



Azure上运行 Kubernetes

Gabe Monroy
首席项目经理，微软Azure容器服务



关于我

- 微软Azure容器服务的首席项目经理
- 代表微软参与开源组织云原生计算基金会Cloud Native Computing Foundation (CNCF)
- Deis公司 (被微软收购) 的创始人兼CTO
- Docker, Mesos, Kubernetes的早期贡献者

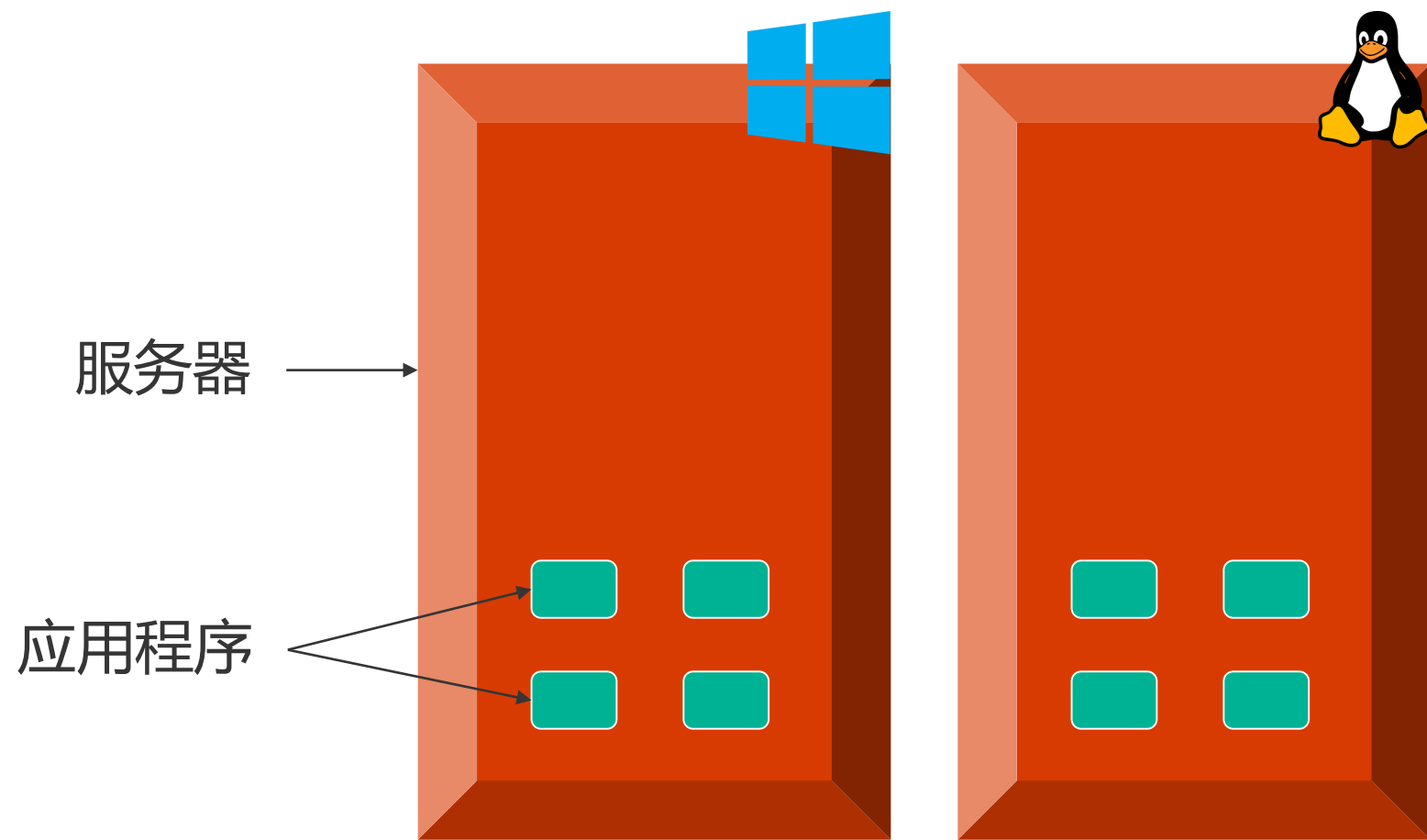
议题

- 技术概览
- 容器服务：Azure Container Service (AKS)
- 容器实例：Azure Container Instances (ACI)
- 用于Kubernetes的ACI Connector
- 未来研究方向

传统服务器



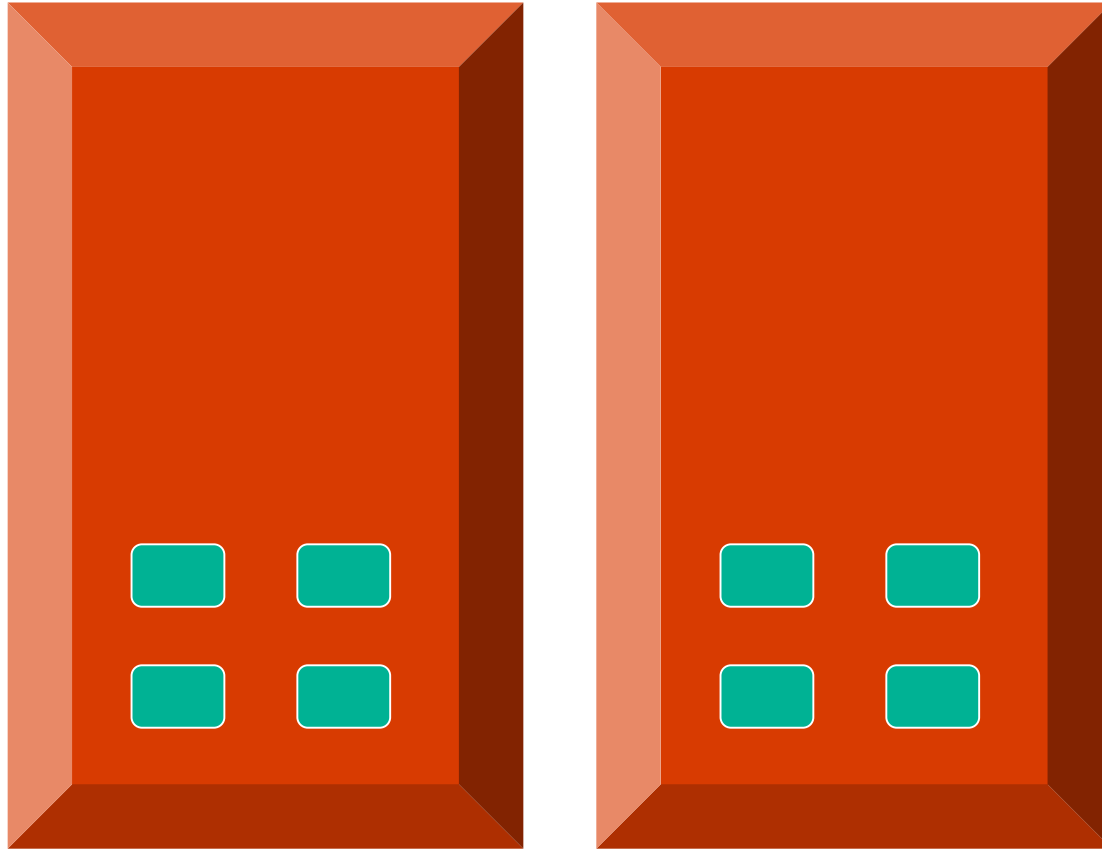
传统服务器



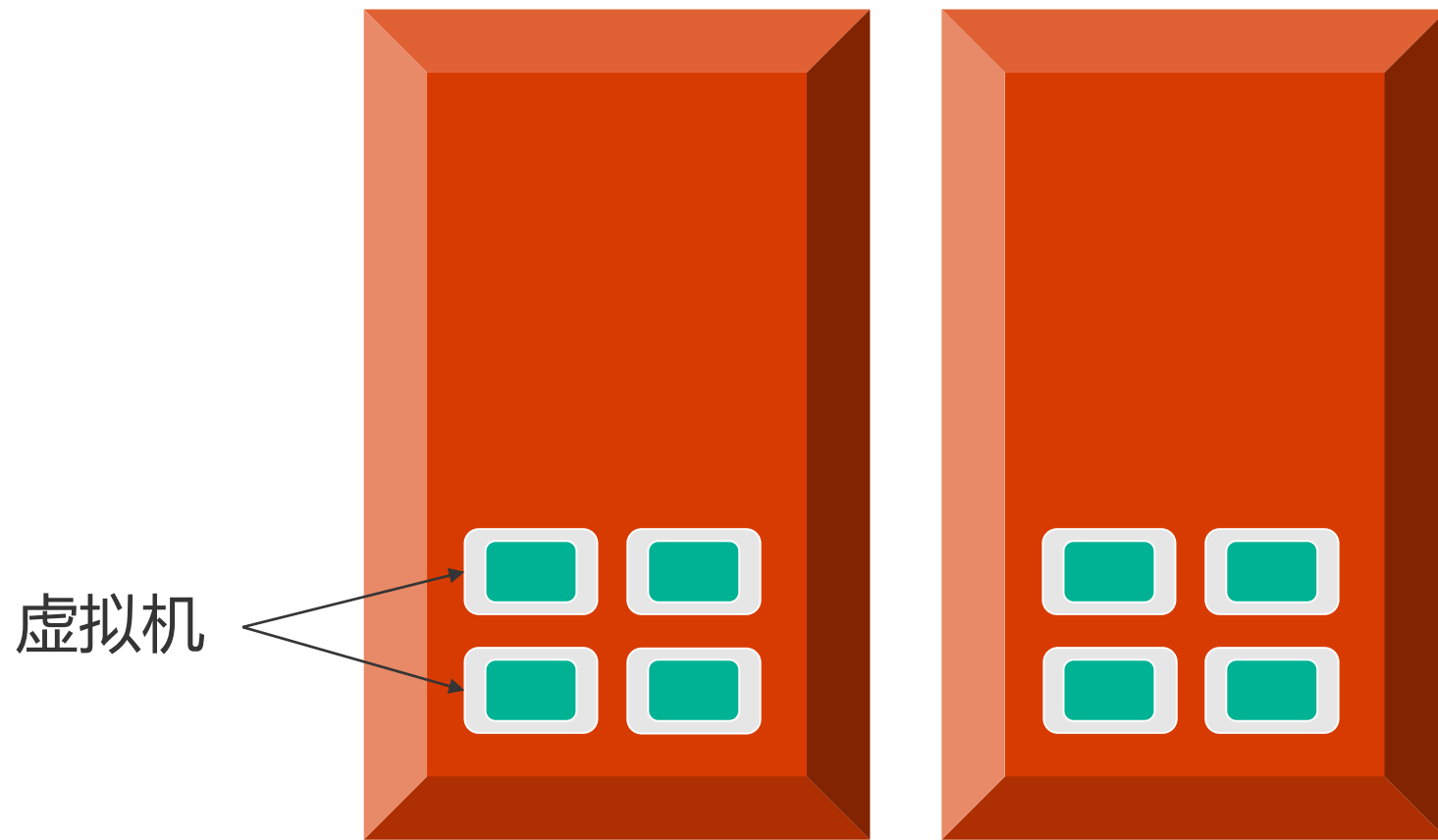
虚拟机



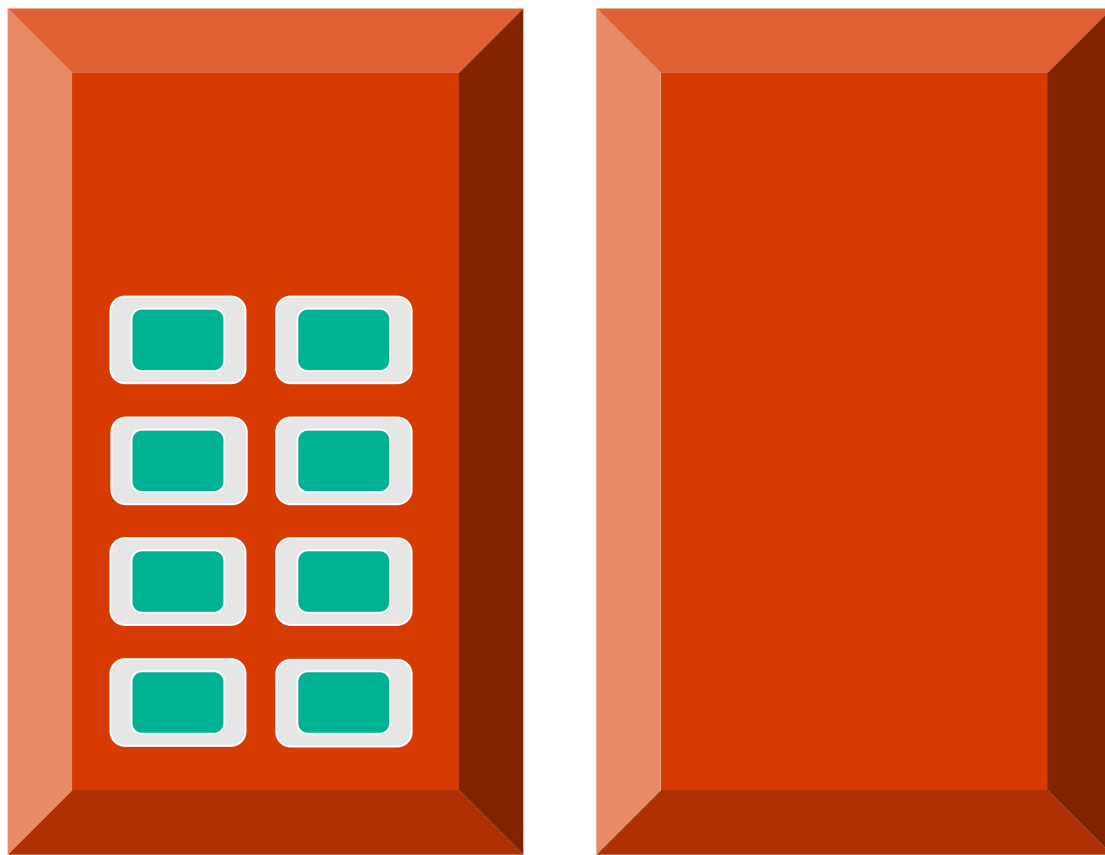
虚拟机



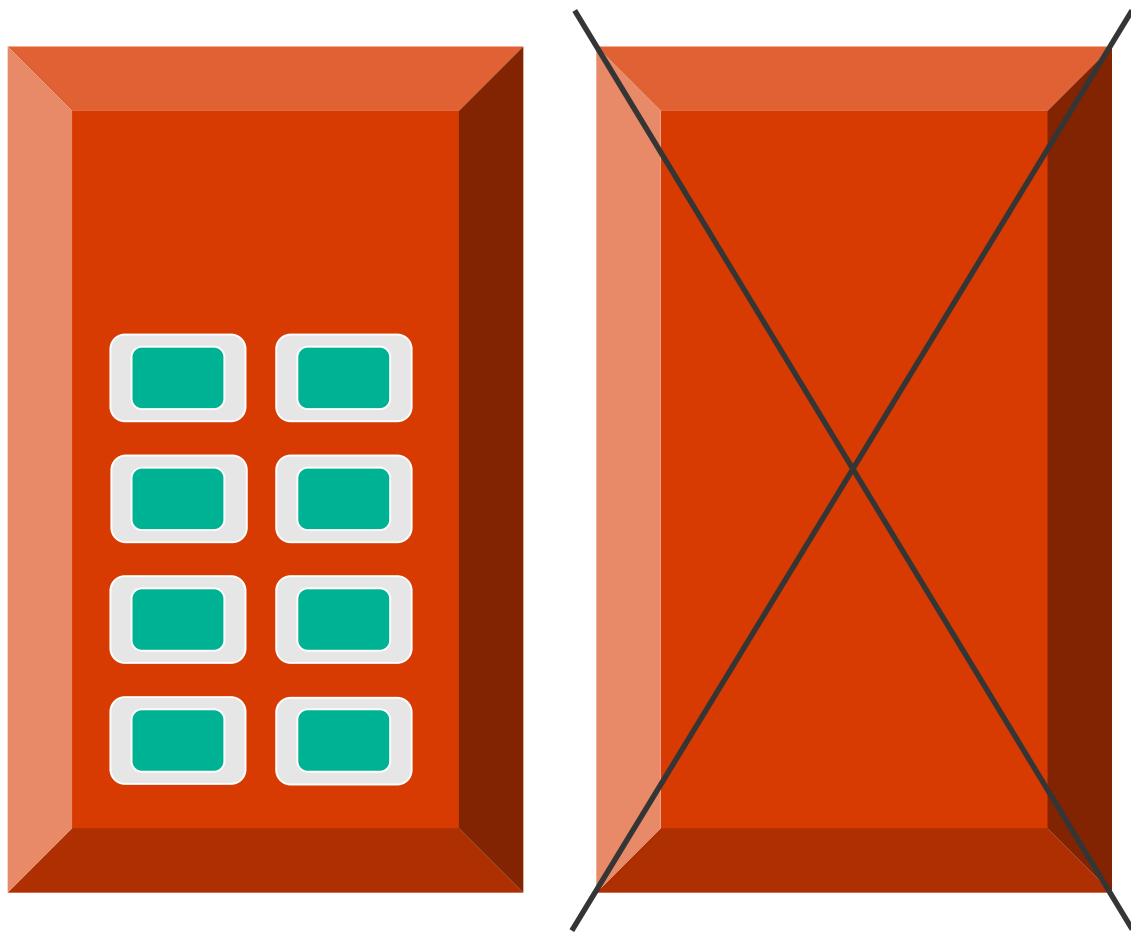
虚拟机



服务器整合



服务器整合



虚拟机蔓延(VM Sprawl)

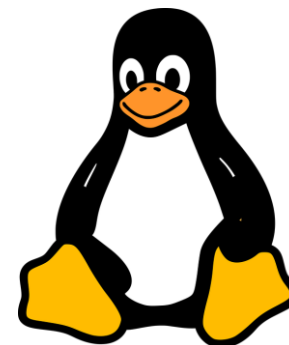


容器



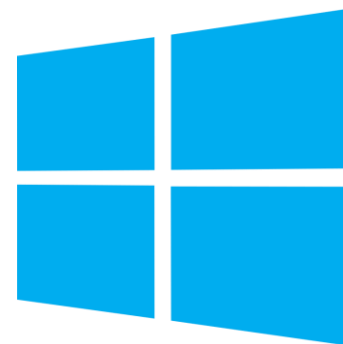
Linux容器

- 命名空间(Namespace)
- Cgroups



Windows容器

- 作业对象 (Job Object)

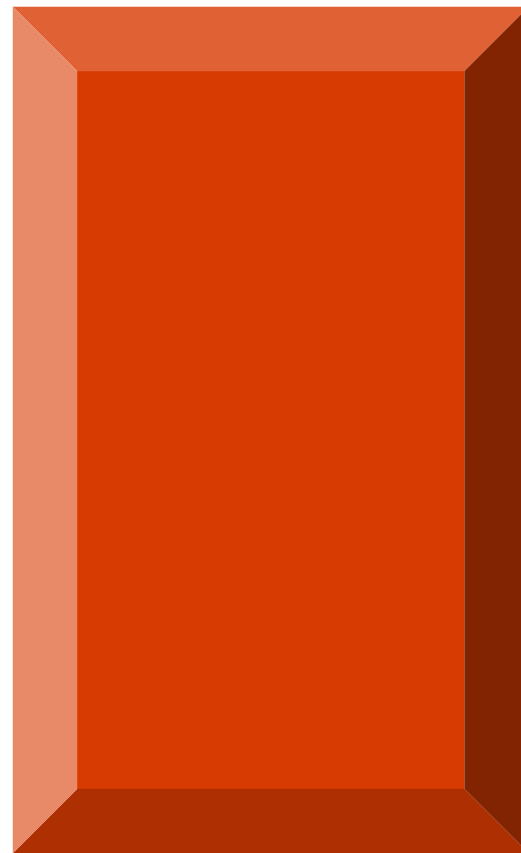
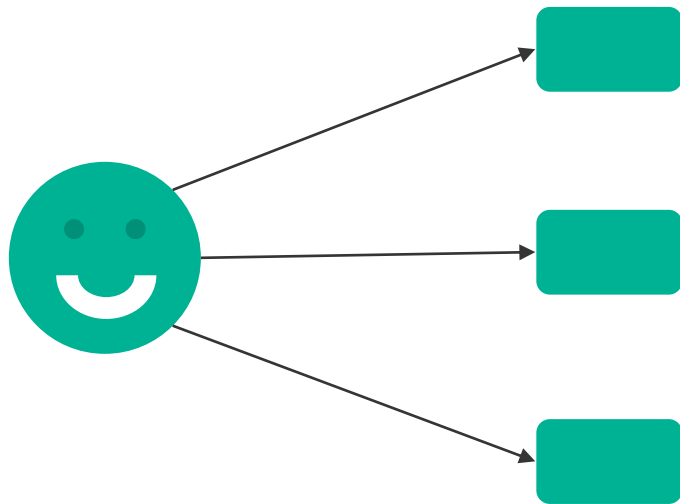


Docker

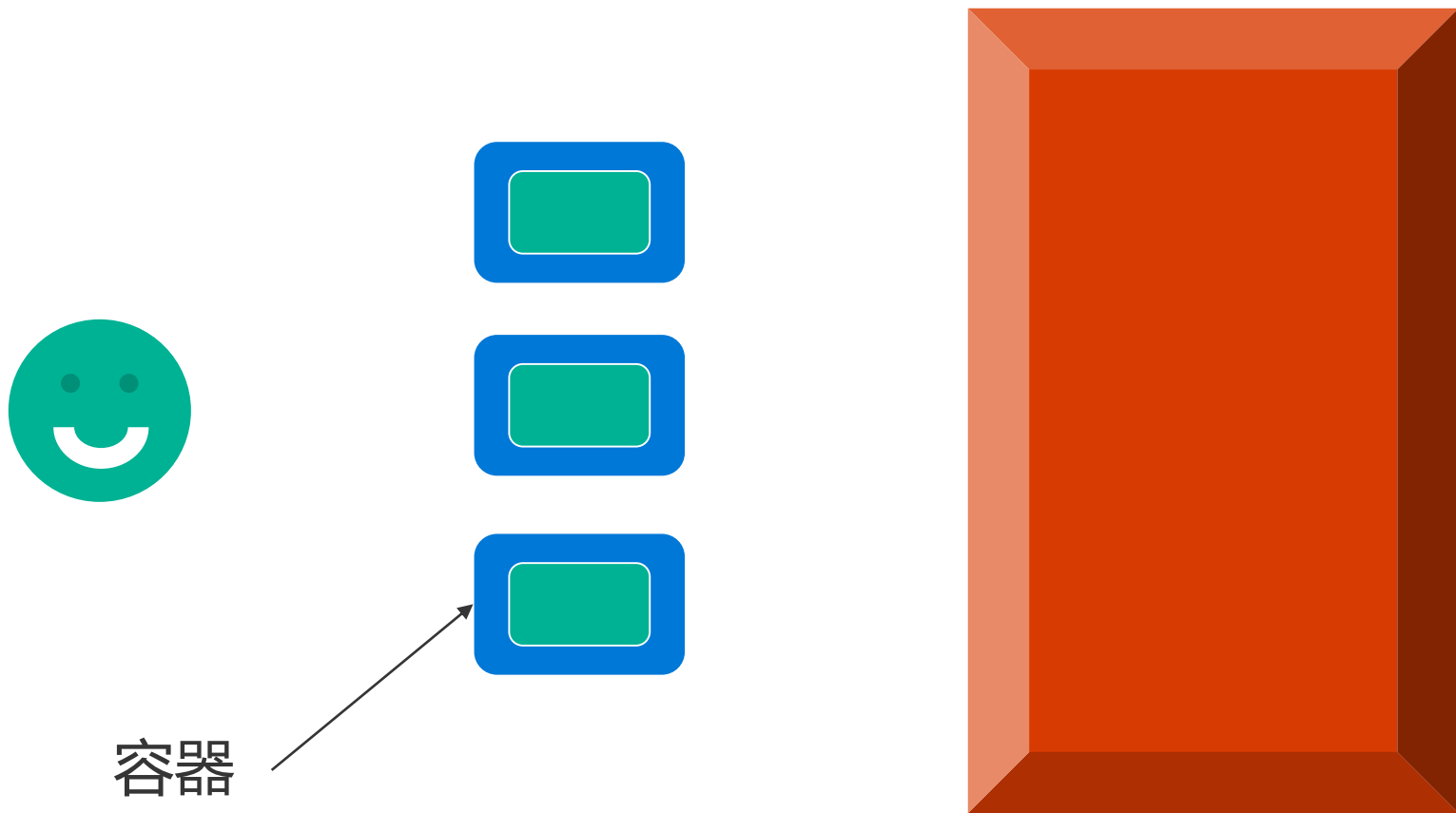
- 联合文件系统 (Union File System)
- 集中的镜像仓库 (Central Registry)
- 易用的工具集



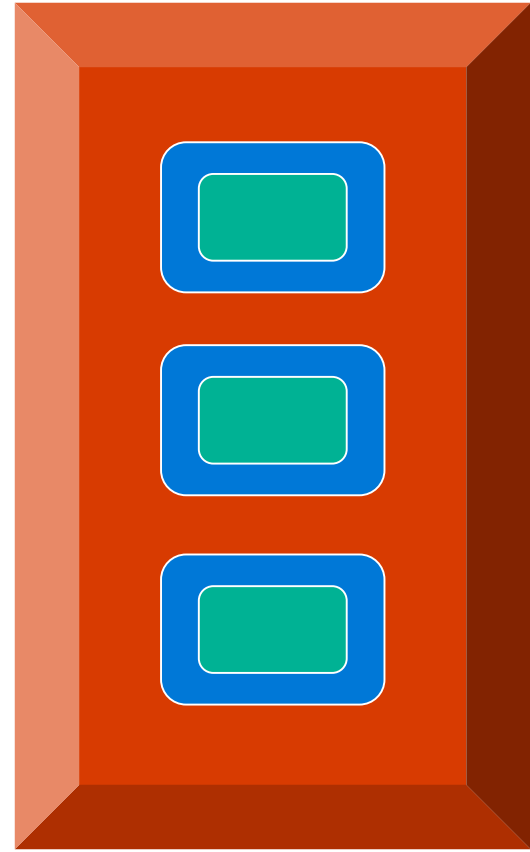
容器采用



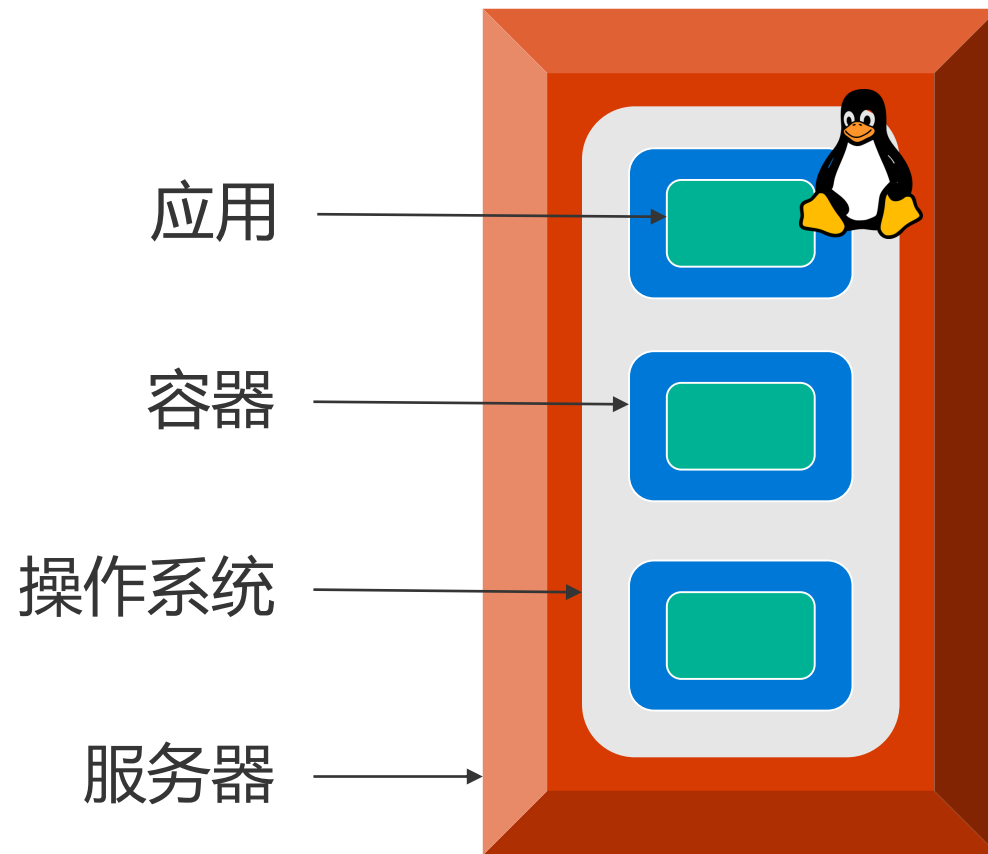
容器采用



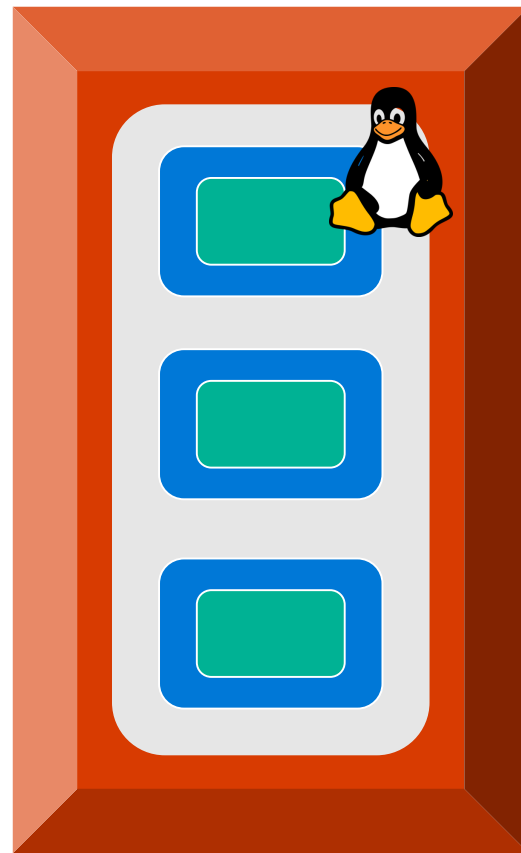
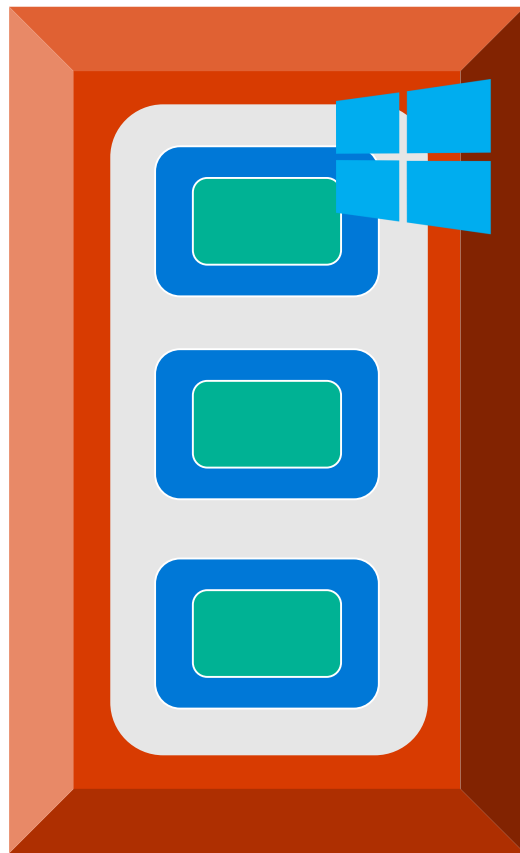
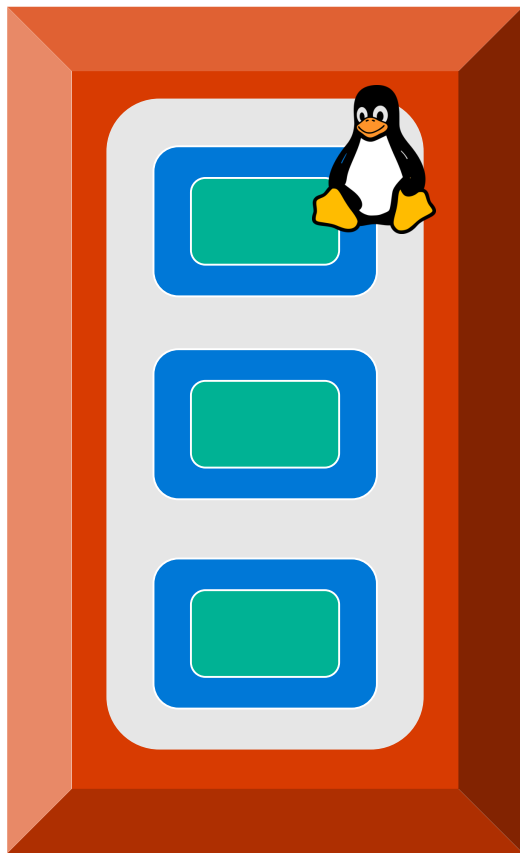
容器采用



容器采用



容器蔓延



容器编排



为什么需要容器编排？

- 灵活性
- 可扩展性
- 可靠性

Kubernetes是什么？

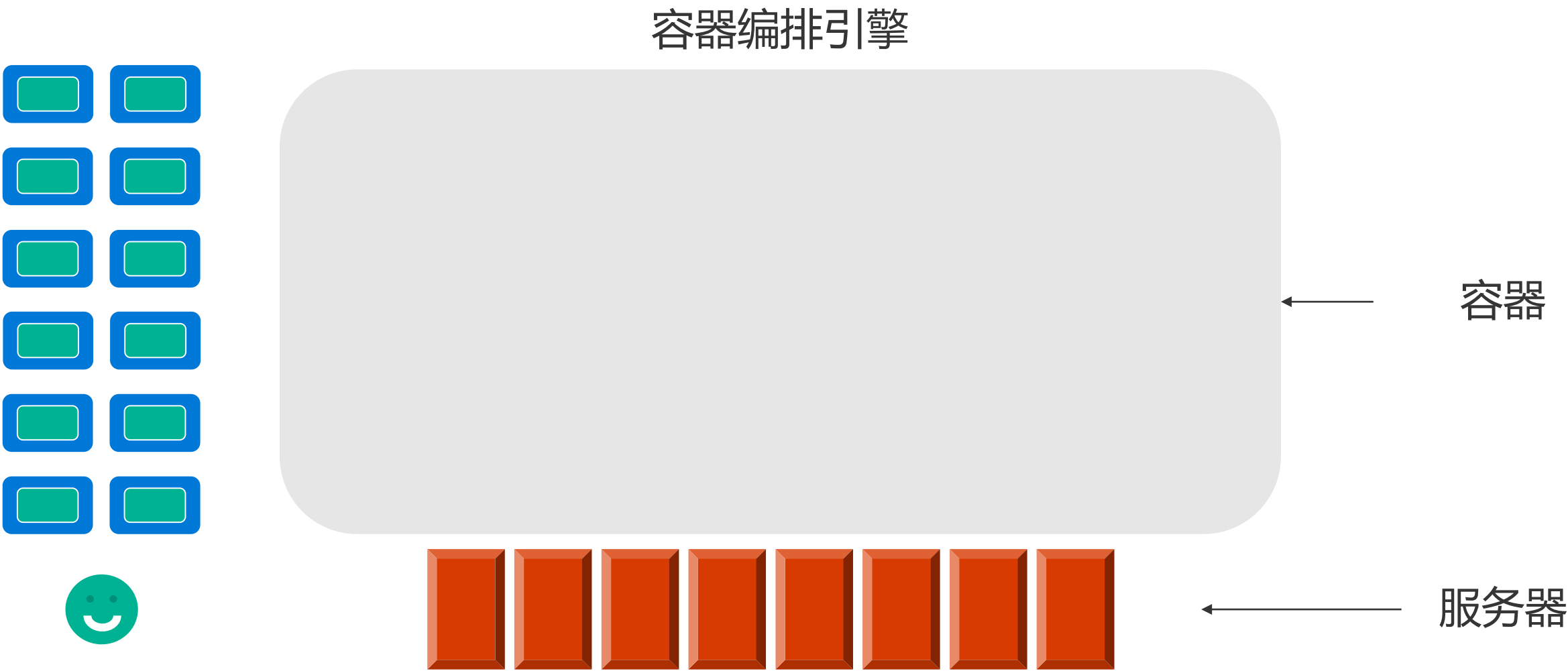
开源的容器编配引擎，可实现应用部署、扩展、管理的自动化

· 功能包括：

- 自动配载
- 自我修复
- 横向扩展
- 服务发现和负载均衡
- 自动部署与回滚
- 安全与配置管理
- 存储编配
- 批处理

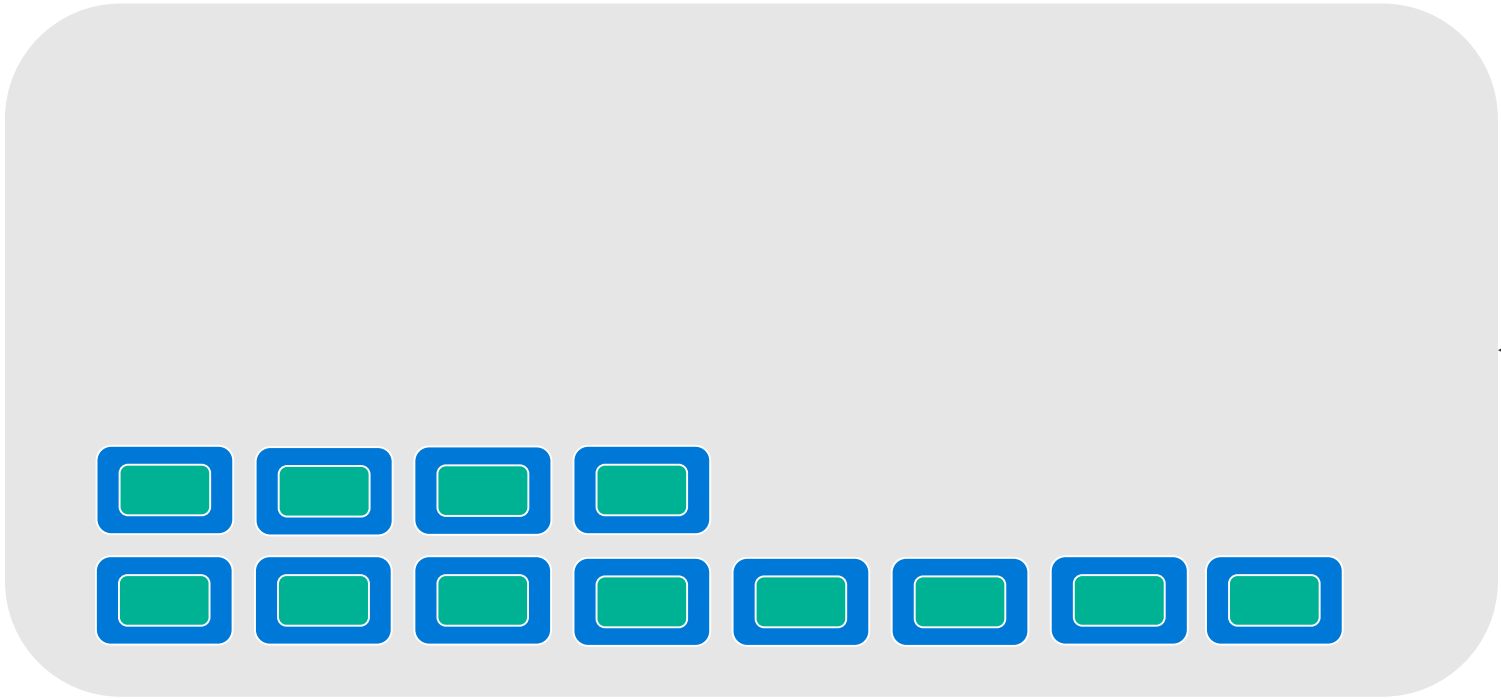


Kubernetes – 灵活性

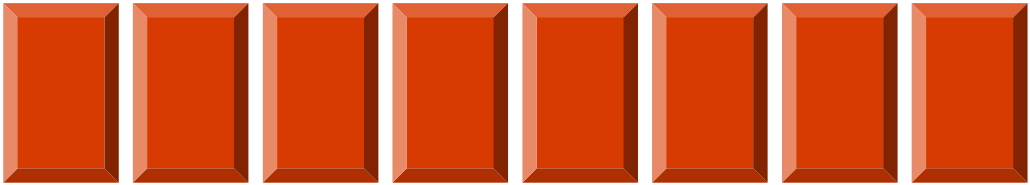


Kubernetes – 灵活性

容器编排引擎



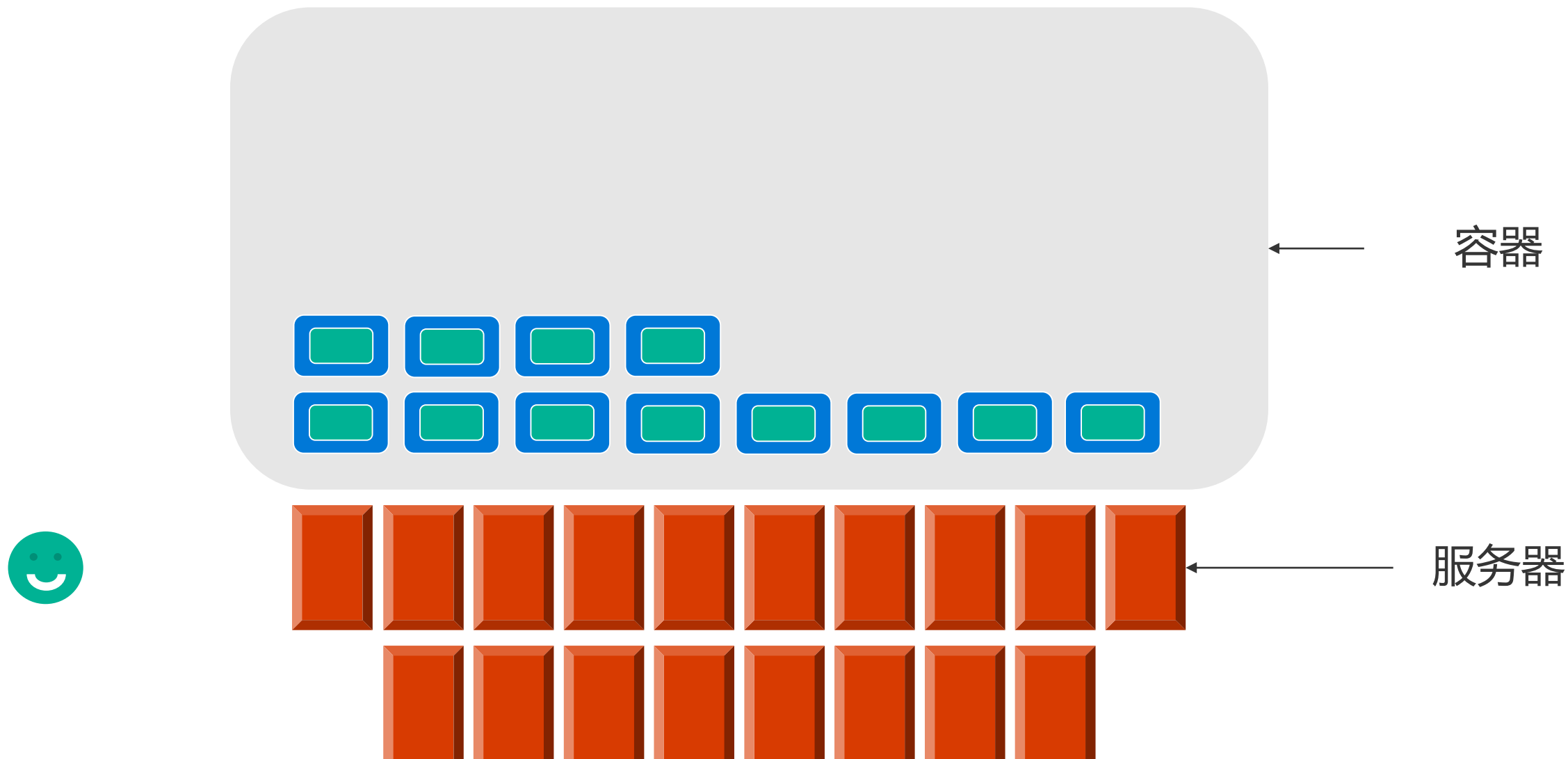
容器



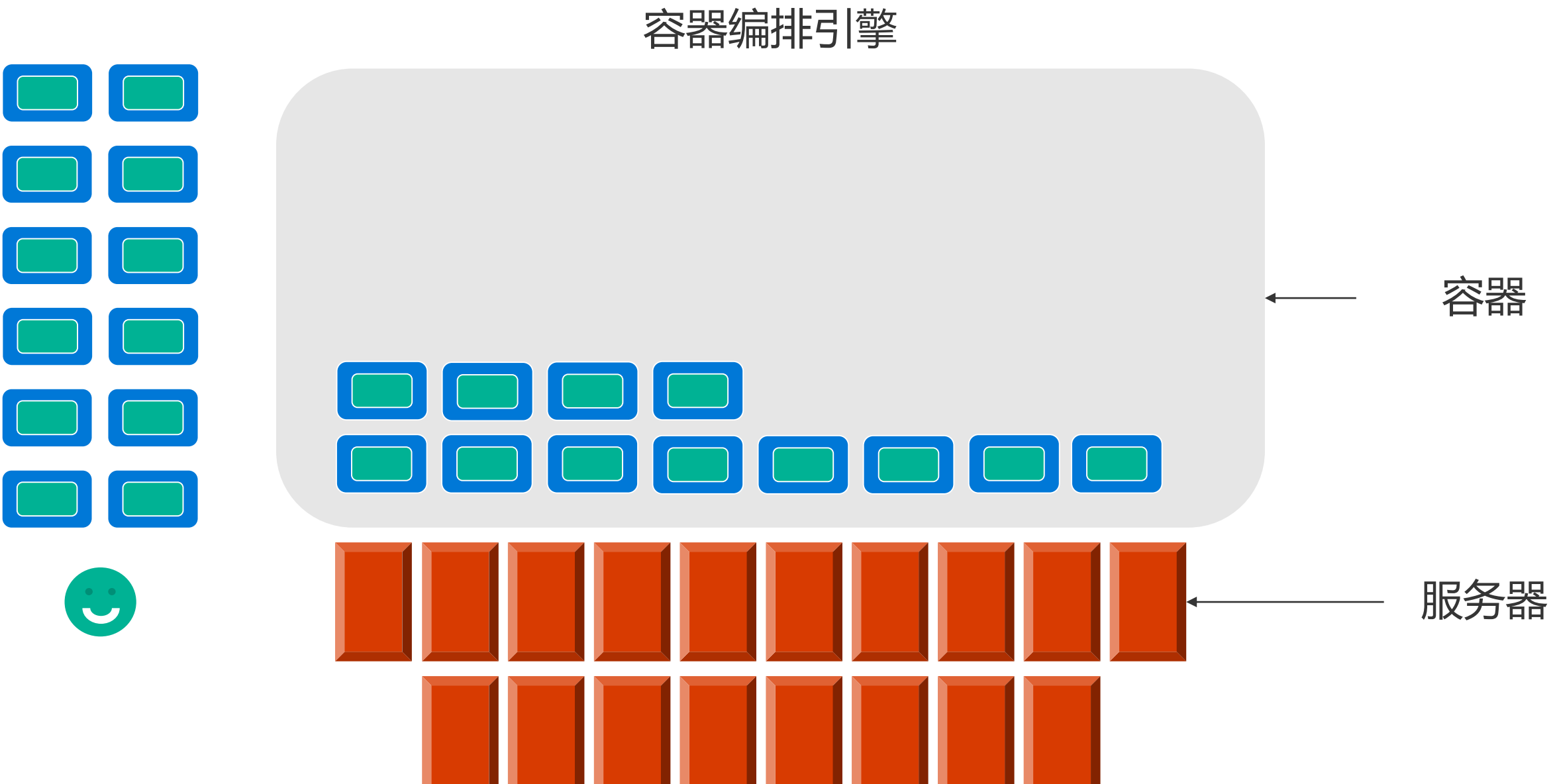
服务器

Kubernetes – 可扩展性

容器编排引擎

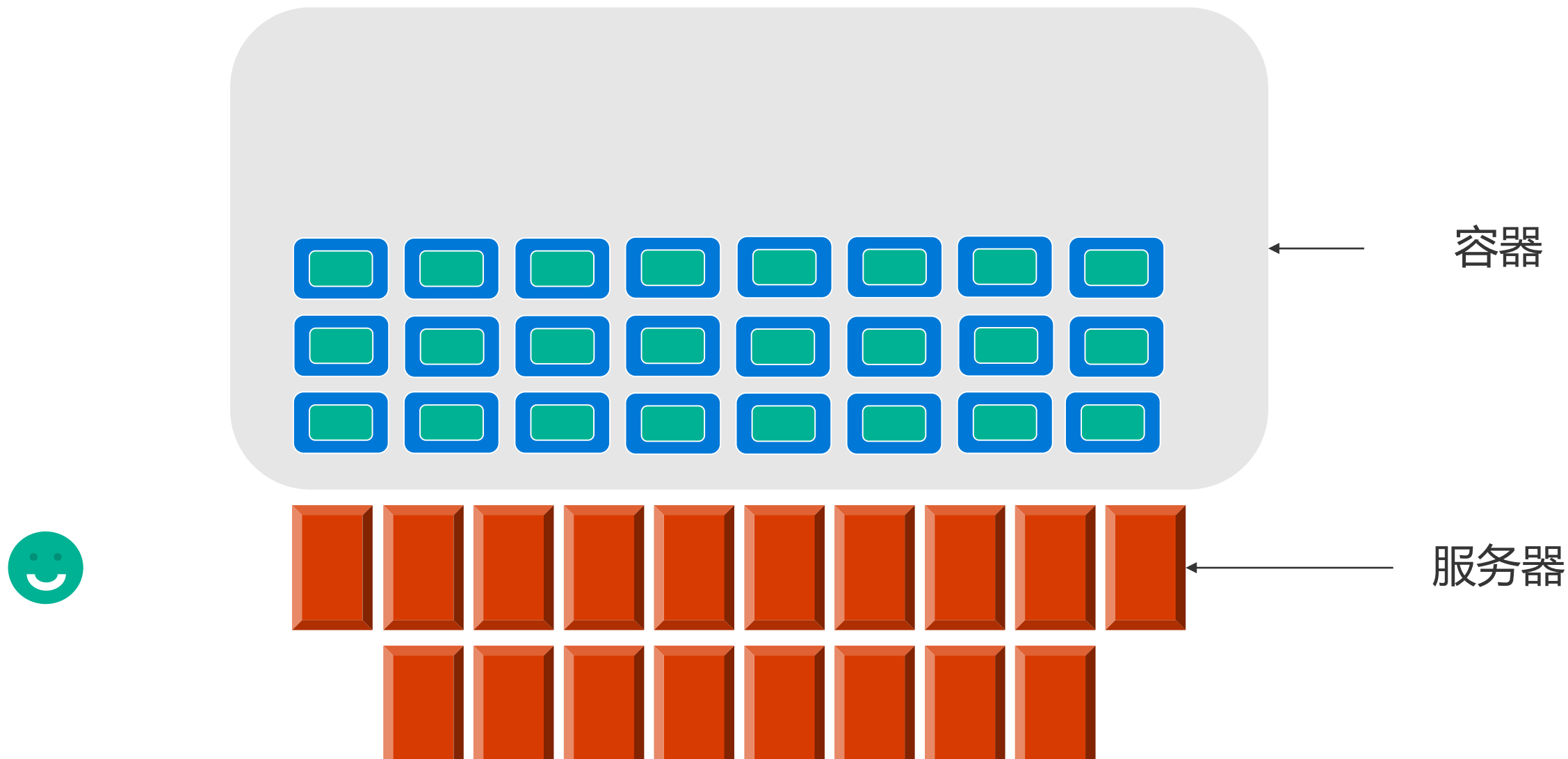


Kubernetes – 可扩展性



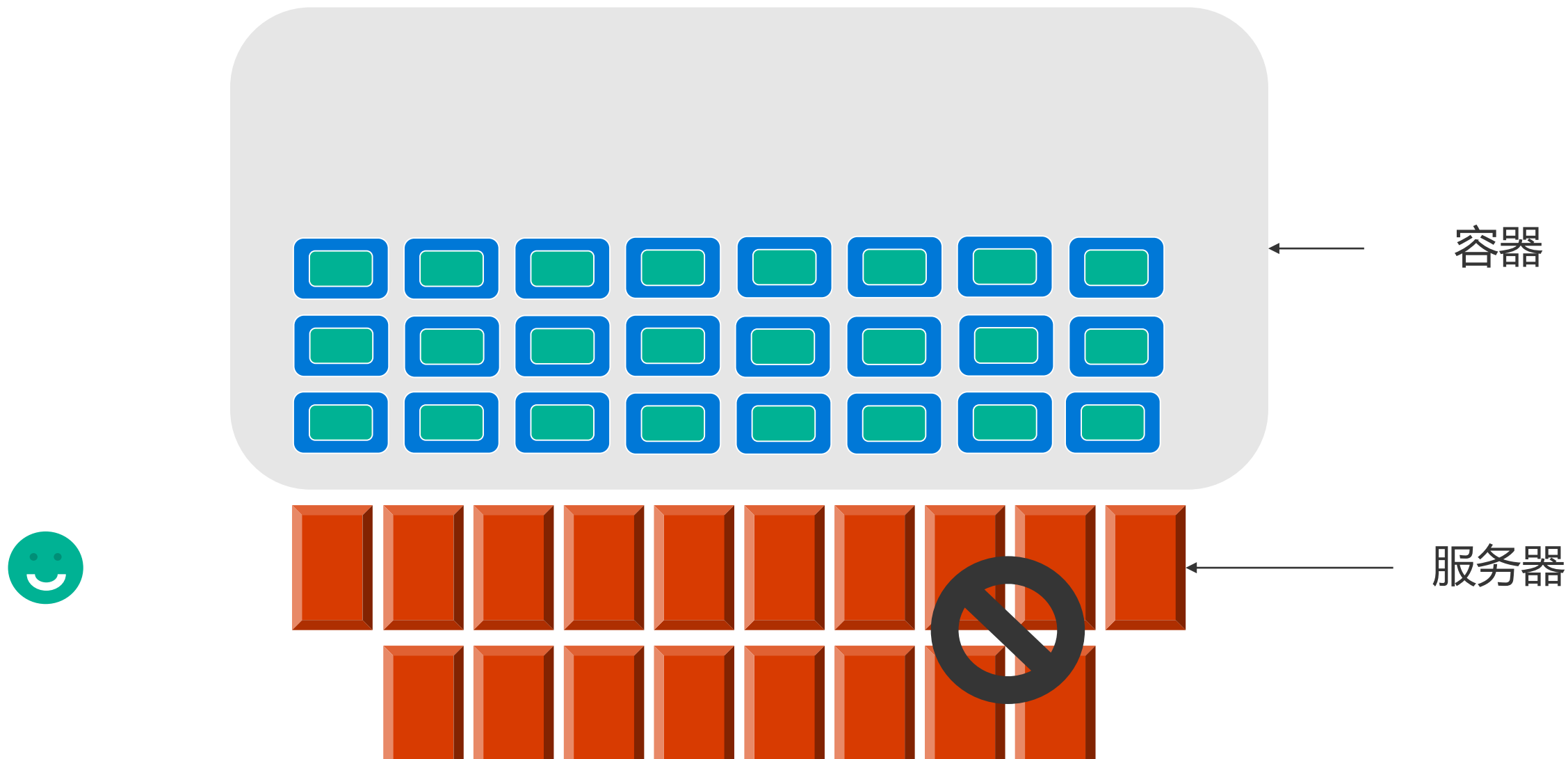
Kubernetes – 可扩展性

容器编排引擎



Kubernetes – 可靠性

容器编排引擎



容器服务： Azure Container Service (AKS)



由Azure管理，供您使用的
Kubernetes集群

为何需要AKS？

- 易用

- 在Azure上使用Kubernetes的最快方式
- 只需3条命令即可启用

- 易管理

- 自动升级、打补丁
- 极易伸缩的集群
- 可自我修复的控制面

- 使用开放的API

- 100% Kubernetes原生

开始使用AKS

```
$ az aks create -g myResourceGroup -n myCluster --generate-ssh-keys  
\ Running ..
```

```
$ az aks install-cli  
Downloading client to /usr/local/bin/kubectl ..
```

```
$ az aks get-credentials -g myResourceGroup -n myCluster  
Merged "myCluster" as current context ..
```

```
$ kubectl get nodes
```

NAME	STATUS	AGE	VERSION
aks-mycluster-36851231-0	Ready	4m	v1.8.1
aks-mycluster-36851231-1	Ready	4m	v1.8.1
aks-mycluster-36851231-2	Ready	4m	v1.8.1

管理AKS集群

```
$ az aks list -o table
```

Name	Location	ResourceGroup	KubernetesRelease	ProvisioningState
-----	-----	-----	-----	-----
myCluster	westus2	myResourceGroup	1.7.7	Succeeded

```
$ az aks upgrade -g myResourceGroup -n myCluster --kubernetes-version 1.8.1
```

\ Running ..

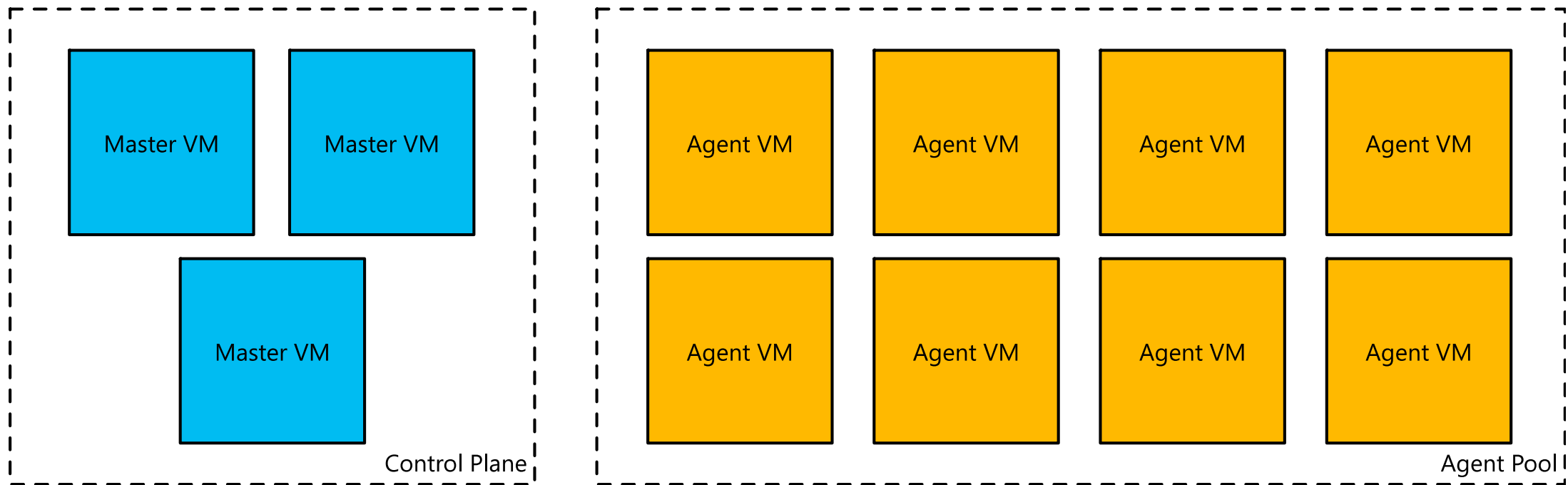
```
$ kubectl get nodes
```

NAME	STATUS	AGE	VERSION
aks-mycluster-36851231-0	Ready	12m	v1.8.1
aks-mycluster-36851231-1	Ready	8m	v1.8.1
aks-mycluster-36851231-2	Ready	3m	v1.8.1

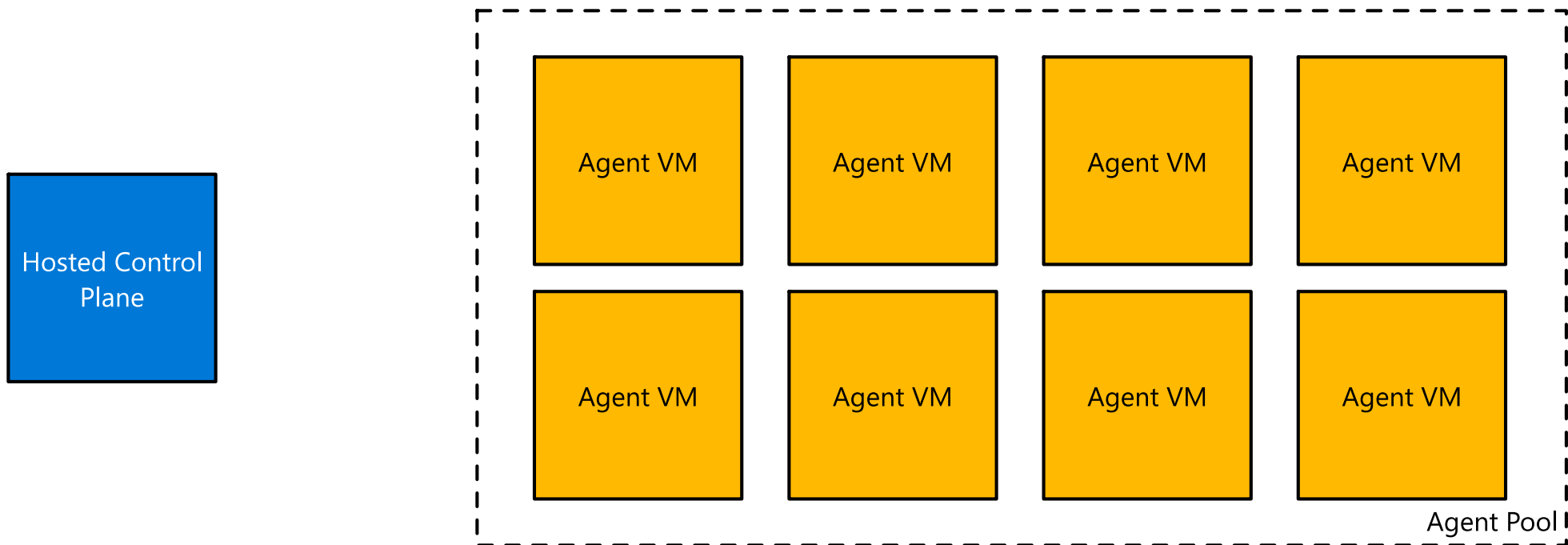
```
$ az aks scale -g myResourceGroup -n myCluster --agent-count 10
```

\ Running ..

没有使用AKS的Kubernetes



使用AKS的Kubernetes





在中国的进度

- AKS预计2018年第二季度在中国Azure上提供
- Preview In Global Azure 目前已经可供使用(全球，含香港、新加坡、日本、美国等区域皆已Preview...etc)
- 目前中国版Azure可使用acs-engine
 - 在中国Azure配置Kubernetes的最简单方法
 - 用户自行管理的解决方案，不同于AKS（由Azure管理）
 - 开源软件，免费提供

容器实例： Azure Container Instances(ACI)



在Azure中运行您自己的容器

为何需要ACI？

- 只需一条命令即可运行容器
- 按秒计费
- 无需管理虚拟机

开始使用ACI

```
$ az container create --name mycontainer --image microsoft/aci-helloworld --resource-group myResourceGroup --ip-address public
```

```
  "ipAddress": {
    "ip": "52.168.86.133",
    "ports": [...]
  },
  "location": "eastus",
  "name": "mycontainer",
  "osType": "Linux",
  "provisioningState": "Succeeded",
```

```
$ curl 52.168.86.133
```

```
<html>
```

```
<head>
```

```
  <title>Welcome to Azure Container Instances!</title>
```

```
</head>
```

费用

项目	费用
创建	每个实例\$0.0025
内存使用	\$0.0000125每GB-秒
CPU使用	\$0.0000125每核心-秒

一个512MB , 1 vCPU的容器每月仅需约\$48(320元人民币)

ACI的使用案例

- 直接迁移到容器 (Lift and Shift)
- 简单的web服务/应用
- 计算密集型负载
- 批处理 (Batch Processing)
- CI代理

在Azure中国的状态

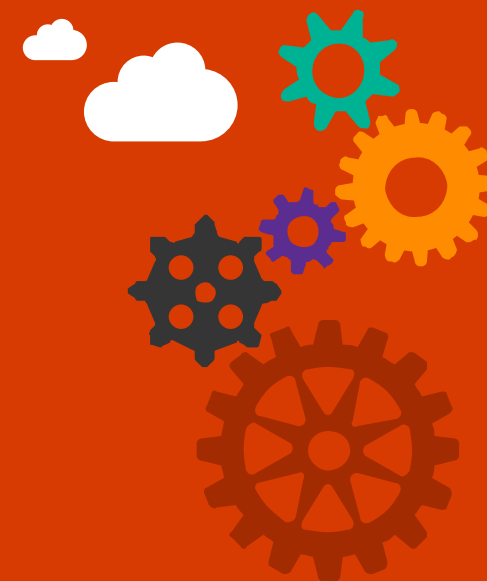
- ACI预计于2018年第二季度提供
- Preview In Global Azure 目前已经可供使用(全球，含香港、新加坡、日本、美国等区域皆已Preview...etc)

固有局限

- 不支持以下特性：
 - 零停机升级
 - 保持大于一个运行副本
 - 批处理 workflow
 - 服务发现
 - 集成的负载均衡
 - 自动扩展

如果您需要以上特性，您应该使用编排引擎(Orchestrator)

用于Kubernetes的ACI连接器



在Azure中
通过Kubernetes
运行您的容器

为何要使用ACI连接器？

- 通过开源的Kubernetes API使用ACI
- 按秒计费
- 无需管理虚拟机（可选）

ACI连接器如何工作？

1. 注册成一个无限容量的虚拟节点
2. 将调度的pods分派到ACI，而不是虚拟机

未来方向



未来方向

- 在Azure Stack上使用Kubernetes
- 服务代理：Azure Service Broker (ASB)
- Kubernetes开发者工具集成
 - Helm
 - Draft
 - Brigade

谢谢



参考资料

- ACS Engine
 - <https://github.com/Azure/acs-engine>
- Azure Container Service (AKS)
 - <https://azure.microsoft.com/en-us/services/container-service/>
 - <https://docs.microsoft.com/en-us/azure/aks/>
- Azure Container Instances (ACI)
 - <https://azure.microsoft.com/en-us/services/container-instances/>
 - <https://docs.microsoft.com/en-us/azure/container-instances/>
- 用于Kubernetes的ACI连接器
 - <https://github.com/Azure/aci-connector-k8s>

