

.NET新思维

Spring Cloud SteelToe + .NET Core微服务架构



.NET新时代开源社区CTO联盟

QQ群 : 618093978

.NET工程师/架构师/CTO 面临的困惑

扎心了，老铁！

- 听说某某公司从.NET转Java了？我们要不要跟风转Java？
- 为何自己每天都在做简单重复的CRUD增删查改？这样的“工作经验”几年后还有价值吗？
- 为何.NET高端职位（20K以上的）这么少？
- 我们公司不差钱，为何招聘靠谱的.NET架构师/工程师这么困难？
- .NET如何开发每秒处理10万单的电商平台？
- .NET有没有成熟先进的开源微服务框架？
- 为何各种技术大会上演讲高大上PPT的大牛很少.NET背景的？
- 为何微软技术大会上很多宣传某某产品多么牛逼，却很少深度技术分享企业真实架构实战？
- 为何有些企业去微软大会上分享，却只是营销宣传PPT，没有分享架构实战干货/开源组件？
- 为何.NET开源社区很少看到真正有影响力的开源架构、开源组件、实战经验分享？

.NET工程师/架构师/CTO 面临的困惑

继续扎心了，老铁！

- 没有兴趣学习新一代.NET Core on Linux/Docker、MySQL、RabbitMQ…… 会不会被淘汰？
- 公司规模较小、业务简单，没有亿级流量、高并发，但不用加班工作轻松，该不该跳槽？
- 国内除了自己所在的公司，是否已经没有其它公司用.NET了？
- .NET Core与Java相比，有哪些优缺点？
- .NET与.NET Core本质的区别是什么？学习曲线高不高？多久能学会？
- 自己学历不行，高薪的公司好像都要学历，怎么办？
- 老板要求公司项目全换成开源免费的数据库、服务器，怎么办？
- 新技术、新框架不断出现，感觉快要学不过来，怎么办？
- 英文水平不行，最新国外技术文档看着很吃力，怎么办？

怎么办？怎么办？怎么办？



分析归类前面的困惑

- .NET成功案例，深度技术架构分享
- 开源社区交流、技术互助
- 开源微服务架构、开源组件
- 社文化.NET招聘、求职渠道（开源社区公众号、QQ群、微信群、技术博客）
- .NET Core on Linux/Docker、Xamarin、Azure学习路线图
- 高校.NET公开课、培训
- 微软、开源社区深度合作

为何很少看到.NET大型成功案例分享？

- 内心错觉：国内除了自己所在的公司，是否已经没有其它公司用.NET了？
 - 真实情况：.NET Core on Linux/Docker在腾讯、网易、同程旅游、龙珠直播.....正式项目使用
 - 成功案例不是没有，是很少分享
-
- 解决方案：
 - 写技术文章、博客分享自己的实战经验、开源组件
 - 加入.NET开源社区、QQ群、微信群，分享技术、求助、招聘、求职
 - 关注知名.NET公众号、网站

.NET Core 适用场景



Desktop



Web and Server



Mobile



Gaming



IoT



AI

使用.NET技术的知名公司、大型网站

- stackoverflow.com（全球最大的程序员社区）
- Azure.com、Azure.cn（世界500强公司90%在使用微软云）、Bing.com搜索引擎、Office365
- 腾讯（互联网金融）、网易绝代双骄游戏事业部、同程旅游、北森云计算、青客公寓、
- 新东方教育科技集团、英孚教育、文都教育、启德教育、跨考教育、昂立教育、云学时代、雄鹰教育、欧佳龙教育、爱迪科森、
- 易车网、中青旅、房多多、碧桂园云平台、中原地产、速8酒店、汉庭酒店、石基信息、广联达、金科股份、国双科技、
- 招商银行、工商银行、华夏银行、花旗银行、东方财富、泰康人寿、宜信、
- 联想利泰、神州泰岳、葡萄城信息、金和软件、三五互联、明基逐鹿、太极集团、
- 达实智能、悦畅科技、新大陆、科通芯城、智联招聘、爱康国宾、慈铭体检、新媒传信、
- 屈臣氏、春秋航空、中国飞机租赁集团、农信科技、大北农集团、同方威视、蓝色星际、
- 掌上通、华夏鑫彩、普华永道、群硕软件、文思海辉、小马科技、亲加通讯云、起点中文网
- 以上公司有的是混合使用多种编程语言，微服务架构+.NET Core/Java/PHP/Go……
- 更多公司列表：<http://www.cnblogs.com/vNext/p/4464466.html>

.NET CORE与JAVA分析比较

	.NET Core	Java
编程语言（语法特性）	C#语法领先、升级进化快、新版本受欢迎	升级进化较慢、Java新版本采用率低
部署到Linux、Docker	.NET Core支持Linux、Docker	支持
原生iOS开发	支持，Xamarin C#原生跨平台	不支持
原生Android开发	支持，Xamarin C#原生跨平台	支持
跨平台游戏开发	支持，Unity3D C#（知名游戏：王者荣耀）	不支持
Desktop桌面应用开发	支持，主流桌面开发平台	支持，但很少人用
IoT、AI、Web、云原生	支持	支持
互联网行业	较少	较多
垂直行业	较多	中等
技术大会成功案例分享	几乎看不到	很多
母公司黑科技实力	微软量子计算、AI、Azure云、MR……领先	Oracle云计算/AI较弱、Java专利费风险
开源社区生态（核心问题）	国内.NET开源生态很弱，全球开源社区活跃	最大优势、开源生态丰富

.NET CORE核心问题（开源社区生态）是否无解？

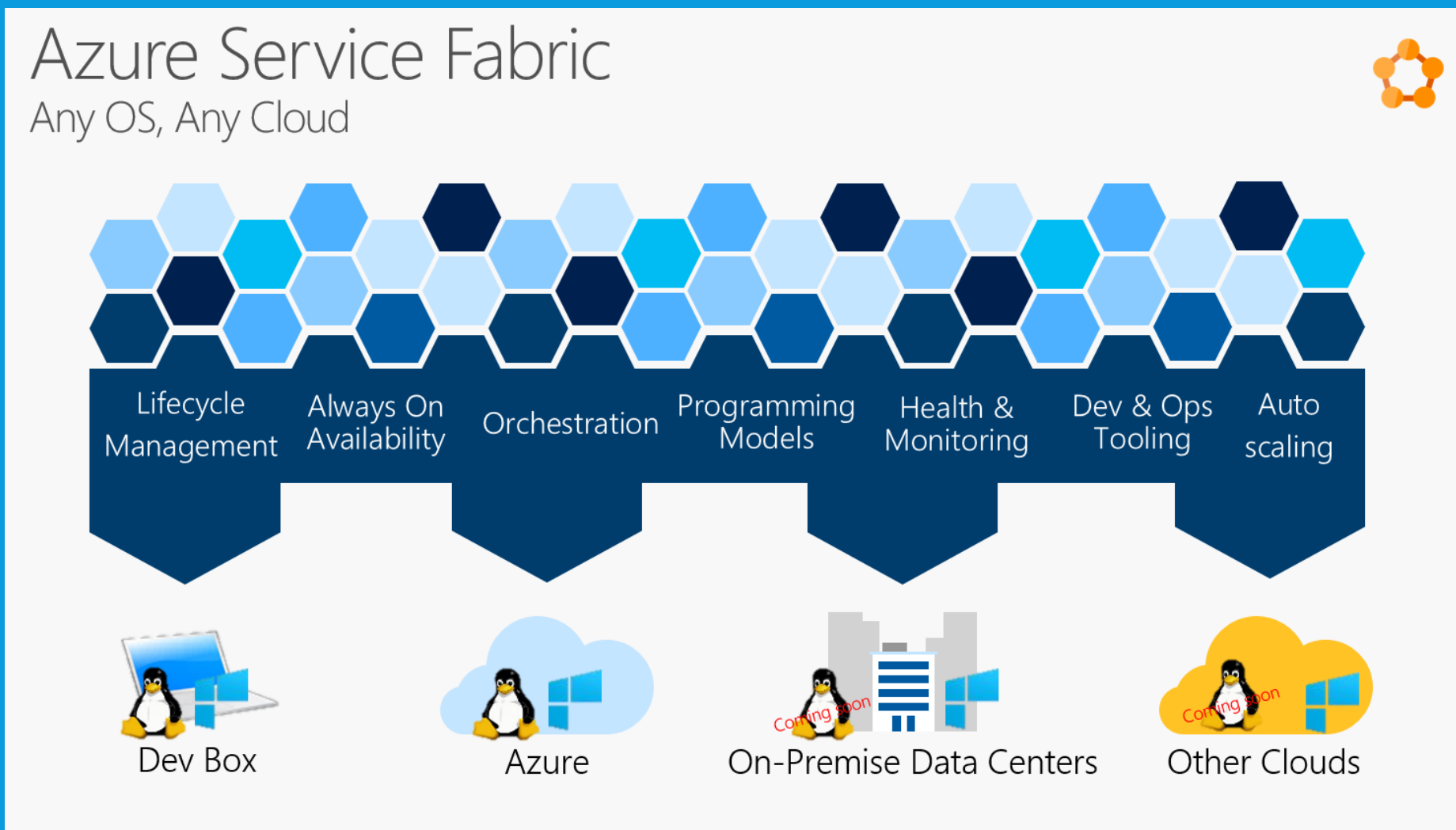
- 开源生态 -> 企业使用开源组件-> 成功案例分享-> 人才招聘-> 高校、社区学习热潮-> 获得高薪职位-> 开源大牛-> 开源生态更繁荣（良性循环，完美）
- 再加上微软黑科技助攻（Visual Studio 2018支持量子计算、AI、MR、云计算.....）
- 整个.NET生态，其实本质问题就是第一步：开源生态如何丰富？
- 答：先完成一个小目标，开源一个主流、成熟的微服务框架+实战案例：
 - Spring Cloud SteelToe + .NET Core/Java/PHP on Linux/Docker + Redis + MySql + PostgreSQL + RabbitMQ/Kafka
- 参考Go语言社区是如何从零开始、不断壮大，某些领域开始抢Java饭碗
- 既然Go开源社区从零开始都能做到，难道.NET社区这么多人都只会增删查改吗？

为何要使用微服务架构？

- 几乎所有大型系统发展多年后，都会到达一个瓶颈：
- 系统越来越庞大臃肿、需求响应越来越慢、
- 模块越来越多，互相关联，经过几波人员变动后，代码质量越来越差
- 双11、大促销等重要场合，不断掉链子，系统不断崩溃，老板也快奔溃，准备挖一个更牛的CTO，终于挖来牛逼CTO，大刀阔斧重构系统（推翻重做）
- 将复杂庞大的单体架构、拆分模块、微服务架构，不断迭代，成功上线，立竿见影，老板很满意，CTO到技术大会分享成功案例（对公司、对个人带来极大好处）
- **注意这里有一个重要细节：**
- 庞大单体架构(.NET) -> 微服务架构(Java/.NET)
- 庞大单体架构(Java) -> 微服务架构(.NET/Java)
- 本质上不是编程语言的问题，而是微服务架构、新的研发团队、系统发展迭代的历史进程
- 微服务架构最大好处：支持多种编程语言、让系统更轻量级、灵活升级部署

微服务框架技术选型

- 最成熟、最知名的两个微服务架构框架：微软Service Fabric、Spring Cloud



Service Fabric是微软云核心组件， 超大规模云平台实战验证

Services Powered by Service Fabric



SQL Database

2.1 million DBs



Cosmos DB

Billions transactions/day



IoT Hub

Millions of messages



Event Hubs

640bn events/day



Event Grid



Skype



Cortana



Intune



Dynamics



Power BI

Designed for mission critical tier 1 workloads

30% of Azure cores run Service Fabric

Service Fabric成功案例，多个世界500强使用

D+H

 Wolters Kluwer

ABBYY

CITRIX®

 OSIsoft®

UNIT4
In business for people.

Schneider
Electric

 MAERSK

Honeywell
THE POWER OF CONNECTED


The Ultimate
Driving Machine

BANG & OLUFSEN

P2
ware

DIGAMORE
entertainment gmbh

 Swiss Re

codit
integrating your business

 AFAS
software

 Bentley®
Advancing Infrastructure

 SOGETI

 SOCIETE
GENERALE

ABB

 quorum

QARNOT
COMPUTING

MESH
SYSTEMS

UX
RISK


LIXAR

Alaska
AIRLINES

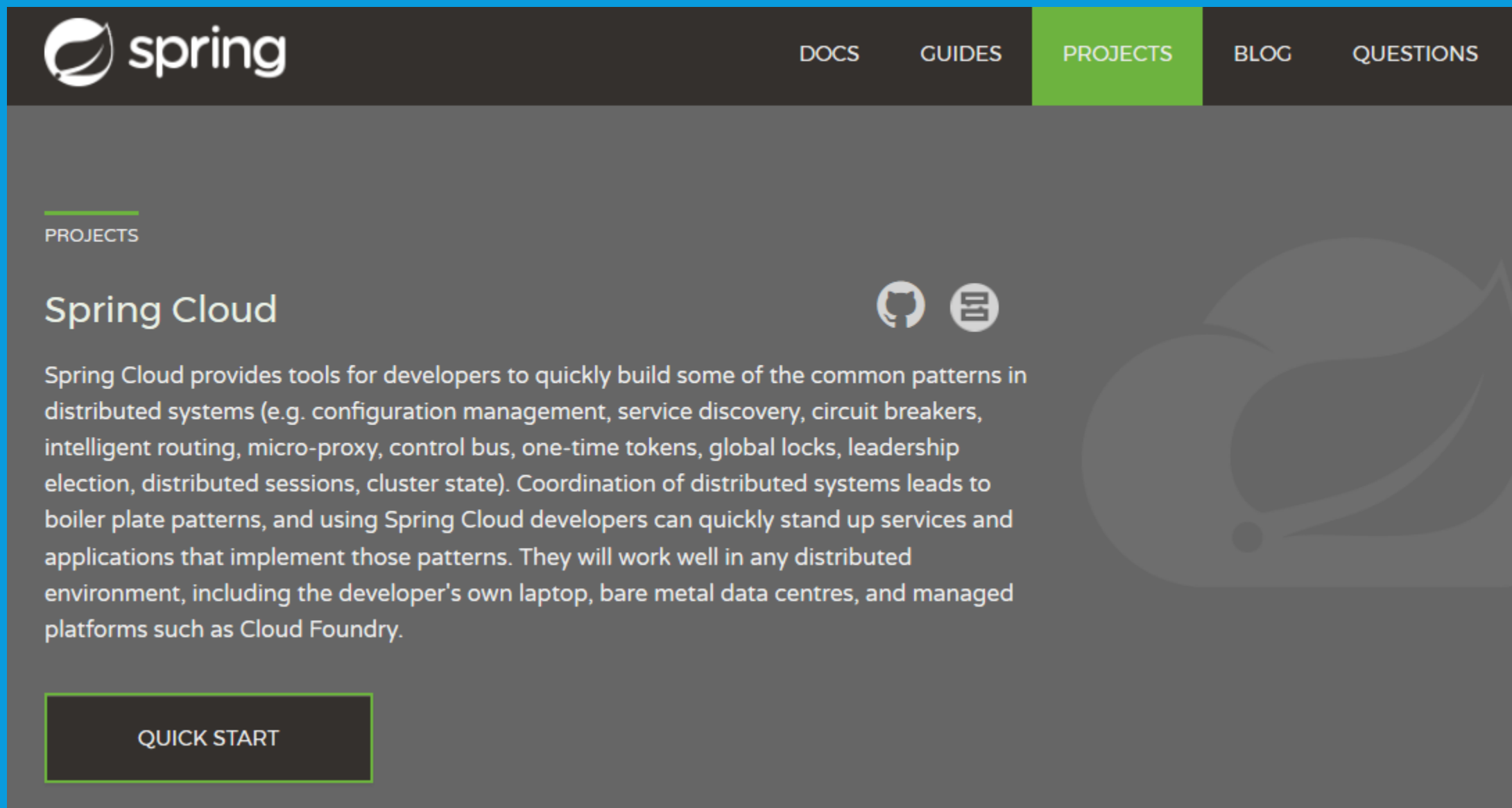
infoSupport
Solid Innovator

 fudura

FUN
ROCK

 Reply
solidsoft

Spring Cloud是Spring厂商Pivotal的微服务框架



The screenshot shows the Spring Cloud project page. At the top is a dark navigation bar with the Spring logo on the left and links for DOCS, GUIDES, PROJECTS (highlighted in green), BLOG, and QUESTIONS on the right. Below the navigation bar, on the left, is a 'PROJECTS' section with a green underline. The main heading 'Spring Cloud' is followed by GitHub and Maven icons. A paragraph describes Spring Cloud as a tool for building common patterns in distributed systems, listing examples like configuration management, service discovery, and circuit breakers. It mentions that these patterns lead to boilerplate code and that Spring Cloud helps developers stand up services and applications in various environments. At the bottom left is a green 'QUICK START' button. On the right side of the page, there is a large, faint, stylized leaf graphic.

spring

DOCS GUIDES **PROJECTS** BLOG QUESTIONS

PROJECTS

Spring Cloud

Spring Cloud provides tools for developers to quickly build some of the common patterns in distributed systems (e.g. configuration management, service discovery, circuit breakers, intelligent routing, micro-proxy, control bus, one-time tokens, global locks, leadership election, distributed sessions, cluster state). Coordination of distributed systems leads to boiler plate patterns, and using Spring Cloud developers can quickly stand up services and applications that implement those patterns. They will work well in any distributed environment, including the developer's own laptop, bare metal data centres, and managed platforms such as Cloud Foundry.

QUICK START

微服务框架技术选型：SF与SC的对决

- 微软Service Fabric（SF）支持多种编程语言、支持Linux、Docker容器编排、经过微软云超大规模实战验证、多个世界500强以及创新公司的大中型系统实战验证
- Spring厂商Pivotal开发了微服务平台Spring Cloud（SC），以及两个编程模型：Spring Boot（Java编程模型）、SteelToe（.NET及.NET Core编程模型，内部名称Iron Boot即.NET版的Spring Boot）来配合Spring Cloud基础设施实现微服务架构
- **SF、SC微服务框架二选一，怎么选？**
- 微软Service Fabric技术实力、成功案例好牛逼，最后的选择是Spring Cloud，原因就一条：
- 微软Service Fabric承诺会开源，但开源动作太慢了！在线等，挺急的！等不了！选SC！
- “可是SF是免费的，虽然还没开源，但也能用啊，你看人家世界500强用SF”
- “**不开源的，坚决不用**”有时候，技术选型就是这么简单（任性）

Spring Cloud SteelToe + .NET Core 微服务架构PPT下载:

http://schd.ws/hosted_files/cfeu17/ea/Building%20Microservices%20with%20.NET%20Core%20and%20Steeltoe.pdf



Building Microservices with .NET Core and Steeltoe

Zach Brown and Matt Horan, Pivotal

Spring Cloud SteelToe + .NET Core 微服务架构PPT下载:

http://schd.ws/hosted_files/cfeu17/ea/Building%20Microservices%20with%20.NET%20Core%20and%20Steeltoe.pdf



Visual Studio



MacOS



Microsoft  Linux

SQL Server  Linux



Microsoft  OpenSource

The .NET Renaissance

a convergence of forces

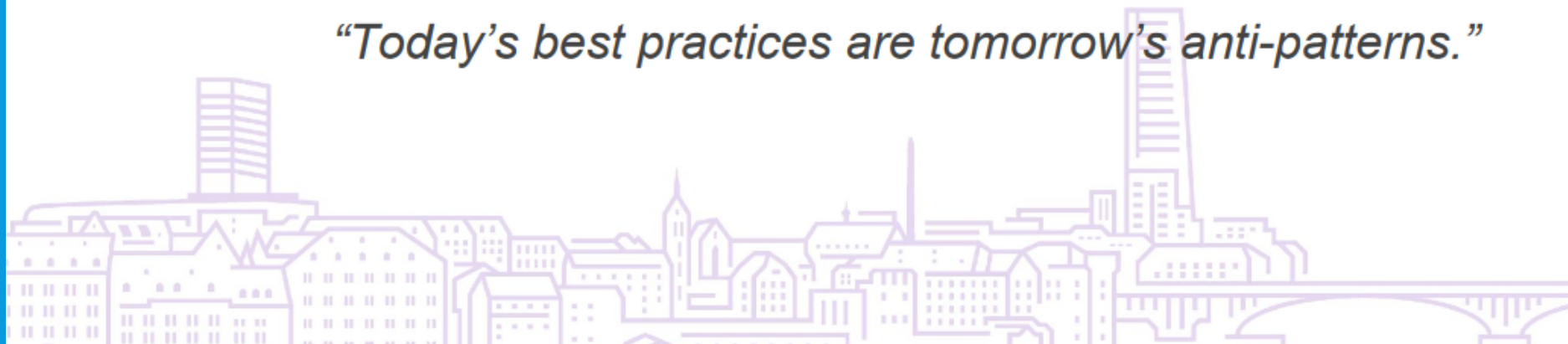
- It's about time...
(.NET's 15th Birthday)
- Cross-platform .NET Core
- Rise of Cloud
- Growing OSS community
- But we need more projects that
support for 12-factor, cloud-native, and microservices



Cloud-Native .NET

- 12 Factor App principles (<https://12factor.net/>)
- Configuration stored in environment variables
- Out of process session state
- No r/w to registry, GAC, Local file system
- Loosely coupled backing services (i.e. RDBMS)

“Today’s best practices are tomorrow’s anti-patterns.”

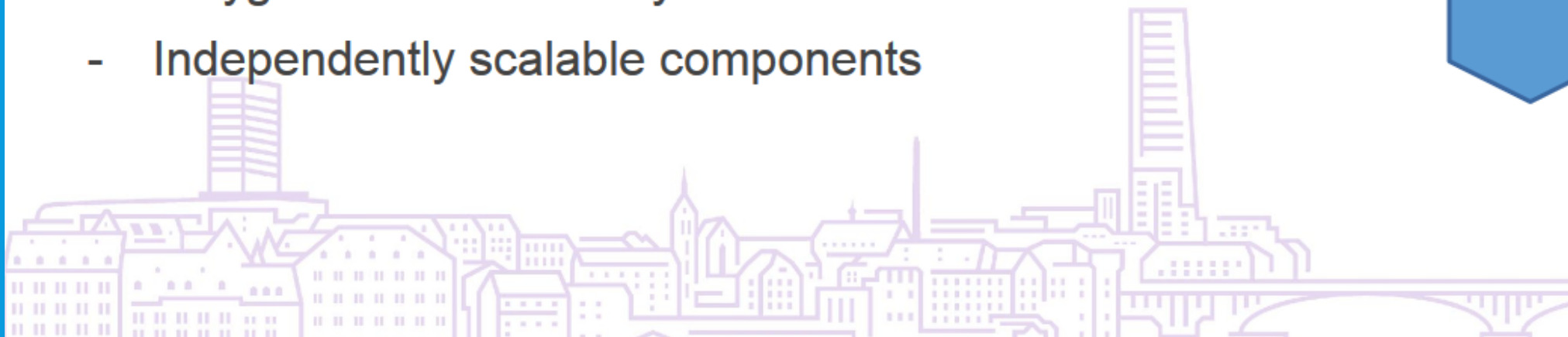
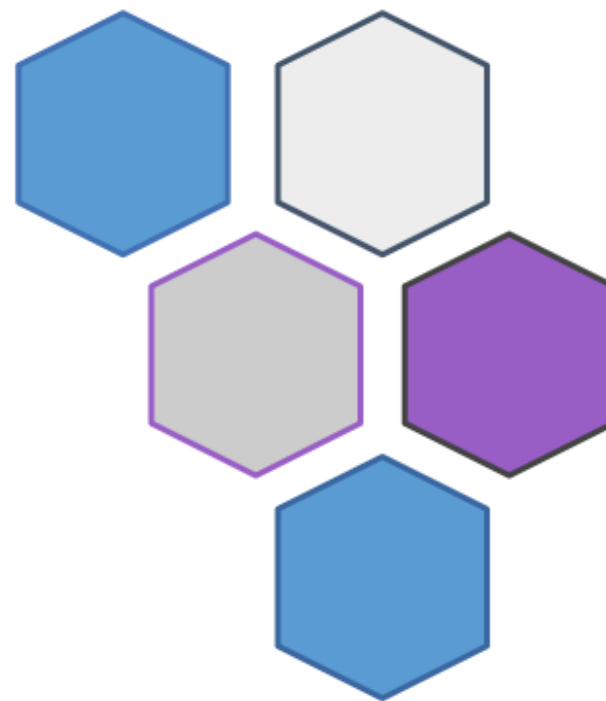


微服务的这些好处，说人话：如果不用微服务重构，老系统已经快撑不下去了！老板也快奔溃了！

Microservices

Benefits of Microservices Architectures

- Quickly iterate and release frequently
- Easy for new devs to join and be productive
- Increased developer velocity
- Shorter test cycles
- Polyglot becomes reality
- Independently scalable components



But Microservices are hard

Microservices = Distributed Systems = Way more complexity

- How to troubleshoot microservices?
- How to set global configuration across the whole application?
- How to look up the dynamic addresses of services you consume?
- What do you do when a service you depend on stops responding?



The Good news is...

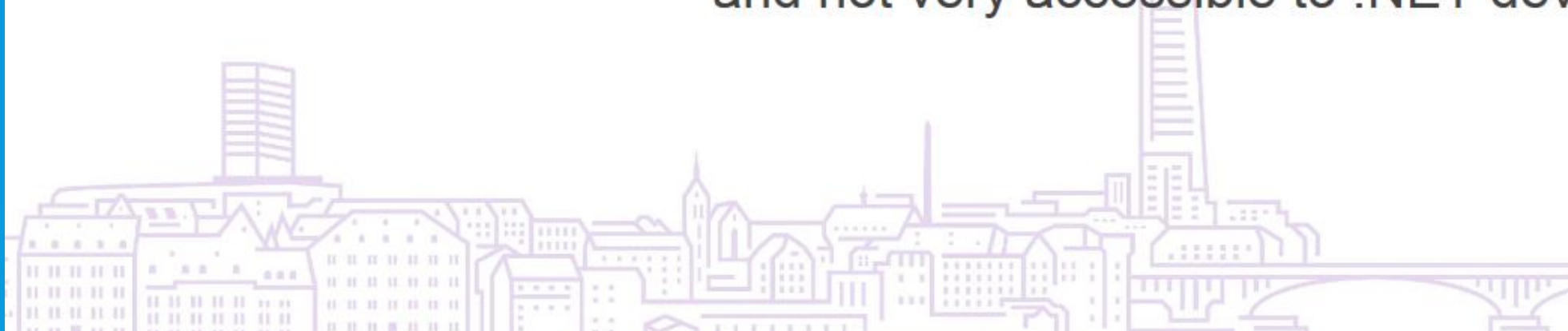
There are some great OSS solutions already available

NETFLIX
OSS

and

 **spring**

But until now they've been focused on Java
and not very accessible to .NET devs...



So that's why we built Steeltoe

To help you build

Cloud native .NET applications

and leverage Spring Cloud tooling for

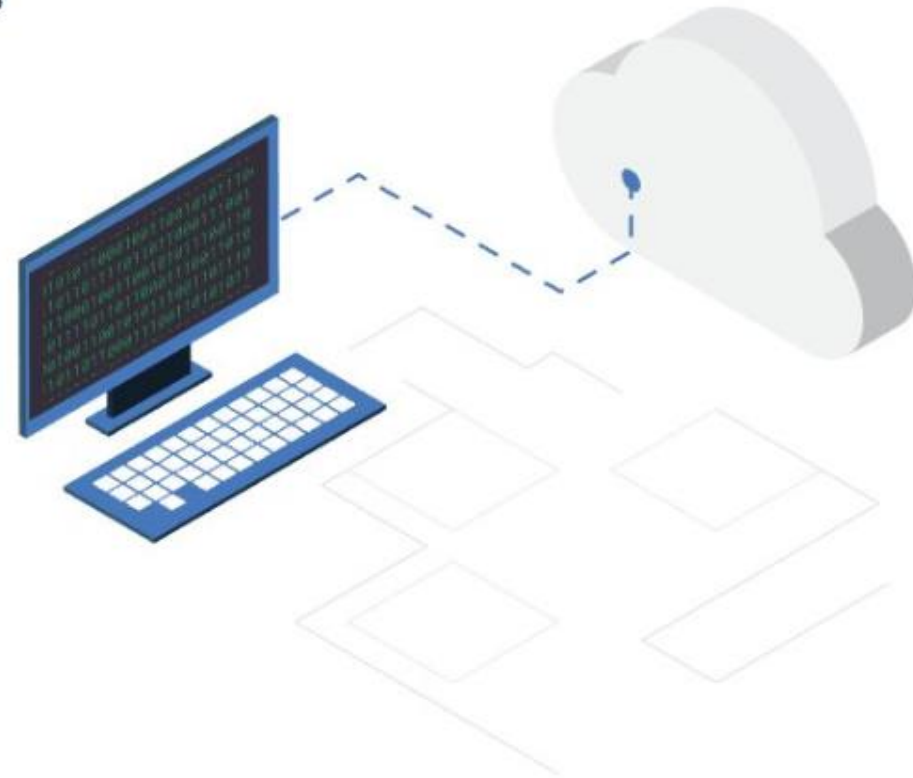
resilient microservices



Cloud-Native .NET

Connectors for common data services

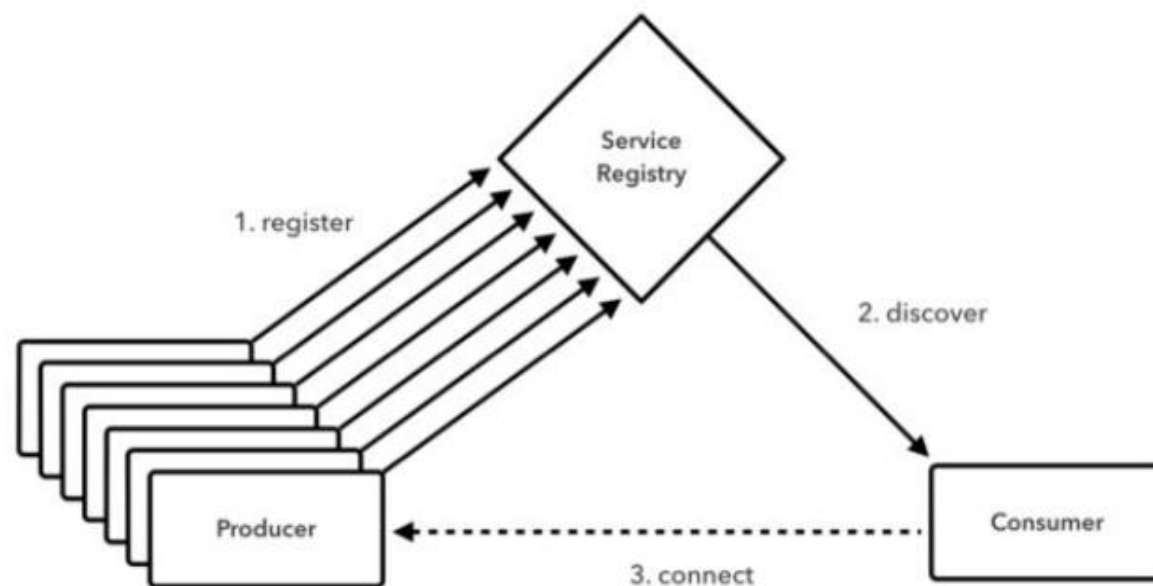
- MySQL
- Postgres
- Redis
- RabbitMQ
- OAuth2



Resilient Microservices

Service Discovery with Eureka

- .NET Client for Netflix Eureka
- Dynamically discover and call registered services
- Supports direct addressing



Resilient Microservices

Circuit Breaker

- .NET implementation of **Netflix Hystrix**
- Bypass failing services with elegant fall-back behavior
(so your users don't see nasty error messages)
- Rich metrics and monitoring



Open and Flexible

Steeltoe works...

with **.NET Core** *and* with the **.NET framework**
on **Windows** *and* on **Linux**
standalone *and* running on **Cloud Foundry**

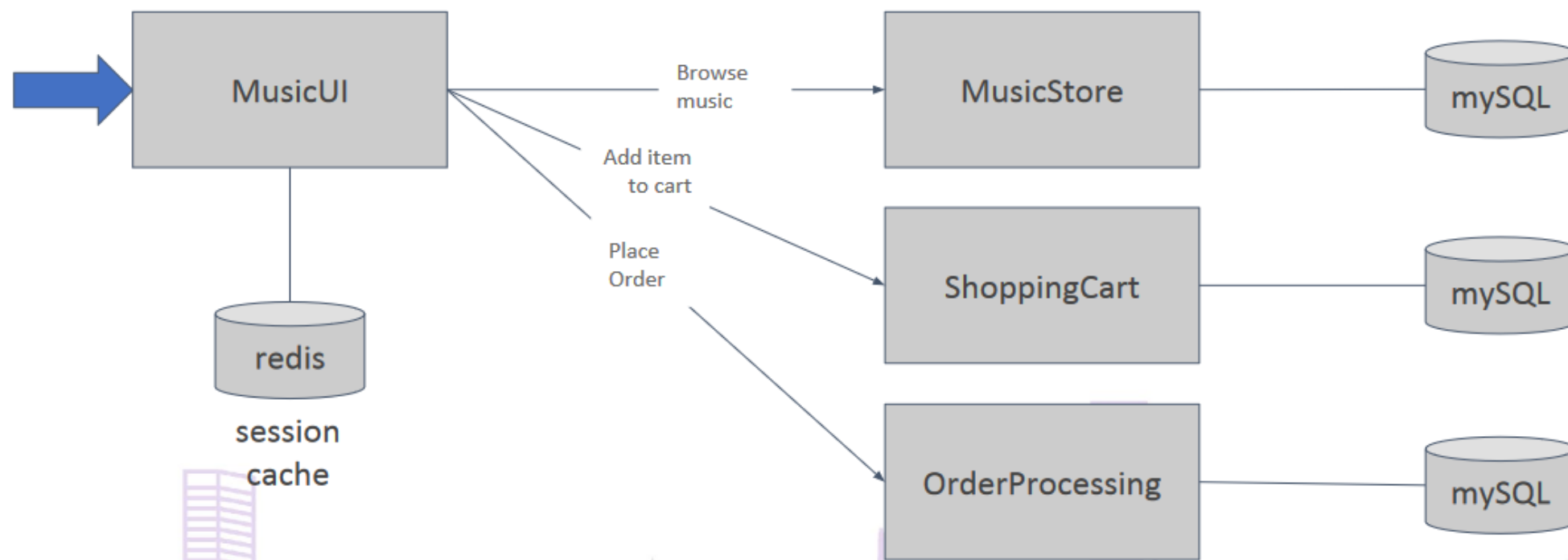




Demo:

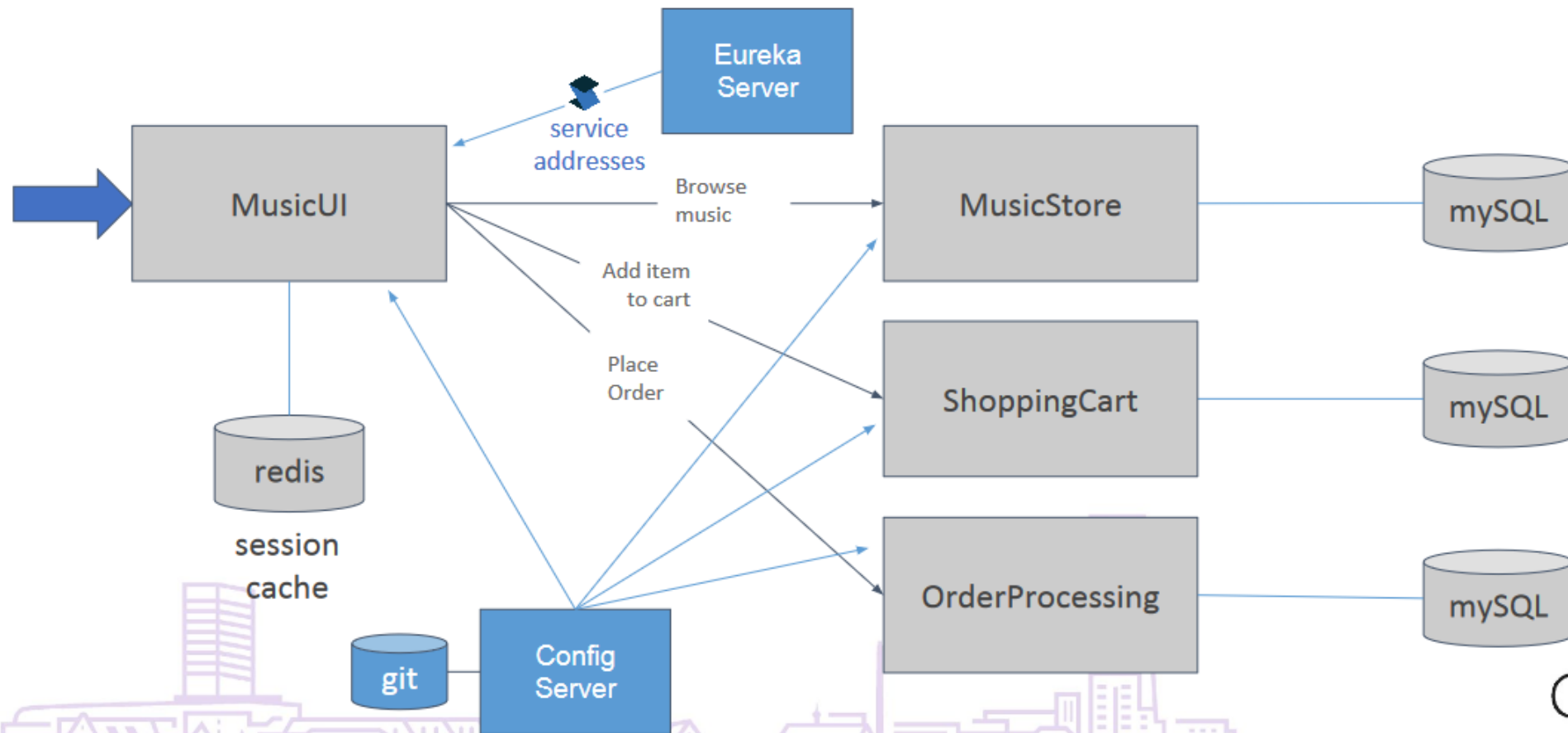
ASP.NET Music Store

ASP.NET Music Store



GitHub repo:
<https://github.com/SteeltoeOSS/Samples/tree/master/MusicStore>

ASP.NET Music Store w/Steeltoe - Discovery



GitHub repo:
<https://github.com/SteeltoeOSS/Samples/tree/master/MusicStore>

How to Get Started

The best way to get started is to take a look at our docs, clone a sample from the GitHub repo, and contact the team in Slack with questions and feedback.



Learn more → <http://steeltoe.io>



Clone a sample → <https://github.com/SteeltoeOSS/Samples>



Get the bits → <https://www.nuget.org/profiles/steeltoe>



Talk to us → <http://slack.steeltoe.io/>



Tell your friends → [@SteeltoeOSS](https://twitter.com/SteeltoeOSS)



Spring Cloud SteelToe + .NET Core 微服务架构

官网: <http://steeltoe.io/> 官方教程: <http://steeltoe.io/docs/>
<https://pivotal.io/cn/platform/pcf-marketplace/microservices-management/steeltoe>

Twitter @SteeltoeOSS @moredeploys

SteelToe PPT:

http://schd.ws/hosted_files/cfeu17/ea/Building%20Microservices%20with%20.NET%20Core%20and%20Steeltoe.pdf

Youtube视频 SteelToe+ .NET Core微服务架构

https://www.youtube.com/results?search_query=Building+Microservices+with+.NET+Core+and+Steeltoe

Building .NET Core Microservices with Steeltoe - Zach Brown & Matthew Horan, Pivotal

<https://www.youtube.com/watch?v=qltci8lLgJw>

Building .NET Microservices — Kevin Hoffmann

https://www.youtube.com/watch?v=-a8v_IRSveY

OSSAT 2017 - Intro to Pivotal Cloud Foundry & Steeltoe by Zach Brown

<https://www.youtube.com/watch?v=GRGoY7iMEUg>

Spring Cloud SteelToe + .NET Core 微服务架构

技术博客:

龙向辉 spring cloud+dotnet core搭建微服务架构：服务治理（一）

<http://www.cnblogs.com/longxianghui/p/7561259.html>

龙向辉 spring cloud+dotnet core搭建微服务架构：服务发现（二）

<http://www.cnblogs.com/longxianghui/p/7576736.html>

龙向辉 spring cloud+dotnet core搭建微服务架构：Api网关（三）

<http://www.cnblogs.com/longxianghui/p/7646870.html>

张占岭 微服务~Eureka实现的服务注册与发现及服务之间的调用

<http://www.cnblogs.com/lori/p/7598058.html>

腾讯张善友 .NET微服务组件精选：

<http://www.csharpkit.com/microservice>

Spring Cloud SteelToe + .NET Core 微服务架构

1、设计目标：每秒10万笔交易，可水平扩展到每秒100万单

参考数据：阿里2016年双十一每秒17.5万笔交易

2、全开源免费解决方案、微服务架构：

Spring Cloud Steeltoe + .NET Core/Java/PHP on Docker + Redis + MySql + PostgreSQL + RabbitMQ/Kafka

3、支持多种编程语言.NET Core/Java/PHP开发不同的业务微服务，统一部署到Docker

4、DevOps自动化编译、测试、灰度发布（蓝绿部署、金丝雀部署）：VSTS/TFS + Jenkins等

5、每个技术组件可灵活替换，例如MQ、NoSql、Sql、微服务组件，每个组件可水平扩展

6、前后端分离、后端接口返回Json、前端页面及JS可以部署到cdn，抗住静态文件高并发流量

7、内外分离，内部运营系统与面向外部用户系统彻底分开，独立数据库，互不影响

Spring Cloud SteelToe + .NET Core 微服务架构

8、对外系统：redis分片集群 + mysql分库分片集群 + RabbitMQ/Kafka异步队列

1) redis当作Nosql（redis不开启持久化，缓存集群保证高可用，实际运行一年没有出现过集群同时重启的问题，不用担心丢数据）

2) redis分片集群，应用中允许配置多个分片redis目标地址

9、对内系统：PostgreSql 业界最领先的开源数据库

1) 多表关联查询性能比MySql快几倍

2) 支持单表大数据量亿级，更大数据量按时间分区、归档历史数据

3) 支持业务处理OLTP、统计分析OLAP混合需求

HTAP（Hybrid Transactional/Analytical Processing）= OLTP + OLAP

4) 支持高级Sql特性、Json类型字段等

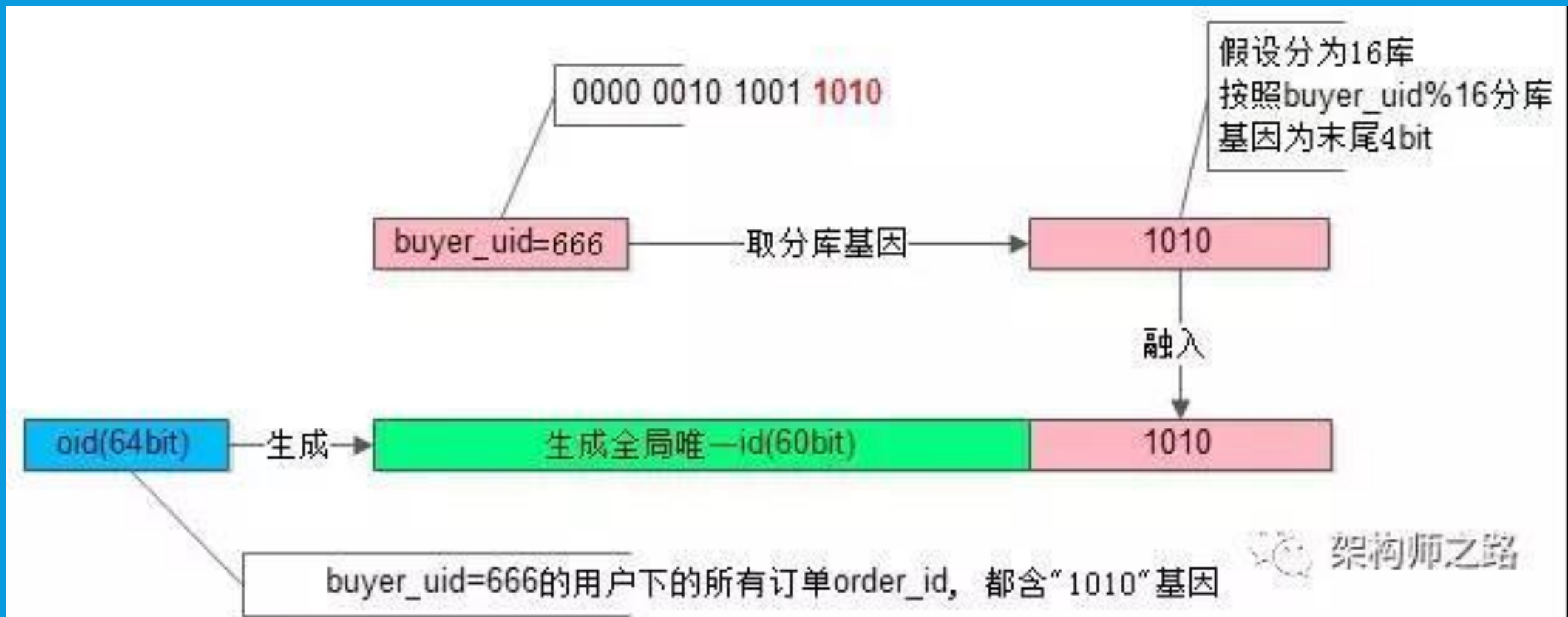
10、运维基础设施：docker容器编排、性能监控、可视化报表、配置中心、Api网关、负载均衡、统一日志监控分析、调用链监控、MySql慢查询监控

11、各业务模块分步骤、分批上线，不用等到全部模块开发完成

12、技术原型Demo验证、提前性能压测，防止后续风险

MySql分片分库 核心算法

- 1、MySql单表1000万条数据之后，性能下降比较严重，建议水平分片分库
- 2、订单号码OrderId、消费者BuyerId，参考58沈剑（58技术委员会主席）的分片分库算法“基因法”
<http://mp.weixin.qq.com/s/PCzRAZa9n4aJwH0X-kAhtA>



MySQL 无限容量数据库架构设计（58沈剑）

<http://mp.weixin.qq.com/s/ad4tpM6cdi9r6vgfbaTzxc>

无限容量数据库架构设计

原创 2017-09-02 58沈剑 架构师之路

花了不少时间，把自己曾经做过的系统，曾经遇到到的问题，曾经实践过的架构方案量“系统的”记录成文字，和大家一起讨论。

本文是不同业务场景下，**体系化**的介绍“**数据库水平切分**”技术，和大家分享。

一、总起

文章：《[每每谈到数据库架构，我们在讨论什么](#)》

内容：

- 单库体系架构
- 数据库**分组**架构
- 数据库**分片**架构
- 数据库**垂直切分**

二、实践一

场景：**单key**业务，如何做到数据库无限容量

文章：《[用户中心，数据库架构优化与实践](#)》

内容：

- 用户中心业务分析
- 用户中心水平切分方案
- “**前台与后台分离**”架构设计思想
- uid分库，name上的查询四种方案

三、实践二

场景：**1对多**业务，如何做到数据库无限容量

文章：《[帖子中心，数据库架构优化与实践](#)》

内容：

- 帖子中心业务分析
- “**索引外置**”架构设计思想
- 基因法，uid分库还是tid分库不再纠结

四、实践三

场景：**多对多**业务，如何做到数据库无限容量

文章：《[好友中心，数据库架构优化与实践](#)》

内容：

- 好友中心业务分析
- 数据冗余的三种方案
- “**最终一致性**”架构设计思想
- 保证数据一致性的四种方案

五、实践四

场景：**多key**业务，如何做到数据库无限容量

文章：《[订单中心，如何做到数据库无限容量](#)》

内容：

- 订单中心业务分析
- “**化繁为简**”架构设计思想
- 订单ID，买家ID，卖家ID究竟应该如何分库

5篇文章超过1万字，架构图超过50副，有点长，建议先收藏，再**转发**，再细细品味。

58沈剑 微信公众号：架构师之路

架构师之路2017半年精选40篇

(原创) 2017-06-24 58沈剑 架构师之路

2017上半年精选索引，点击标题阅读。

【特别推荐】

《架构师之路2016年精选66篇》

发起一个活动，一起**携手见证**“架构师之路”的**第一篇10w+**，还差3000阅读，邀您一起转发。

【通用设计与方法论】

《单KEY业务，数据库水平切分架构实践》

《架构设计中常见“反向依赖”与解耦方案》

《互联网架构如何实现“高可用”》

《互联网架构如何实现“高并发”》

《典型数据库架构设计与实践》

【典型架构实践】

《计数系统架构设计一次搞定》

× 架构师之路

【典型架构实践】

《计数系统架构设计一次搞定》

《session一致性架构设计实践》

《TCP接入层的负载均衡、高可用、扩展性架构设计》

《配置中心架构设计演进》

《跨公网调用的大坑与架构优化》

《DNS在架构设计中的巧用》

《互联网智能广告系统简易流程与架构》

【数据库】

《100亿数据量1万属性10万并发数据库架构设计》

《数据库秒级平滑扩容架构方案》

《58到家数据库30条军规》

《再议数据库军规》

《业界难题-“夸库分页”的四种方案》

《100亿数据平滑迁移却不影响服务》

《MySQL-proxy数据库中间件架构》

《用uid分库，那uname上的查询怎么办》

× 架构师之路

《1分钟理解负载》

《1分钟理解领导者-跟随者线程模型》

【一致性问题】

《库存扣多了，到底怎么整》

《再议库存扣减多种方案》

《浅谈CAS在分布式ID生成方案上的应用》

《CAS下ABA问题及优化方案》

【搜索架构】

《深入浅出搜索引擎架构、方案与细节》

《如何迅猛的实现搜索需求》

《58同城如何检索到1秒前发布的帖子》

【消息总线架构】

《到底什么时候该使用MQ》

《MQ如何实现消息幂等》

《MQ如何实现削峰填谷》

《定时任务如何高效触发》

《快速实现“延迟消息”功能》

《MQ如何实现消息必达》

让人震撼的PostgreSql，业界最先进的开源数据库

1、这篇文章太震撼了，看完这个，估计90%的人都想把Oracle、SqlServer全换成开源免费的PostgreSql数据库：

https://github.com/digoal/blog/blob/master/201702/20170209_01.md

难以用语言来形容这一篇文章的震撼程度！ 不信你自己看！

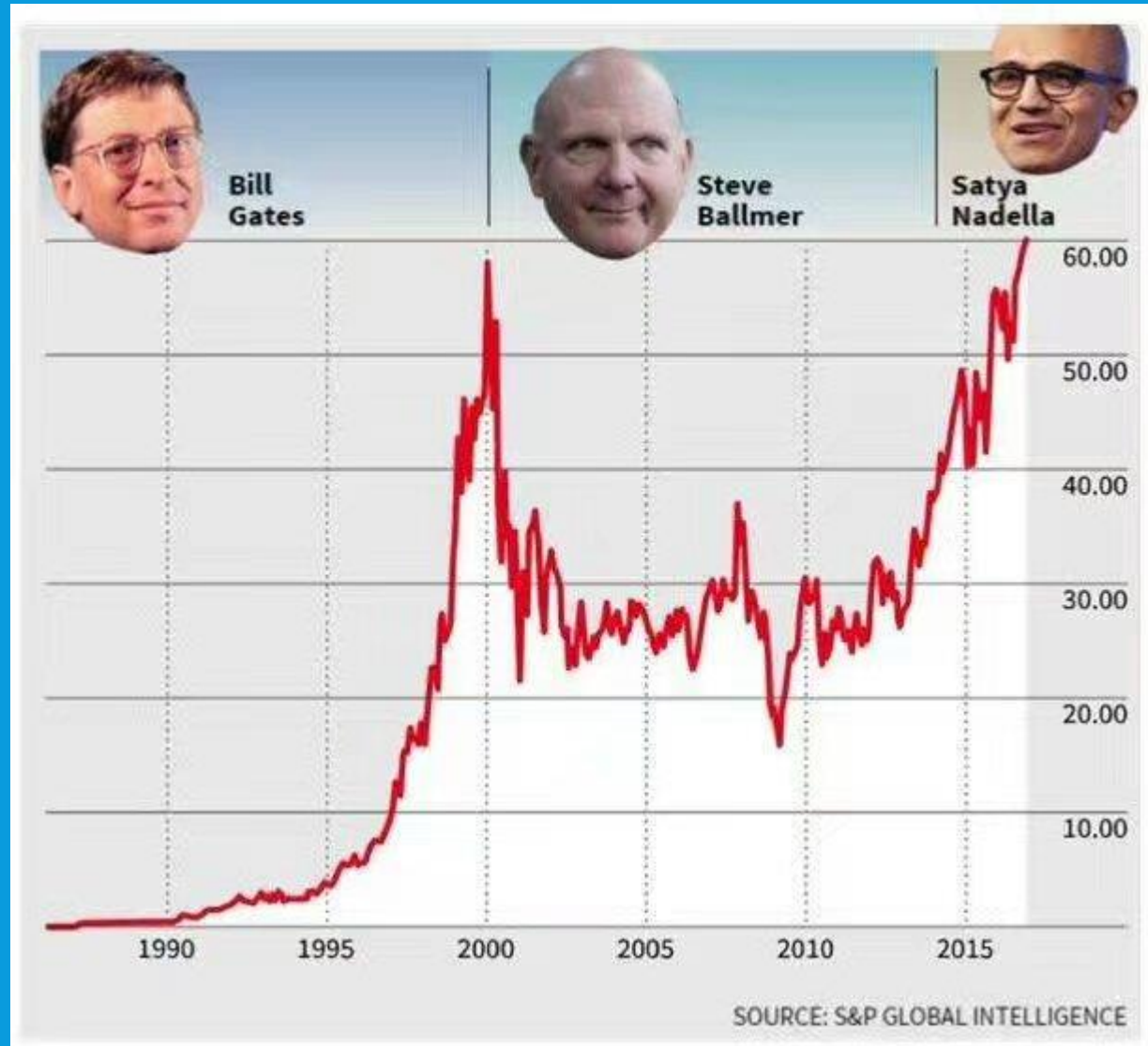
主流云厂商（微软云、腾讯云、阿里云……都支持PostgreSql数据库云服务）

MySql轻量级用于简单业务，分片分库，单表建议1000万条数据以下，
PostgreSql可用于替代Oracle、SqlServer等重量级数据库，单表10亿条数据很轻松

如果大家都把Oracle、SqlServer换成开源免费PostgreSql、把WinServer换成Linux、Docker，微软是否会损失大量利润，微软会不会破产，导致.NET社区群龙无首？

“你想多了，微软Azure云上面的Linux虚拟机比例已经超过50%、微软云MySql、PostgreSql云数据库、AI人工智能服务、量子计算、MR……等方面大把赚钱，一点不比传统方式少，看看微软的股价涨幅”

微软股价市值变化



重要技能：技术+营销造势（Chui Niu B）能力

技术营销造势（Chui Niu B）能力：

营销造势 + 无法技术实现 = 吹牛

营销造势 + 技术实现落地 = 牛人

这些好书、网站、公众号，能帮助你成为真正的 技术+营销造势 牛人

《亿级流量网站架构核心技术》

《大型网站技术架构：核心原理与案例分析》

《尽在双11：阿里架构演进与超越》

《企业IT架构转型之道》

《Spring Cloud微服务实战》翟永超

《Spring Cloud与Docker微服务架构与实战》周立

<http://www.infoq.com/cn/>

微信公众号：架构师之路、聊聊架构、dotnet新时代、dotnet跨平台、InfoQ

开放心态，同时参考学习.NET Core、Java、Go社区技术资源

微信公众号：架构师之路、聊聊架构、dotnet新时代、dotnet跨平台、XamarinUG、InfoQ

中国.NET新时代 开源社区QQ群：618093978（开源社区、技术交流、招聘、求职、技术分享PPT……）

注意上面的QQ群申请加入时需要写明真实公司名称、职业，否则无法审核通过

中国.NET新时代 CTO CEO联盟微信群：加入微信群方式请看QQ群公告

微信公众号：DotNet新时代

知识星球：.NET新时代



开源、开源、开源

Spring Cloud Steeltoe + .NET Core/Java/PHP on Docker + Redis + MySql + PostgreSQL + RabbitMQ/Kafka + VSTS/TFS Jenkins DevOps

每一个组件都需要深入研究，这次分享相当于一个提纲目录

我们也刚刚开始，前方征途是星辰大海，建议大家一起来，

以后会逐步将研究成果、Demo代码开源、并写成技术文章分享

本次演讲PPT、未来研究成果、开源代码都会分享到Q群：

中国.NET新时代开源社区QQ群：618093978（只研究技术，不闲聊）

Thank You

Make .NET Great Again

