

www.qconferences.com
www.qconbeijing.com
www.qconshanghai.com

QCon

伦敦 | 北京 | 东京 | 纽约 | 圣保罗 | 上海 | 旧金山
London · Beijing · Tokyo · New York · Sao Paulo · Shanghai · San Francisco

QCon全球软件开发大会

International Software Development Conference

InfoQ^{ueue}



@InfoQ



infoqchina

软件
正在改变世界!

搜狗商业平台Java生态演化之路

搜狗商业平台研发部

刘建

关于我



刘 建

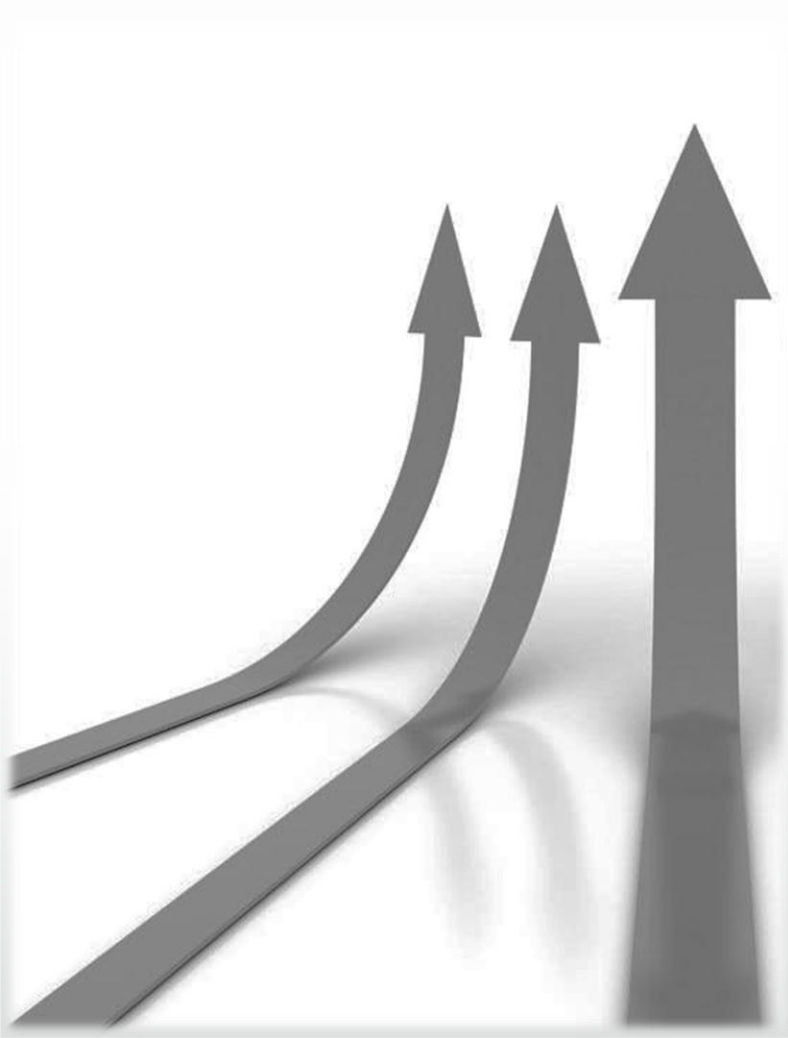
搜狗 架构师

十年Java相关研发经验

目前负责商业平台架构设计、商业平台
安全业务、组件的评审和引入

@kersus

主要内容



Java生态演化历程

组件选型漫谈

—— “原则” “评估” “流程”

组件演化实践之旅

—— “那些奋斗的日子”

未来展望

无处不在的Java

900万开发人员

30+亿设备

开源、成熟的软件体系

主流公司开源社区支持

Oracle(Sun) Eclipse

IBM JBoss

Apache Spring

QCon



● 安全 ● 移动 ● 大数据 ● Web ● 网络

渗入软件体系结构全层次

全行业渗透 全语言渗透

互联网

银行业

Scala

Clojure

Sogou 搜狗

搜狗商业平台业务

搜索广
告投放

精准广

网盟广

品牌广

无线广

广告形式

Java > 70%

广告工具

无线建
站工具

广告客
户管理

客服业
务支持

创意内
容专家

联盟站
长管理

搜狗商业平台Java生态系统

线上系统、平台

研发、测试、运维监控工具集

基础技术、组件集

生态“演化”的烦恼与挑战

**开源组件太多
技术体系繁杂**

SSH vs Spring MVC
Hibernate vs MyBatis

...

迫不得已的“升级”

2013/7/17 Struts2安全漏洞

...

环境调整

安全访问策略/权限
私有云

...



技术人员成就感

技术升级
组件推广

...

组件成熟度

有“坑”么？
好用么？
好维护么？

...

从“无序生长”到“有序进化”

目标

- 知识库

手段

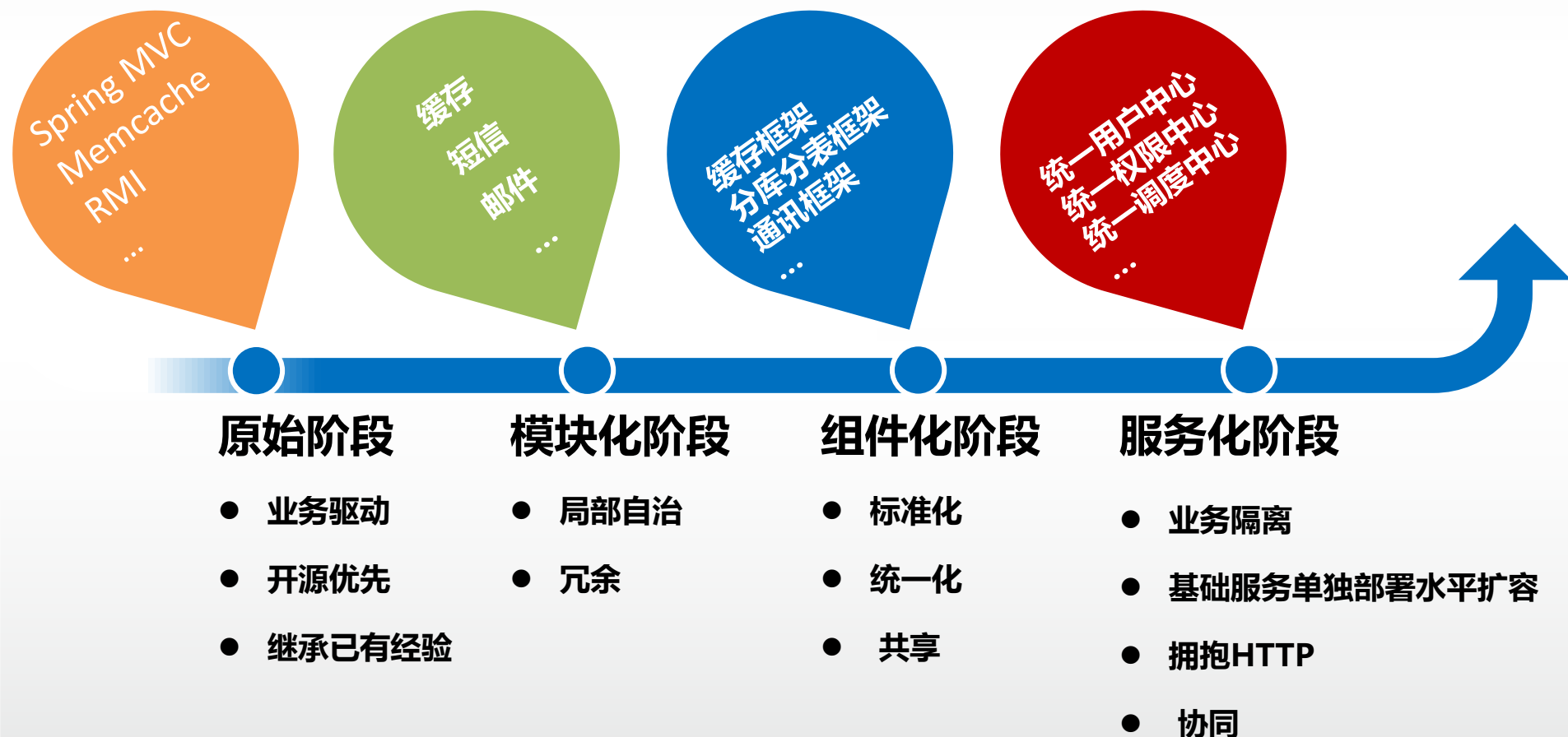
- 组件化/服务化

收益

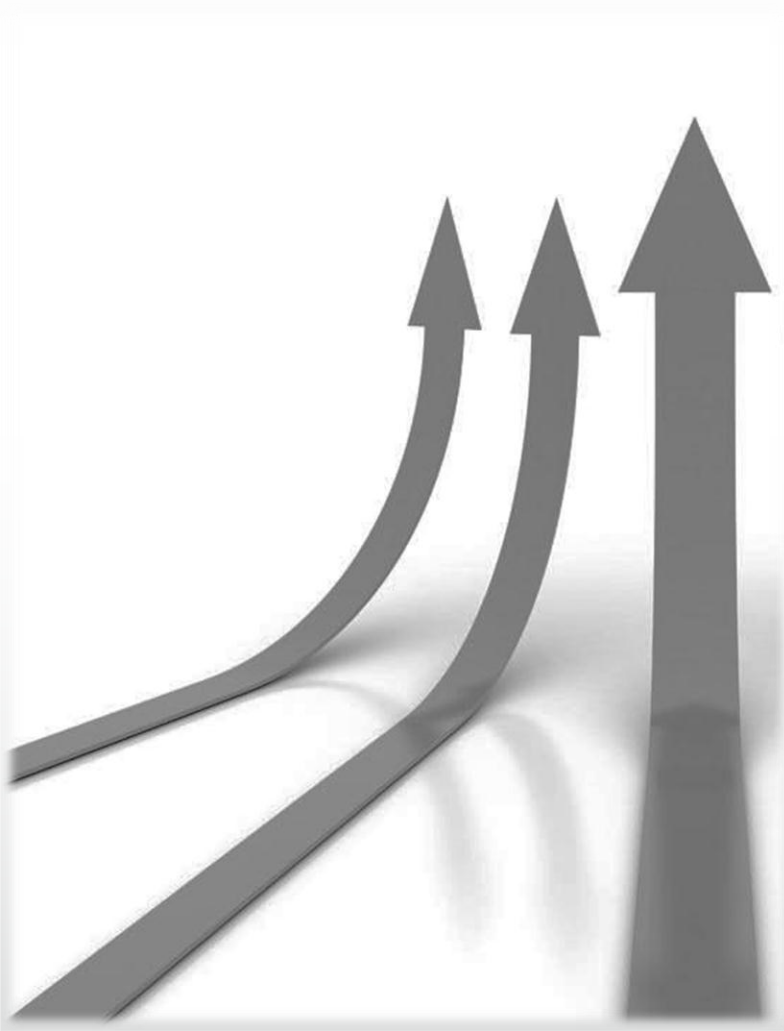
- 更健壮的代码
- 更少的整体成本
- 更低的耦合性
- 统一的技术集



搜狗商业平台Java生态演化阶段



主要内容



Java生态演化历程

组件选型漫谈

—— “原则” “评估” “流程”

组件演化实践之旅

—— “那些奋斗的日子”

未来展望

组件/服务设计原则概述

面向对象

单一职责原则

开放封闭原则

依赖倒置原则

接口隔离原则

里氏替换原则

面向组件

重用-发布等价原则

共同重用原则

共同封闭原则

无环依赖原则

稳定依赖原则

稳定抽象原则

面向服务

边界是显式的

服务具有**自治性**

共享架构和合约

不共享类

服务兼容性基于策略

商业平台组件引入原则

实现层

- 接口稳定
- 兼容性
- 插件化

使用层

- 单一职责
- 易用性高
- 性能卓越
- 依赖简单
- 安全
- 自治

运维层

- 可监控性
- 故障容忍
- 故障恢复

组件评估原则



高性能

稳定性

易用性

可维护性

安全性

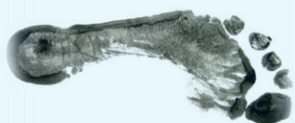
组件引入流程 “五部曲”

需求评审



- * 需求来源
- * 需求凝练

组件研发



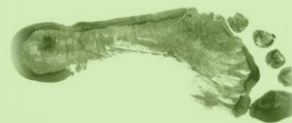
- * 选型调研
- * 选型评估
- * 过程管理

灰度实验



- * 试点项目
- * 双路对比

反馈优化



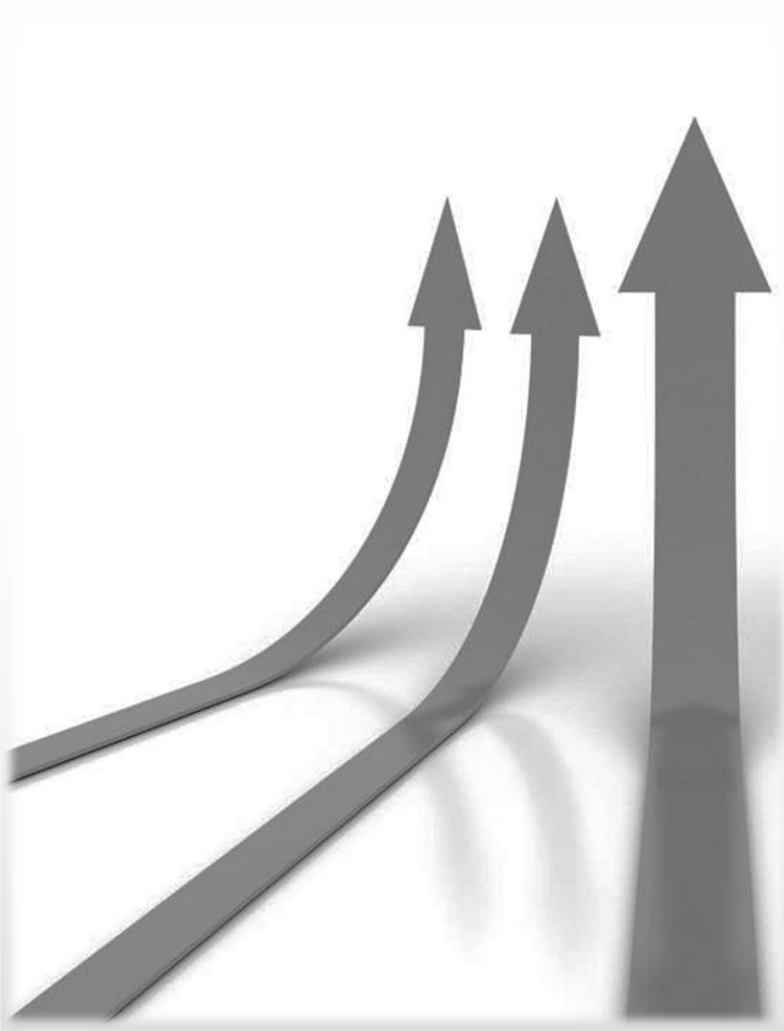
- * 效果评估
- * 优化改进

全面推广



- * 全面切换
- * 知识库

主要内容



Java生态演化历程

组件选型漫谈

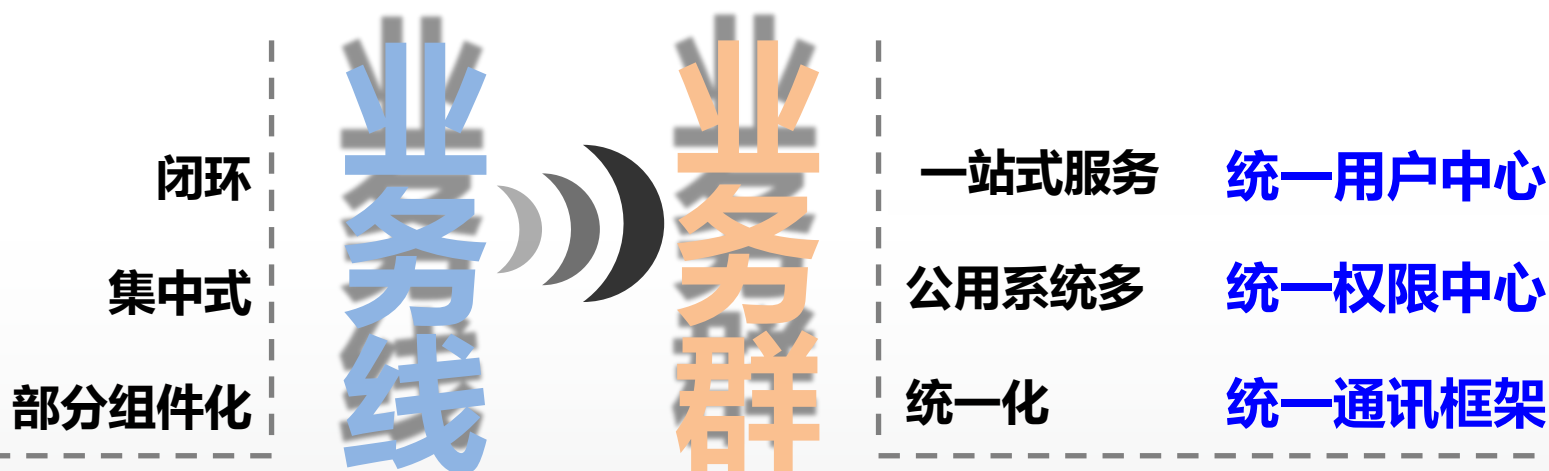
—— “原则” “评估” “流程”

组件演化实践之旅

—— “那些奋斗的日子”

未来展望

组件演化实践概述



案例1：安全之路—统一用户中心

目 标

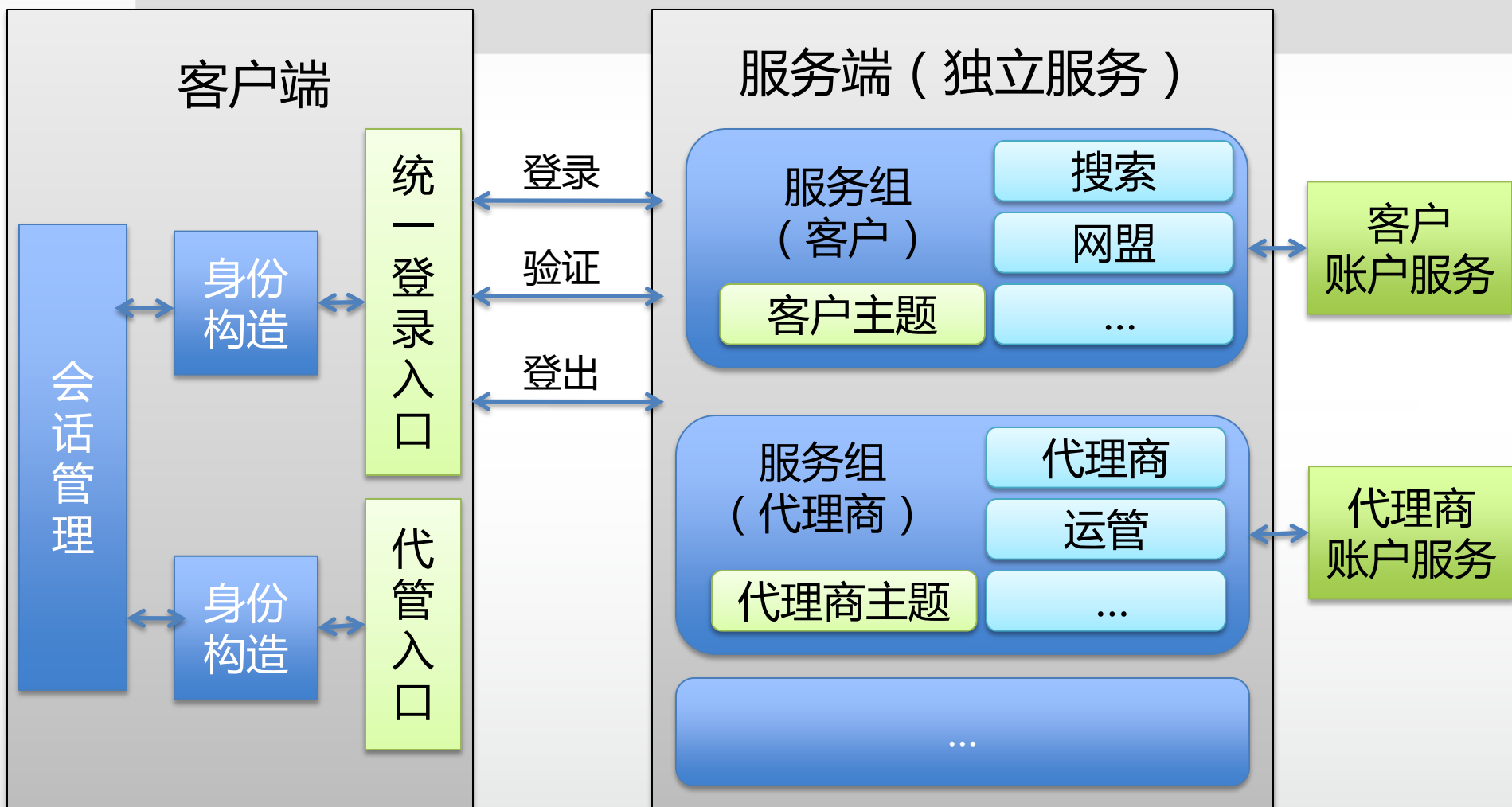
用户统一接入需求

用户安全性需求

简单的服务接入流程

不同类业务间的服务隔离

案例1：安全之路—统一用户中心



案例1：安全之路—统一用户中心

① 技术点

单点登录/登出

安全

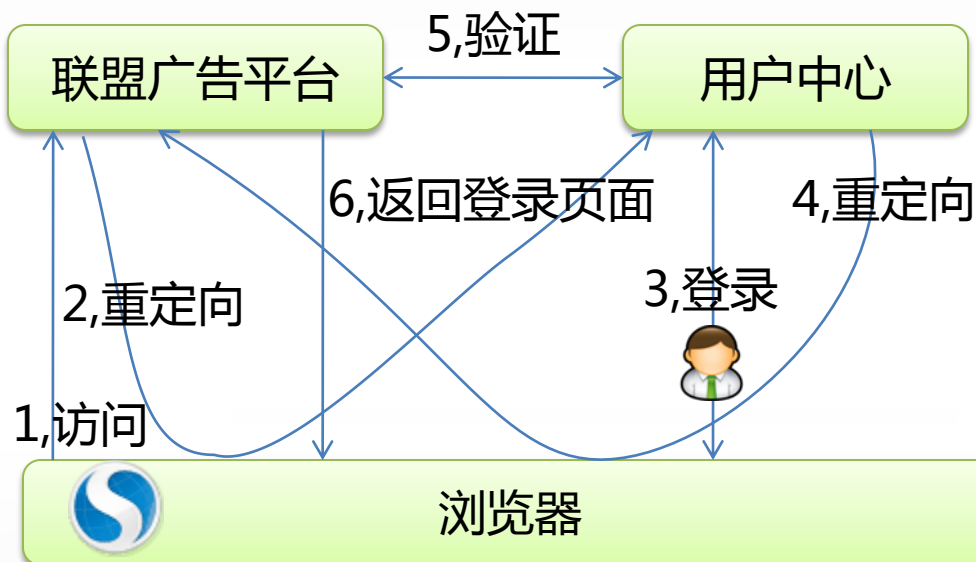
② 基础组件选型-CAS

简单

功能全面

广泛应用

应用会话隔离



方案	考察点	说明
Cookie	易用性	不支持跨域 会话统一管理
SAML	性能	性能损失较高

案例1：安全之路—统一用户中心

组件研发

- 易用性：登出优化；简化客户端配置（支持登录和代管）
- 安全性：验证接口安全性调整；登出使用内网；服务隔离；

灰度实验

- 第一阶段：网盟平台少量用户通过业务推动测试；
- 第二阶段：（二周后）搜索广告平台进行切换；

反馈优化

- 分布式会话（基于Memcache），支持故障转移

全面推广

- 知识库：组件集成使用说明文档
- 系统集成：已集成数十个内外部Web系统。
- 安全中心、安全分析；

案例2：组件化之路—统一权限中心

目 标

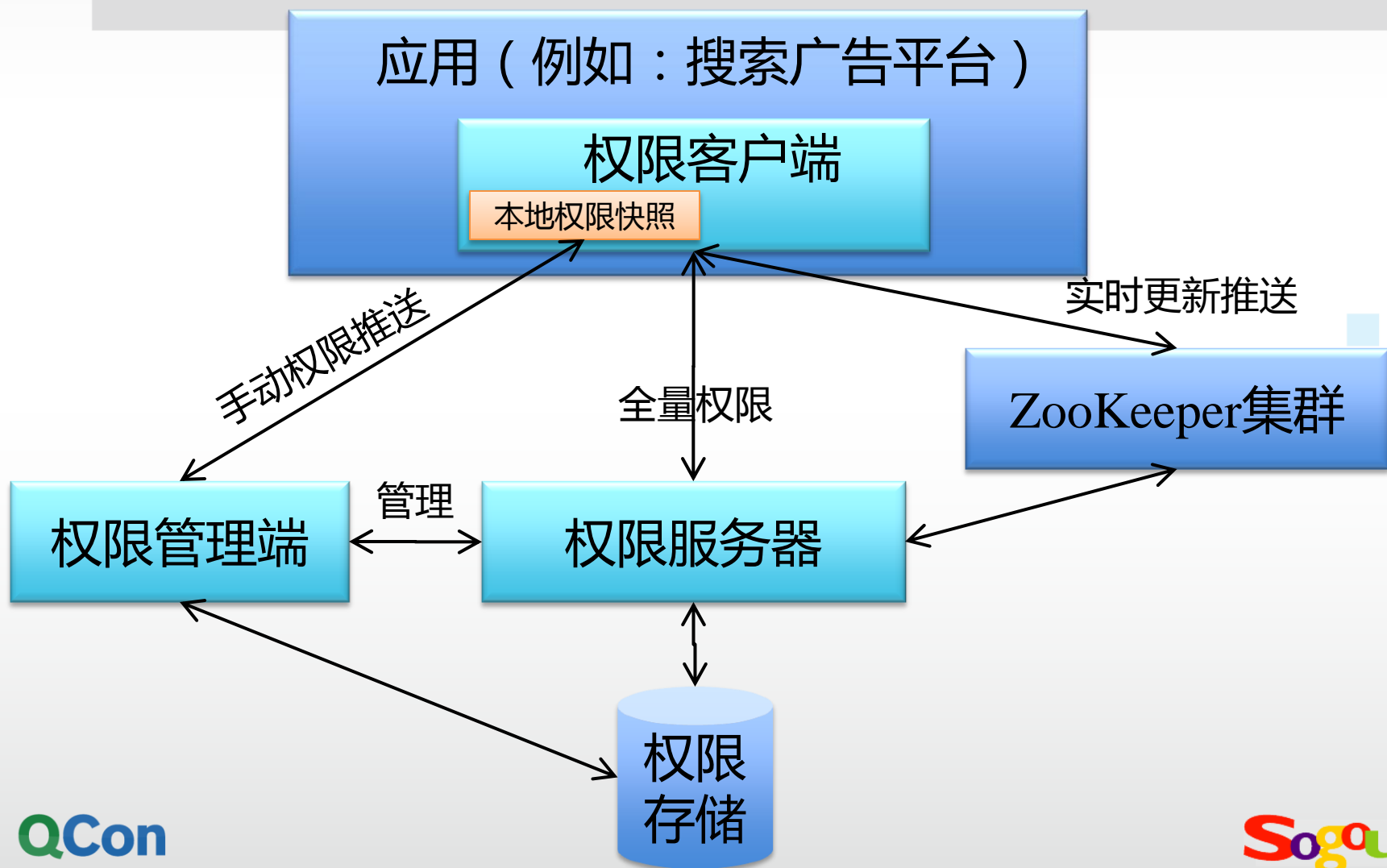
统一功能权限框架

权限实时推送

统一权限管理

高可靠性、高性能

案例2：组件化之路—统一权限中心



案例2：组件化之路—统一权限中心

- 权限模型

权限组合：“&”
默认权限：拒绝

权限协议

RBAC

ABAC

小流量

权限配置

权限配置

权限配置

权限配置

功能1

功能2

案例2：组件化之路—统一权限中心

① 技术点

RBAC

ABAC

② 缺失点

权限组合（小流量）

权限管理/更新

③ 自主实现

方案	考察点	说明
Spring Security	易用性	RBAC，集成用户认证 静态配置，不支持动态更新
Apache Shiro	易用性	RBAC
JBoss XACML	性能 易用性	ABAC

案例2：组件化之路—统一权限中心

组件研发

- 提供可拔插的框架支持RBAC、ABAC、小流量等权限的组合（“与”关系）；
- 权限实时推送；权限管理；标准化权限接入流程；

灰度实验

- 权限灰度接入流程：线下比对→线上比对（校正权限）→线上切换（原权限框架下线）
- 网盟业务线优先推广

反馈优化

- 管理端可用性优化；
- 新增IP白名单控制权限；
- 权限客户端权限快照，手工推送，支持灾难场景；

全面推广

- 知识库：权限集成手册
- 全业务线推广，已集成几十个业务线系统

案例3：服务化之路—统一通讯框架

目标

接口可描述性高

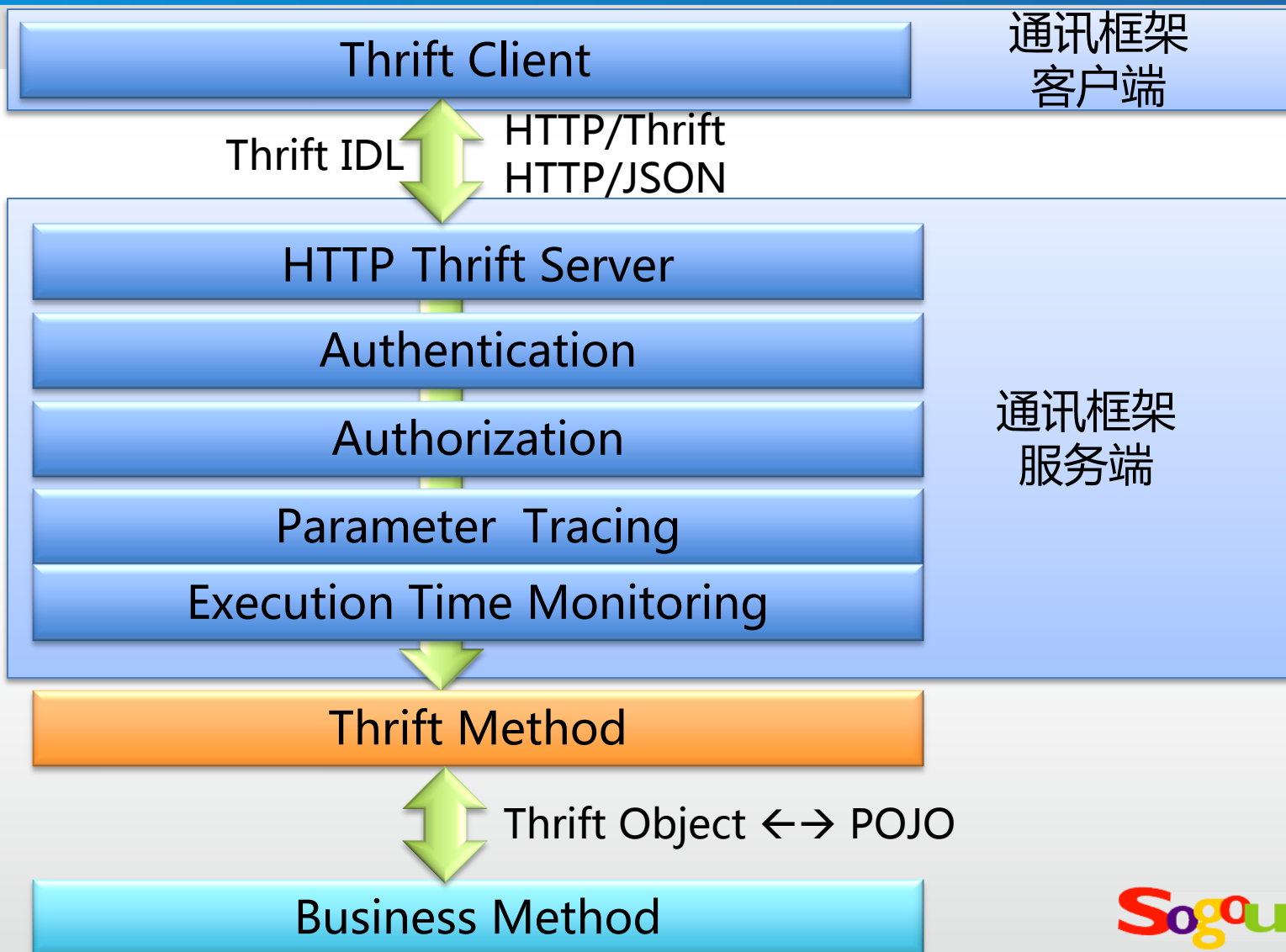
高性能

支持内外部API

POJO友好

降低接口管理成本

案例3：服务化之路—统一通讯框架



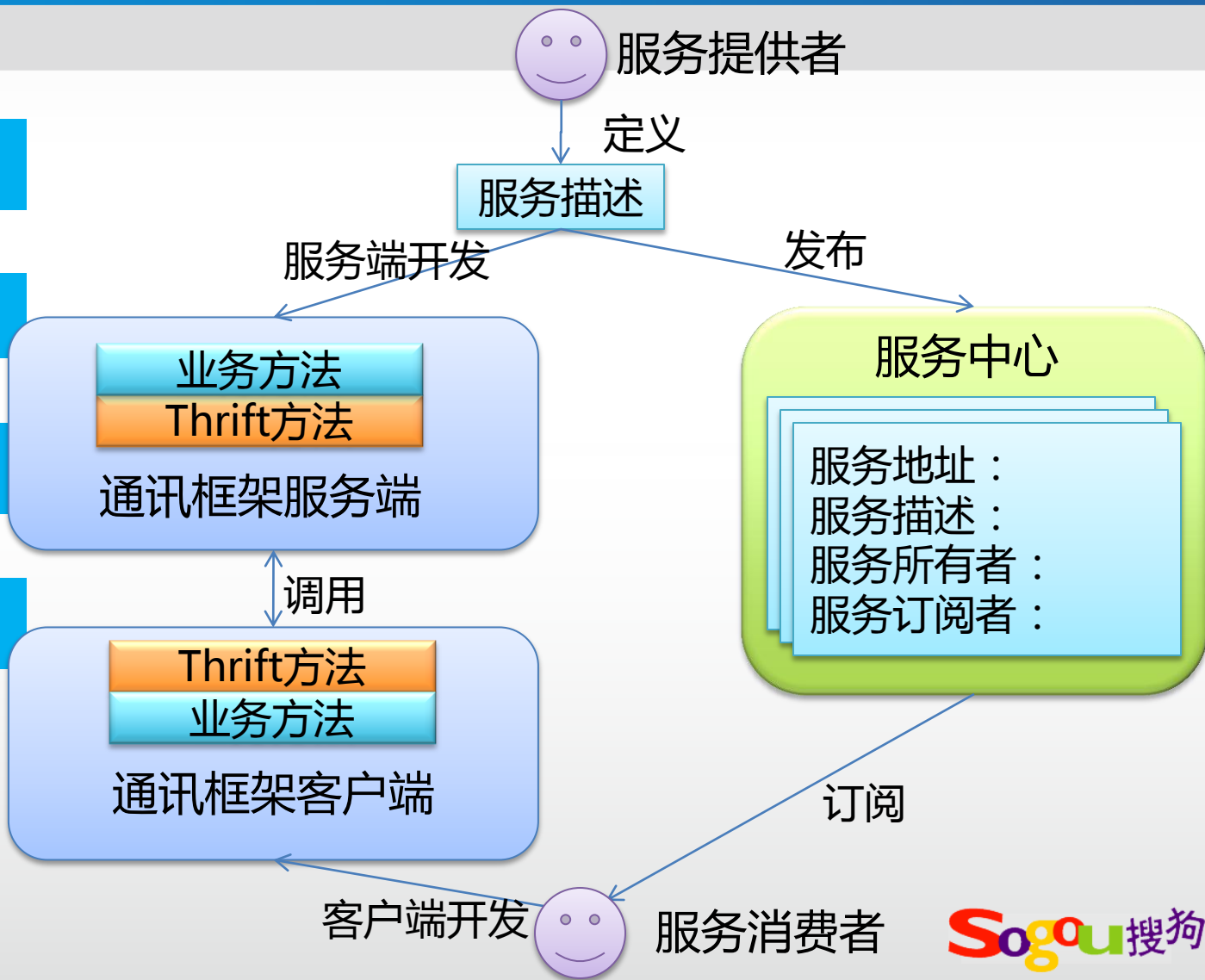
案例3：服务化之路—统一通讯框架

借鉴“SOA”思想

松耦合、无依赖

服务版本管理

服务依赖管理



案例3：服务化之路—统一通讯框架

① Thrift

Facebook / Apache

② 优点

接口定义文档

更多的数据类型支持

跨语言

多种编码方式

社区活跃度高

③ 缺点

无服务地址描述

服务管理

类型侵入

备选方案	考察点	说明
Web Services	性能	XML性能较差
Protocol Buffer	易用性	无接口描述
HTTP/JSON	易用性	无服务接口/地址描述
Dubbo	兼容性 易用性	框架庞大

案例3：服务化之路—统一通讯框架

易用性

Thrift服务端框架：简化
Thrift服务发布成HTTP服务的
流程/防重发

Thrift客户端框架：失败重
试/类RPC调用

简化测试：本地测试；

可维护性

监控日志报警

服务中心

安全性

认证

授权（集成统一权限框架）

演化经验总结



演化风险控制

模式名称

风险

实例

组件 → 服务演化

风险适中

可通过服务（一般是新系统）

提前上线，小规模实验

统一用户中心

短信服务

邮件服务……

组件 → 组件演化
（双路成本低）

风险较小

双路期间性能稍低

统一权限中心

统一配置中心

……

组件 → 组件演化
（双路成本高）

风险较大

一般出现在涉及有状态的功能点；

优先级低→优先级高；

准备完善的回滚方案。

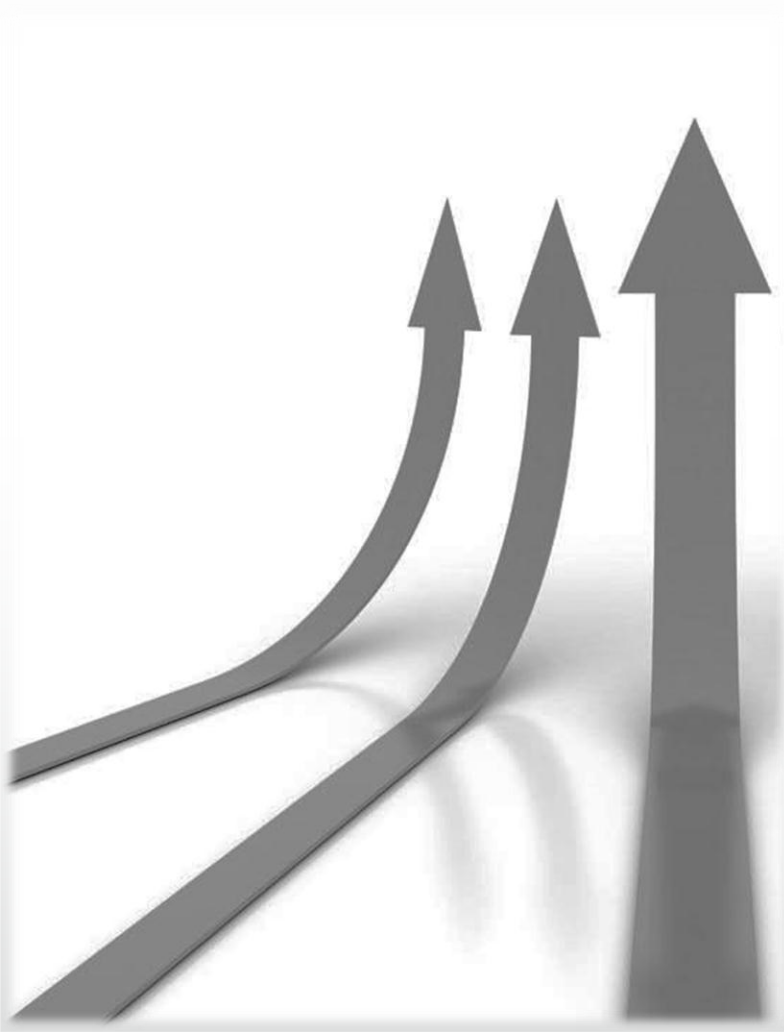
统一通讯框架

分布式会话

分库分表框架

……

主要内容



Java生态演化历程

组件选型漫谈

—— “原则” “评估” “流程”

组件演化实践之旅

—— “那些奋斗的日子”

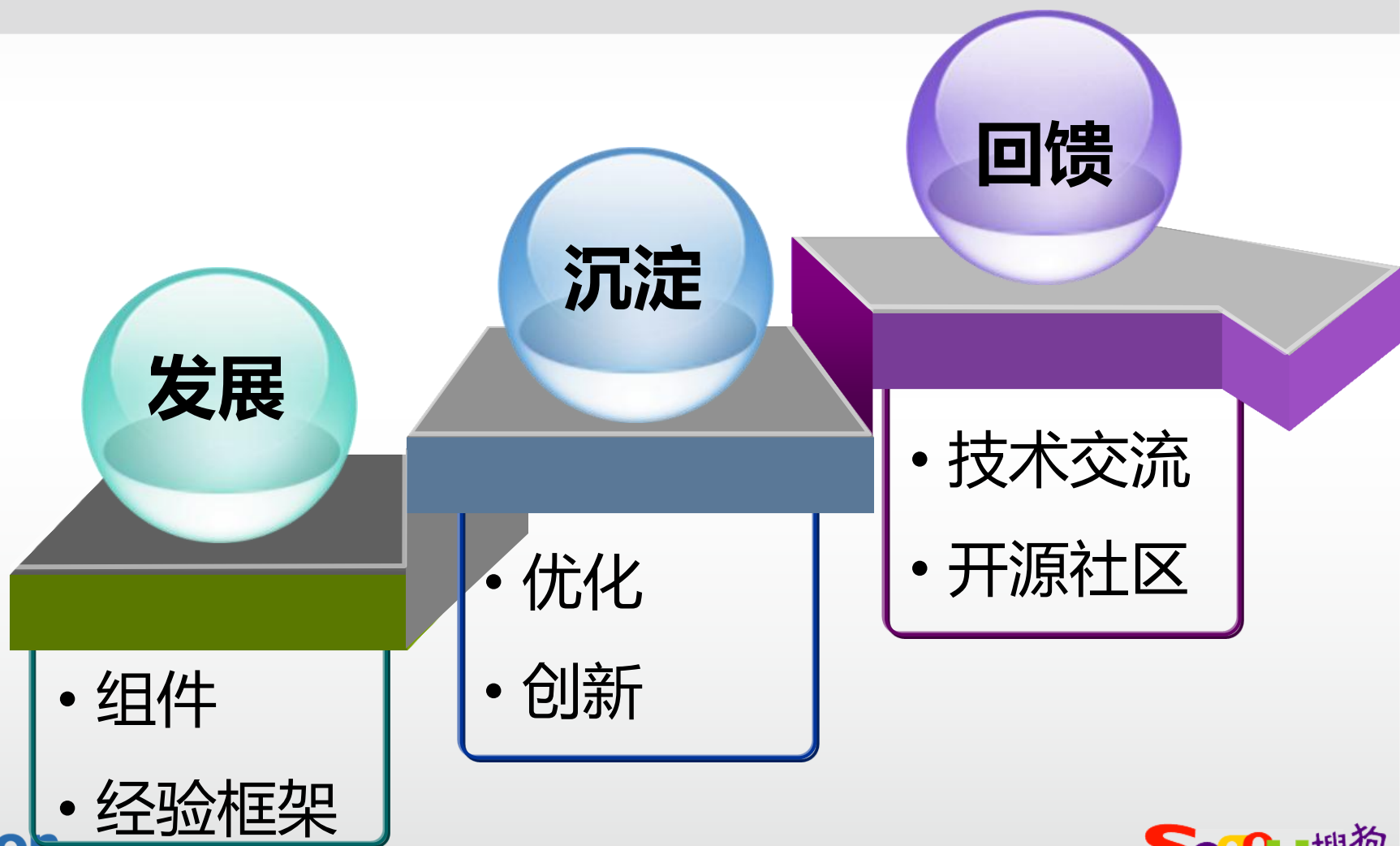
未来展望

商业平台Java生态知识库



基础服务层	SMS	Mail	Image	Messaging
	Authentication	Authorization	Configuration	Scheduling
接口交互层	Thrift	Web Services	HTTP/JSON	
数据计算层	Offline Computing	Streaming Computing		
数据访问层	Datasource Sharding	MongoDB Sharding	ORM	
数据存储层	RDB	MongoDB	DFS	Logging
基础组件层	Distributed Session	Distributed Cache	IoC	

未来展望



Q/A

谢谢大家！

刘建
[@kersus](https://github.com/kersus)

特别感谢合作伙伴



特别感谢媒体伙伴（部分）

網易科技

51CTO.com

福布斯中文网
Forbes
FORBESCHINA.COM

腾讯大讲堂
DJT.QQ.COM

腾讯视频
V.QQ.COM

网易云阅读

腾讯精品课

手机腾讯网
4G.QQ.COM

新浪微盘
vdisk.weibo.com

华章科技
HZ BOOKS

第一财经
C B N

TECH
开发者社区

懒汉互联
汇集/分享 www.lanzhan.cn

w3ctech

ifeve

DOIT
中国 IT 新媒体

ChinaZ.com
China Webmaster 站长之家

OSGi
中文社区

FT 中文网
FTchinese.com

通信产业报
COMMUNICATIONS WEEKLY

ZDNet.com.cn
云计算第一门户

TURING
图灵教育

动点科技
cn.technode.com

IT168.com
www.it168.com