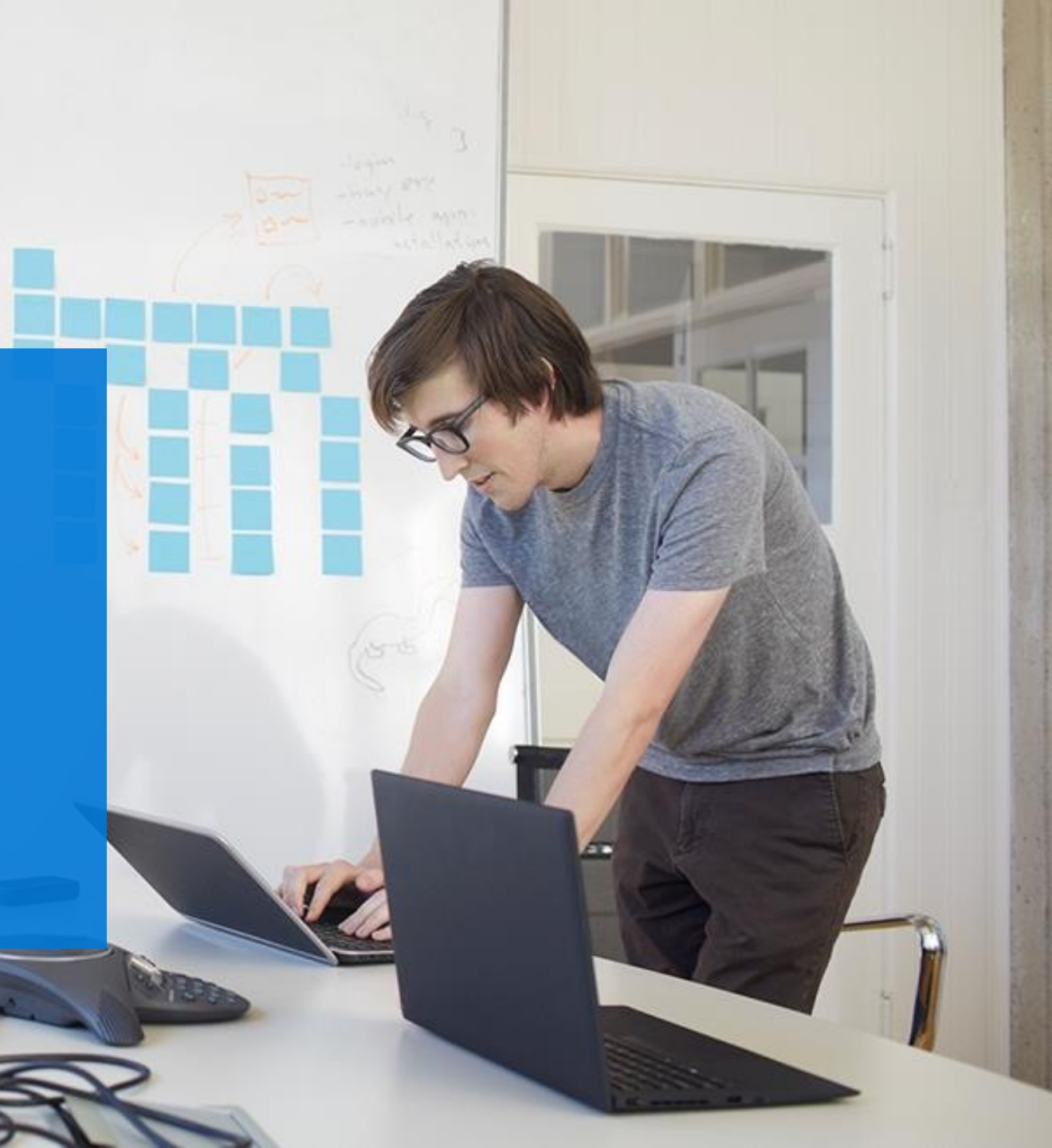




# 微软Azure云助力微服务

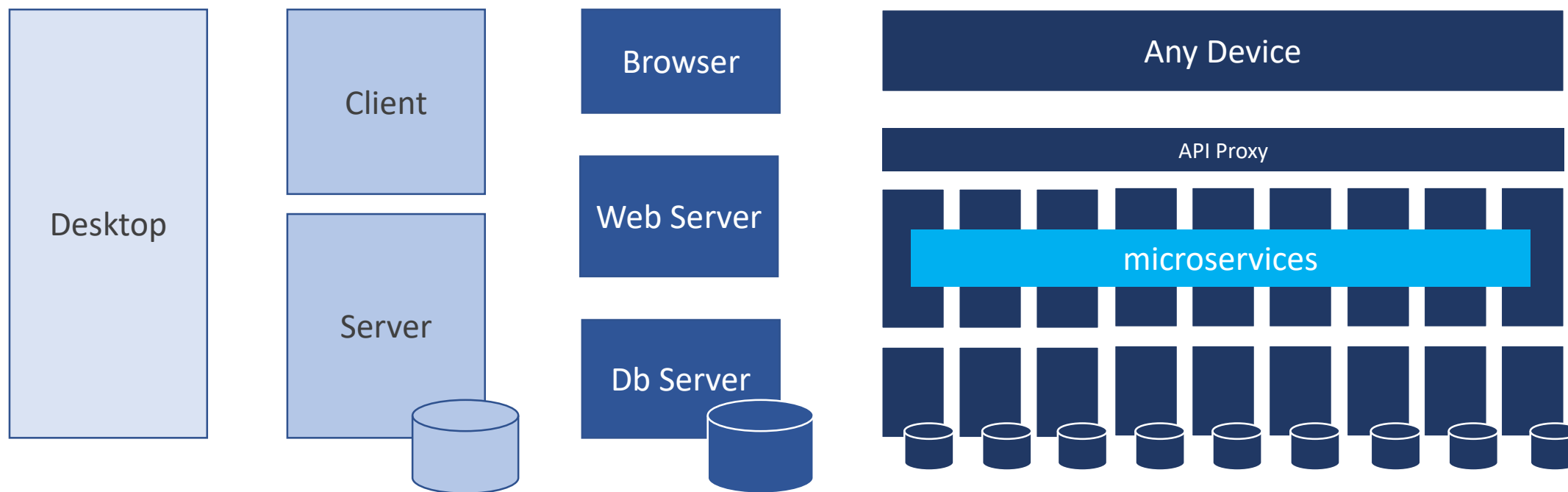
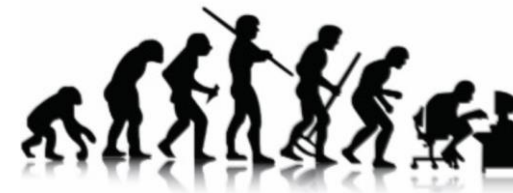
赵文婧  
微软大中华区创新技术合作事业部(CSE)  
技术顾问



# 概览

- 微服务的演变过程
- 容器与微服务
- Azure容器服务生态
- 持续集成和持续交付

# 软件架构的演变

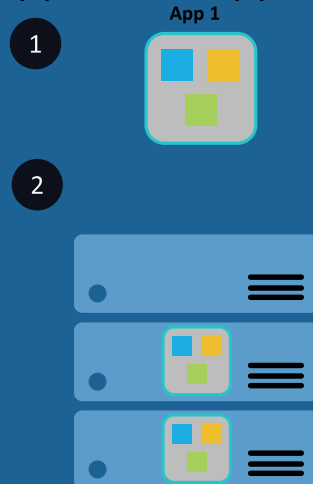


From Monolithic to Microservice

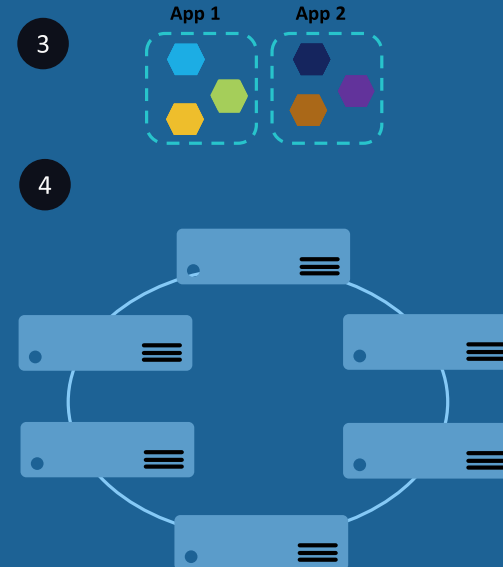
# 集群部署：单体应用 VS 微服务应用

- 分开编译和部署
- 小的独立执行的任务
- 集成时调用published API
- 细粒度松耦合的应用程序

Monolithic application approach



Containerized application approach

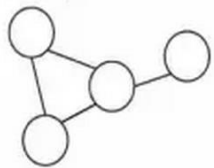


# 微服务是什么？

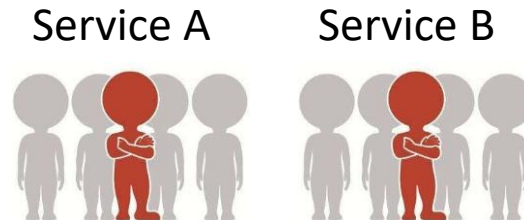
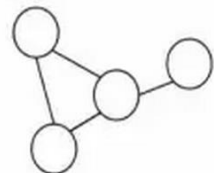
## “小而专业”

conway's law

new system:



organization:



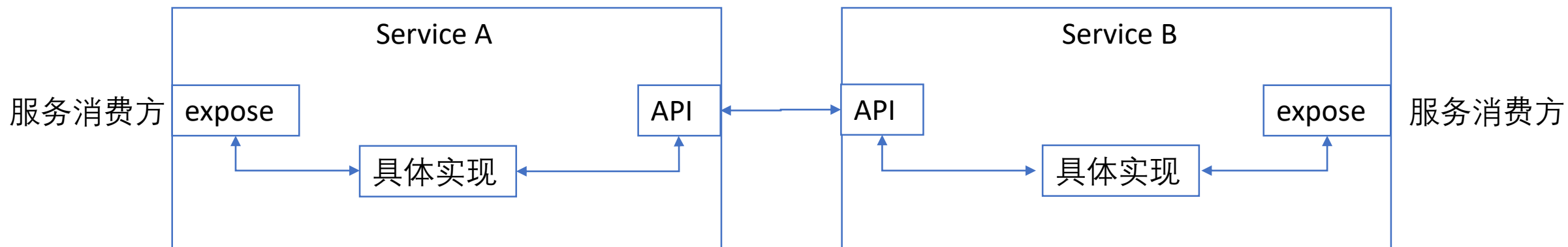
Service A <-> Function A

Service B <-> Function B

Service C <-> Function C

.....

## “独立自主”



# 微服务带来什么好处

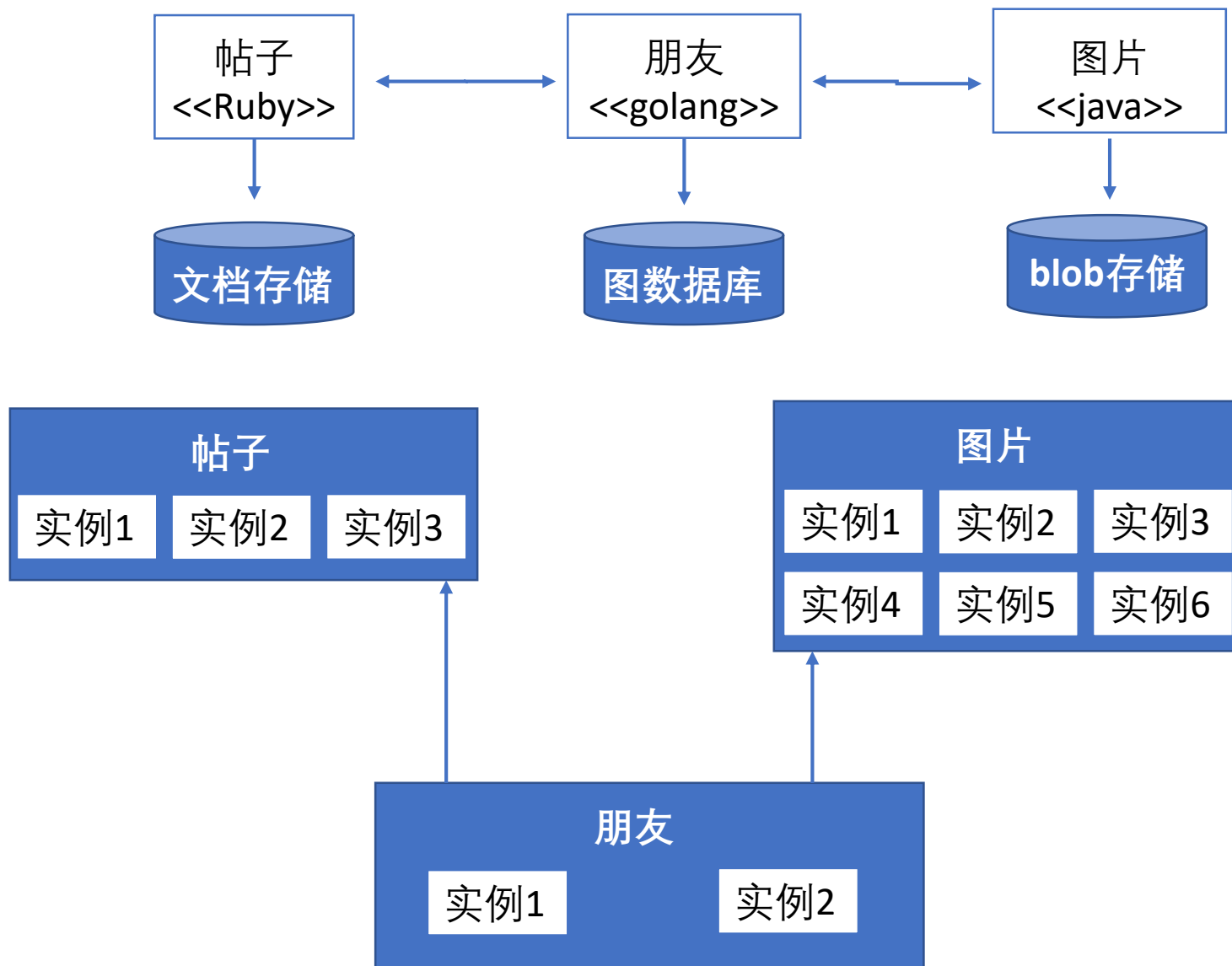
技术异构

容错

扩展

简化部署

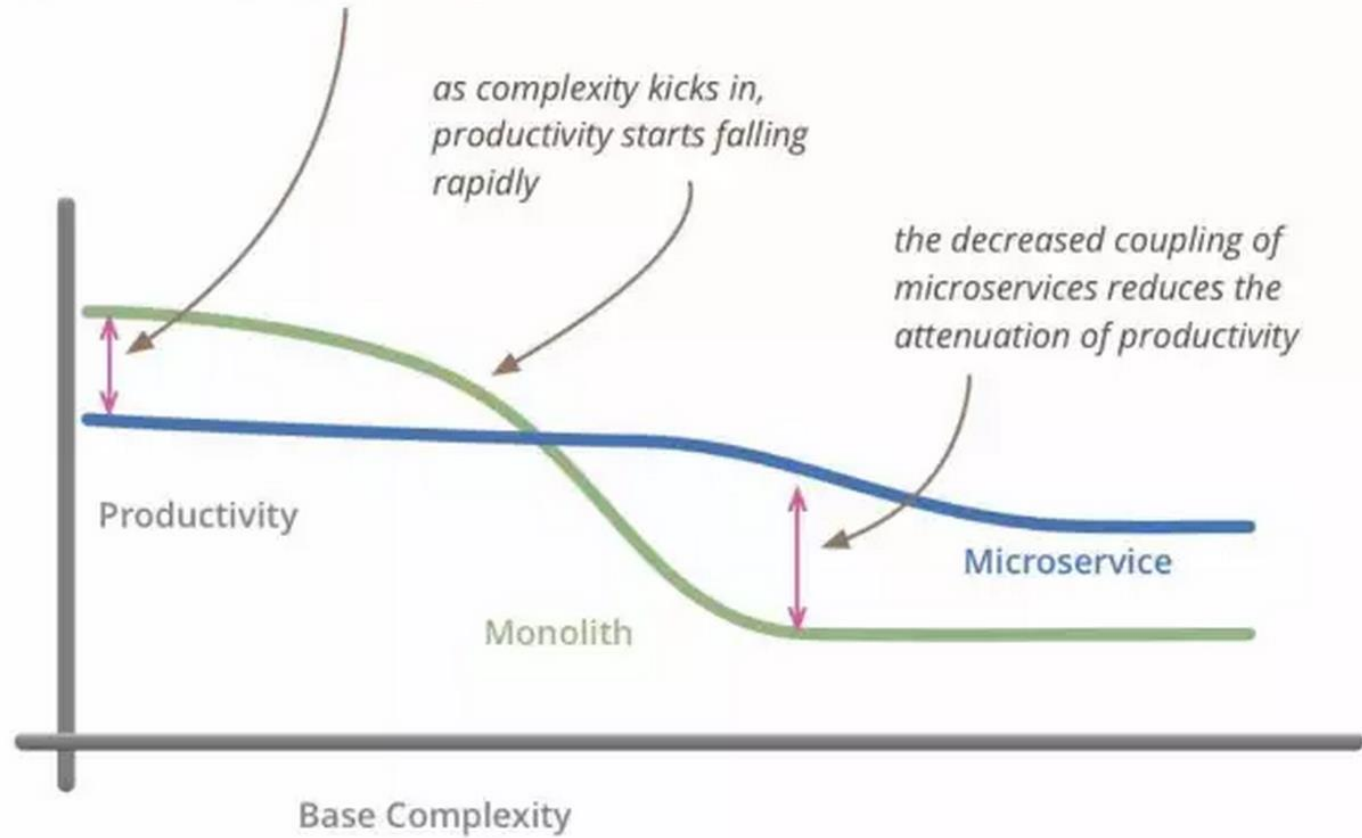
高可控性



# 没有银弹

*for less-complex systems, the extra baggage required to manage microservices reduces productivity*

什么时候应该使用微服务？



*but remember the skill of the team will outweigh any monolith/microservice choice*

# 容器和微服务-完美的结合

## Microservices and containers?

微服务：一种软件技术架构

容器：可以在微服务实现过程中的实现方式

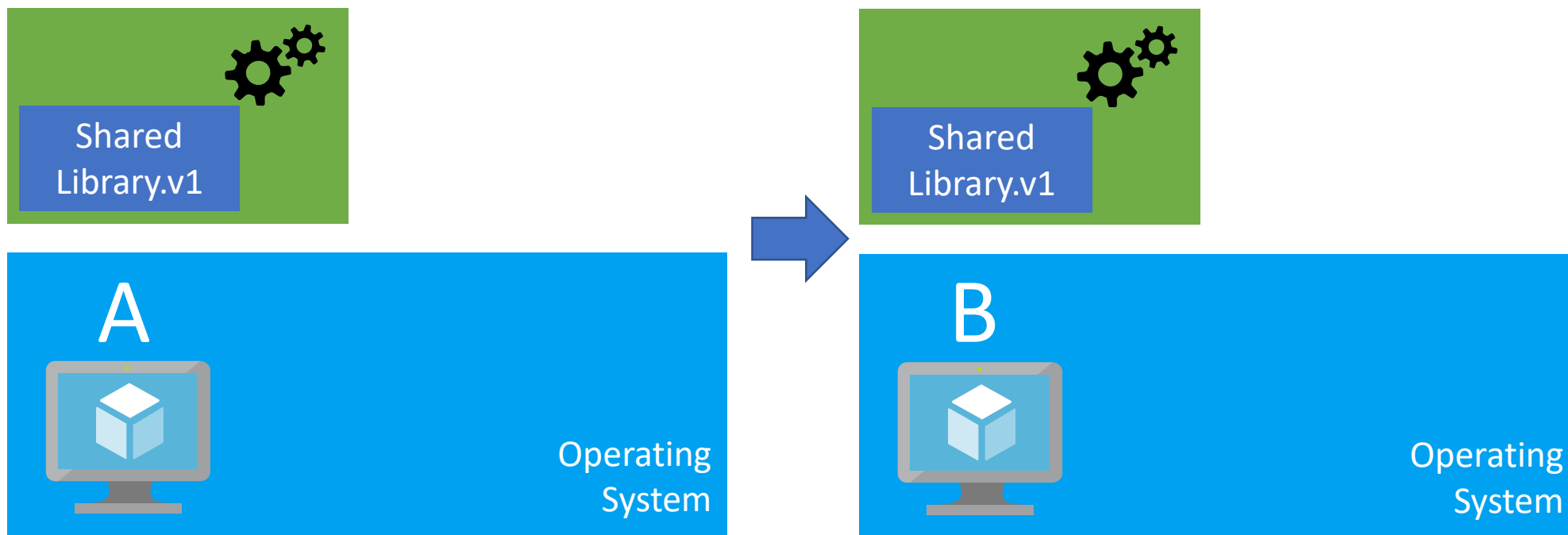
通常，每一个服务使用一个容器

容器	小	隔离性	快且密集	绿色部署
微服务	小	独立	分布式特性	快速方便的部署



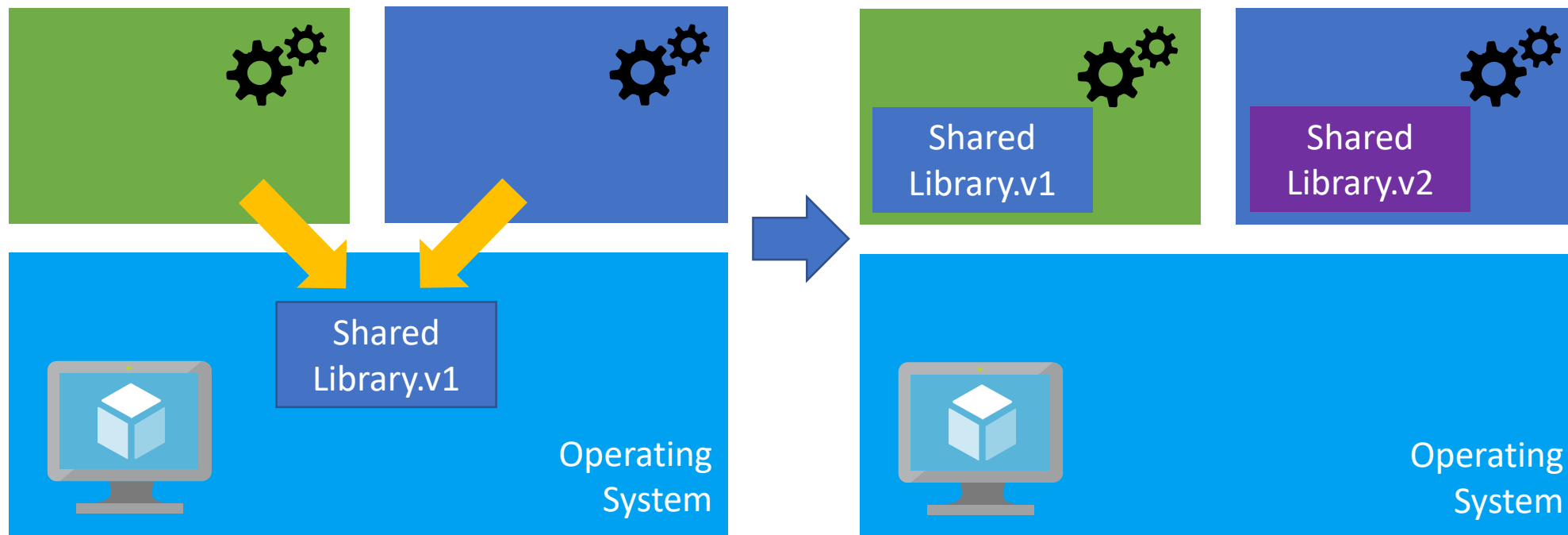
# 容器特性

- 应用程序打包和部署的机制

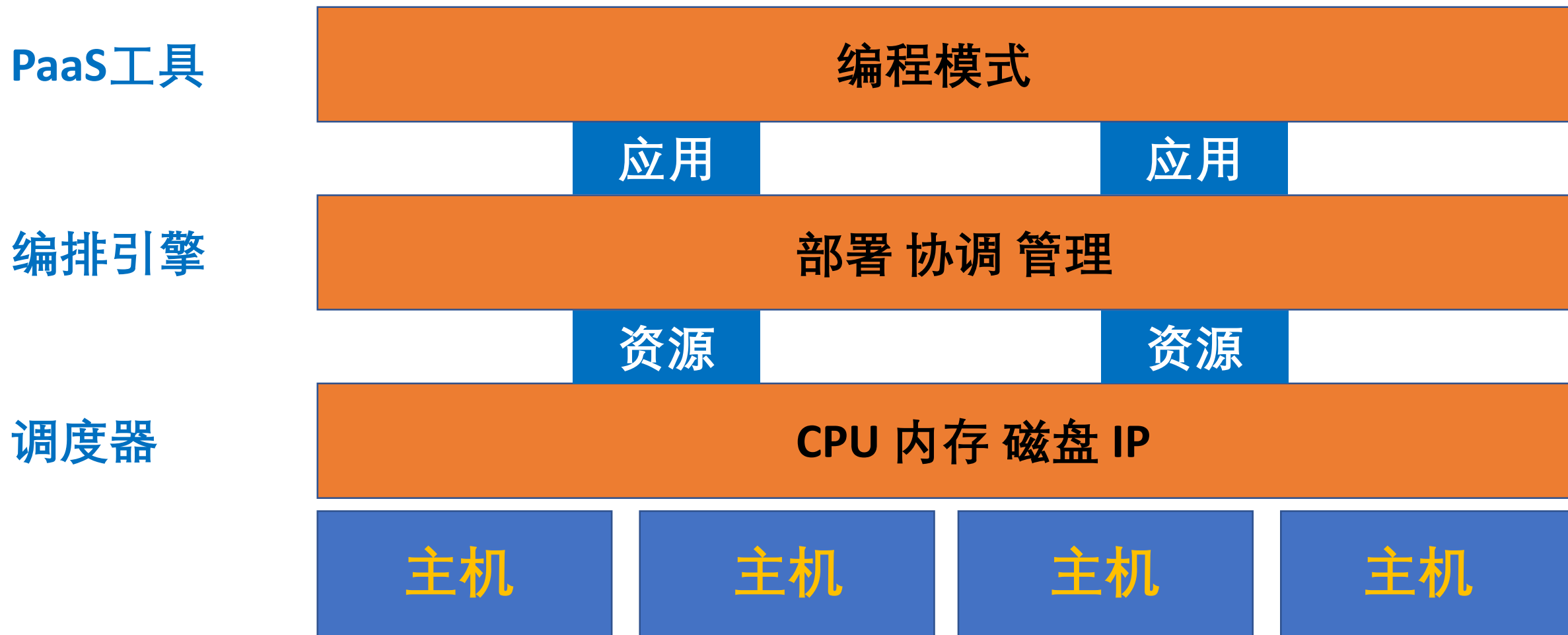


# 容器特性

- 隔离性环境，清晰的边界



# 云端微服务应用部署架构解析



# 集群和编排引擎

## Cluster

- 由一组计算机节点组成，可以被看作一个单独的系统
- 由网络互联
- 用于高性能分布式计算

## Orchestrators (Schedulers)

- 在一个集群中分配任务或者分配容器
- 在服务或容器失效时重启
- 基于资源的消耗重新分配服务或者容器
- IaaS: 网络 存储 负载均衡
- PaaS: 发现 缩放 容错 监控 安全

# Azure 容器生态



Hyper-V  
Container



Windows Server  
Container



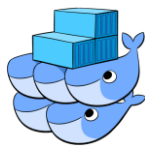
Docker



Windows  
Container



Nano  
Server



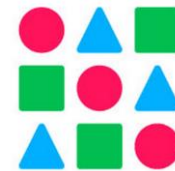
Docker Swarm



kubernetes



DC/OS



Deis



Service Fabric

容器化

编排工具

PaaS工具



Azure Container Service

Azure Container Service(AKS)



Azure Marketplace

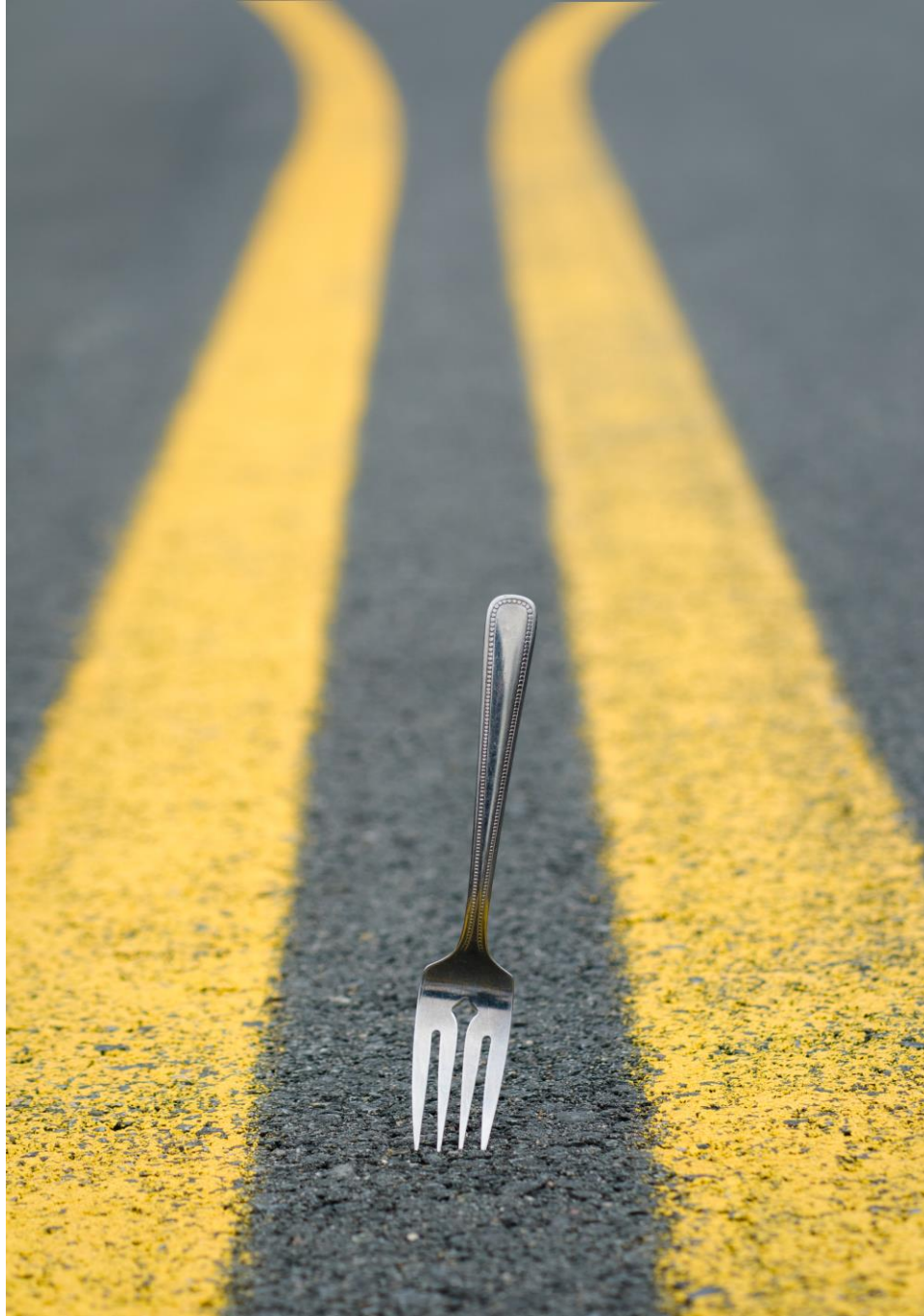


Azure Resource Manager

- Demo

- 部署 windows container 到 k8s 集群

# 持续集成持续交付 CICD



<http://aka.ms/Jenkins-on-azure>

- **Thanks**