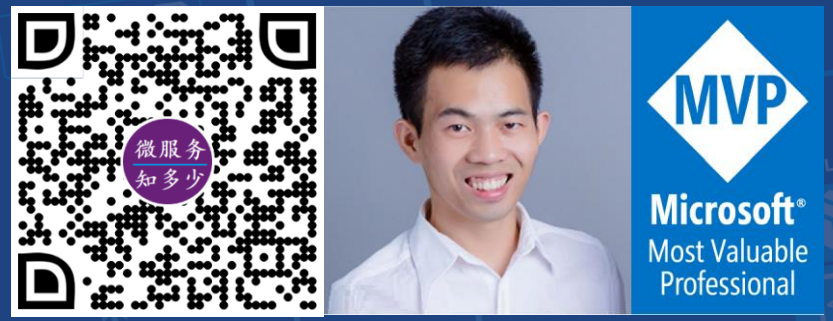
AKS 赋能 .NET Core 微服务

eShopOnContainers + AKS

颜圣杰

@sheng-jie



STUDY4

Study For Love

2019 Global Azure BOOTCAMP

目录

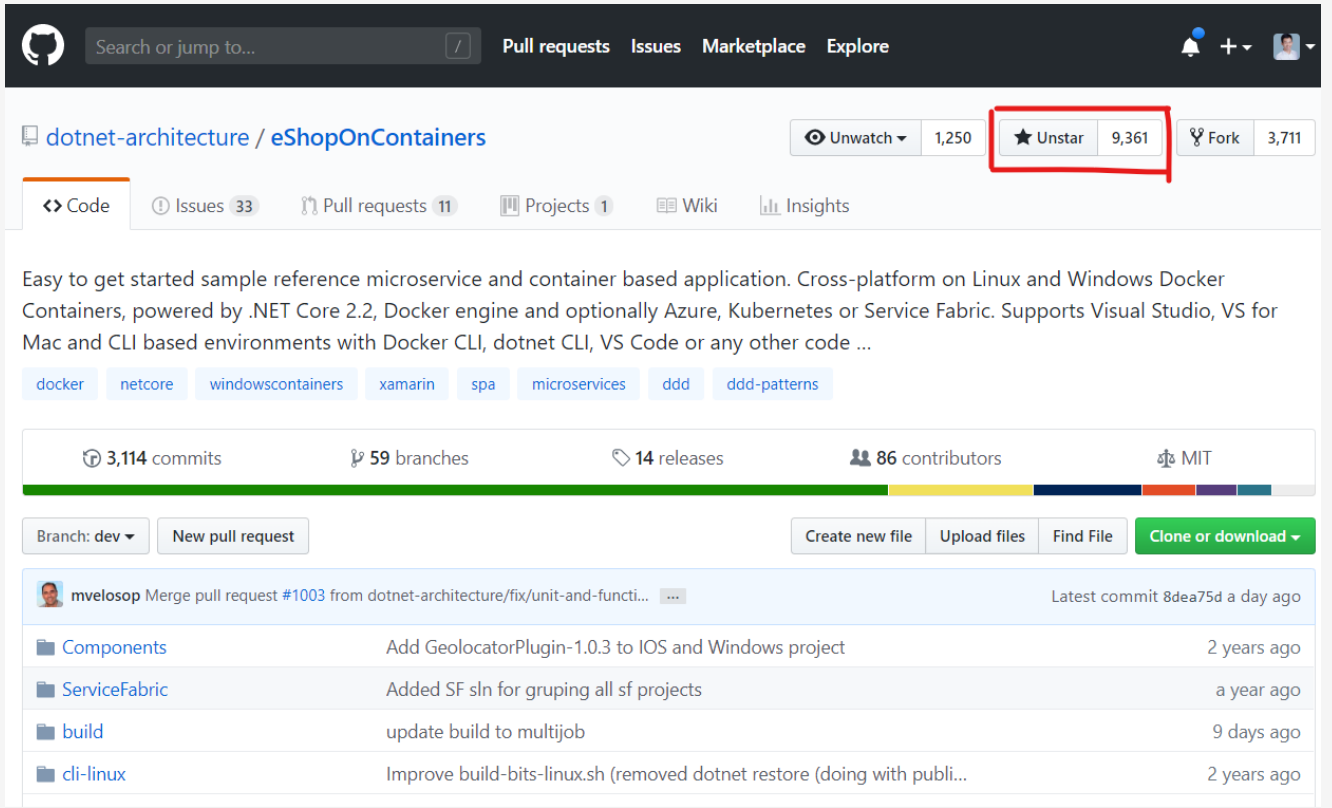
eShopOnContainers 知多少

eShopOnContainers 简介

1. .NET Core 开源微服务示例项目（简版在线商城应用）

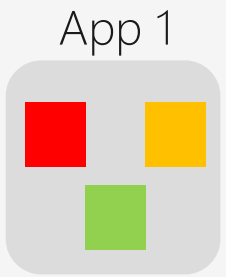
2. 简化版的基于.NET Core和Docker等技术开发的面向微服务架构的参考应用。

3. 基于Xamarin的移动端、Mpa、Spa



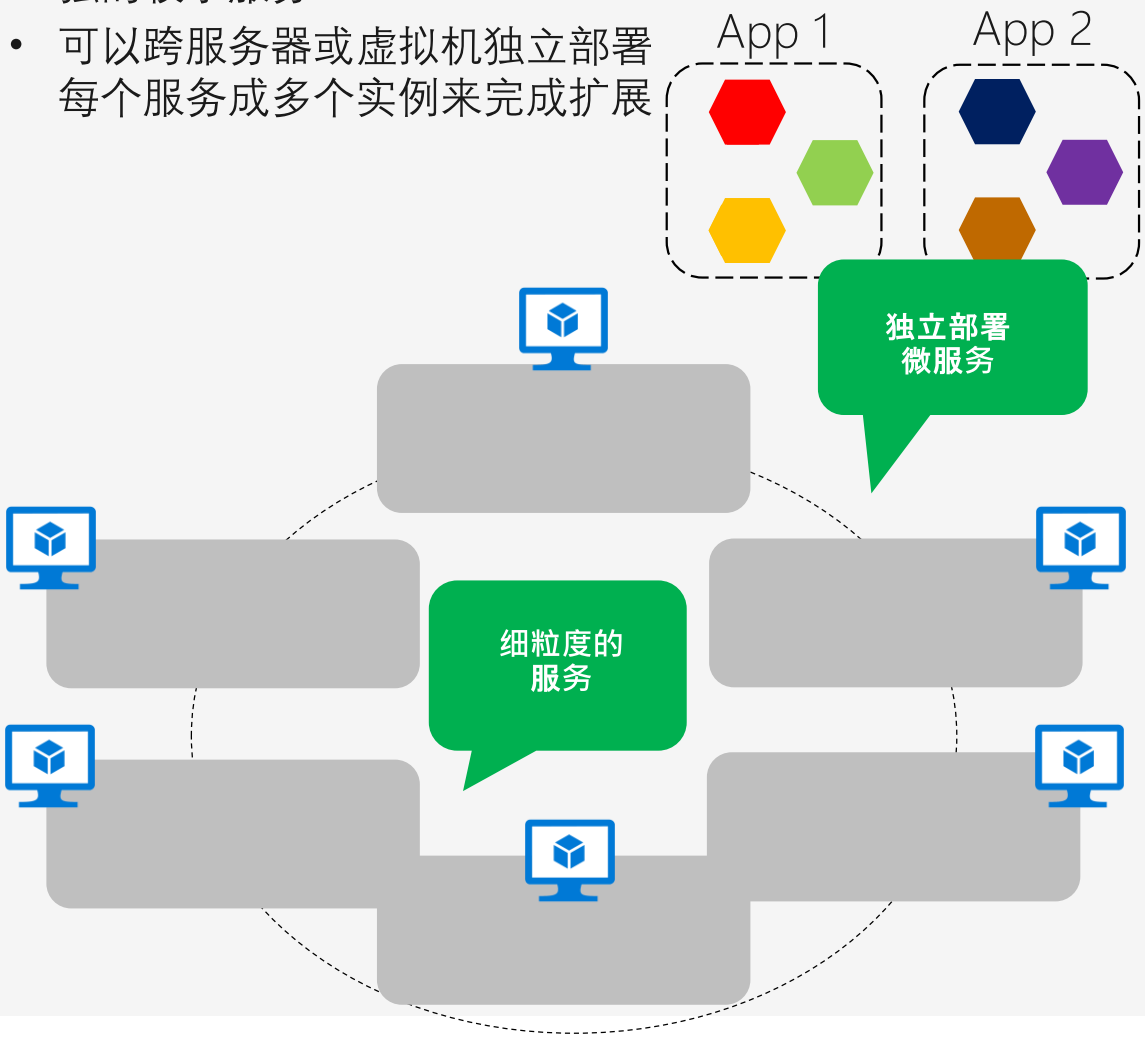
传统应用

- 传统应用一般通过分层和类库构造的多个组件进程提供功能
- 需要通过复制整个应用到多台服务器或虚拟机上完成扩展



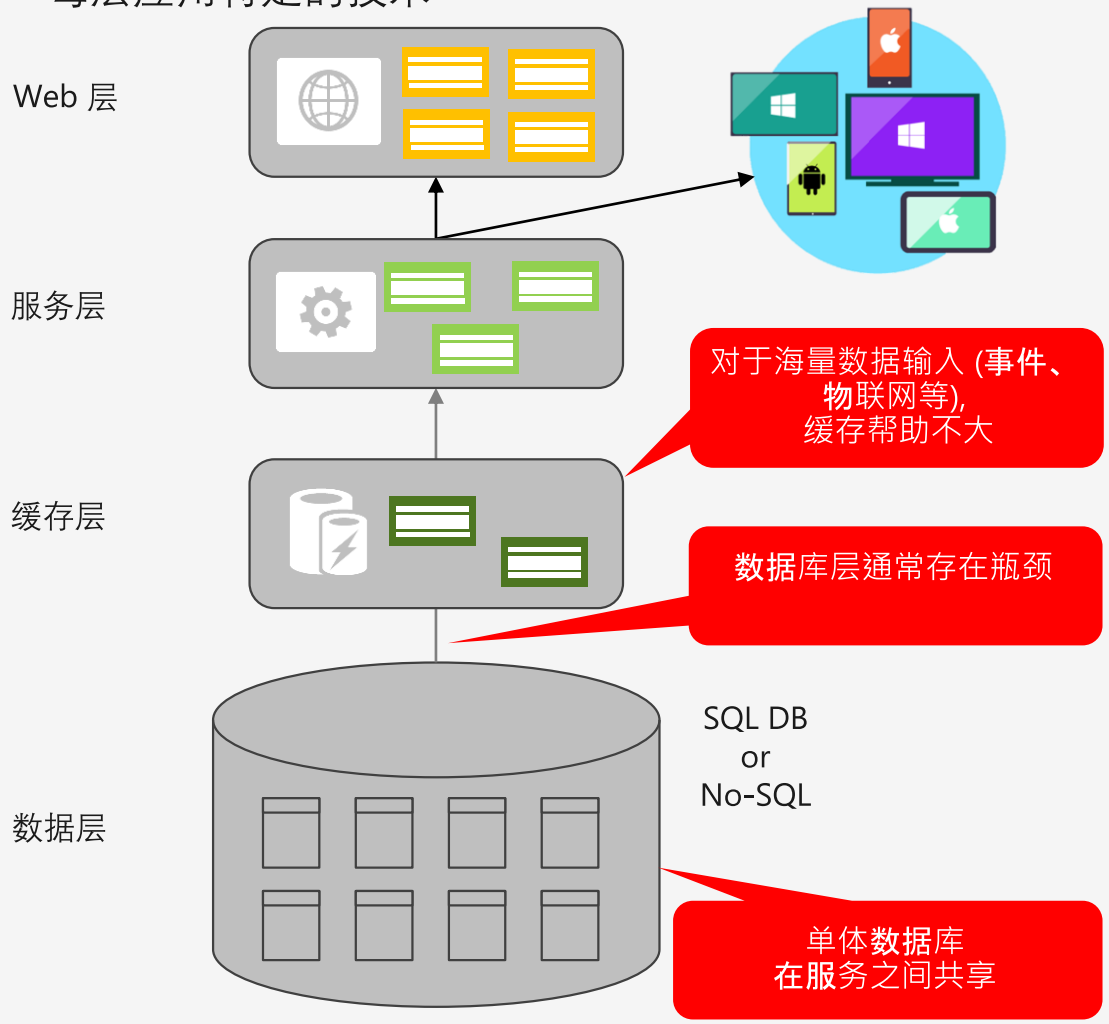
微服务应用

- 微服务应用程序将功能分离为单独的较小服务
- 可以跨服务器或虚拟机独立部署每个服务成多个实例来完成扩展



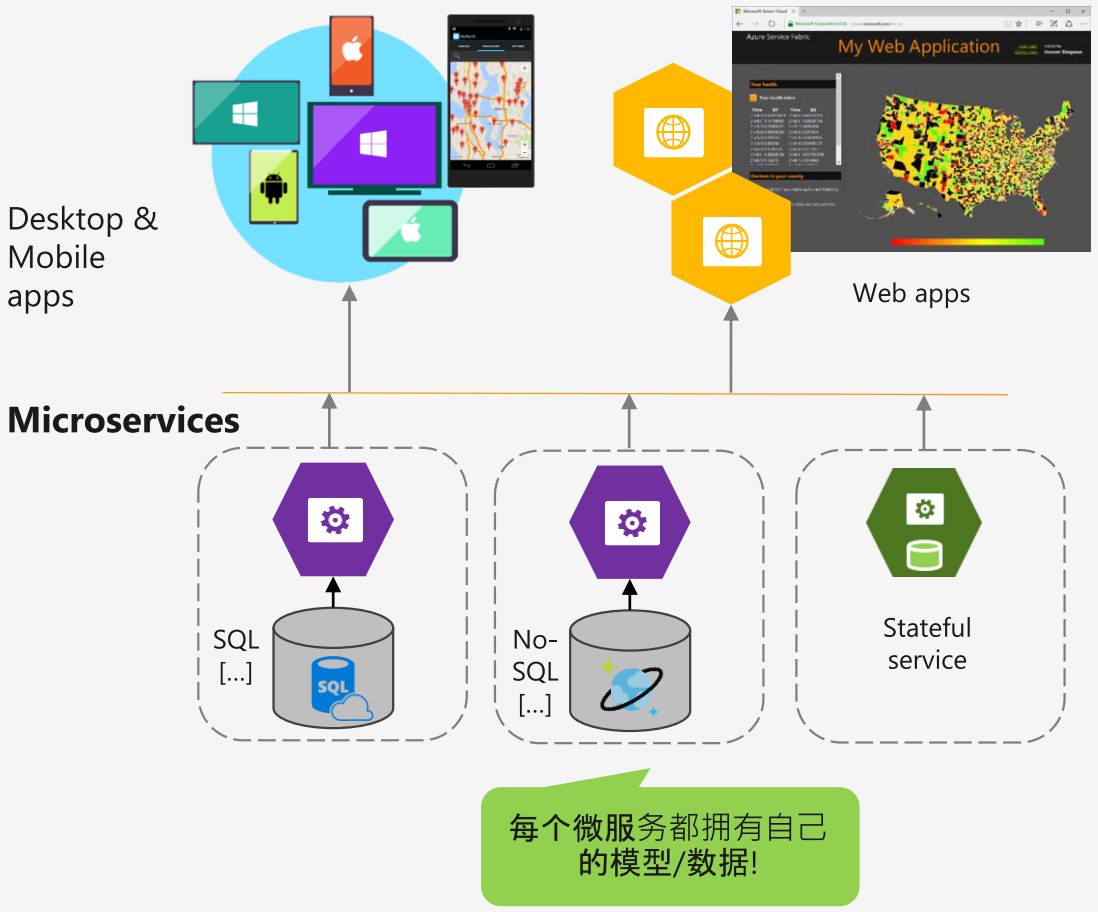
传统应用

- 单体数据库
- 每层应用特定的技术



微服务应用

- 微服务之间互联
- 每个微服务都有独立的状态/数据作用域
- 冷数据的远程存储



新的模式和技术

Patterns

Technologies

Microservices

Autonomous

Bounded Context

Nomad & addressable services

Isolated

API Gateway

Decoupled

Events

Async. communication

Event Bus

Message Brokers

Service Discovery

Health Checks

Transient Failures Handling

Resiliency

Commands

Retries with Exponential Backoff

Circuit Breakers

Domain-Driven Design

CQRS simplified

Aggregates

Domain Entity

Domain Events

Mediator

Docker Containers

Linux Containers

Docker Image

Docker Host

Windows Containers

Docker Registry

Docker Hub

RabbitMQ

Hyper-V Containers

Azure Service Bus

Azure Container Registry

NServiceBus

Orchestrators

CI/CD

MassTransit

Brighter

Azure Service Fabric

Stateful Services

Actors

Polly

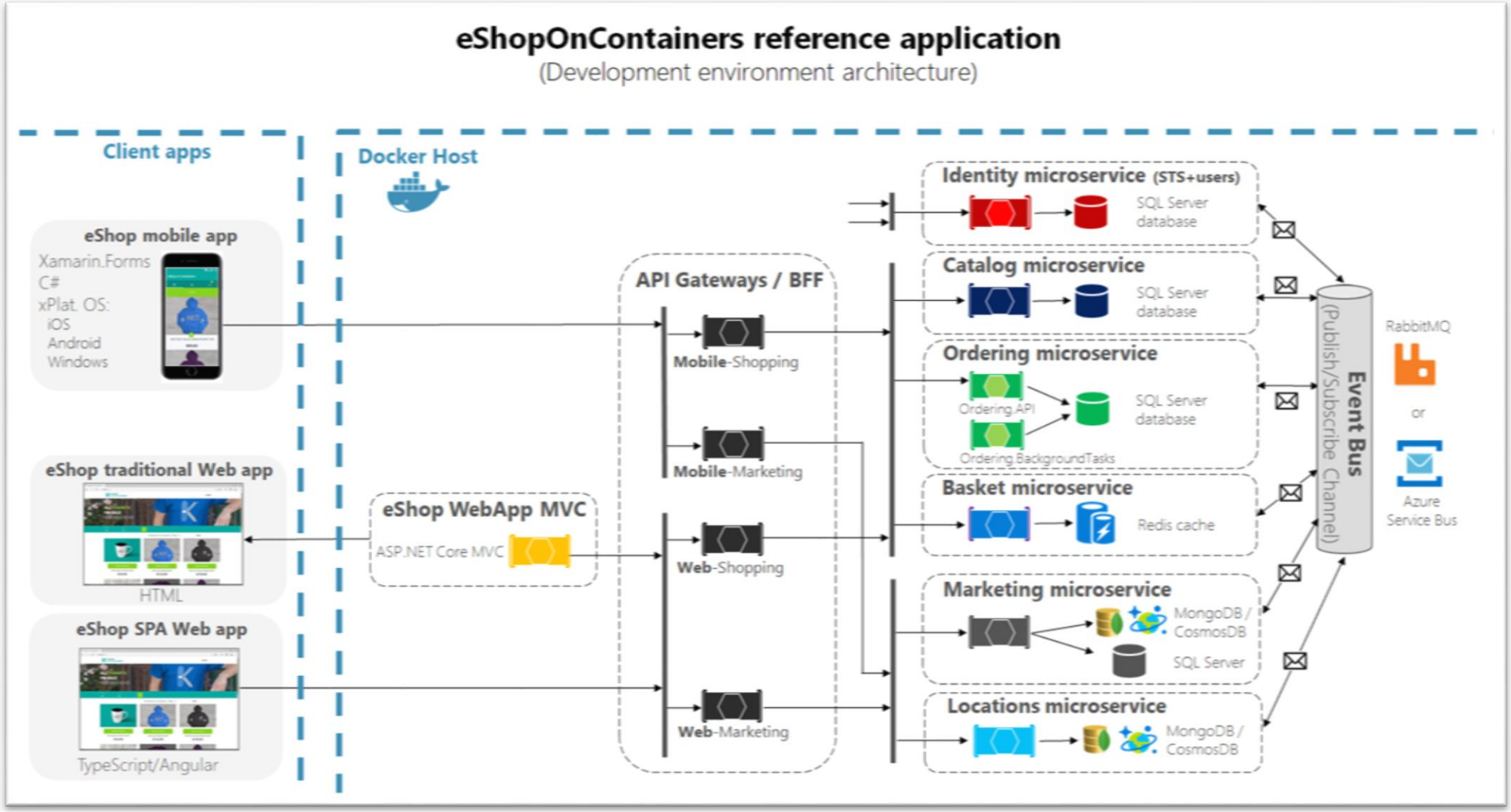
Azure Container Service

Kubernetes

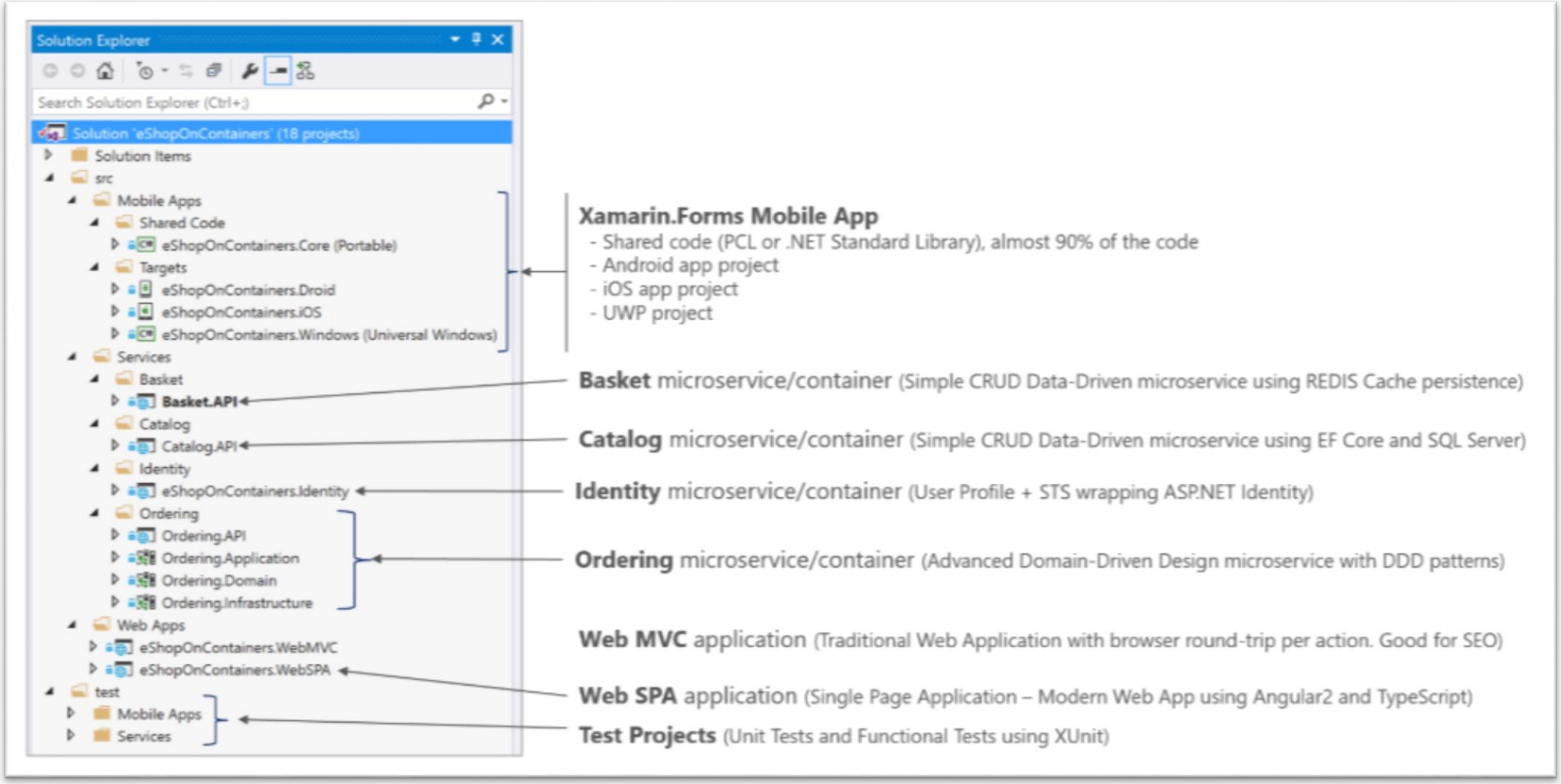
Docker Swarm

Mesos DC/OS

整体架构


eShopOnContainers reference application
(Development environment architecture)
Client apps
Docker Host
eShop mobile app
Xamarin.Forms
C#
xPlat. OS:
iOS
Android
Windows
eShop traditional Web app
HTML
eShop SPA Web app
TypeScript/Angular
API Gateways / BFF
Mobile-Shopping
Mobile-Marketing
eShop WebApp MVC
ASP.NET Core MVC
Web-Shopping
Web-Marketing
Identity microservice (STS+users)
SQL Server database
Catalog microservice
SQL Server database
Ordering microservice
Ordering.API
Ordering.BackgroundTasks
SQL Server database
Basket microservice
Redis cache
Marketing microservice
MongoDB / CosmosDB
SQL Server
Locations microservice
MongoDB / CosmosDB
Event Bus
(Publish/Subscribe Channel)
RabbitMQ
or
Azure Service Bus

代码结构


Solution Explorer
Search Solution Explorer (Ctrl+;)
Solution 'eShopOnContainers' (18 projects)
Solution Items
src
Mobile Apps
Shared Code
eShopOnContainers.Core (Portable)
Targets
eShopOnContainers.Droid
eShopOnContainers.iOS
eShopOnContainers.Windows (Universal Windows)
Services
Basket
Basket.API
Catalog
Catalog.API
Identity
eShopOnContainers.Identity
Ordering
Ordering.API
Ordering.Application
Ordering.Domain
Ordering.Infrastructure
Web Apps
eShopOnContainers.WebMVC
eShopOnContainers.WebSPA
test
Mobile Apps
Services
Xamarin.Forms Mobile App
- Shared code (PCL or .NET Standard Library), almost 90% of the code
- Android app project
- iOS app project
- UWP project
Basket microservice/container (Simple CRUD Data-Driven microservice using REDIS Cache persistence)
Catalog microservice/container (Simple CRUD Data-Driven microservice using EF Core and SQL Server)
Identity microservice/container (User Profile + STS wrapping ASP.NET Identity)
Ordering microservice/container (Advanced Domain-Driven Design microservice with DDD patterns)
Web MVC application (Traditional Web Application with browser round-trip per action. Good for SEO)
Web SPA application (Single Page Application – Modern Web App using Angular2 and TypeScript)
Test Projects (Unit Tests and Functional Tests using XUnit)

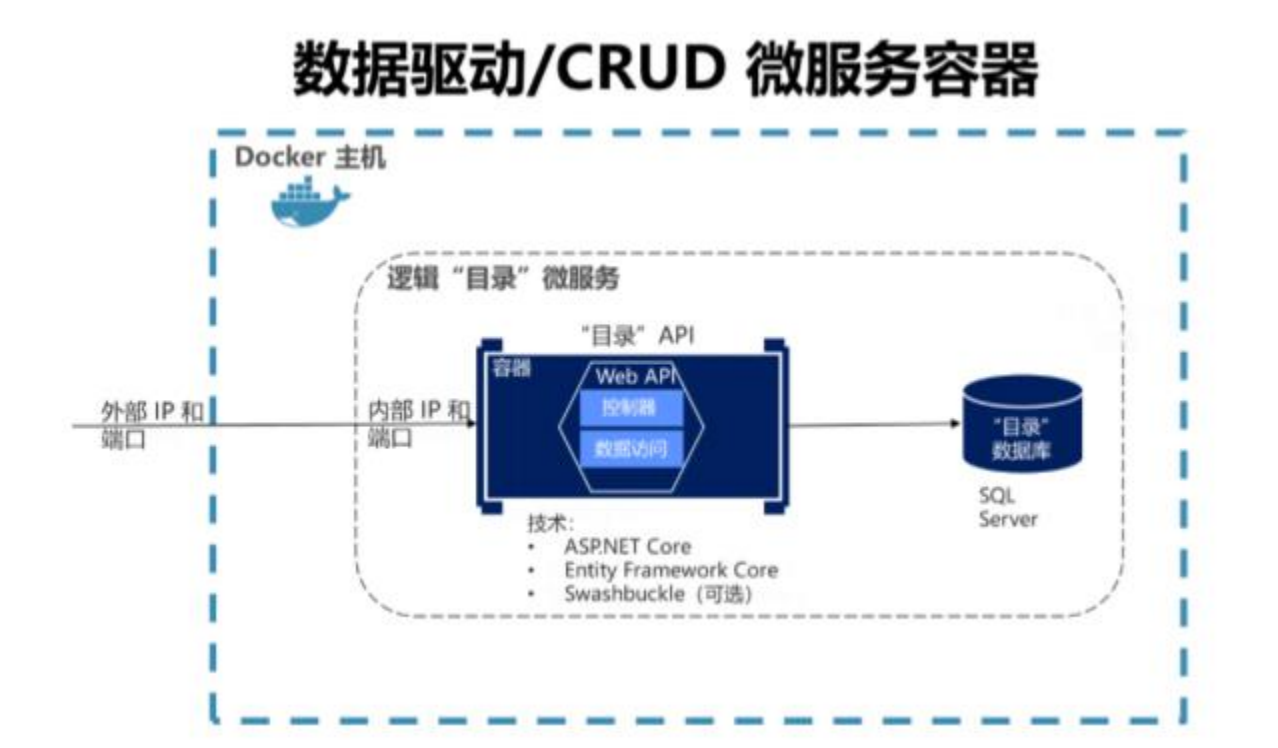
架构模式

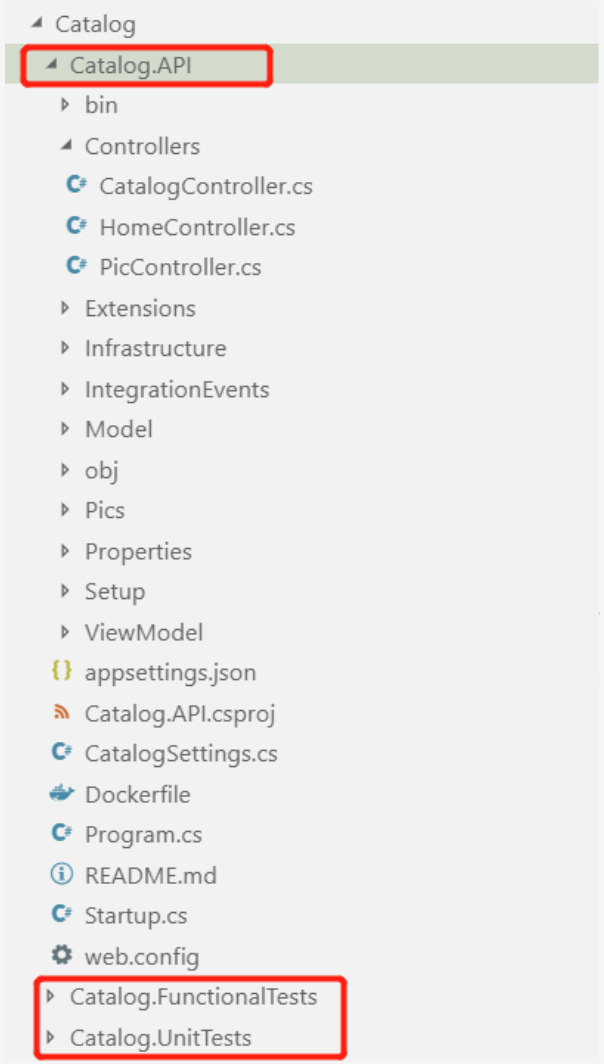
基于数据驱动的CRUD微服务

基于DDD的微服务

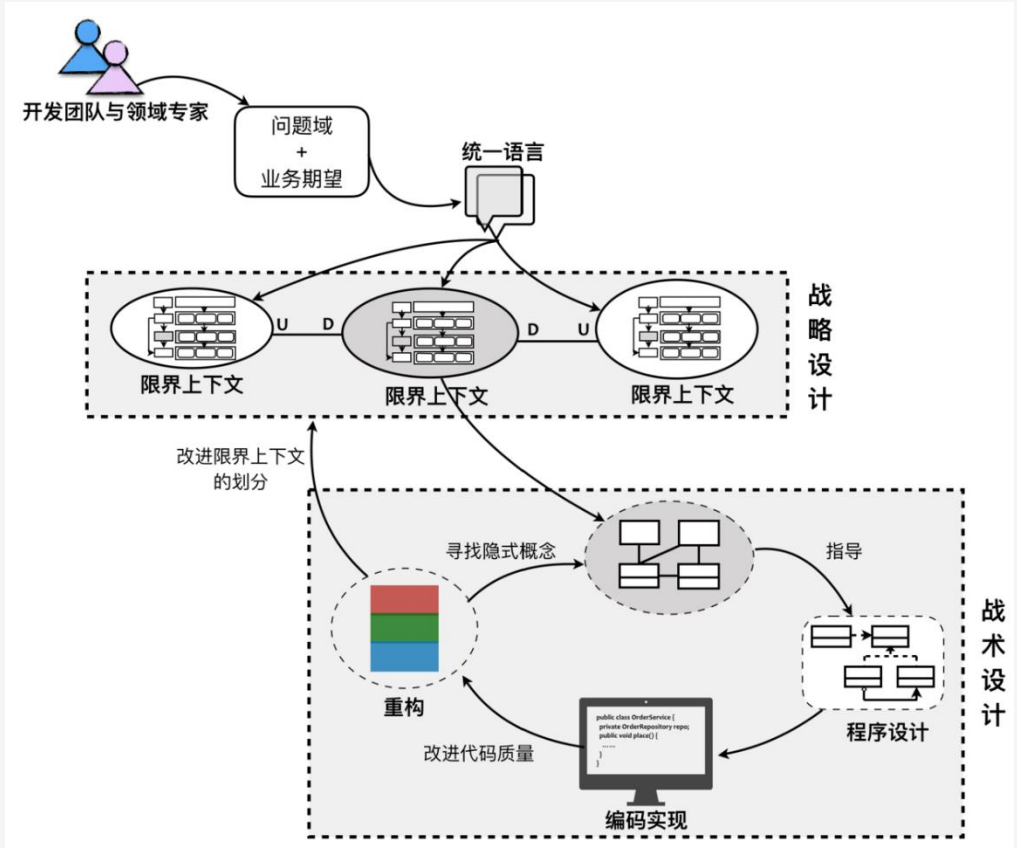
基于事件驱动的微服务

基于数据驱动的CRUD微服务





基于DDD的订单微服务



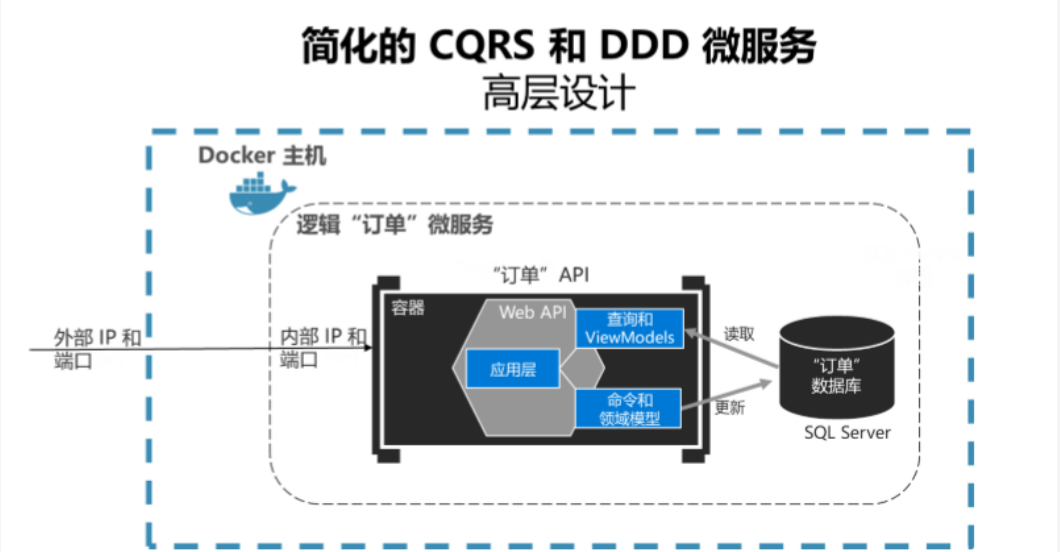
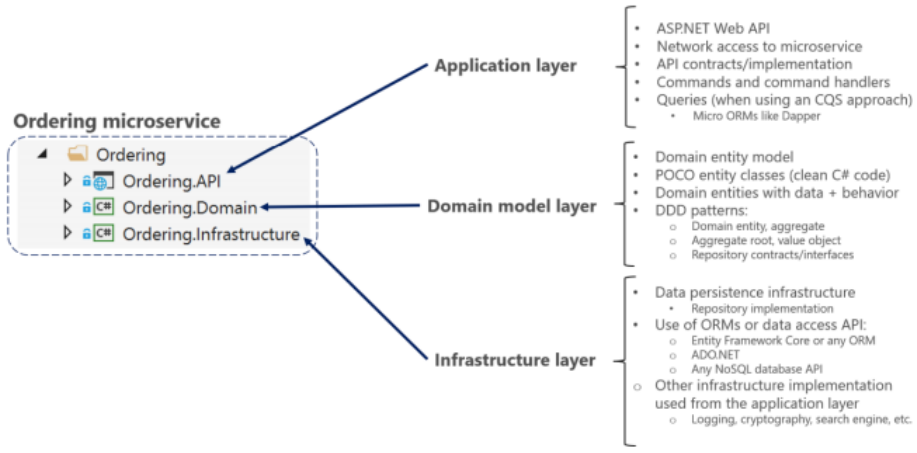


图 9-2. 基于简化的 CQRS 和 DDD 的微服务

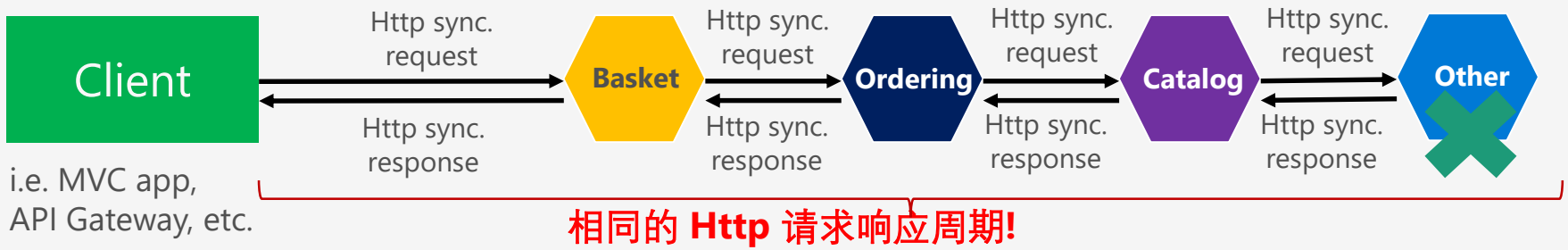


基于MediatR 和Dapper实现CQRS

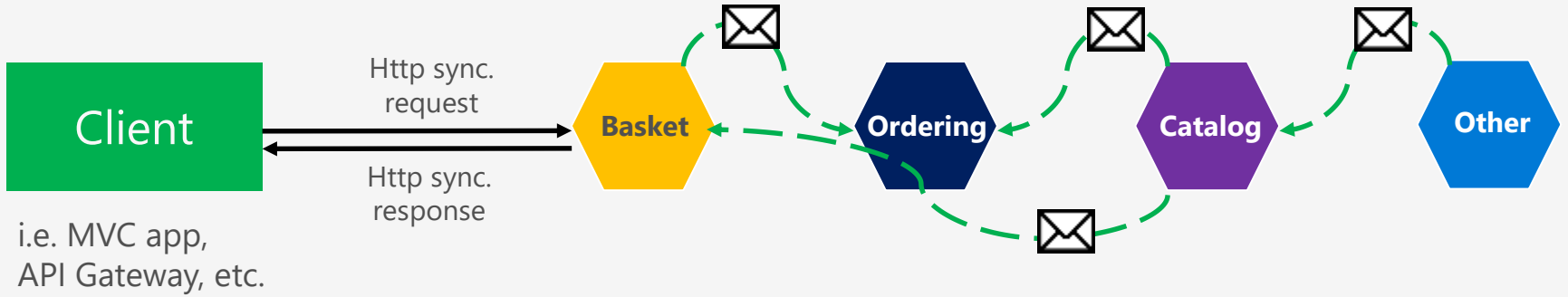
微服务间的通信方式：同步 VS 异步

反模式

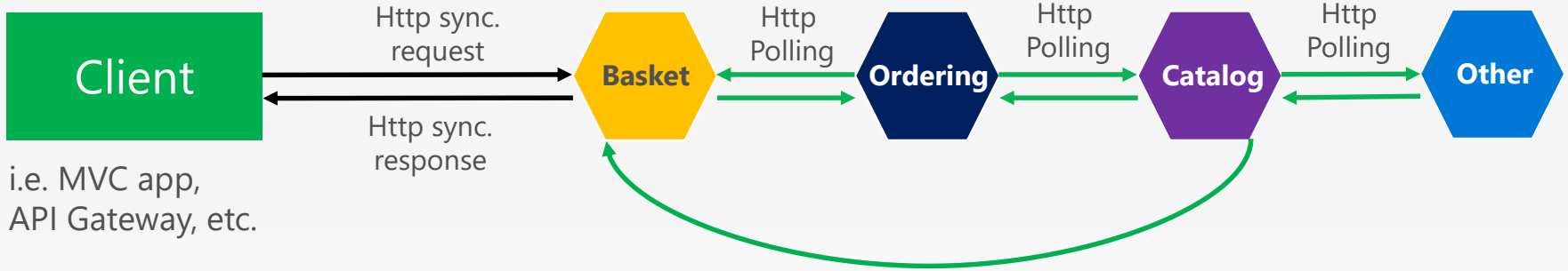
同步
all req./resp. cycle



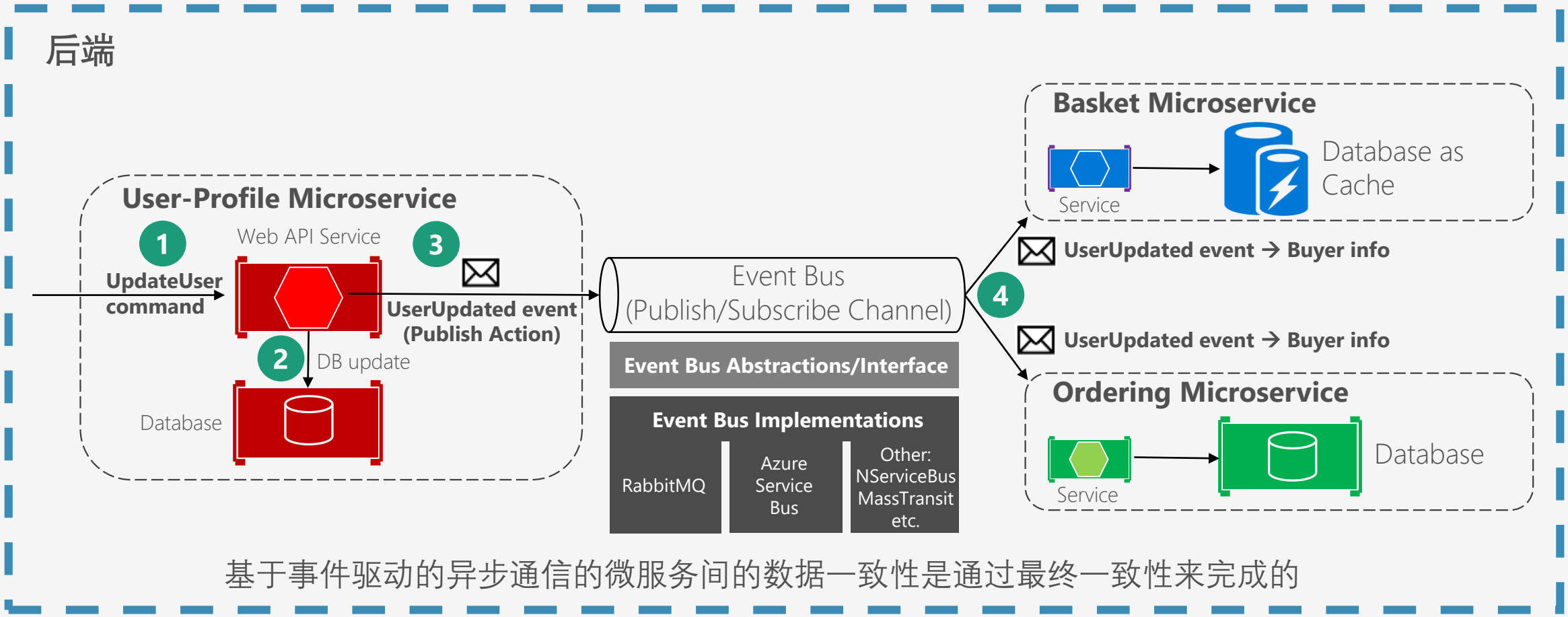
异步
跨内部微服务 (事件总线: AMPQ)



“异步”
跨内部微服务(轮询: **Http**)

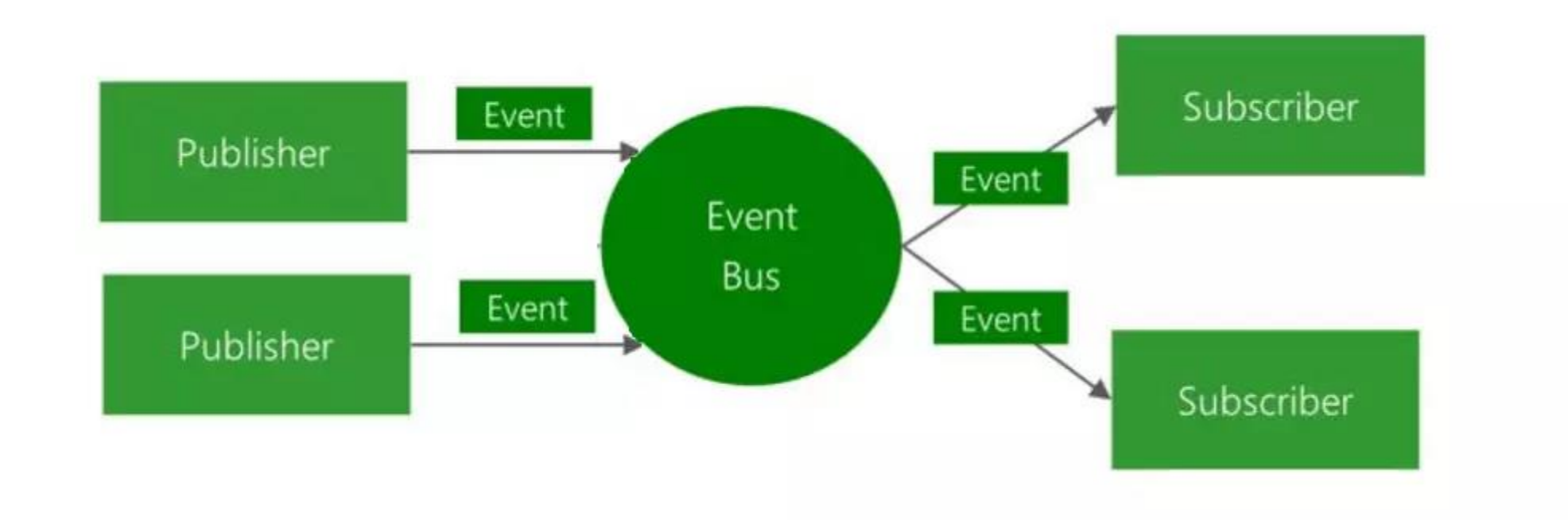


基于事件驱动的微服务 -- EventBus的应用

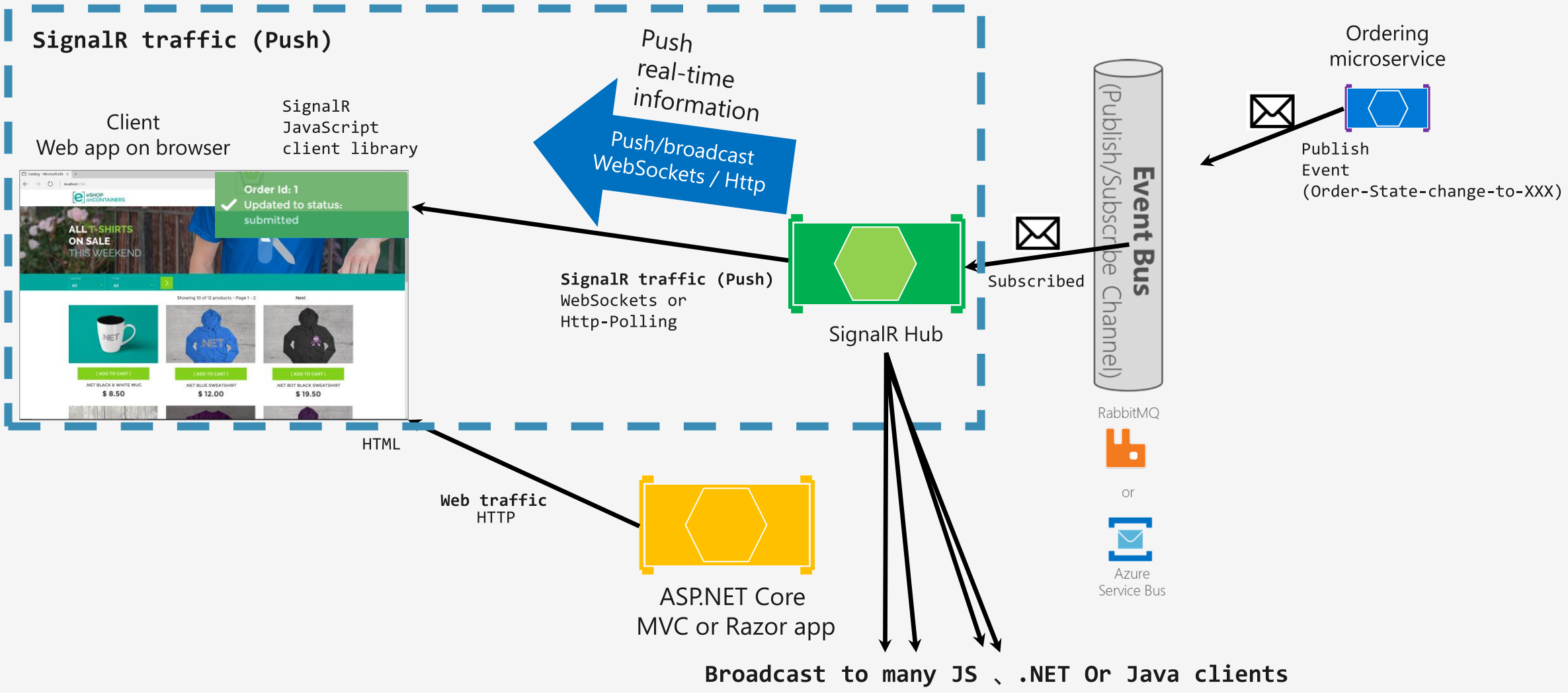


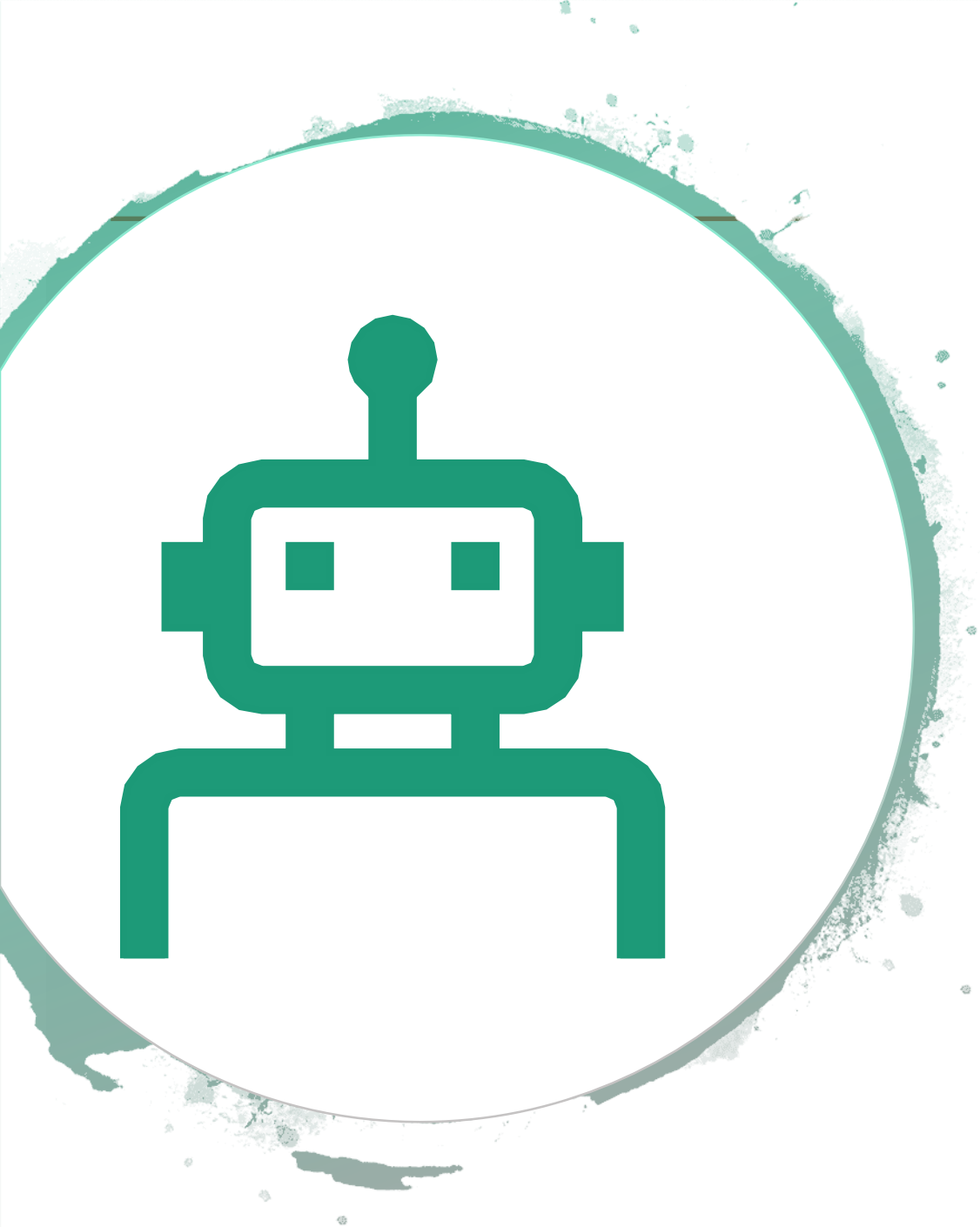
基于事件驱动的异步通信的微服务间的数据一致性是通过最终一致性来完成的

Event Bus简介


Publisher
Publisher
Event
Event
Event Bus
Event
Event
Subscriber
Subscriber

基于事件驱动的微服务 -- SignalR的应用

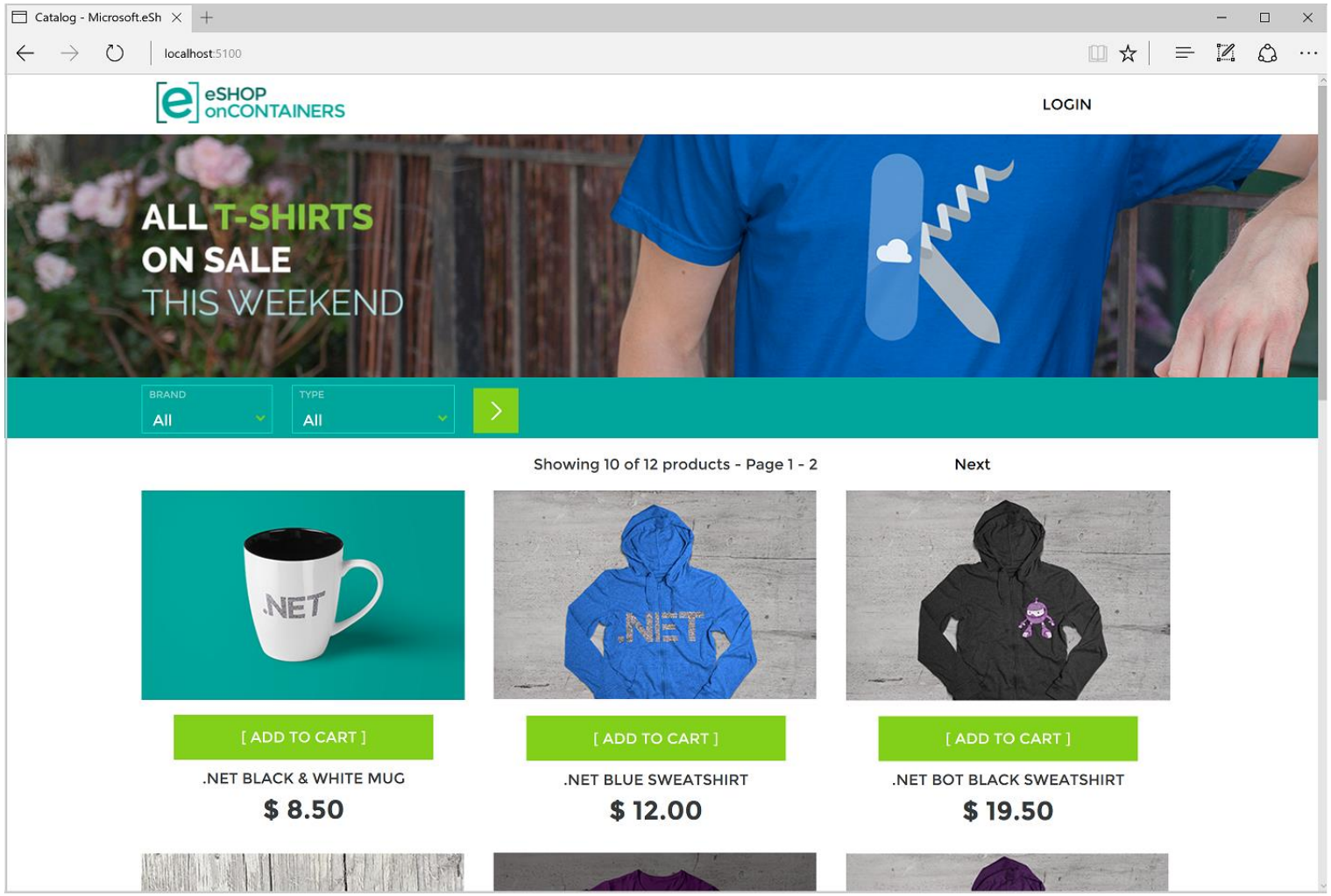




其他技术选型

- Redis
- RabbitMQ
- Polly
- MediatR（CQRS)
- Ocelot
- Autofac
- Mogodb

Demo


Catalog - Microsoft.eSh
localhost:5100
eSHOP onCONTAINERS
LOGIN
ALL T-SHIRTS
ON SALE
THIS WEEKEND
BRAND
All
TYPE
All
Showing 10 of 12 products - Page 1 - 2
Next
[ADD TO CART]
.NET BLACK & WHITE MUG
$ 8.50
[ADD TO CART]
.NET BLUE SWEATSHIRT
$ 12.00
[ADD TO CART]
.NET BOT BLACK SWEATSHIRT
$ 19.50

相关资料

eShopOnContainers 知多少[1]：总体概览
eShopOnContainers 知多少[2]：Run起来
eShopOnContainers 知多少[3]：Identity microservice
eShopOnContainers 知多少[4]：Catalog microservice
eShopOnContainers 知多少[5]：EventBus With RabbitMQ
eShopOnContainers 知多少[6]：持久化事件日志
eShopOnContainers 知多少[7]：Basket microservice
eShopOnContainers 知多少[8]：Ordering microservice
eShopOnContainers 知多少[9]：Ocelot gateways
eShopOnContainers 知多少[10]：部署到 K8S | AKS

Azure Kubernetes Service (AKS)

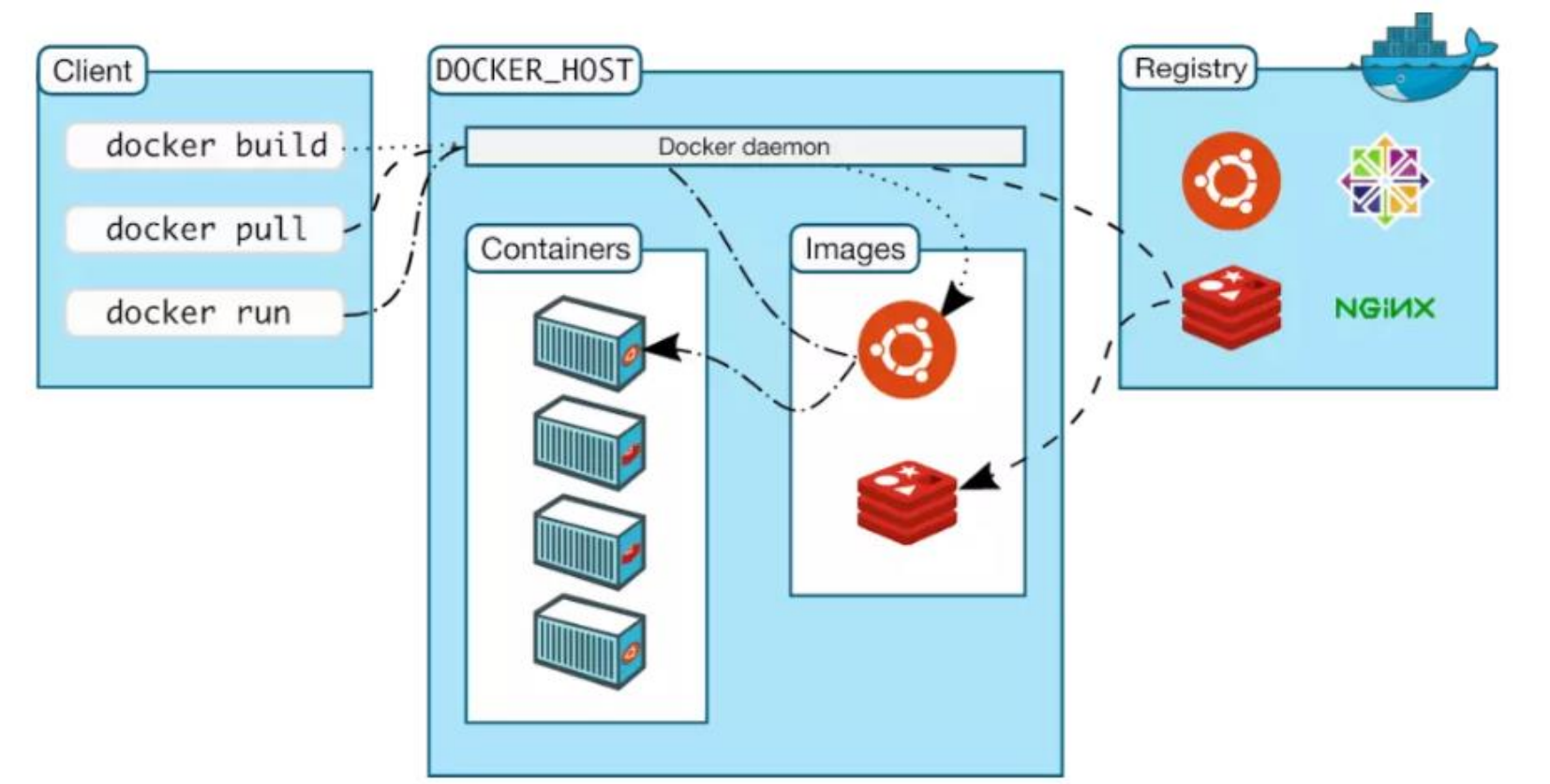
Azure Kubernetes Service 知多少

容器技术的发展历程


Virtual Machine
Docker
Kubernetes
Azure Kubernetes Service

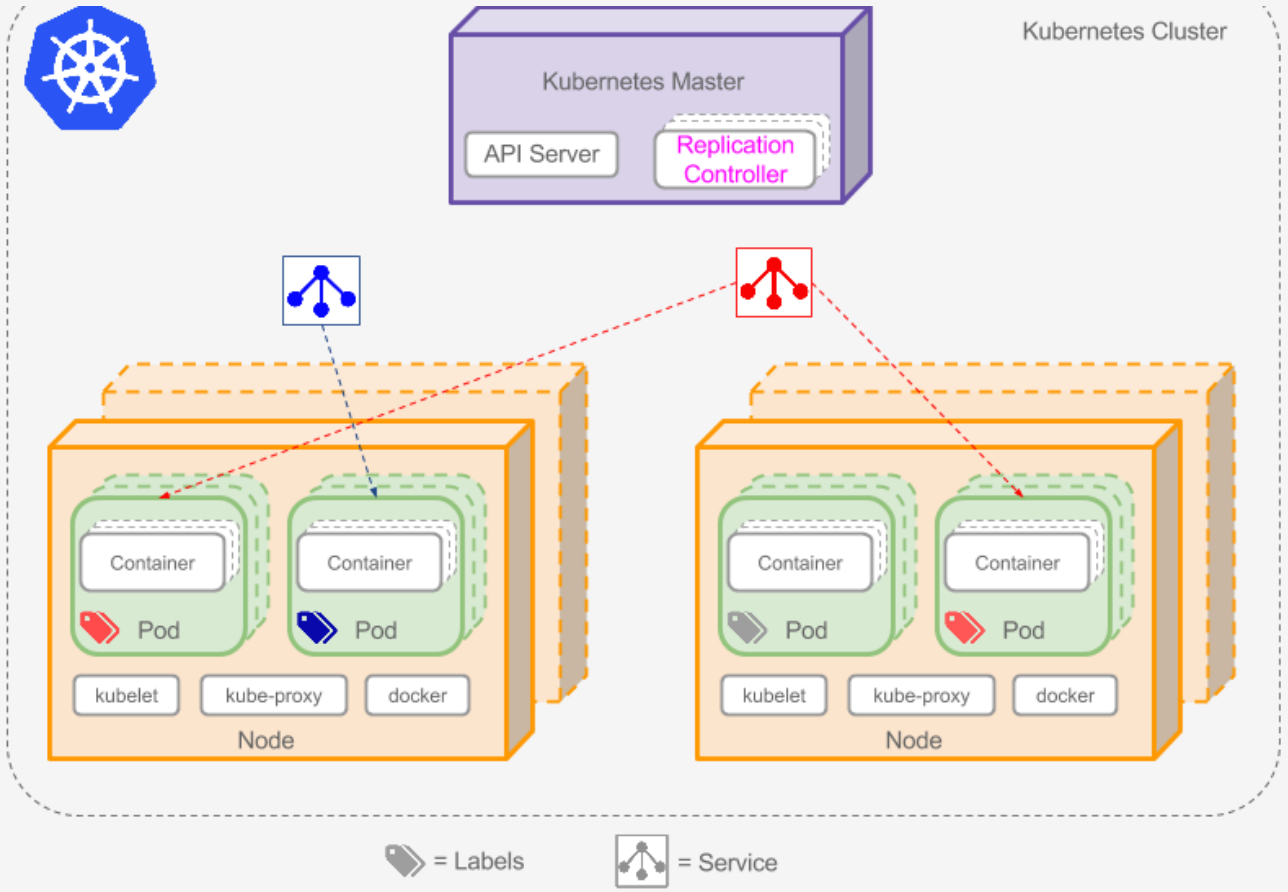
Docker 简介 （Build, Ship and Run Any App, Anywhere)

解决了应用程序从构建到分发，再到运行的**环境**问题



K8S 简介（Automated container deployment, scaling, and management）

- 解决了容器的难于管理、编排和调度的问题



K8S 特性

- 服务发现和负载均衡
- 自动装箱
- 存储编排
- 自愈能力
- 自动部署和回滚
- 密钥和配置管理
- 批处理
- 水平扩容

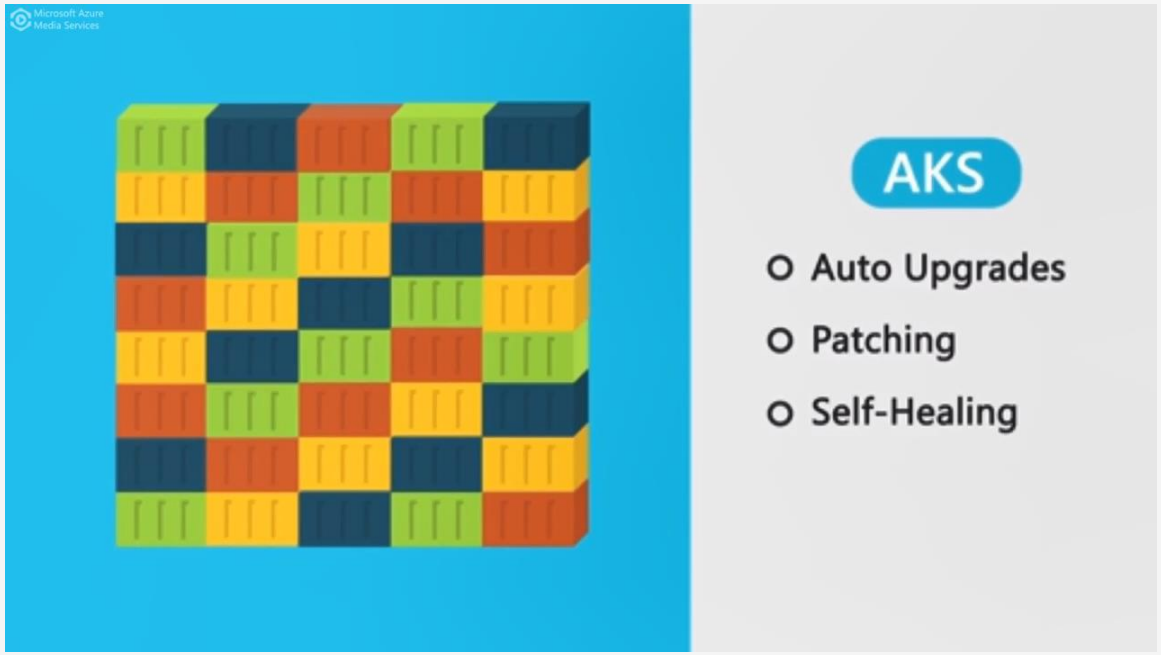
Azure + K8S = AKS

Azure Kubernetes Service (AKS) 简介

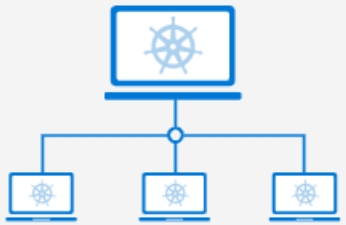
AKS主要解决了跨区域集群的快速搭建和进一步简化K8S的管理

1. 简化Kubernetes管理，部署和运营

2. 使用完全托管的Kubernetes容器编排服务



AKS 特性

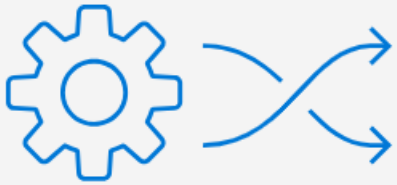


轻松部署和管理 Kubernetes

放心缩放和运行应用程序

保护 Kubernetes 环境

加速容器化应用程序开发

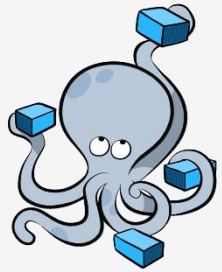


使用开源工具和 API，随心所欲地工作

只需单击几下即可设置 CI/CD

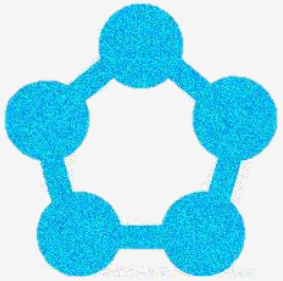
部署eShopOnContainers到AKS

部署方式



Docker-compose

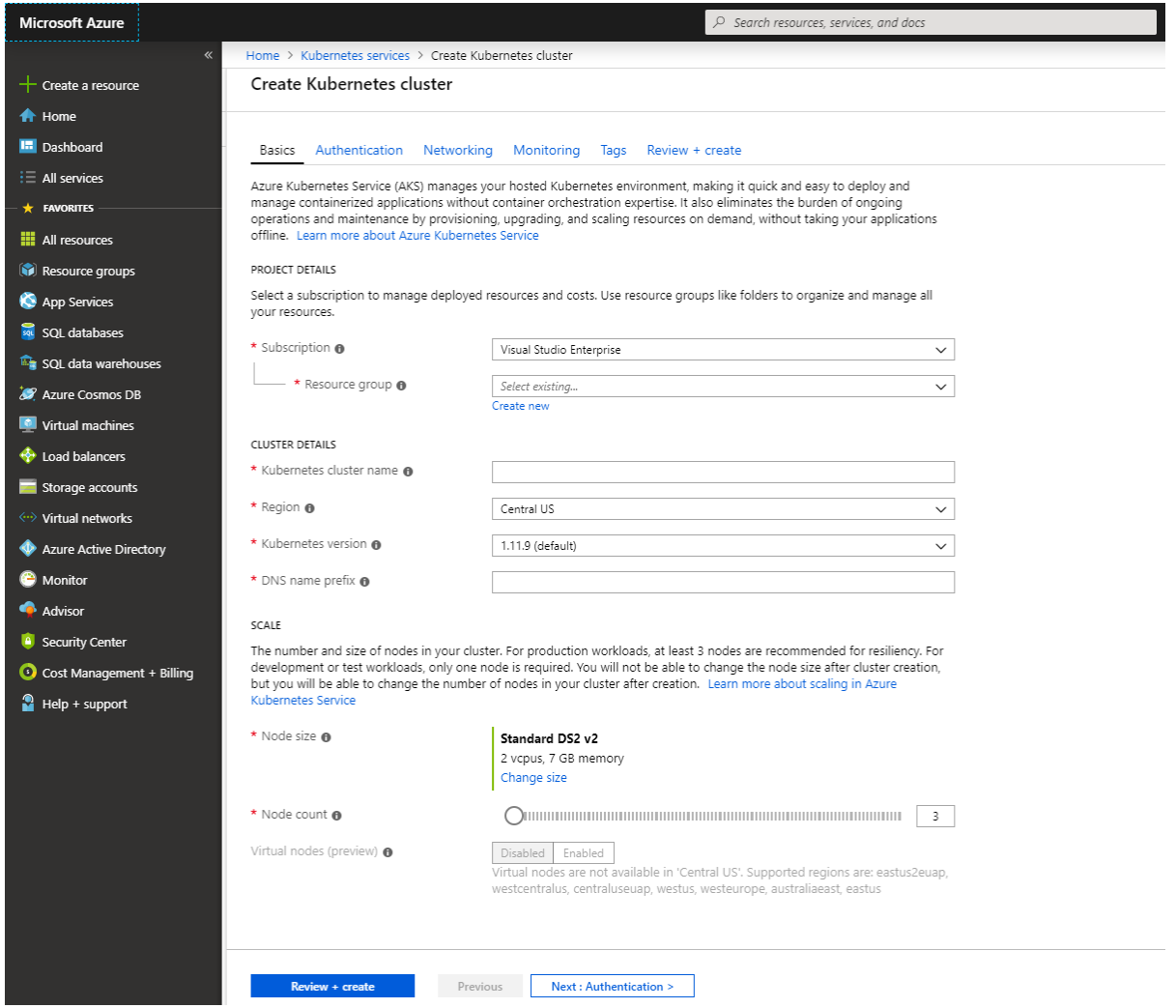
K8S|AKS



Service Fabric

Demo

使用门户创建AKS集群



使用Helm发布到K8S

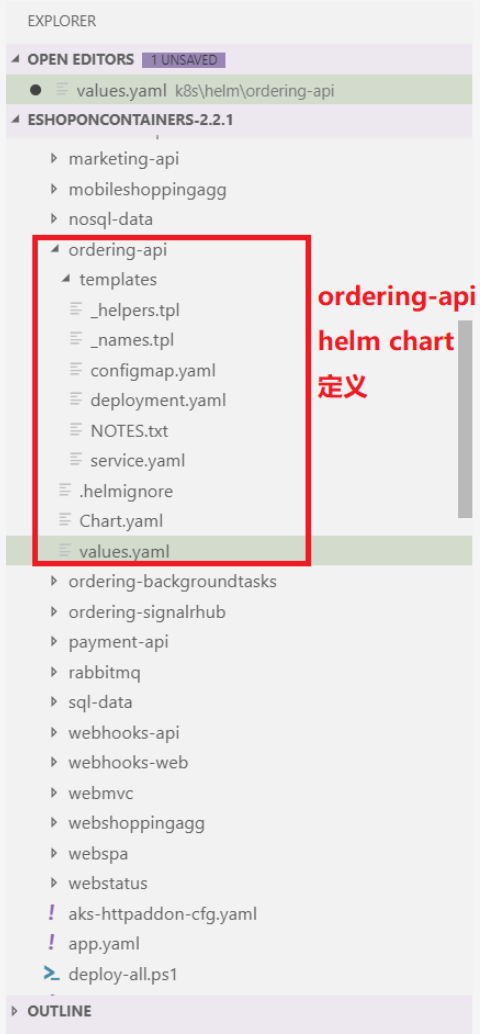
Helm is THE package manager for Kubernetes:

...Kubernetes deployments with just Kubectl.exe and .yml files are not standard but custom & complex...

使用Helm的好处:

- 使应用程序部署变得简单、标准和可重用。
- 易于安装应用程序, 更新, 回滚和删除。包是在 “Helm Chart" 中以声明方式定义的
- Charts可以共享和公开发布 (https://github.com/helm/charts/tree/master/stable)
- 版本控制

Helm Chart



ordering-api helm chart 定义

values.yaml

```yaml
replicaCount: 1
clusterName: eshop-aks
pathBase: /ordering-api

image:
  repository: eshop/ordering.api
  tag: latest
  pullPolicy: IfNotPresent

service: ...
ingress: ...
resources: {}

# env defines the environment variables that will be dec
env:
  urls:
  # configmap declares variables which value is taken fr
  configmap:
    - name: ConnectionString
      key: ordering__ConnectionString
    - name: ApplicationInsights__InstrumentationKey ...
    - name: EventBusConnection ...
    - name: AzureServiceBusEnabled ...
    - name: UseLoadTest ...
    - name: IdentityUrl ...
  values:
    - name: ASPNETCORE_ENVIRONMENT
      value: Development
    - name: OrchestratorType
      value: 'K8S'
probes:
  liveness:
    path: /liveness
    initialDelaySeconds: 10
    periodSeconds: 15
    port: 80
  readiness: ...
nodeSelector: {}
```

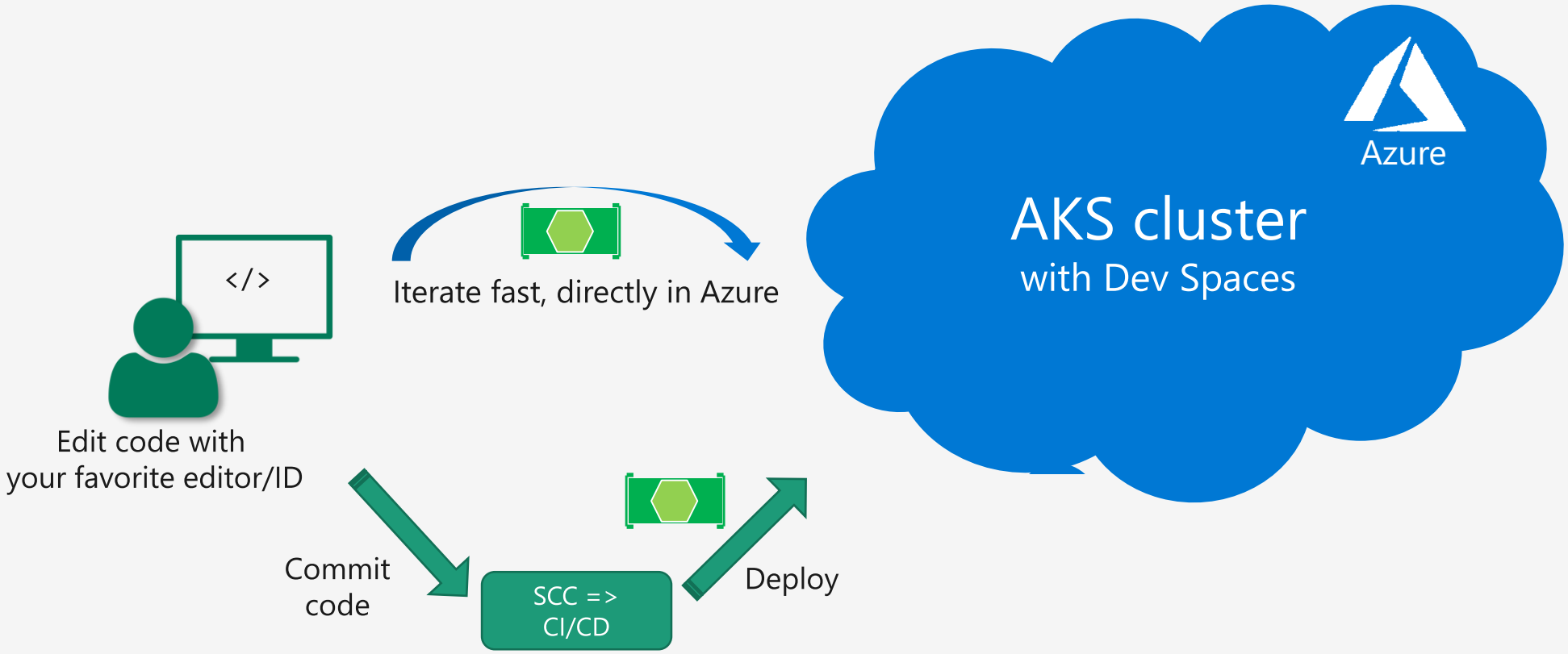
指定使用的镜像

指定环境变量的引用

指定存活探针和就绪探针

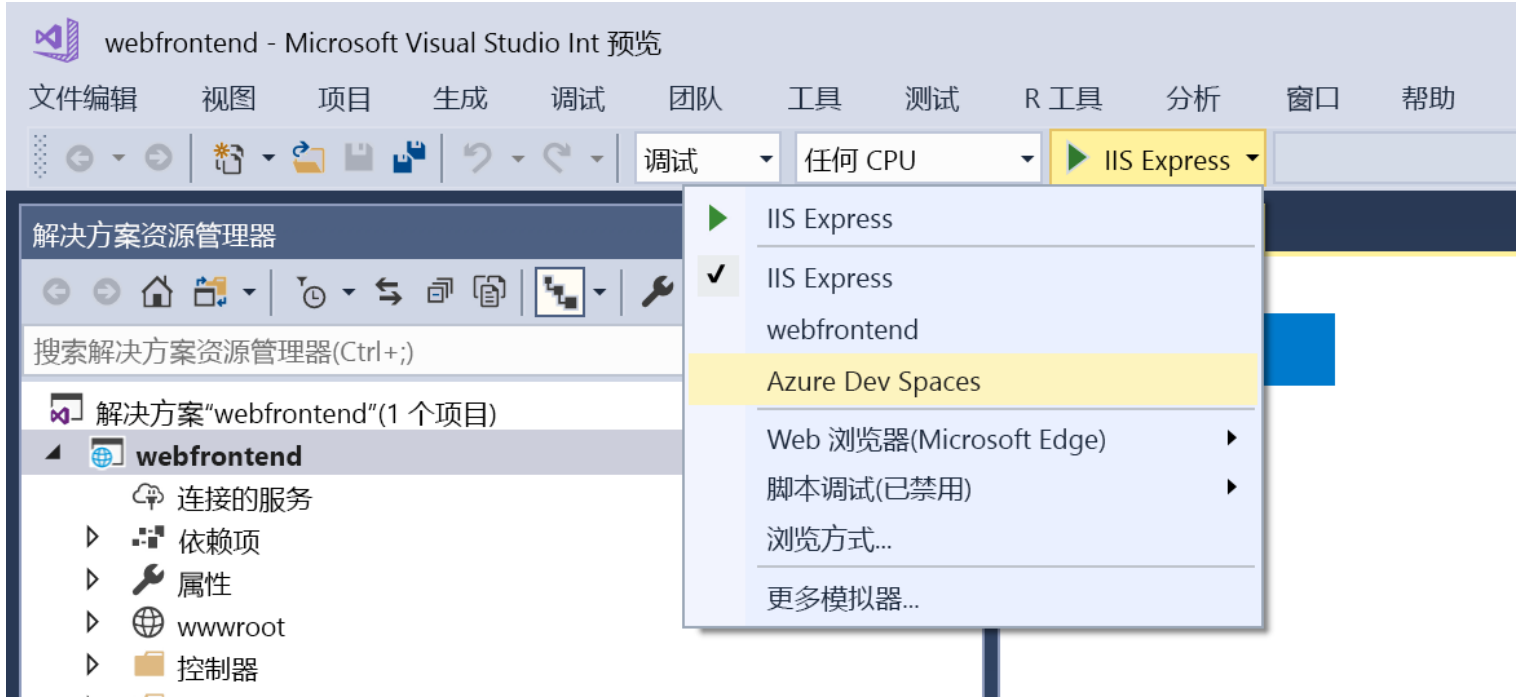
```
$ helm create ordering-api
$ tree ordering-api
mongodb
├── Chart.yaml #Chart本身的版本和配置信息
├── charts #依赖的chart
├── templates #配置模板目录
│   ├── NOTES.txt #helm提示信息
│   ├── _helpers.tpl #用于修改kubernetes objcet配置的模板
│   ├── deployment.yaml #kubernetes Deployment 定义
│   └── service.yaml #kubernetes Serivce 定义
└── values.yaml #kubernetes 配置源
```

Azure Dev Spaces



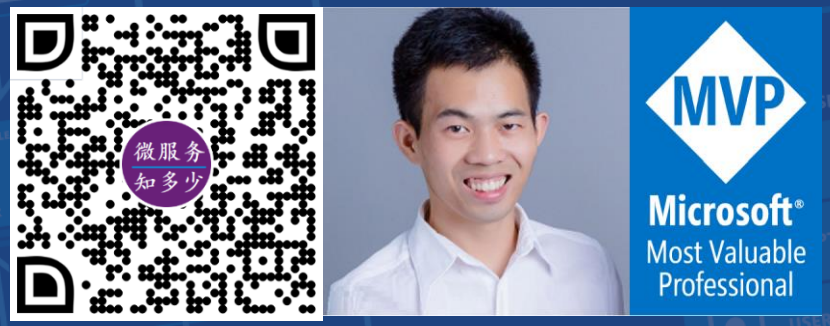
Demo

Azure Dev Space 启用在线调试



Thank you !

颜圣杰
@sheng-jie



STUDY4
Study For Love

2019
Global Azure
BOOTCAMP

特别感谢

STUDY4
Study For Love

MVP Microsoft® Most Valuable Professional

2019 Global Azure BOOTCAMP

广东职业教育信息化研究会

CloudRiches
宏富云信息科技有限公司

.NET foundation