ArchSummit全球架构师峰会北京站2015

1号店订单系统稳定与优化-从SOA规划与治理开始

张立刚 2015.12.18

Geekbang.
极客邦科技

整合全球最优质学习资源，帮助技术人和企业成长

Growing Technicians, Growing Companies

InfoQ
专注中高端技术人员的
技术媒体

EGO EXTRA GEEKS' ORGANIZATION NETWORKS
高端技术人员
学习型社交网络

StuQ
实践驱动的
IT职业学习和服务平台

GiT GEEKBANG INTERNATIONAL TRAINING 极客邦培训
一线专家驱动的
企业培训服务

地址：北京市朝阳区洛娃大厦C座6层1607室　　网址：www.geekbang.org

1号店，沃尔玛网上超市。

2008年7月 11日，“1号店”正式上线。

世界500强沃尔玛子公司，专注于零售和购物者研究的权威咨询机构Kantar Retail发布2015年度中国电商力量综合实力排行榜单中，京东、天猫与1号店位列三甲，第三名超出第四名将近50个百分点。

- 张立刚
  2012年7月加入1号店，架构部技术总监
- 负责1号店订单、库存、拆单、运费、第三方平台订单等电商核心交易系统。
- 致力于构建新一代电子商务核心系统 – 智能OMS订单管理平台。

目录

双十一背后的技术较量 – 系统的稳定与流畅

- 稳定与流畅是用户最直观的体验
- 稳定与流畅的背后是技术的较量
- 技术的背后是什么？ -- 架构
- IT架构、基础架构、应用架构
- 应用（业务）架构 – 从SOA规划和治理开始

双十一背后的技术较量 – 系统的稳定与流畅

如何保证系统稳定，提升可用性？ -- 事前、事中、事后

1、硬件&网络升级

2、架构层面：动态/静态、垂直/水平、读/写分离、解耦/异步、热点/削峰

3、Healthcheck/监控/预警（系统层面/应用层面）

4、分流/限流/降级（主动/被动）

5、踢出与复活/ fail-over/灾备切换/多活

SOA

百度百科对SOA的定义：

面向服务的体系结构（Service-Oriented Architecture，SOA）是一个组件模型，它将应用程序的不同功能单元（称为服务）通过这些服务之间定义良好的接口和契约联系起来。接口是采用中立的方式进行定义的，它应该独立于实现服务的硬件平台、操作系统和编程语言。这使得构建在各种这样的系统中的服务可以以一种统一和通用的方式进行交互。

我们理解的SOA：

SOA是一种架构模式，是设计原则，不是技术规范。

SOA是一个伟大的思想，它试图定义一个大家（各种软件厂商）都“认”的、都“遵循”的法则，大家都使用这样的方法来进行互联互通，从而实现无界限的联通，以及服务组件库的继承和复用，解放无效和重复劳动。

狭义的SOA：Service化 - 标准化、模块化、组件化

广义的SOA：模式、原则、思想

用SOA的模式、原则、思想进行架构的规划和落地实践。

目录

1号店SOA产品规划路线图

1号店Service架构规划

1号店SOA治理之 Service设计原则

1、分层结构
基础Service不含业务逻辑，只封装基本的数据操作
业务（聚合）Service封装业务逻辑甚至是全部的业务逻辑

2、Service层次调用
上层可以调用下层、下层不可调用上层、同层间可互相调用
调用链长度不超过3级、不循环调用

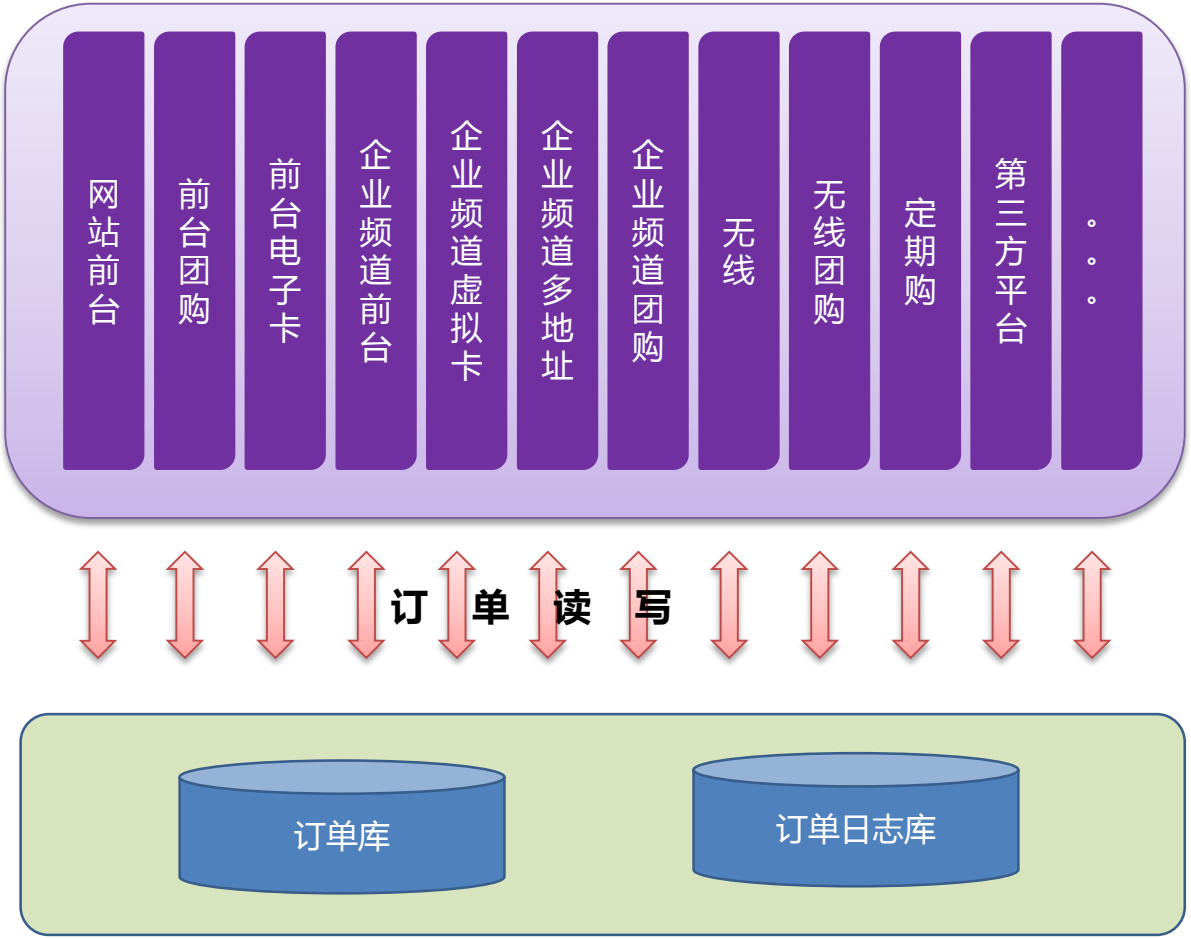
3、离线数据抽取不走Service

4、有日志、监控、授权、流量控制

5、性能基准
读<20ms
写<50ms

目录

早期的订单系统



■ 问题

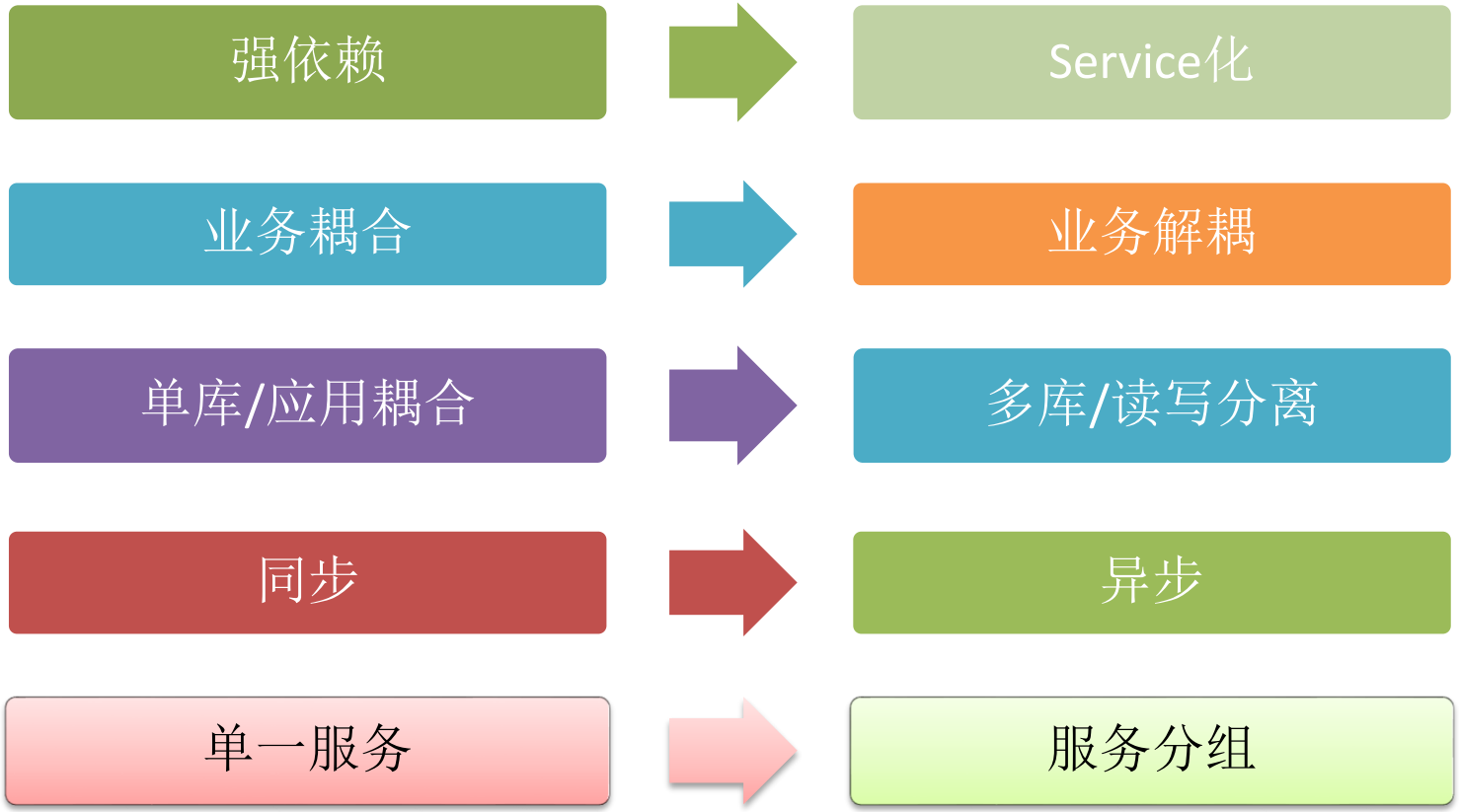
- ✓ 代码耦合
- ✓ 业务互相影响
- ✓ 责任不清
- ✓ 问题定位慢
- ✓ 表结构混乱
- ✓ 监控预警单一

■ 根本原因

- ✓ 原架构已不适应业务的高速发展

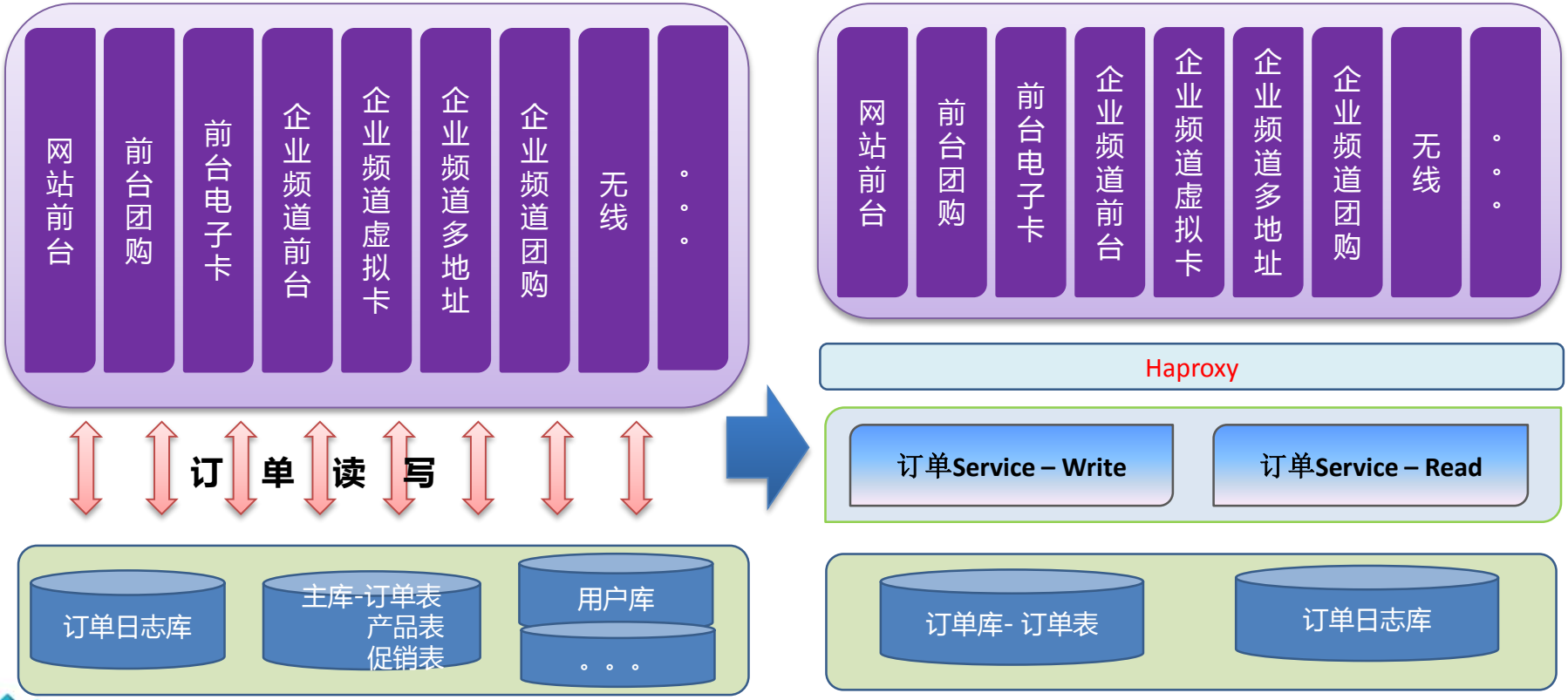
1号店SOA治理之 订单系统解耦

解耦维度



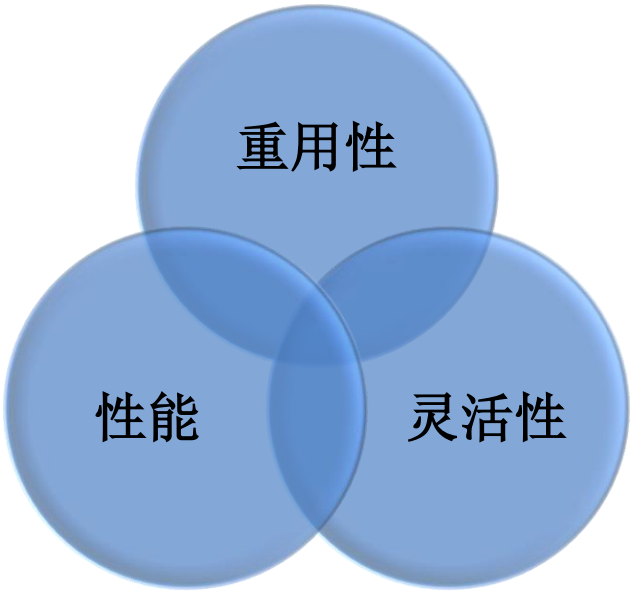
1号店SOA治理之 强依赖->订单Service化

- 合久必分：订单业务逻辑剥离，独立Service化
- 分久必合：接口统一，入口唯一，应用上读写分离



SOA治理之 SOA服务粒度划分

SOA服务粒度设计原则



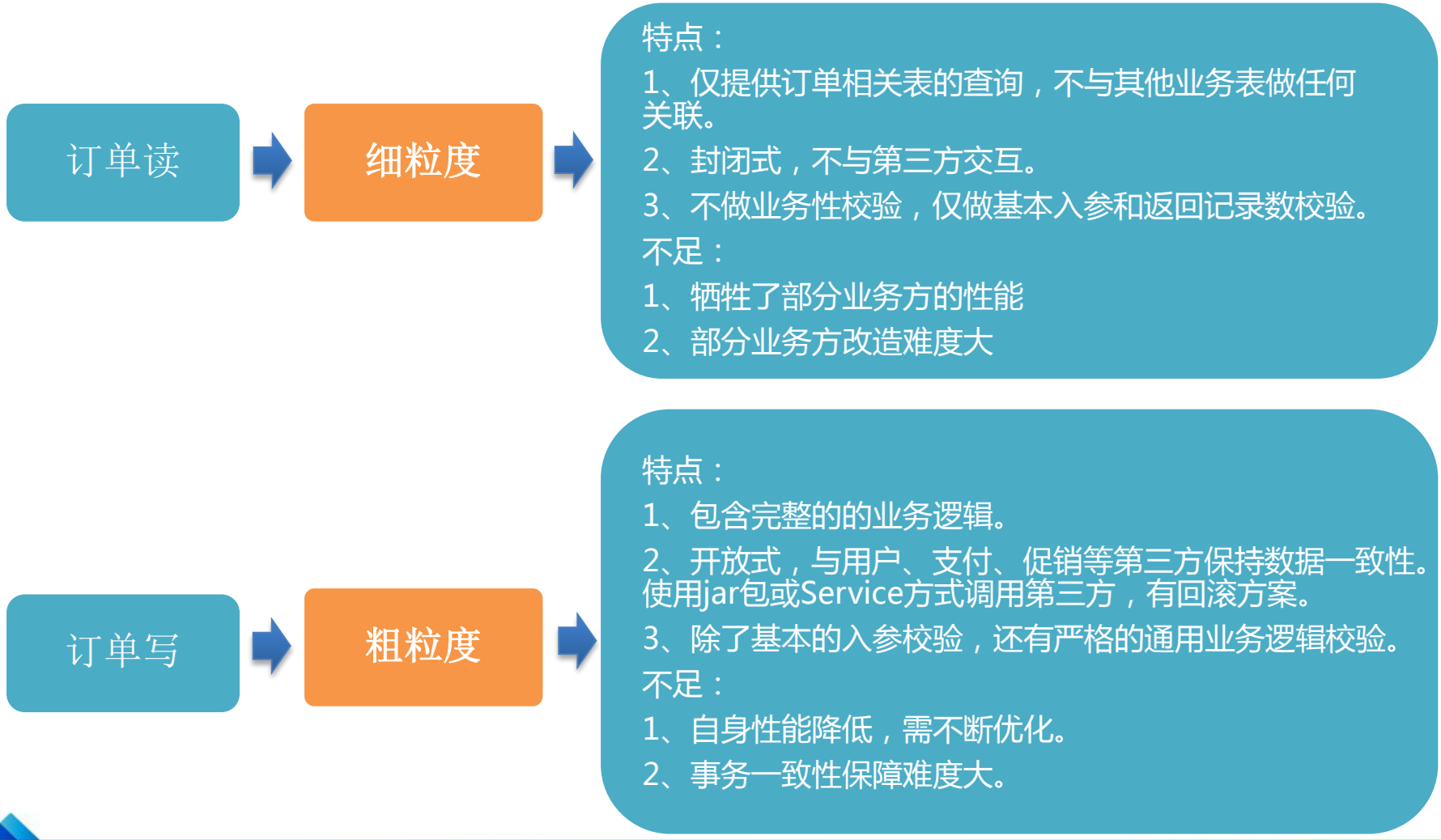
细粒度&粗粒度

- 迷你裙定律“mini-skirt theory”告诉我们：
  一个出色的演讲应该
  “short enough to keep people interested, but long enough to cover the important part”

- 细粒度的服务（fine-grained）提供相对较小的功能单元，或交换少量的数据。
  细粒度的服务使服务更容易被组装

- 粗粒度的服务（coarse-grained）则是在一个抽象的接口中封装了独立的业务/技术能力，减少服务请求交互的次数。
  粗粒度的服务适合更广泛的需求

1号店SOA治理之 订单Service服务粒度划分



1号店SOA治理之 订单Service化历程

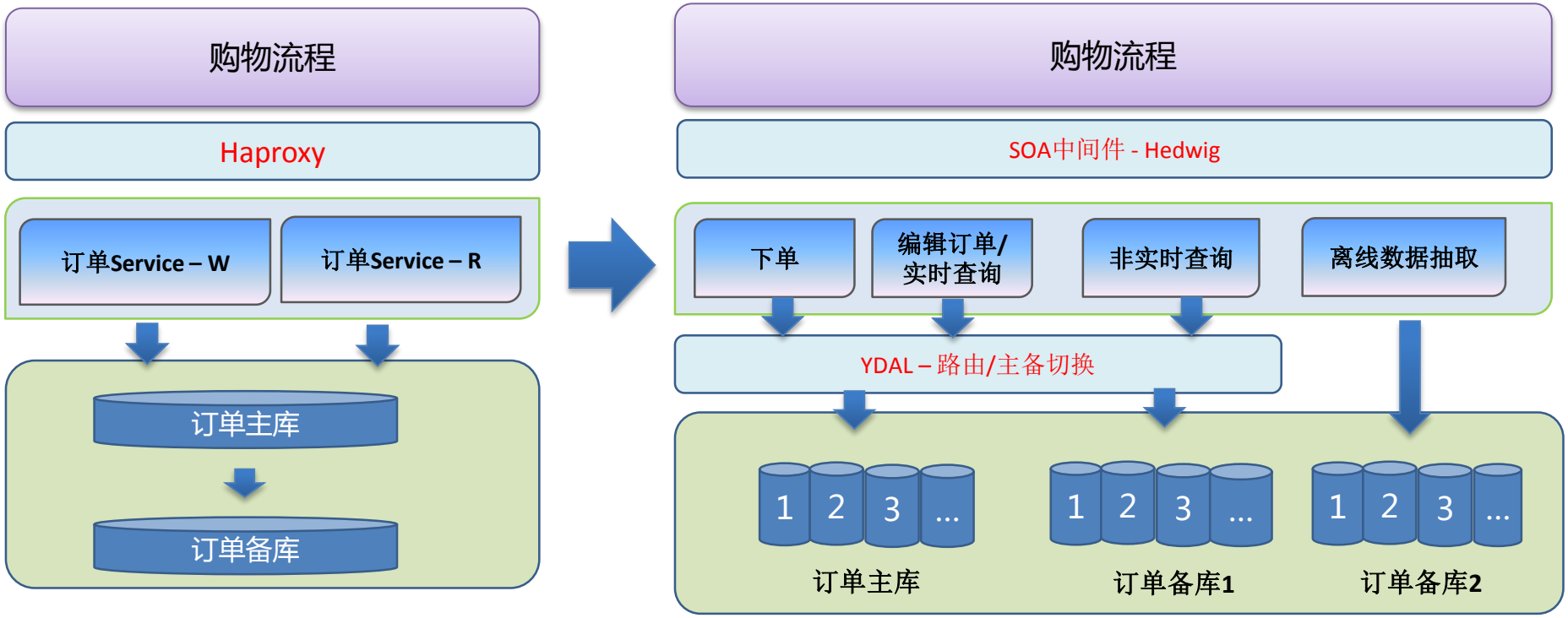
1、SOA规划，业务垂直划分
2、Service日志系统规划
3、SOA服务粒度划分
4、代码规范统一
5、从最复杂的业务-下单入口开始
6、调用方改造、接入订单Service
7、订单Service迁移新Schema，未接入订单Service走原Schema
8、查漏补缺，保证所有调用方接入订单Service
9、订单库物理独立
10、实现应用&DB读写分离

订单系统开启新的历程

目录

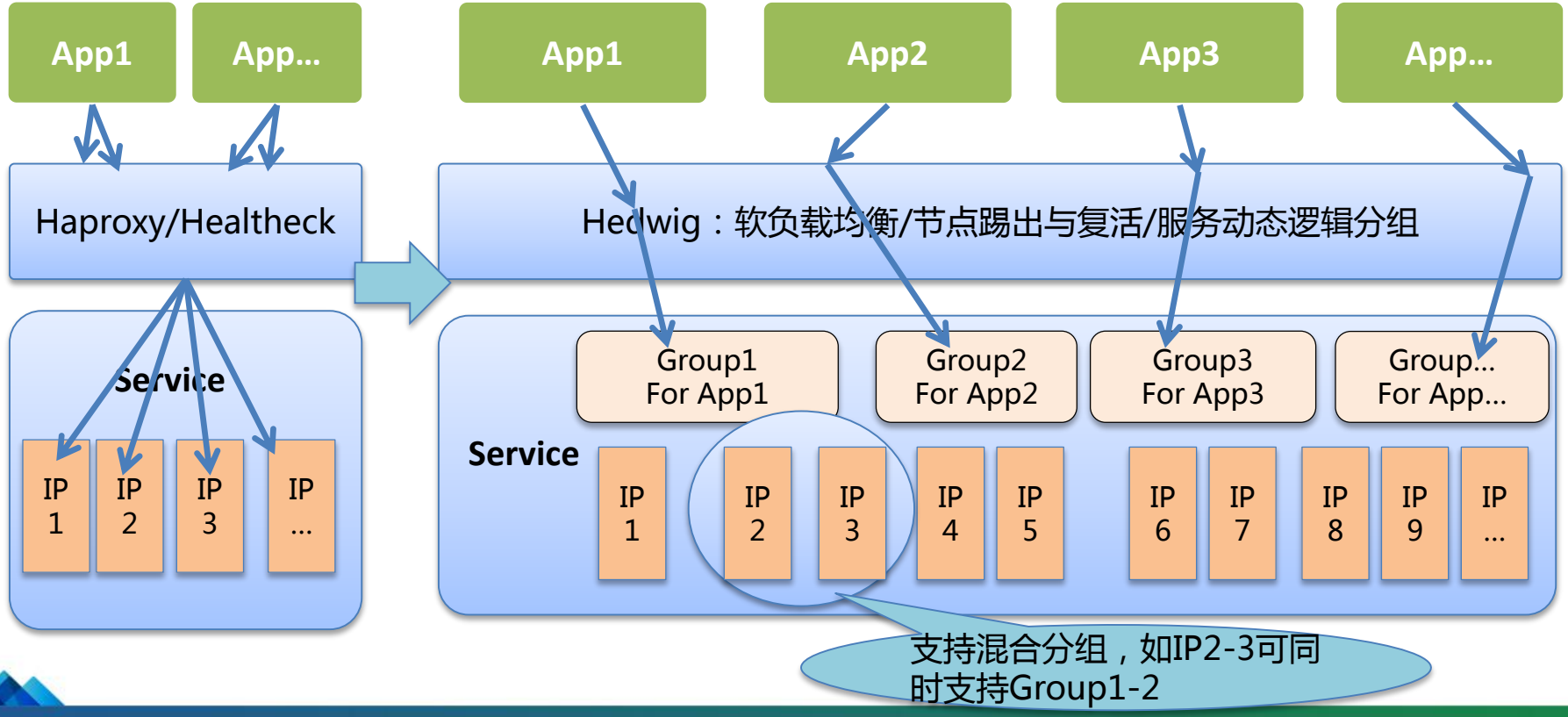
1号店SOA架构之 订单Service服务架构演进

- 读写分离
- 水平拆库
- 服务分组



1号店SOA中间件 - Hedwig

Hedwig提供了服务自动注册发现、软负载均衡、节点踢出与复活、服务动态逻辑分组、请求自动重试等众多SOA框架所需的强大功能，支持并行请求、灰度发布，其背后提供的调用链路及层次关系、日志分析、监控预警等更是为SOA治理提供了强大的后勤保障



1号店SOA架构之 水平拆分

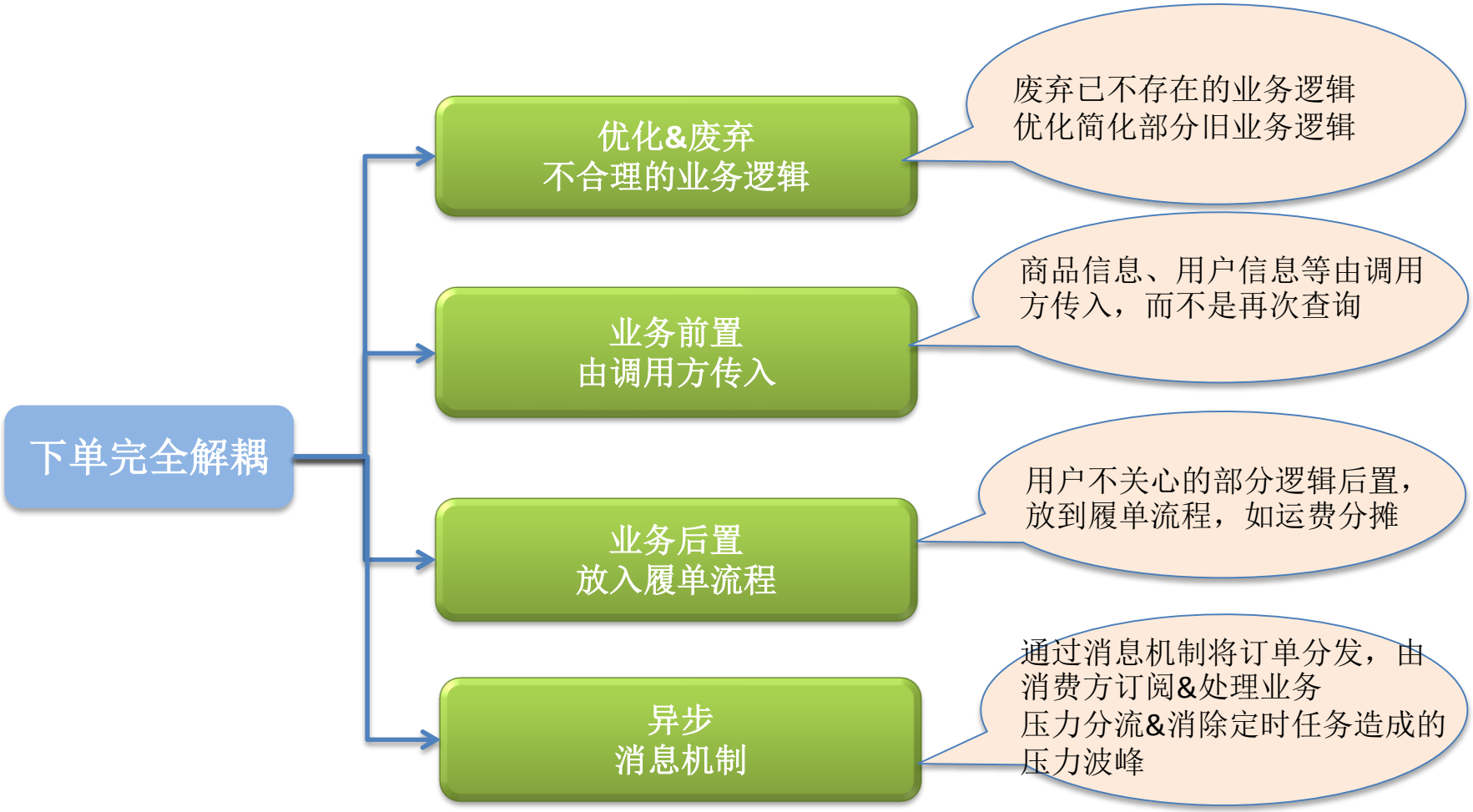
- 拆分原则

数据类型	记录数量	数据增量	读数据量	写数据量	硬盘空间	读缓存时效	写事务一致性要求	水平拆分维度	热点数据可能性
订单	大	很大	大	大	很大	无	强	订单ID/用户/商家	小/一般/大

1号店SOA治理之 下单系统解耦

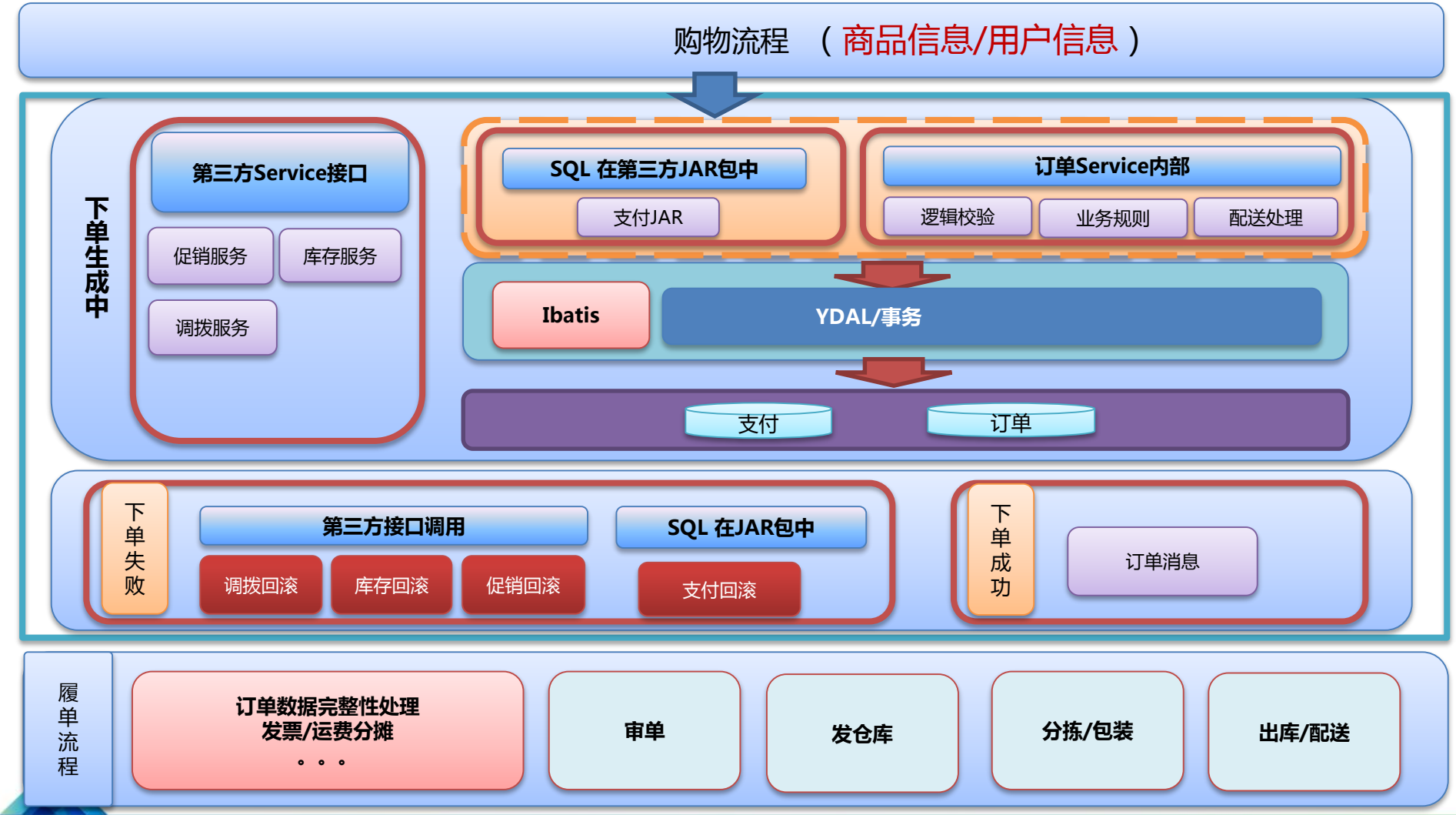
订单Service化初完成后的下单接口服务 – 太多耦合&依赖

1号店SOA治理之 订单系统业务

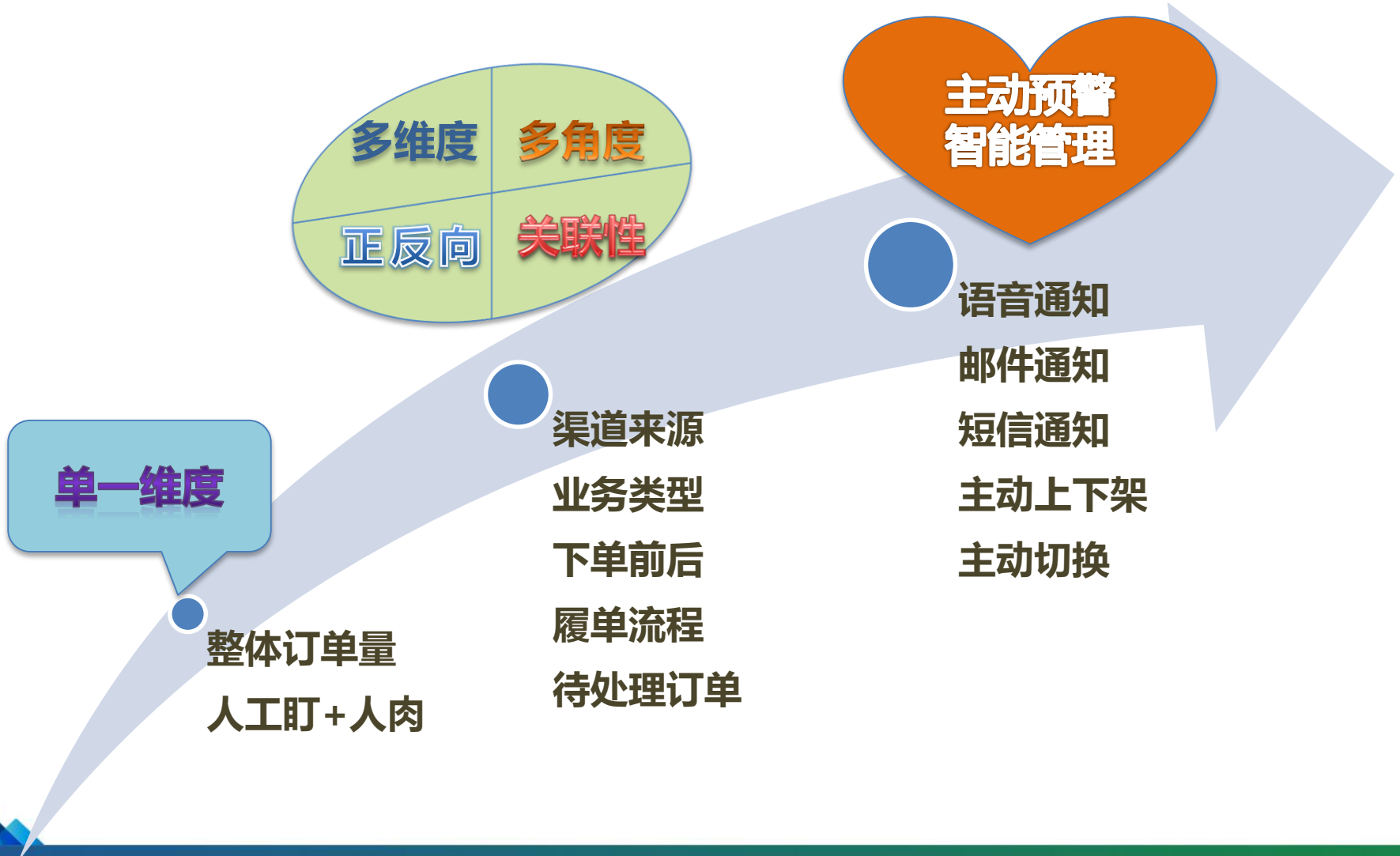


1号店SOA治理之 下单业务解耦

订单Service化，优化后的下单接口服务



1号店SOA治理之 业务监控



目录

总结：稳定与性能是分不开的

Service化只是开始，用SOA的思想去做业务的规划和治理是关键。

如何保证系统稳定，提升可用性？

1、硬件&网络升级

2、架构层面：动态/静态、垂直/水平、读/写分离、解耦/异步、热点/削峰

3、Healthcheck/监控/预警（系统层面/应用层面）

4、分流/限流/降级（主动/被动）

5、踢出与复活/ fail-over/灾备切换/多活

6、应用架构层面 - 降低出错概率：
业务逻辑优化，少调用第三方接口，少同步读写操作，业务分流、异步，业务前置/后置。

如何提升系统性能？

1、硬件&网络升级

2、架构层面：动态/静态、垂直/水平、读/写分离、解耦/异步、热点/削峰

3、Healthcheck/监控/预警（系统层面/应用层面）

4、分流/限流/降级（主动/被动）

5、代码优化、SQL优化，使用新技术。能提升多少？

6、应用架构层面 – 减少直接调用：
业务逻辑优化，少调用第三方接口，少同步读写操作，业务分流、异步，业务前置/后置。

Thanks!