

Azure Service Broker and Helm

让应用部署变得更简单

Rita Zhang

Principal Software Engineer @ Microsoft



ritazh



ritazzhang

Hi, I'm an Open Source Engineer

- 我在美国三藩市工作
- 与开源社区一起工作，解决我们客户的问题
- 我参与贡献的项目：Kubernetes, Cloud Foundry, Azure container service engine, service broker, Helm charts



议题

- Service brokers
- Helm
- Demo

Azure Service Broker (ASB)



让你容器内的应用更容易使用 Azure Service

传统企业中，向运维部门申请新的服务
还是一件挺痛苦的事情

程序员的效率



从头搭建一个能经受住生产环境考验的
数据服务并不容易

更别提去运营，
还有各种认证权限管理



Manage Complexity 管理复杂环境

为什么要开发 Azure Service Broker (ASB) ?

- 基于容器的数据服务
 - 运维太困难和复杂
 - 没有 SLA
- Azure 数据服务
 - Strong operational characteristics
 - 商业 SLA

*ASB 在基于容器的应用和Azure数据服务之间建立了一个桥梁
提高开发者的效率，降低运维复杂度*

开放 Service Broker API

- 最早始于 Cloud Foundry
- 通用的接口/API，横跨所有的 Cloud Native Platforms

支持的平台

- Cloud Foundry
- Service Fabric
- OpenShift
- Kubernetes (AKS)



服务目录



用户通过服务目录可以注册和配置自己的应用所需要的服务

Service Broker



- Advertising service offerings and plans to marketplace
- Acting on requests from the marketplace for provisioning, binding, unbinding, and de-provisioning

目前ASB支持的服务

- Azure CosmosDB
- Azure EventHub
- Azure KeyVault
- Azure SQL
- Azure MySQL
- Azure PostgreSQL
- Azure Redis
- Azure Search
- Azure Service Bus
- Azure Blob Storage



Helm



Helm

Kubernetes的包管理器

为啥要用 Helm?

- Manage Complexity
- Easy Updates
- Simple Sharing
- Rollbacks



目前的进展

Azure Service Broker

- Alpha 阶段，即将发布

Helm

- 已正式发布



WordPress Demo



资源

- Azure Service Broker
 - <https://github.com/Azure/azure-service-broker> (to be released)
- Kubernetes Service Catalog
 - <https://github.com/kubernetes-incubator/service-catalog>
- Helm
 - <https://helm.sh/>
- Helm charts
 - <https://github.com/kubernetes/charts/>
- WordPress chart using Service Broker for to provision Azure Database for MySQL
 - <https://github.com/Azure/helm-charts/tree/master/wordpress>
- 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/>

Q&A