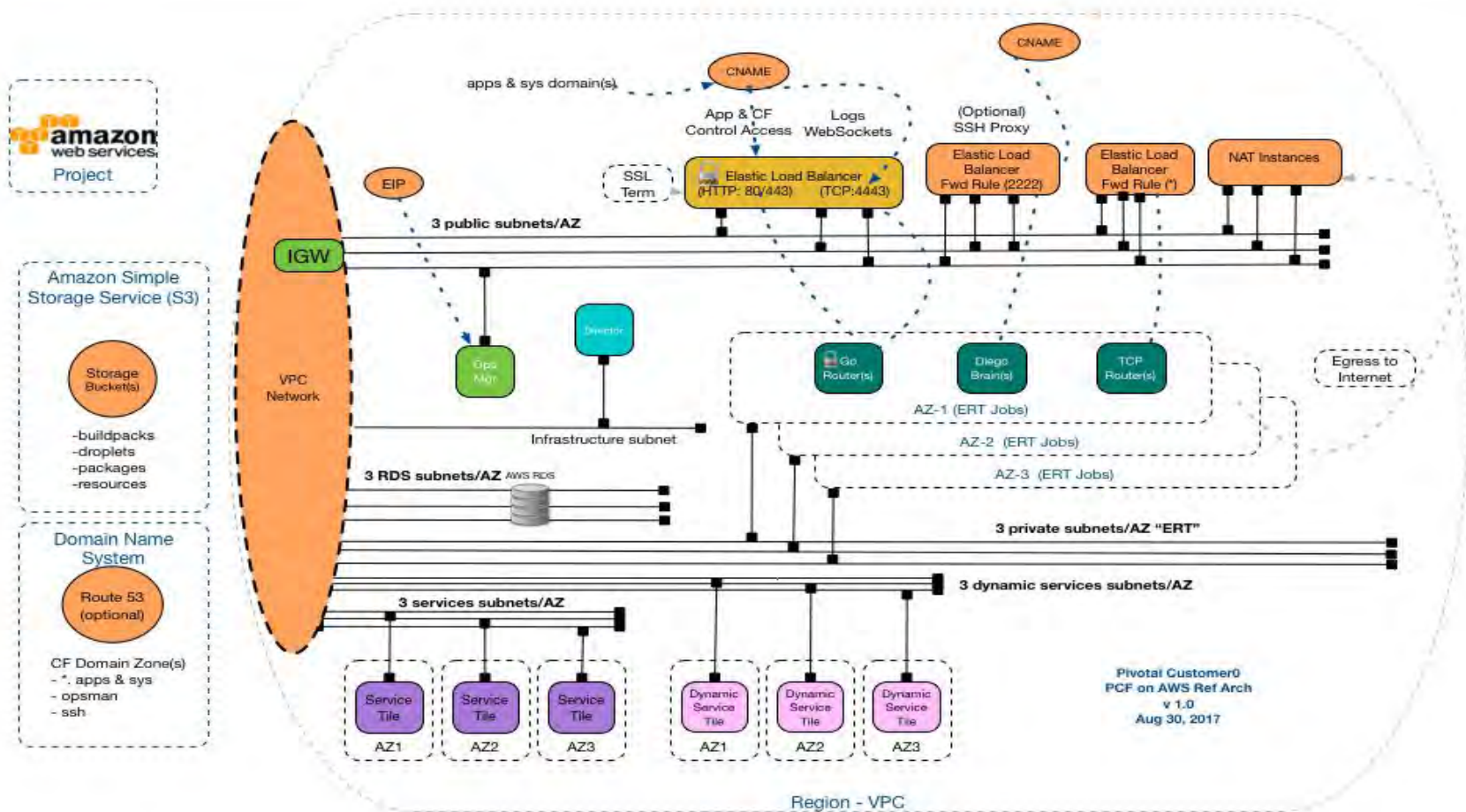
Microservices

独立扩展

快速应用  
新技术



## Pivotal Web Service 架构图



# 微服务架构的优势

快速发布

快速试错

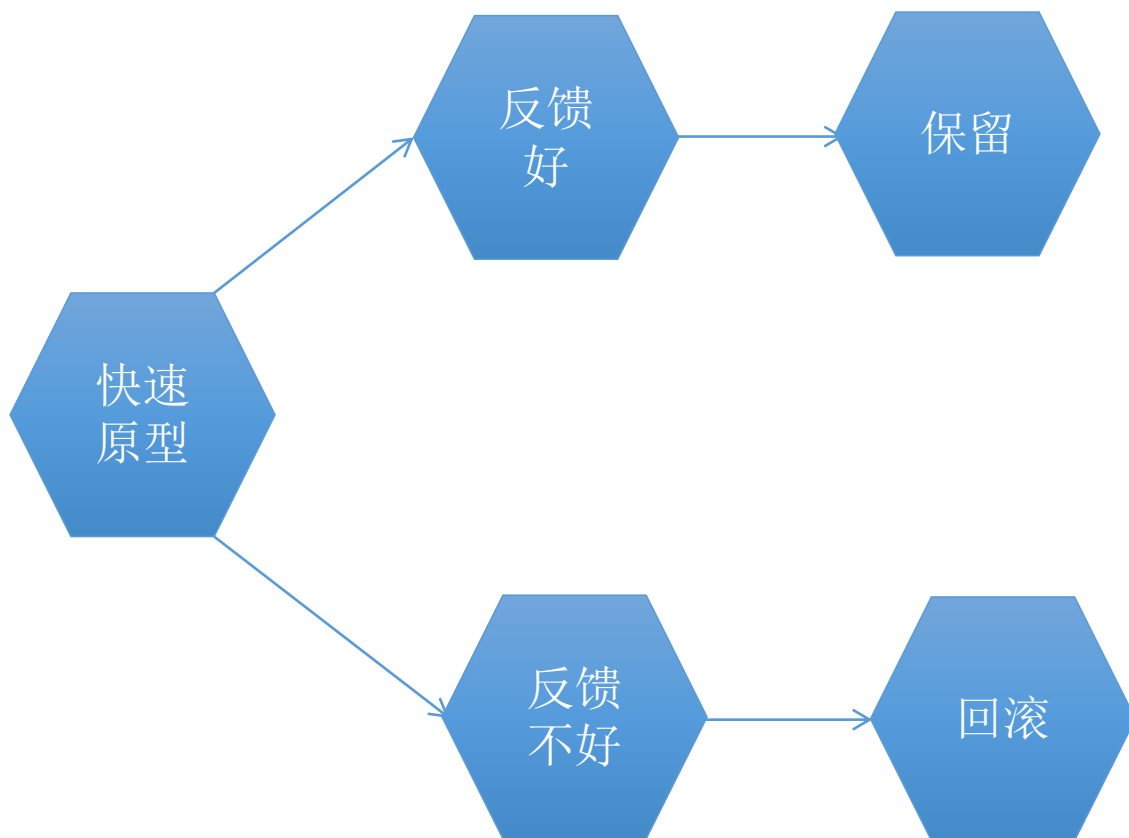


Microservices

独立扩展

快速应用  
新技术

# 微服务架构的优势——快速试错



# 微服务架构的优势

快速发布

快速试错

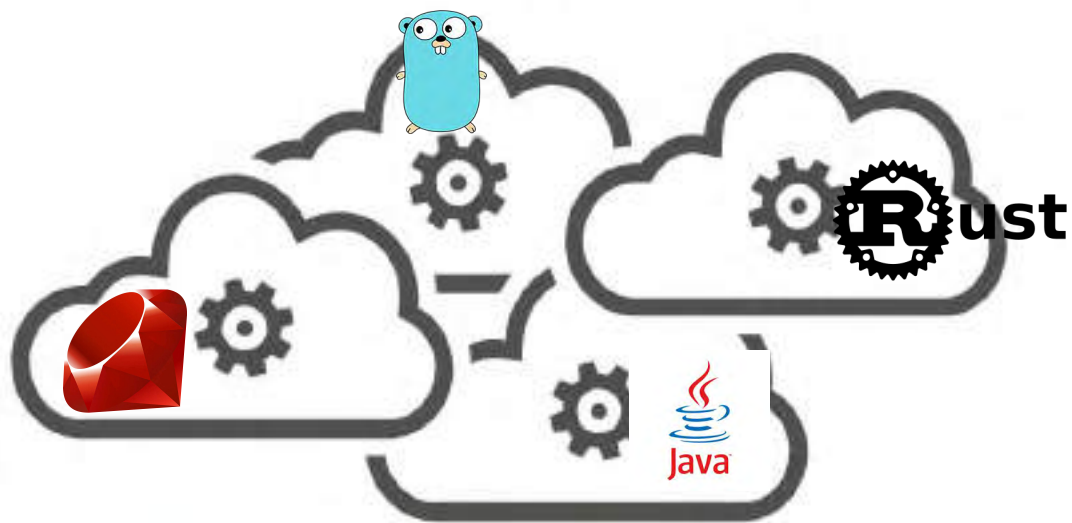


Microservices

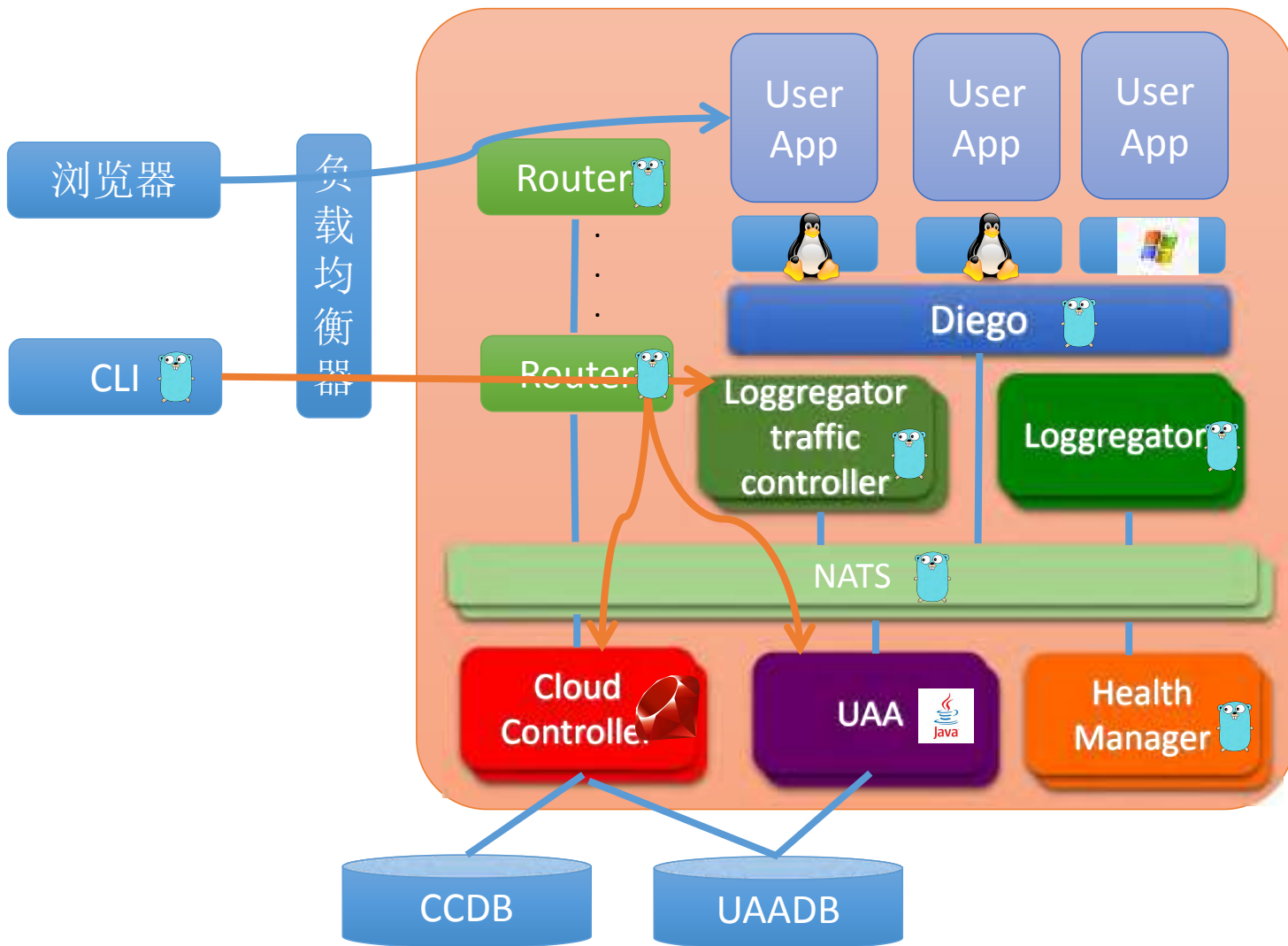
独立扩展

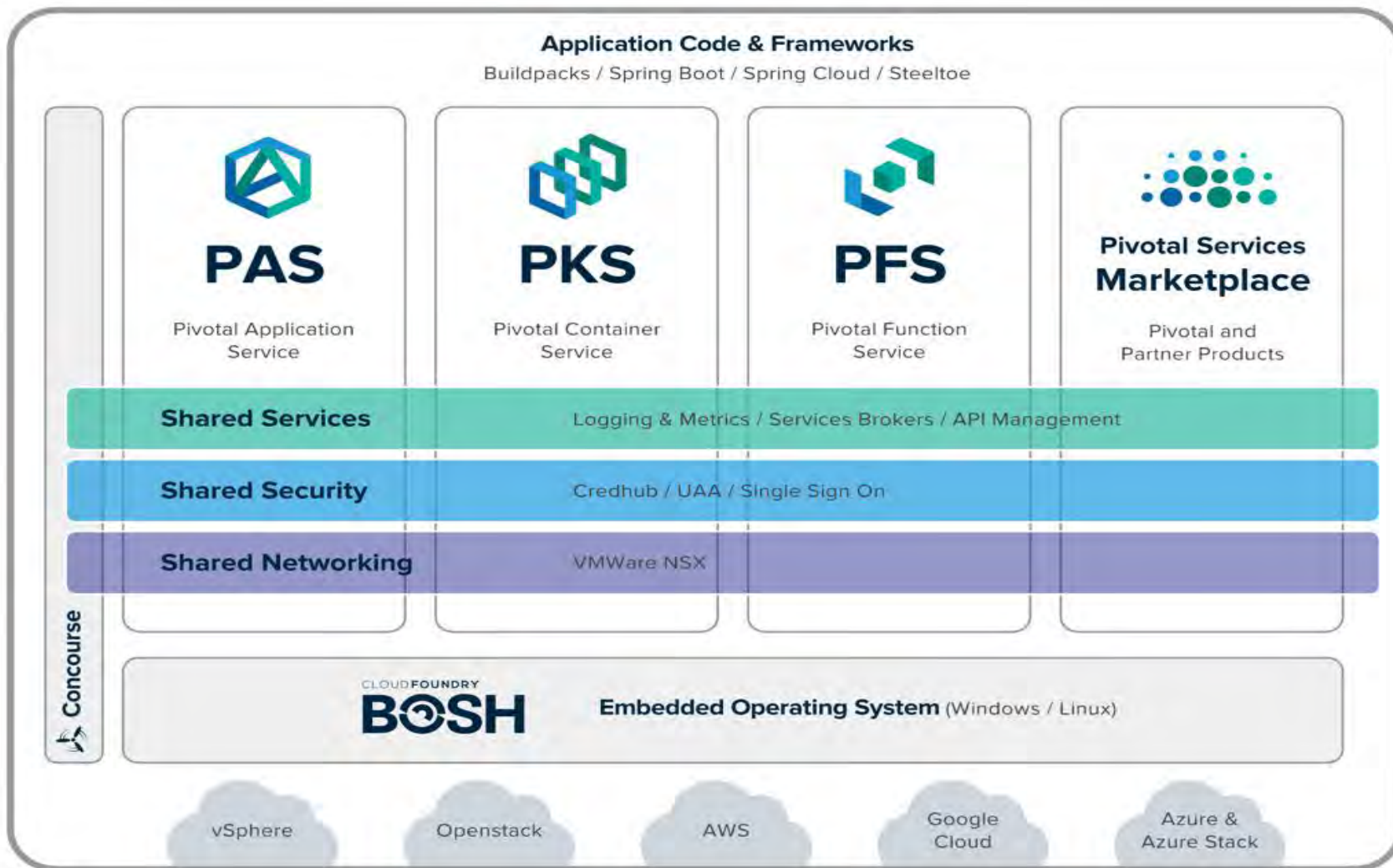
快速应用  
新技术





# Microservices





# 微服务架构的劣势以及应对

系统复杂

难于调试



Microservices

性能损失

难于监控

# 微服务架构的劣势以及应对

系统复杂

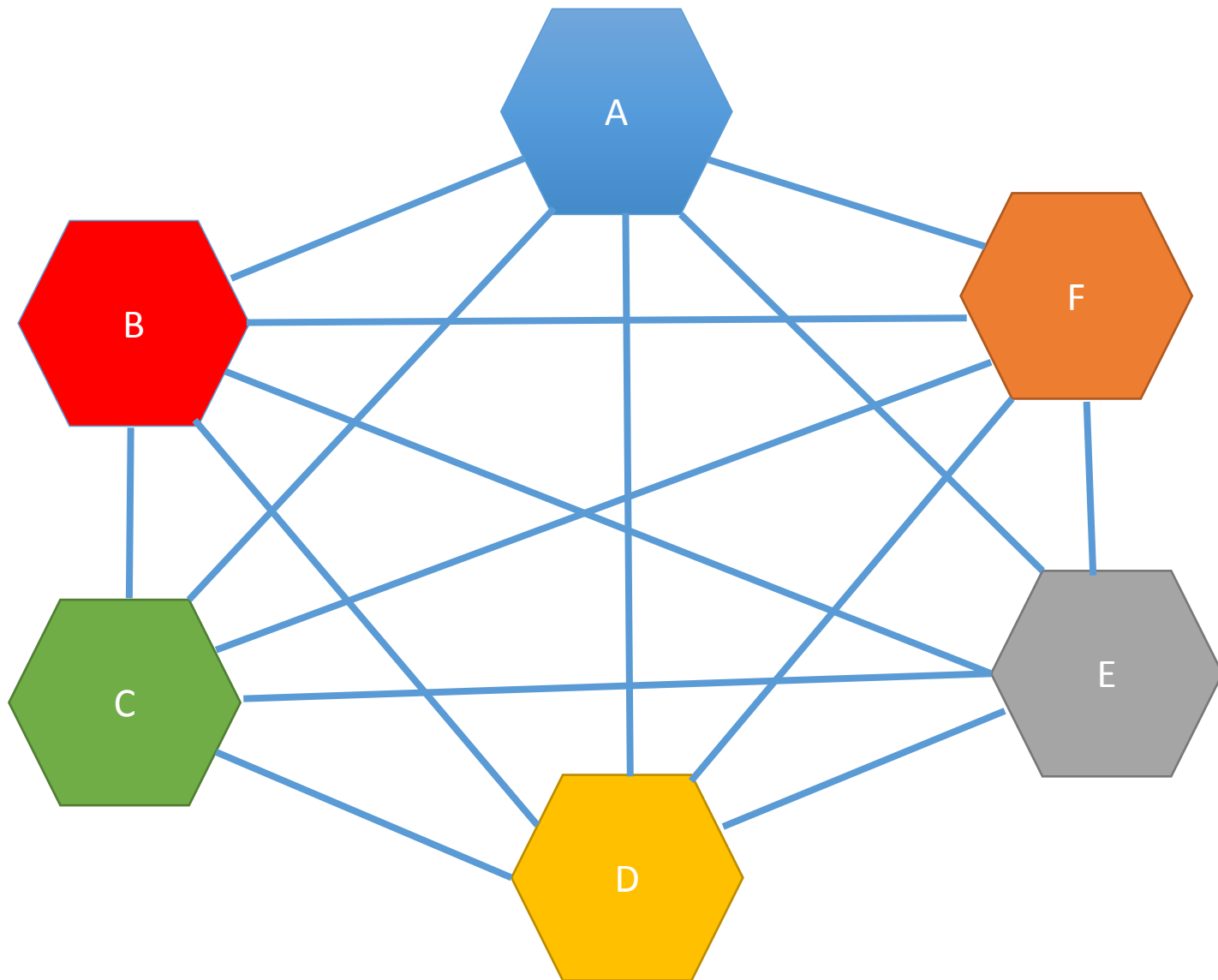
难于调试



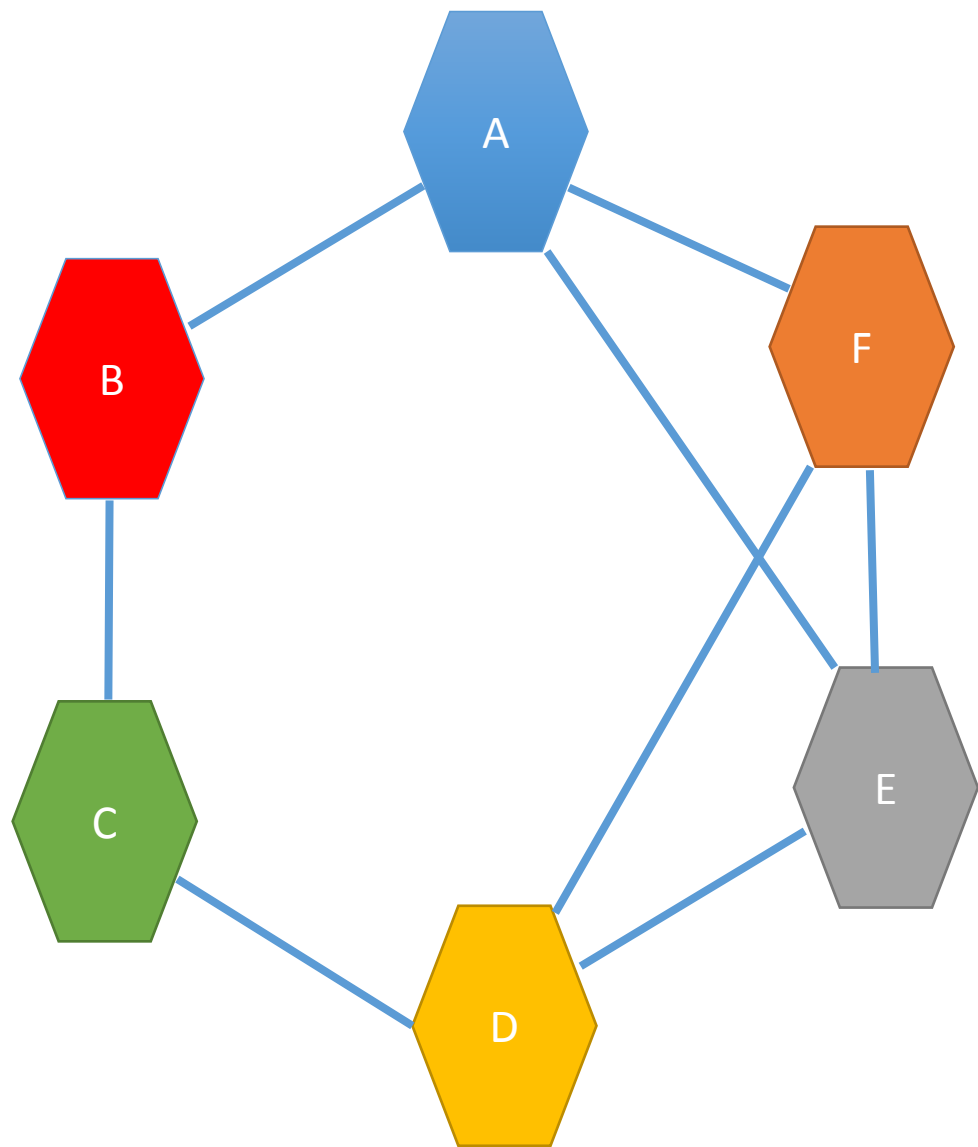
Microservices

性能损失

难于监控



- 基于业务领域建立模型
- 减少服务之间的依赖关系



# 微服务架构的劣势以及应对

系统复杂

难于调试



Microservices

性能损失

难于监控



# 分布式系统的代价——性能损失

- 微服务架构适用于
  - 对单机性能要求不高
  - 通过水平扩展提高性能和吞吐量
  - 例如：Web
- 微服务架构不适用于
  - 对单机性能要求高
  - 难于水平扩张的
  - 例如：操作系统，数据库

# 微服务架构的劣势以及应对

系统复杂

难于调试



Microservices

性能损失

难于监控

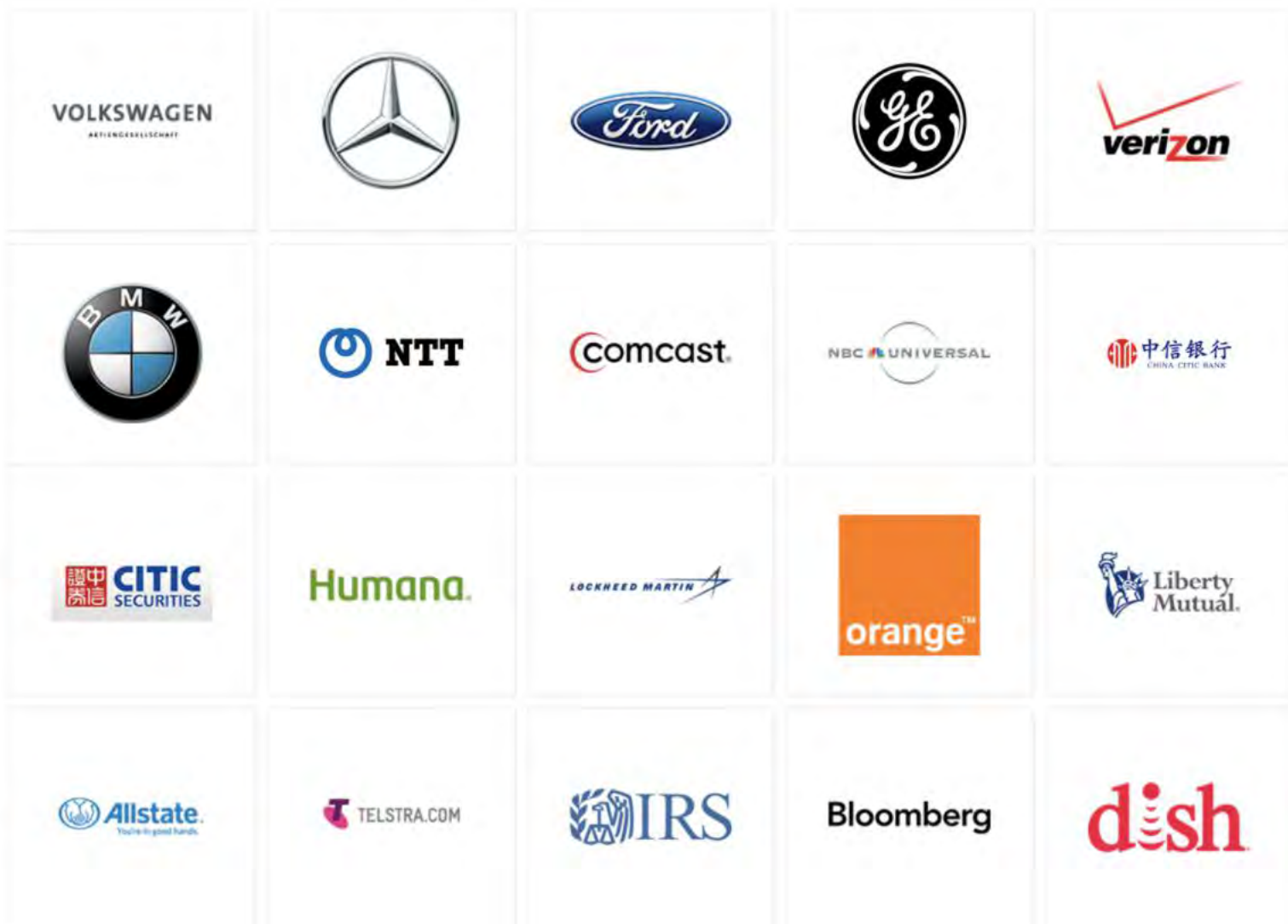
# 分布式系统的代价——难于调试监控

## • Cloud Foundry的解决方案

- 详尽的日志系统：
  - 虚拟机日志：bosh agent
  - 微服务日志：loggregator
- 专用的监控系统
  - Ops Metrics
  - PCF Metrics

## 结论

- Cloud Foundry采用了微服务架构
  - 快速发布新功能
  - 多可用区水平扩展
  - 易于引入新技术
  - 为云原生应用提供了高可用的应用平台



## 参考资料

- Cloud Foundry社区
  - <http://www.cloudfoundry.org>
- Cloud Foundry代码库
  - <http://github.com/cloudfoundry>
  - <http://github.com/cloudfoundry-incubator>
- Pivotal公司文档库
  - <http://docs.pivotal.io/pivotalcf>
- Pivotal公司Blog
  - <http://content.pivotal.io/blog>
- Pivotal中国研发中心微信公众号：pivotalrnd

GIAC | 全球互联网架构大会  
GLOBAL INTERNET ARCHITECTURE CONFERENCE

GIAC

全球互联网架构大会

GLOBAL INTERNET ARCHITECTURE CONFERENCE



扫码关注GIAC公众号

2017.thegiac.com