

微软Azure最佳实践

陈品嘉
技术解决方案专家
微软(中国)有限公司



Agenda



- 面向云服务的应用开发
- 架构设计要点
- 代码开发要点
- 部署运维要点
- 典型案例

云时代的应用开发有何不同

编程模型：面向过程 → 面向对象 → 面向组件 → 面向服务 → 面向云服务

代码可移植性：不可移植 → 源代码可移植

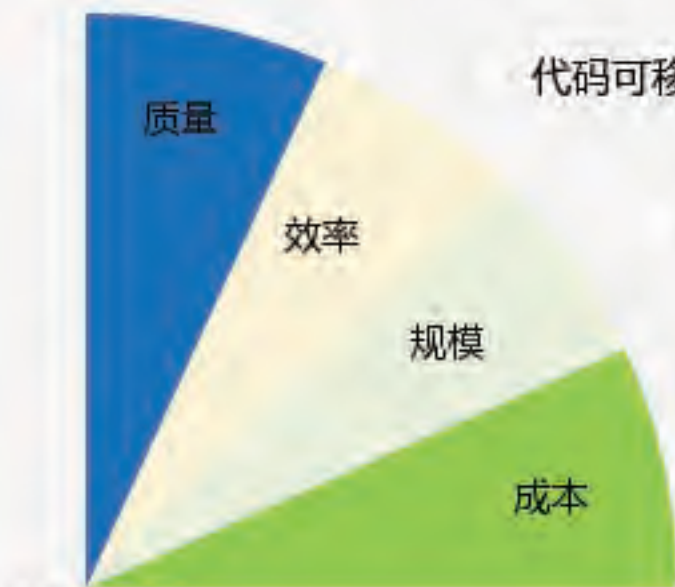
- 编译后可移植
- 即时运行(脚本)

开发效率：瀑布式开发 → 敏捷开发 → 在线开发

运维管理：手动管理 → 自动管理 → 自主管理

扩展方式：垂直扩展 → 水平扩展

软件发布：分发介质 → 分发入口



驱动力

过去

今天

微软云的显著特点

1. Truly Hybrid Solution



2. IaaS & PaaS Flexibility



3. Security, Compliance & SLAs



4. Commercial Benefits



5. Proven track record



6. Hyper-Scale & Innovation



7. Support for Microsoft Workloads



8. Open & Flexible



微软云服务概览

<http://azure.microsoft.com/en-us/services/>

Platform

- Classic PaaS
- Cache
- Service Bus
- Service Fabric
- Web Worker
- Docker

Apps

- App Service
- Web Apps
- API Apps
- Mobile Apps
- Mobile Engagement
- Visual Studio
- VS Online

DevOps

- VS Online
- VS Online

Integration

- Logic Services
- API Management

Media

- Media Services
- Direct Connect

Big Compute

- Azure Data
- HPC Pack

IoT

- Azure IoT Suite

Identity

- Azure AD
- Cloud Identity
- MFA
- RM

Management

- Automation
- Application Insights
- Cloud Management
- Operational Insights
- Key Vault
- ARM
- Portals
- Hybrid Tools
- Open Stack

Marketplace

- SAP
- IBM
- ORACLE
- Box
- CloudStack
- django

Database

- Database-as-a-service
- SQL Database
- DocumentDB

Analytics

- Big Data
- ML
- Decision Analytics
- ADW
- Data Factory
- Stream Hub
- Search
- Data Lake

Storage

- Azure Storage Account
- Blob
- Queue
- Table
- Queue
- Premium
- Files

Infrastructure

- Classic IaaS
- Compute
- Network
- Storage
- Load Balancing

Connectivity

- Connectivity
- Global Reach
- Internet VPN
- IPSec VPN
- SSL VPN
- ExpressRoute
- Traffic Manager
- Azure DNS

Hybrid

- Backup
- Site Recovery
- Workload
- Migration Accelerator
- Azure Stack

合规 - 微软云获得的认证

- Certifications covering physical security, operational security, incident management & response, monitoring and logging, update management, penetration testing & DDoS protection.



- ISO 27018 is the first international set of PII privacy controls in the cloud.
- Microsoft is the only cloud provider to receive joint approval from the European Union's Article 29 Working Party for our strong contractual commitments to comply with rigorous EU privacy laws.

架构设计需要考虑的因素：Azure上的基本常数



Azure Compute (Web, Worker, IaaS)

- 1-8 CPU cores
- 2-14 GB RAM
- 5-800 Mbps network

- 单实例SLA: 99.9%
- Role SLA: 99.95%



Azure Storage单账户

- 200 TB storage
- 20000 操作/秒
- In 5/Out 10 Gbps异地复制
- In 10/Out 15 Gbps本地复制

- SLA: 99.9%



Azure SQL

- 150 GB
- 305 threads
- 400 并发请求

- SLA: 99.9%

架构设计的基本要点

单一资源存在性能和容量瓶颈

应用规模可能快速增长

单一资源可用性有限

站点存在停机风险

分布式系统难于管理



采用多节点水平扩展

模块之间松耦合

采用多活方式构建集群

将应用部署在多个站点

采用自动管理或自主管理

可扩展性

可用性与容灾

可管理性

以下哪种是好的基于云平台的应用架构

Web Role通过TCP直连
Worker Role

安装数据库到两个VM上，
配置成主备形式

Session状态存储在单机
内，不做复制。负载均衡
启用Session Affinity

Web Role将用户数据存
储在本地磁盘以提高性能

Azure上各种服务和技术的对比

推荐使用

不推荐使用

服务	单站点可用性	容灾级别	可扩展性	易用性
结构化数据				
SQL Azure	高(3实例互为冗余)	中(数据容灾)	高(通过Federation水平扩展)	高(0部署)
在IaaS上建DB	低(需手动配置HA)	低(手动备份)	低(手动配置集群)	低(手动配置/优化)
半结构化数据				
Azure Table	高(3份拷贝)	高(RPO=30秒, RTO接近0, 对用户透明)	高(通过Partition水平扩展)	高(0部署)
开源代码, 如 MongoDB, Cassandra	高(多份拷贝)	中(数据容灾)	高(水平扩展)	低(手动配置/优化)
非结构化数据				
Azure Blob	高(3份拷贝)	高(RPO=30秒, RTO接近0, 对用户透明)	高(通过Container/Account水平扩展, 通过CDN提升并发)	高(0部署)
本机磁盘	低(无冗余)	低(无容灾)	低(无法扩容)	高(成熟)

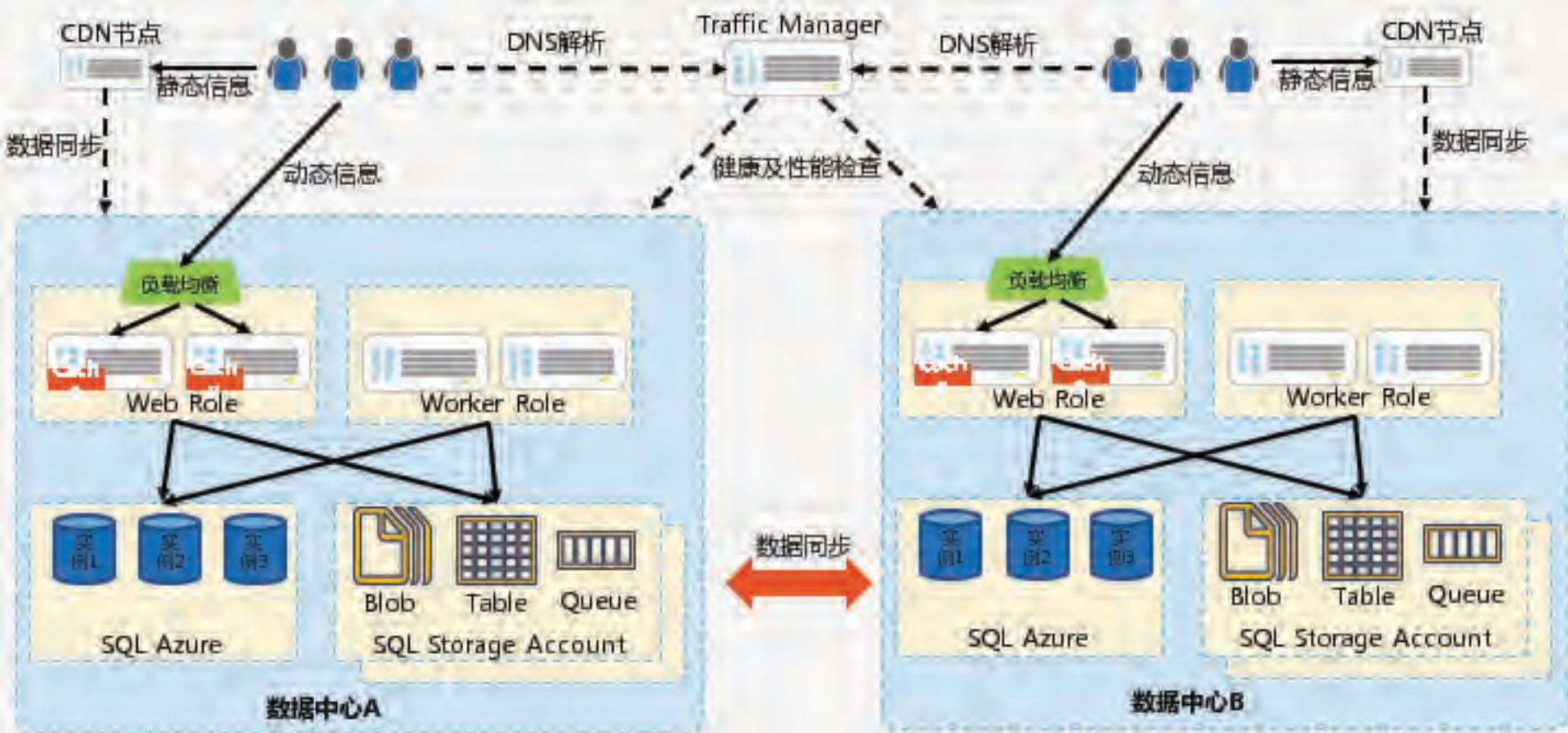
Azure上各种服务和技术的对比(2)

推荐使用

不推荐使用

服务	单站点可用性	容灾级别	可扩展性	易用性
业务逻辑				
Cloud Service/Website/Mobile	高(集群)	中(通过Azure Traffic Manager实现灾难切换)	高(集群水平扩展)	高(自动/自主管理)
IaaS	高(集群)	中(通过Azure Traffic Manager实现灾难切换)	高(集群水平扩展)	低(手动部署/配置)
消息				
Azure ServiceBus/Queue	高(集群)	高(RPO=30秒, RTO接近0, 对用户透明)	高(通过Account水平扩展)	高(0部署)
传统消息队列	低(手动配置集群)	低(无容灾)	低(手动配置集群)	低(手动部署/配置)
网络				
域名	高(静态指定)	中(通过Azure Traffic Manager实现灾难切换)		
IP	低(动态分配)	低(不可跨站点)		

典型的基于Azure的高可用、高可扩展应用架构



一个例子：PotterMore

<http://www.microsoft.com/en-us/news/features/2012/jun12/06-06Pottermore.aspx>

APMCon



上线3个月1百万注册用户

上线两周10亿点
击量(峰值1.1亿
/天)

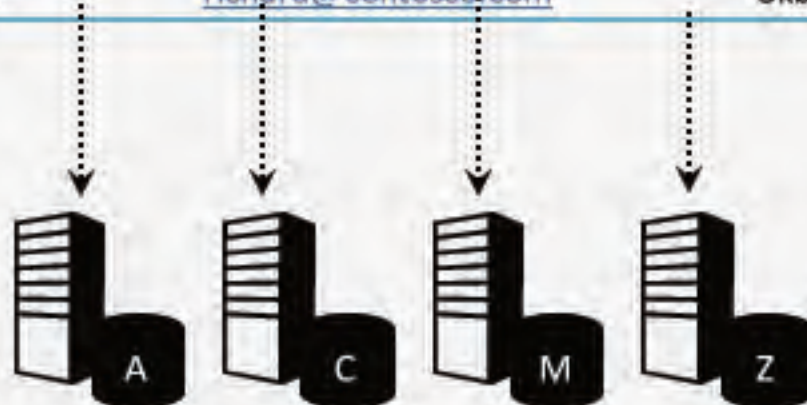
每天新注册
25000用户

1000个核

500个
数据库

如何做数据切分进行水平扩展？水平切分

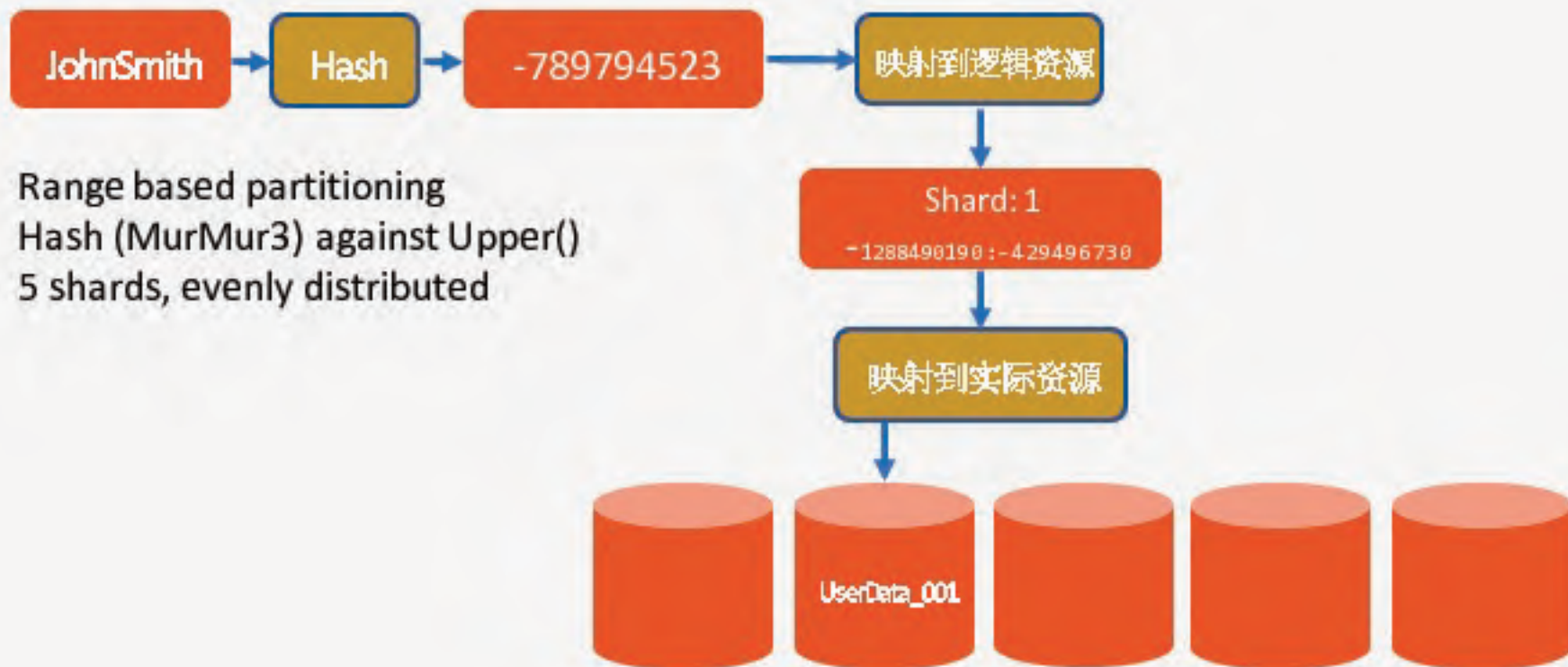
First Name	Last Name	Email	Thumbnail	Photo
David	Alexander	davida@contoso.com	3kb	3MB
Jarred	Carlson	jaredc@contosco.com	3kb	3MB
Sue	Charles	suec@contosco.com	3kb	3MB
Simon	Mitchel	simonm@contoso.com	3kb	3MB
Richard	Zeng	richard@contosco.com	3kb	3MB



First Name	Last Name	Email	Thumbnail	Photo
David	Alexander	davida@contoso.com	3kb	3MB
Jarred	Carlson	jaredc@contosco.com	3kb	3MB
Sue	Charles	suec@contosco.com	3kb	3MB
Simon	Mitchel	simonm@contoso.com	3kb	3MB
Richard	Zeng	richard@contosco.com	3kb	3MB



切分规则举例



Range based partitioning
Hash (MurMur3) against Upper()
5 shards, evenly distributed

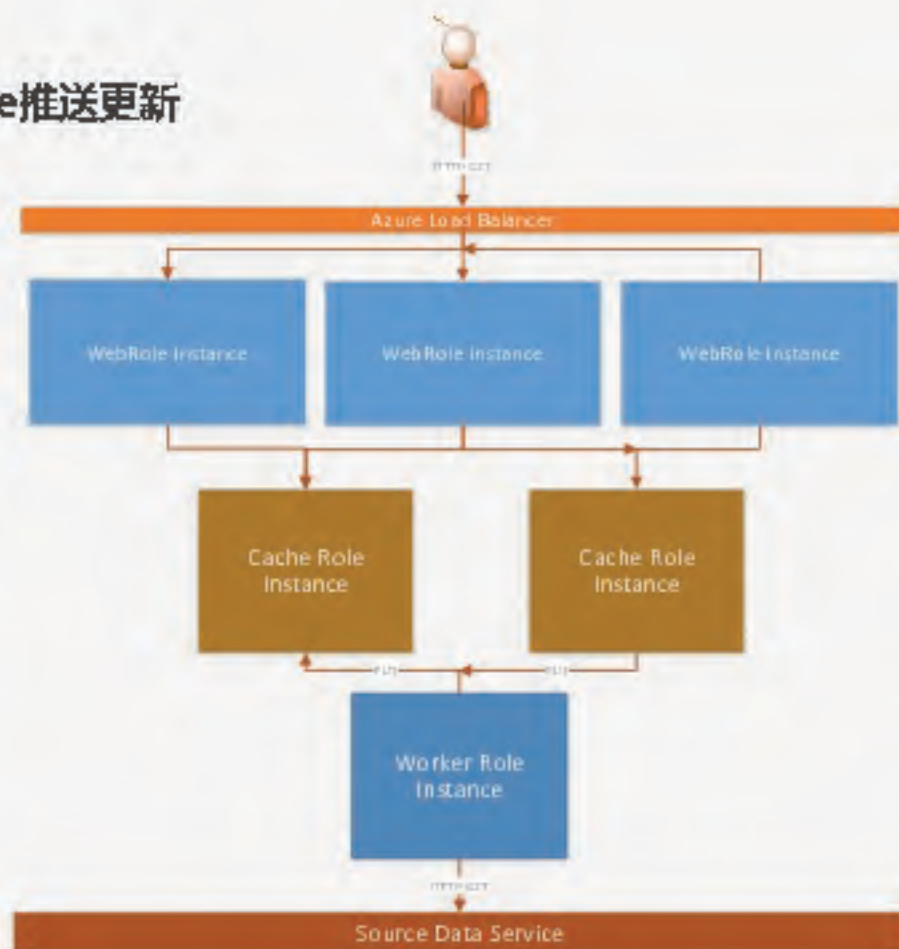
另一个例子：一个网络媒体

采用Cache服务缓存数据，用Worker Role推送更新

8个站点

每天20亿
点击

每秒5万请
求



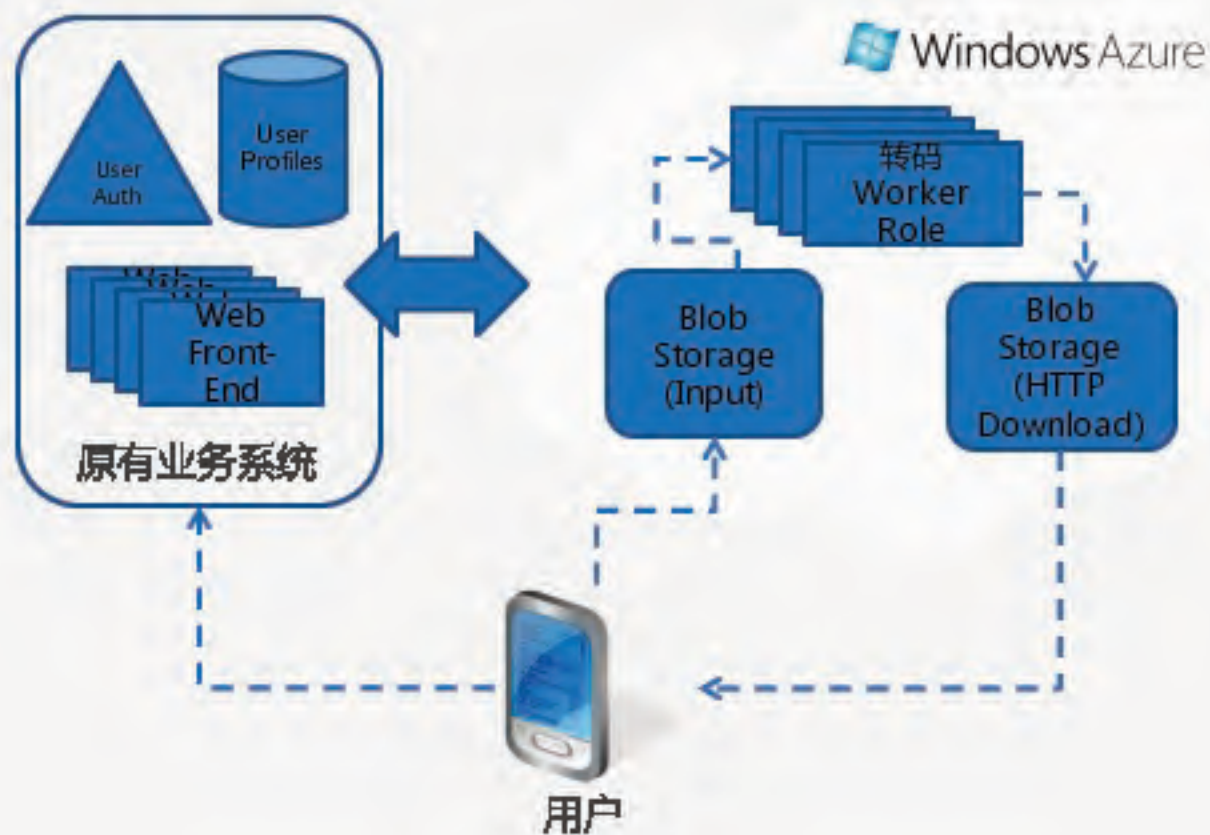
选择了Azure Cache而不是Memcached

	Memcached	Azure Cache
Deployment Model	<ul style="list-style-type: none">Provisioned as custom code in worker role	<ul style="list-style-type: none">Provisioned as part of your service via SDK\ToolsDistinct role or part of existing role
Scale Approach	<ul style="list-style-type: none">Add\Remove role instancesMultiple Worker Roles	<ul style="list-style-type: none">Add\Remove InstancesMultiple Roles
Partitioning Approach	<ul style="list-style-type: none">Client Driven – some APIs have a choice of hashing algorithms	<ul style="list-style-type: none">Server driven
Client APIs	<ul style="list-style-type: none">Many 3rd party APIs	<ul style="list-style-type: none">Microsoft provided .Net APIMemcached interoperability
Client Side Cache	<ul style="list-style-type: none">No	<ul style="list-style-type: none">Yes – can work with memcached clients
Manageability	<ul style="list-style-type: none">Managed as custom roleRequires custom code to expose server stats	<ul style="list-style-type: none">Managed as custom rolePerformance counters available
High Availability	<ul style="list-style-type: none">No	<ul style="list-style-type: none">Windows Azure awareHigh Availability options available

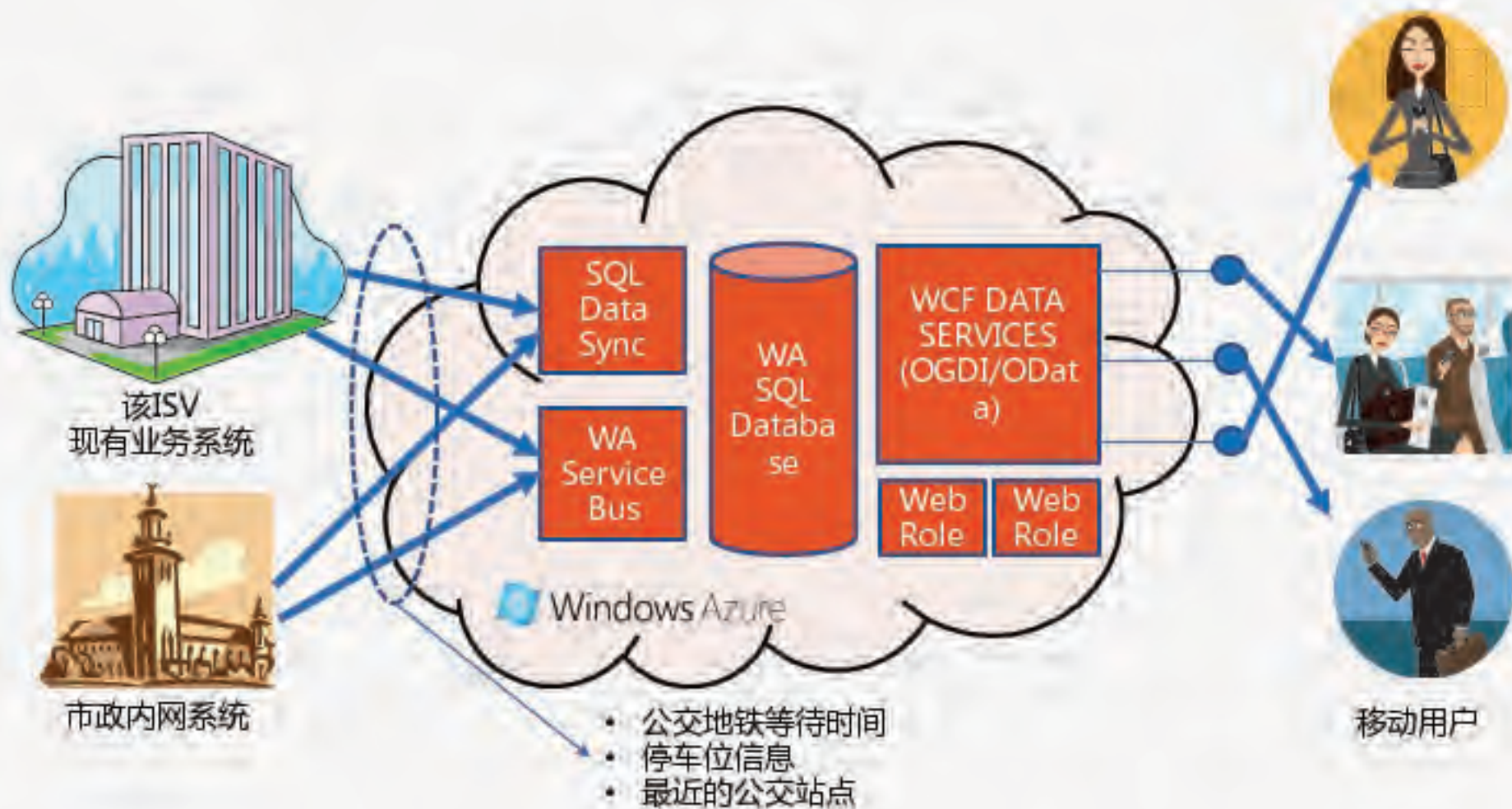
如何根据自己的需求选择合适的Azure服务? APMCon



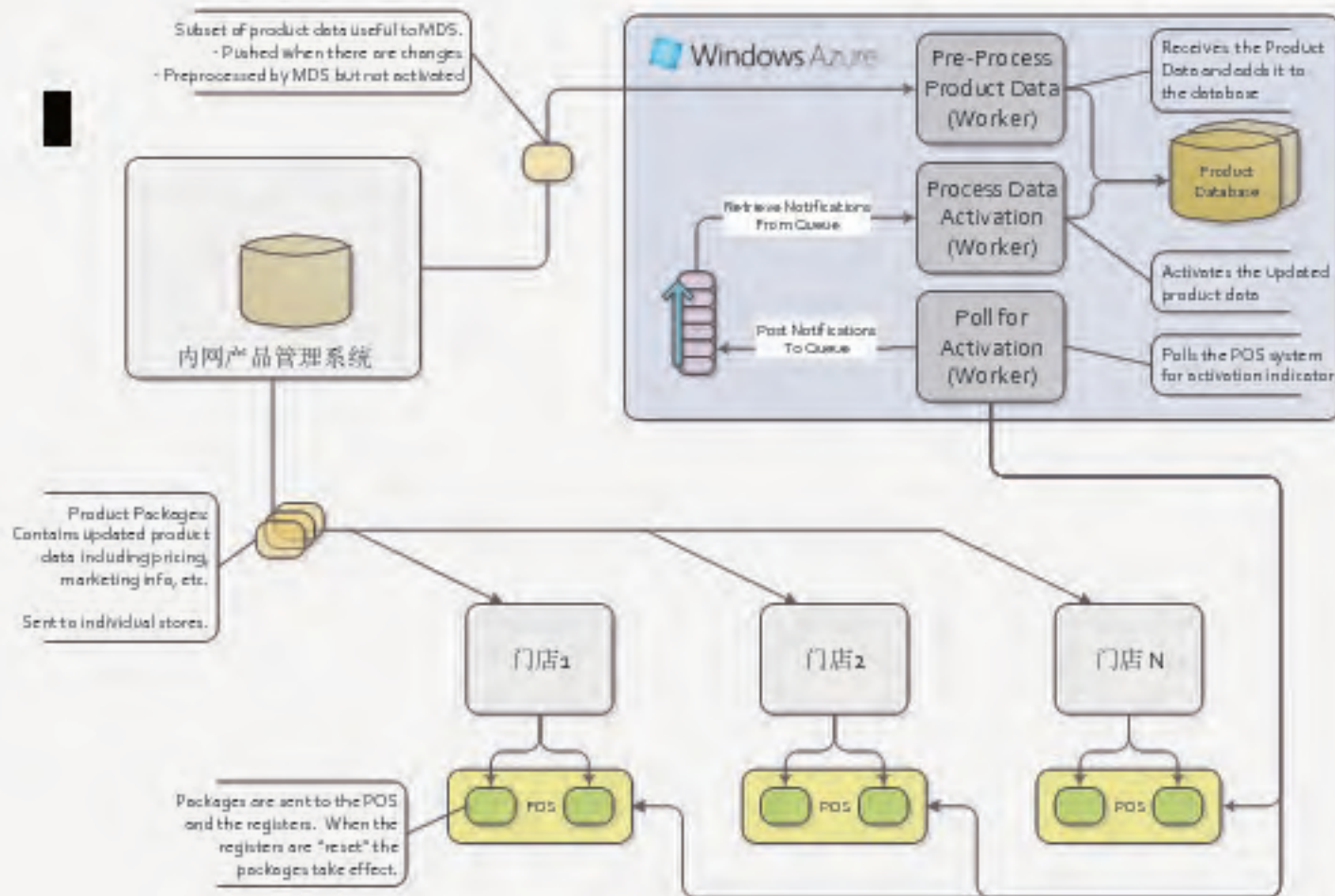
应用实例：欧洲电信运营商为用户提供网盘 APMCon



应用实例：为公众用户提供城市交通信息



应用实例：零售企业门店信息发布



应用实例：实现150万辆车的云端监控



存在多个实例同时运行

实例可能随时创建销毁

IP地址是变化的

单一资源可用性有限

分布式系统部署复杂



处理好对同一数据的并发访问

区分需持久化的数据和临时数据

引用外部服务时, 使用域名

使用合适的重试策略

自动化编译、测试、打包、上传过程

可扩展性

可用性

可管理性

在遇到服务异常时使用合适的重试策略

[http://msdn.microsoft.com/en-us/library/hh680934\(PandP.50\).aspx](http://msdn.microsoft.com/en-us/library/hh680934(PandP.50).aspx)

任何服务可能会出现一些临时性故障：

- 负载突然增加导致临时阻塞
- 硬件故障、网络故障、软件故障
- 人为故障

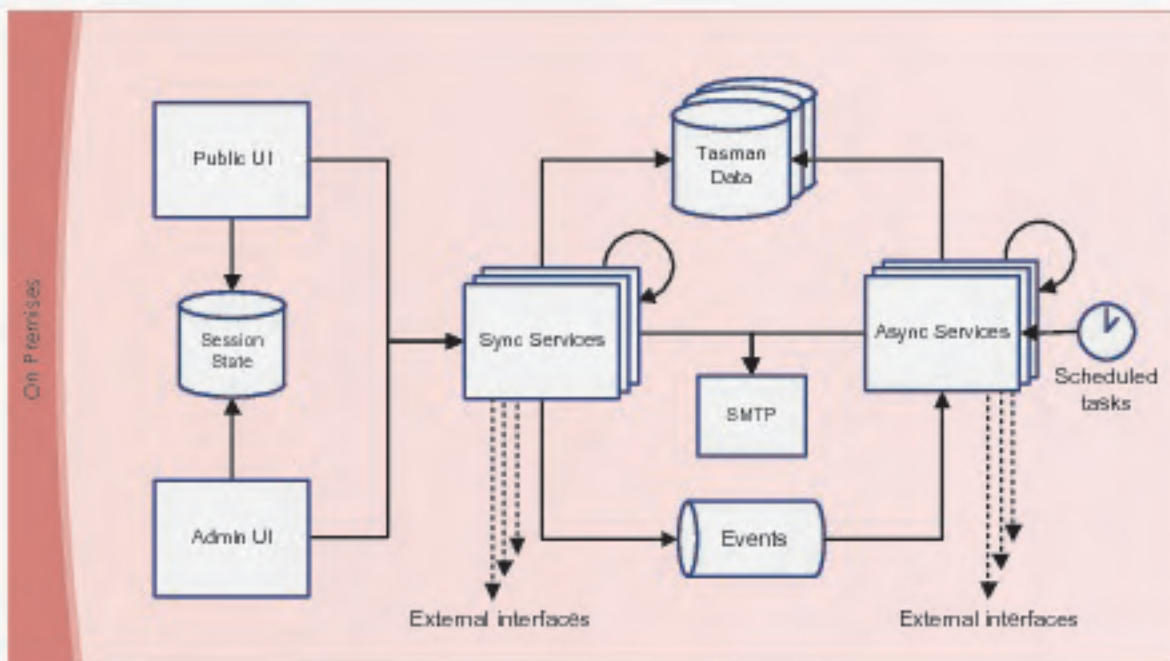
Azure提供了一个代码框架(Transient Fault Handling Application Block)按策略进行重试

```
public void DemoHandleTransient()
{
    var connString = "some database connection string";
    var policy = RetryPolicy.Create<AzureSqlDatabaseError>({
        retryCount: 3,
        delay: TimeSpan.FromSeconds(5));

    policy.Execute(() =>
    {
        using (var conn = new SqlConnection(connString))
        {
            conn.Open();
        }
    });
}
```



应用迁移实例——原应用

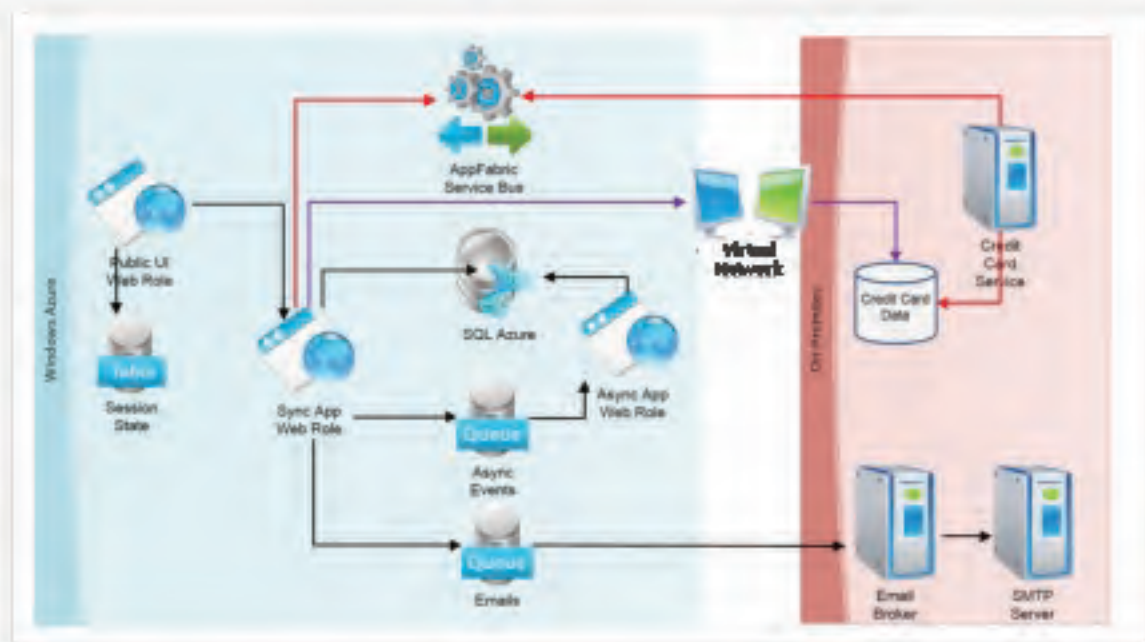


代码变化

- 修改Worker Role的任务触发逻辑
- 将敏感数据存放在企业内网
- 改变发邮件的处理方式
- 更换Session的Provider
- 使用Azure的Diagnostics

配置修改

- 将部分配置信息从web.config移到.cscfg



实现应用自动化发布

- 代码：打包上传至TFS或Azure Blob
- 配置信息：写入.cscfg或者将配置文件写入Azure Blob
- 数据：在Azure Blob上备份
- 编写安装配置脚本，实现Web/Worker Role启动后自动配置

实现应用的自动化备份

采用平滑的应用升级

- 利用Cloud Service VIP切换功能实现生产环境和Staging环境的切换

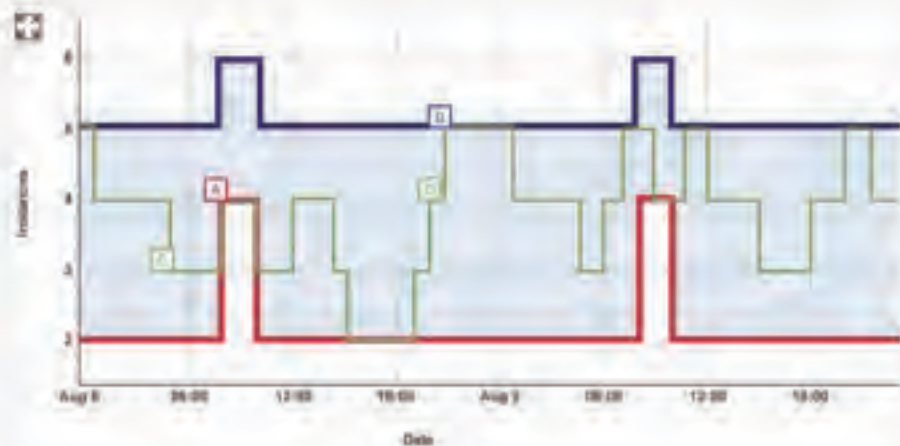
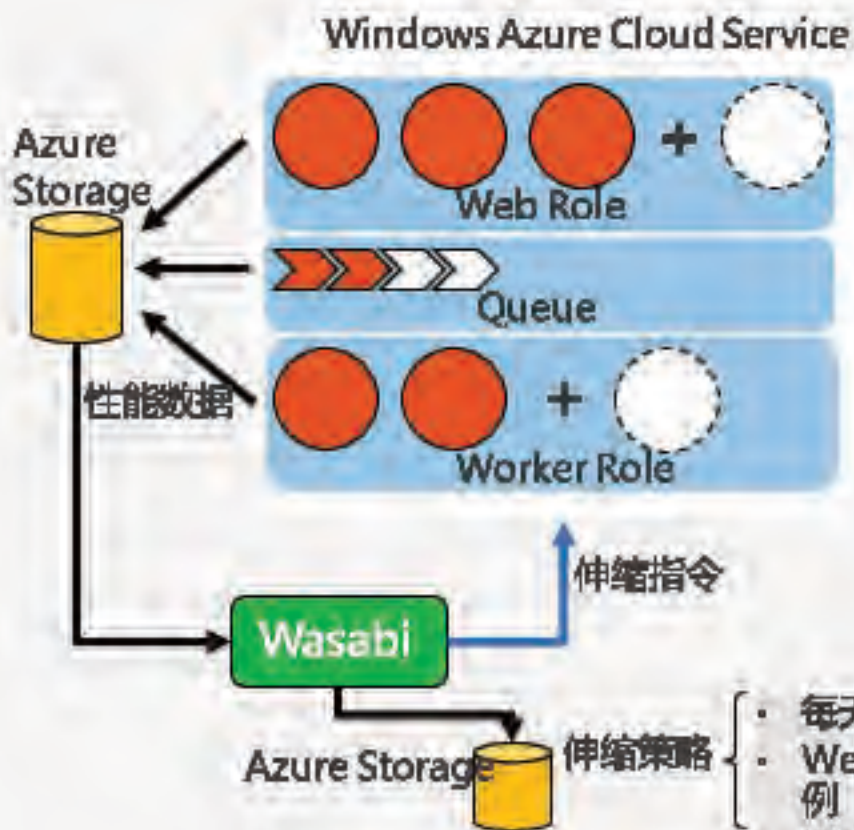
建立监控机制

- 采用管理门户进行监控
- 采用工具进行监控 (System Center, New Relic, AppDynamic)

弹性伸缩模块(Wasabi)可实现自主管理

介绍[http://msdn.microsoft.com/en-us/library/hh680892\(v=pandp.50\).aspx](http://msdn.microsoft.com/en-us/library/hh680892(v=pandp.50).aspx)

下载<http://www.microsoft.com/en-us/download/details.aspx?id=28189>



- 每天早上8至10点, Web Role最少要4个, 最多6个, 其余时间..
- Web Role平均CPU利用率大于50%时, Web/Worker各增加1个实例
- Web Role平均CPU利用率小于20%时, Web/Worker各减少1个实例

THANK YOU

