

微服务架构设计与实践

知识原理篇



孙玄@58集团

关于我

✓ 58集团技术 **QCon** 全球软件开发大会 主席



✓ 58集团高级 **SDCC** 中国软件开发大会 架构师
Software Developer Conference China



✓ 转转架构师 **PROGRAMMER** 程序员 负责人



✓ 百度高级工程师

✓ 毕业于浙江大学

✓ 代表公司多次对外分享

✓ 企业内训&公开课



关于我

企业内训

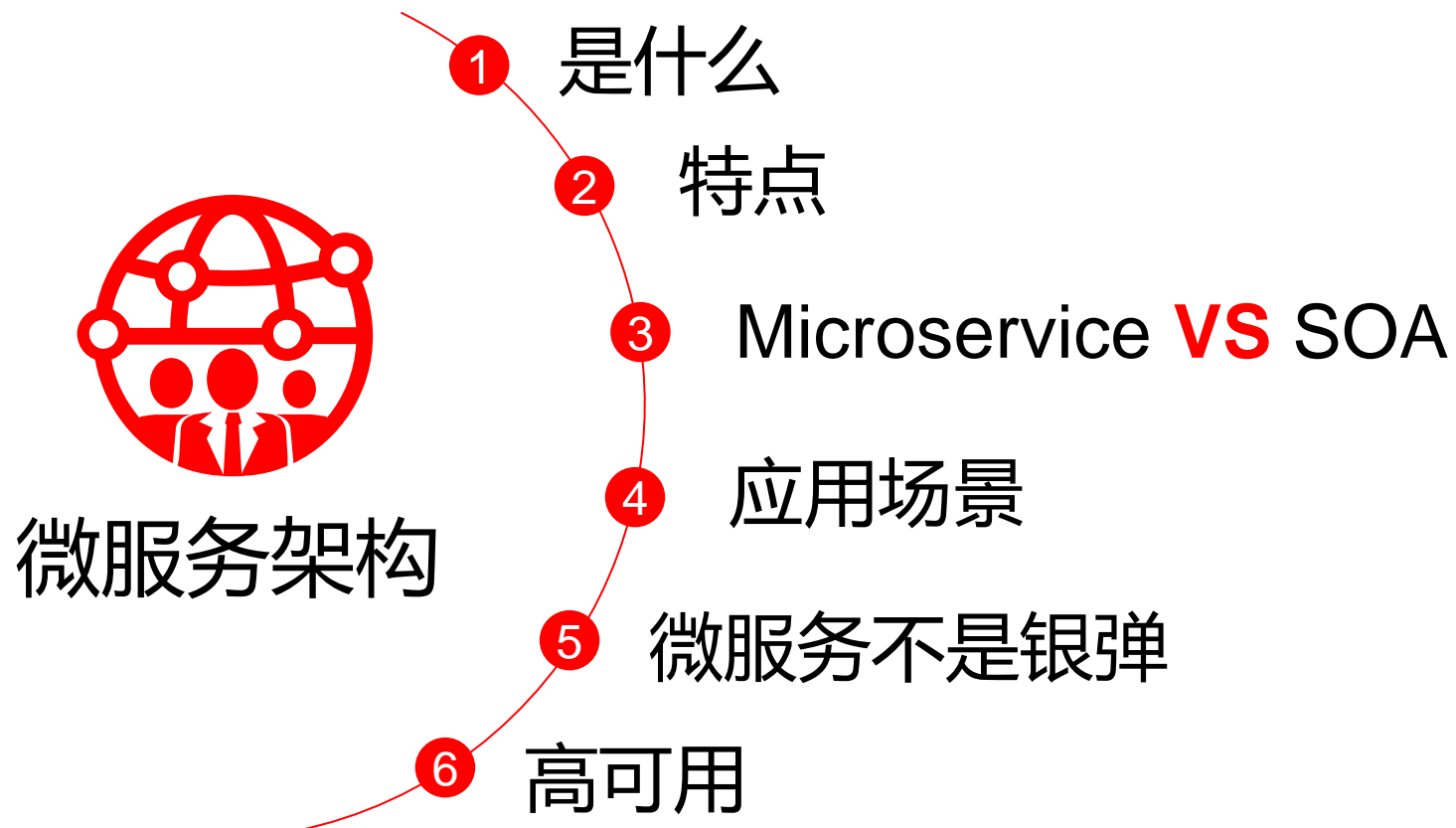
- ✓ 华为
- ✓ 中航信
- ✓ 平安
- ✓ 银联
- ✓ 华泰证券
- ✓ 思科

- ✓ 云南电力
- ✓ 深信服
- ✓ 新华社
- ✓ 民生银行
- ✓ 招商银行
- ✓

公开课

- ✓ 北京
- ✓ 上海
- ✓ 深圳
- ✓ 广州
- ✓ 成都
- ✓

分享要点



微服务架构

- Microservice

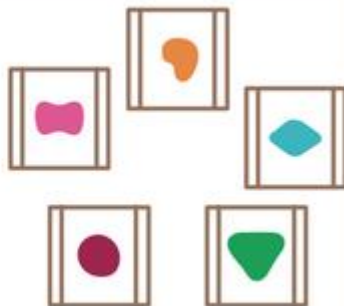
In short, the microservice architectural style is an approach to developing a single application as a **suite of small services**, each **running in its own process** and communicating with lightweight mechanisms, often an HTTP resource API. These services are **built around business capabilities** and **independently deployable** by fully automated deployment machinery. There is a **bare minimum of centralized management** of these services, which may be written in different programming languages and use different data storage technologies.

-- James Lewis and Martin Fowler

Microservices

common characteristics of this architectural style

by James Lewis and Martin Fowler



微服务

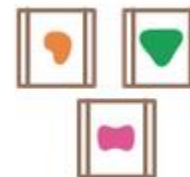
- Monoliths VS MicroServices

- 架构

A monolithic application puts all its functionality into a single process...

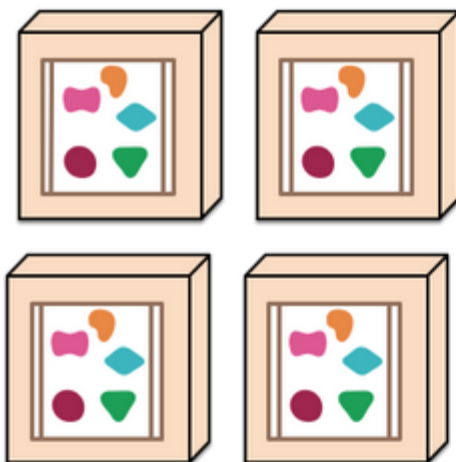


A microservices architecture puts each element of functionality into a separate service...

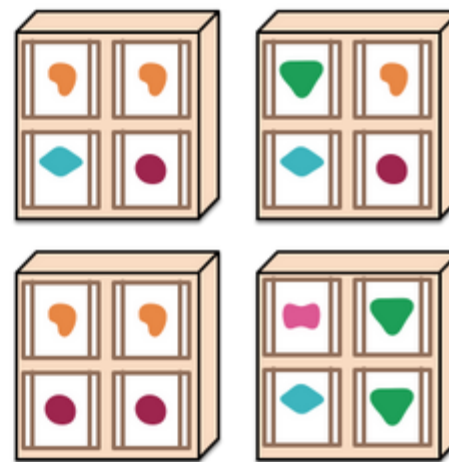


- 扩展

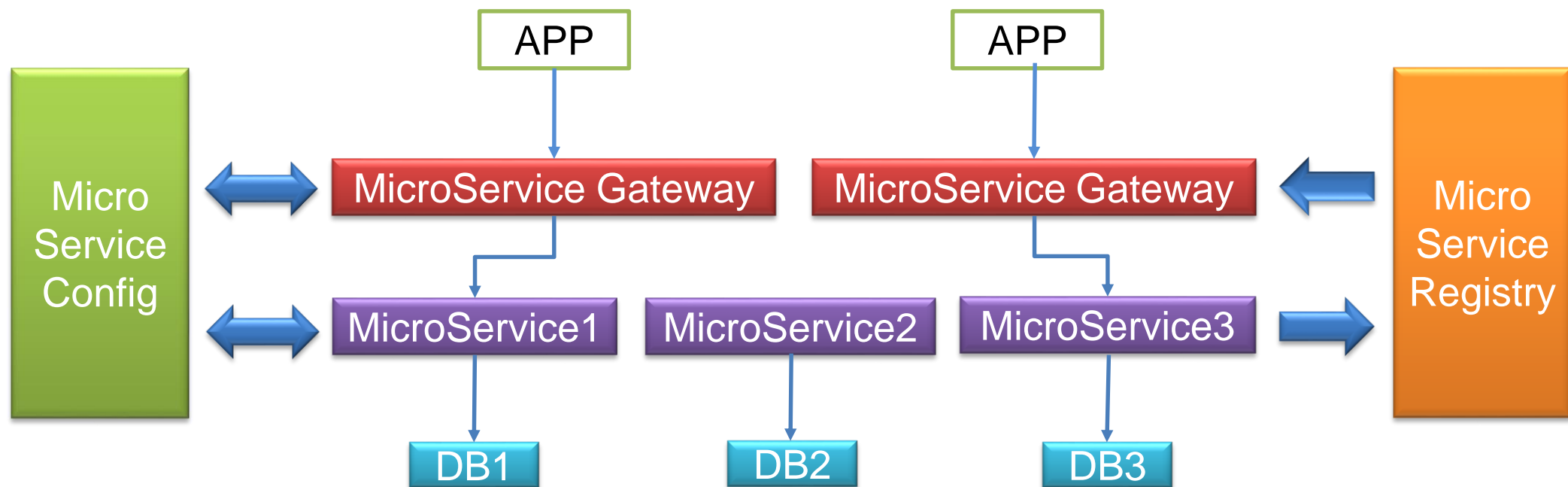
... and scales by replicating the monolith on multiple servers



... and scales by distributing these services across servers, replicating as needed.



微服务架构



特点



微服务架构



微服务粒度



独立进程



围绕业务建模



轻量级通信



去中心化管理



特点

微服务颗粒化

职责单一化

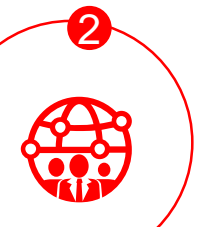
进程自主化

通信轻量级化

管理自动化

团队自主化

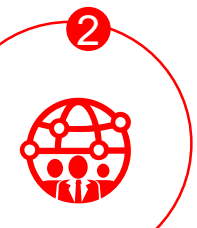
2



Microservice **VS** SOA

✓ SOA(Service Oriented Architecture)

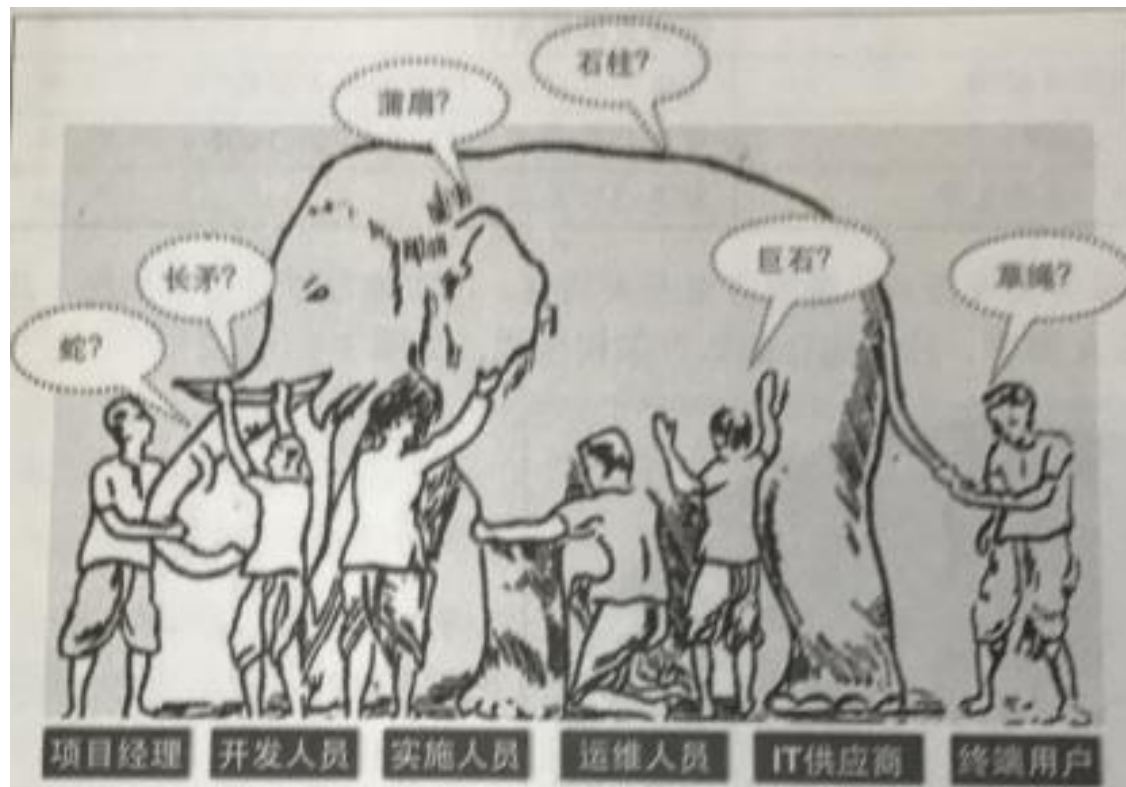
对于复杂的企业IT系统，应该按照不同的、可复用的粒度划分，将相关的一组功能提供者组织在一起为消费者提供服务。



Microservice **VS** SOA

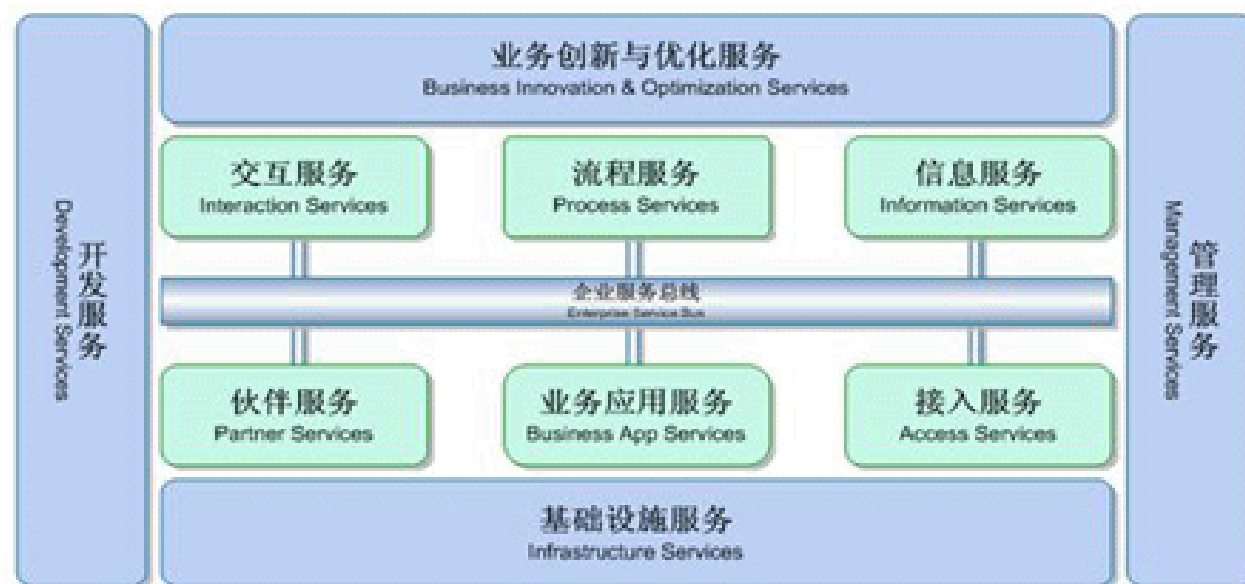
✓ SOA(Service Oriented Architecture)

- ✓ 1996年提出
- ✓ 不同认知和理解
- ✓ 2000年逐步落地
 - ✓ ESB
 - ✓ WebService
 - ✓ SOAP
 - ✓



Microservice **VS** SOA

✓ SOA(Service Oriented Architecture)



Microservice **VS** SOA

✓ Microservice

✓ 互联网高速发展

✓ 快速迭代/MVP

✓ 基础设施完备

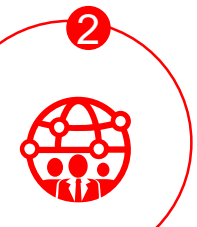
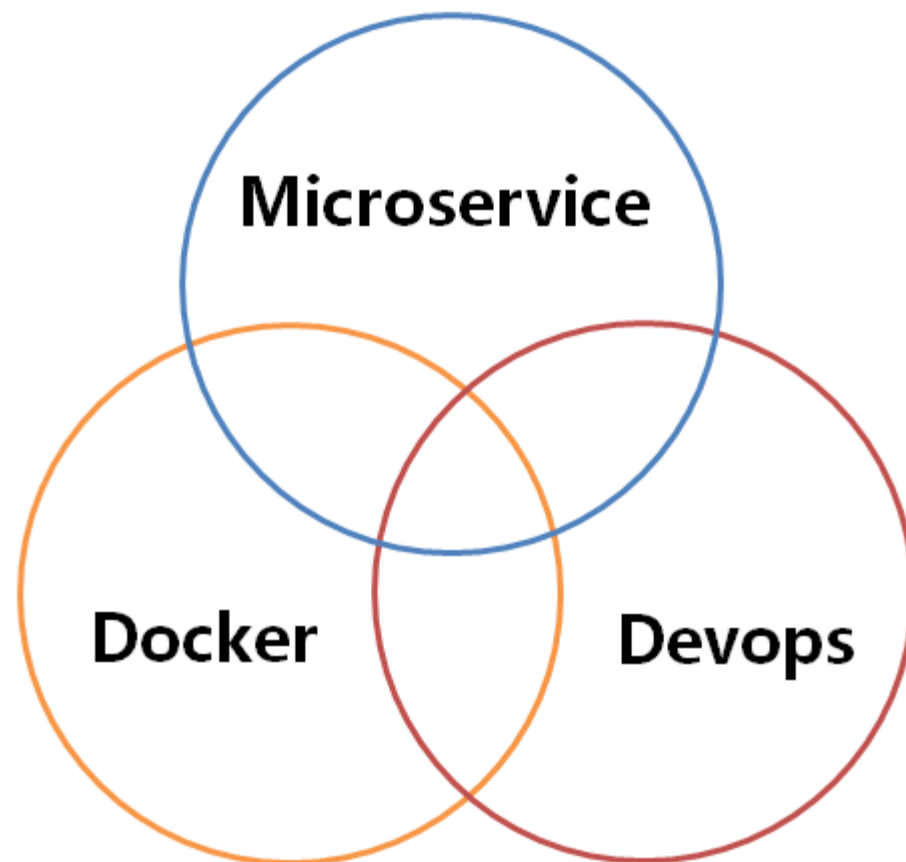
✓ 敏捷开发

✓ 持续集成

✓ 持续交付

✓ DevOps

✓ 虚拟化



Microservice **VS** SOA

**微服务架构不再使用SOA架构里比较重的ESB企业服务总线，
同时SOA的思想进入到微服务内部实现真正的组件化。**



Microservice **VS** SOA

✓ 区别

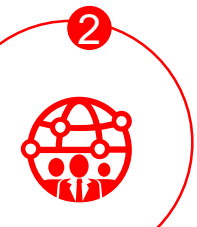
MicroServices	SOA
自底向上实施落地（团队级）	自顶向下实施落地（公司级）
一个系统被拆成多个服务（粒度细）	服务由多个子系统组成（粒度粗）
松散式服务架构（无ESB）	集中式服务架构（有ESB）
集成方式简单（REST/HTTP/RPC）	集成方式复杂（ESB/WS/SOAP）
Monoliths（子系统相互依赖，部署复杂）	微服务独立部署
团队规模小	团队规模大



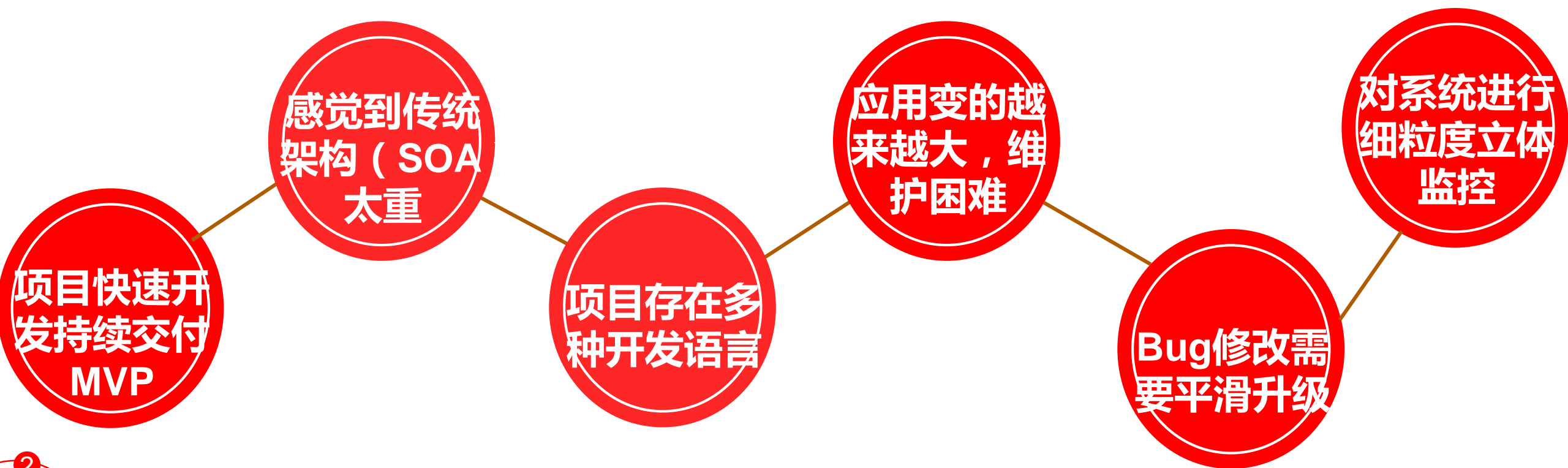
Microservice **VS** SOA



微服务架构是SOA架构一种轻量级的实施方案



使用场景



使用微服务架构比较合适



微服务不是银弹



分布式系统复杂性



运维成本高



部署自动化



微服务依赖测试



微服务依赖管理

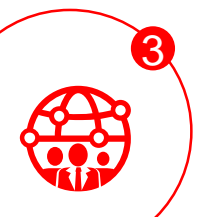


DevOps与组织架构

微服务不是银弹

分布式
系统
复杂性

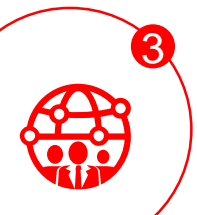
- ✓ 微服务架构是分布式系统
- ✓ 性能开销（跨进程、跨网络、多次调用）
- ✓ 可靠性难度大（微服务多、调用失败）
- ✓ 数据一致性（最终一致/分布式事务）
- ✓ 异步通信（提升吞吐量/提升实现复杂度）
- ✓ 开发调试（工具开发调试分布式系统支持较差）



微服务不是银弹

运维
成本高

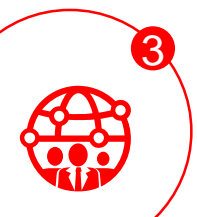
- ✓ 配置（配置中心）
- ✓ 部署（微服务部署到制定环境）
- ✓ 日志采集（App、Server）
- ✓ 监控与报警（进程监控、语义监控、错误日志监控）



微服务不是银弹

部署 自动化

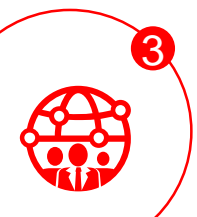
- ✓ 微服务独立部署
- ✓ 互联网高速发展、迭代速度快、部署频繁
- ✓ 手动部署成本高，不再适用
- ✓ 自动化部署流水线，降低部署成本，提高部署效率



微服务不是银弹

微服务
依赖测试

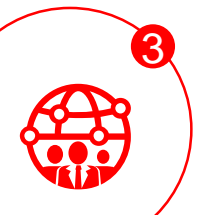
- ✓ 接口测试
- ✓ 按照约定接口正常工作
- ✓ 微服务多
- ✓ 测试面临挑战



微服务不是银弹

微服务
依赖管理

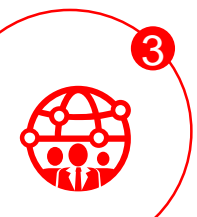
- ✓ 微服务多个独立服务
- ✓ 清晰有效展示服务依赖关系
- ✓ 挑战大



微服务不是银弹

DevOps 组织架构

- ✓ 微服务不仅是架构模型，也是组织模型
- ✓ Monoliths职能组织架构
- ✓ 践行DevOps文件价值
- ✓ 构建全功能团队
- ✓ 挑战较大



高可用

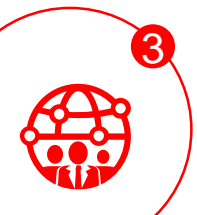
✓ 微服务依赖基础不高可用

硬件

- ✓ 生命周期
- ✓ 硬件故障

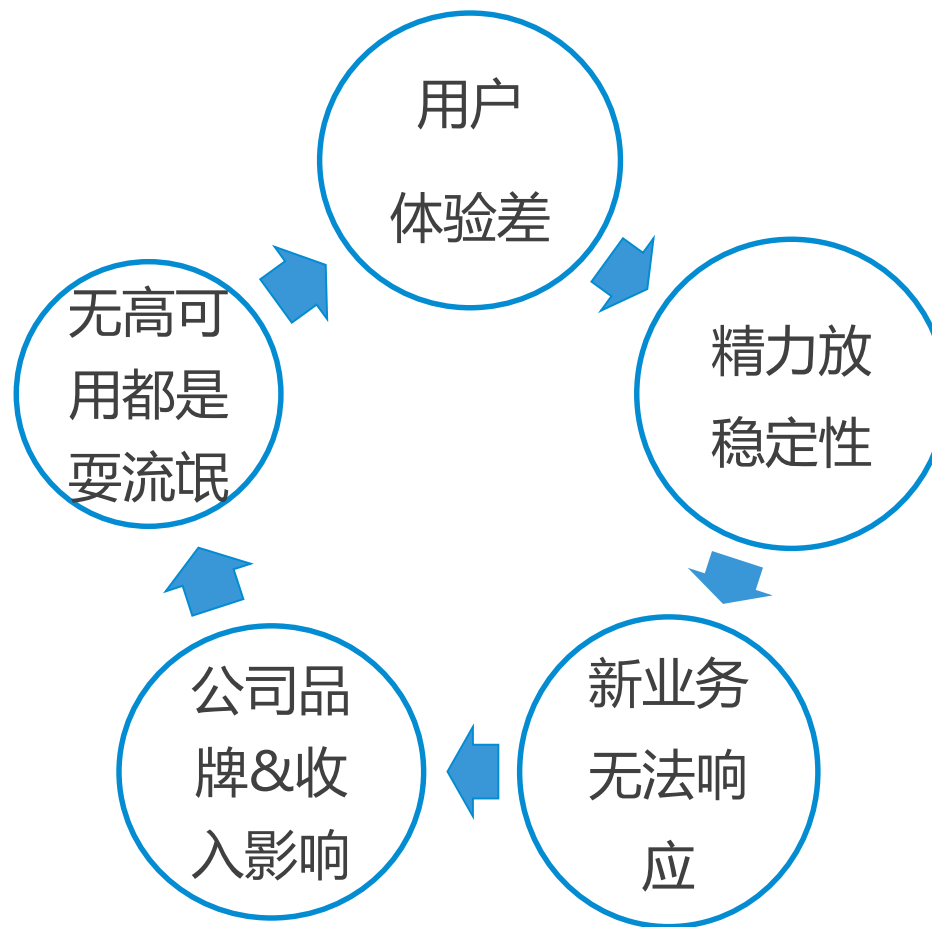
软件

- ✓ 软件Bugs
- ✓ 软件间资源抢占



高可用

高可用重要性



高可用

高可用手段

微服务冗余

1

无状态化

负载均衡

2

幂等设计

超时机制

3

异步化

微服务降级

4

分级管理

5

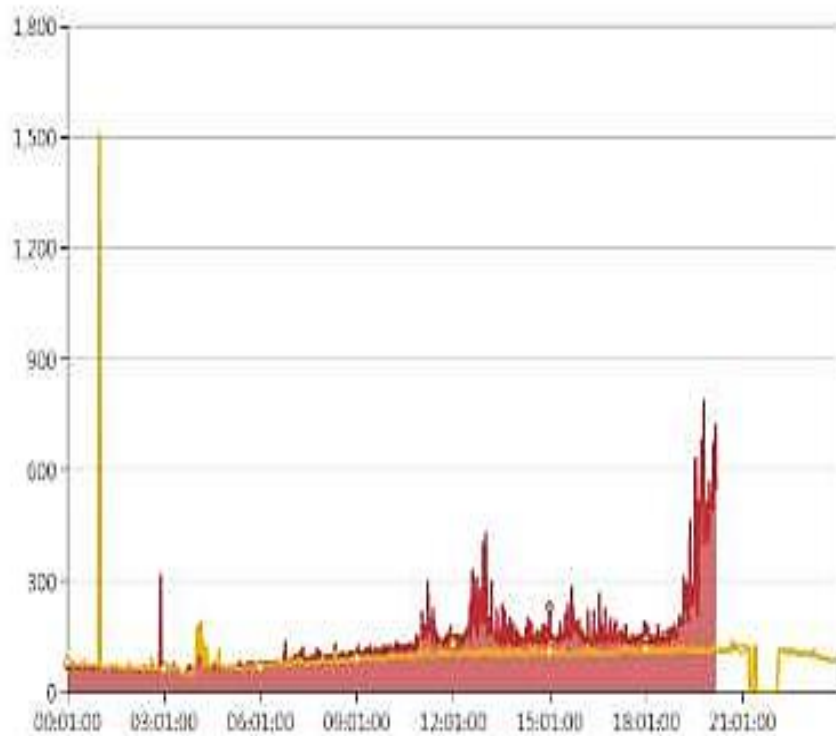
微服务治理



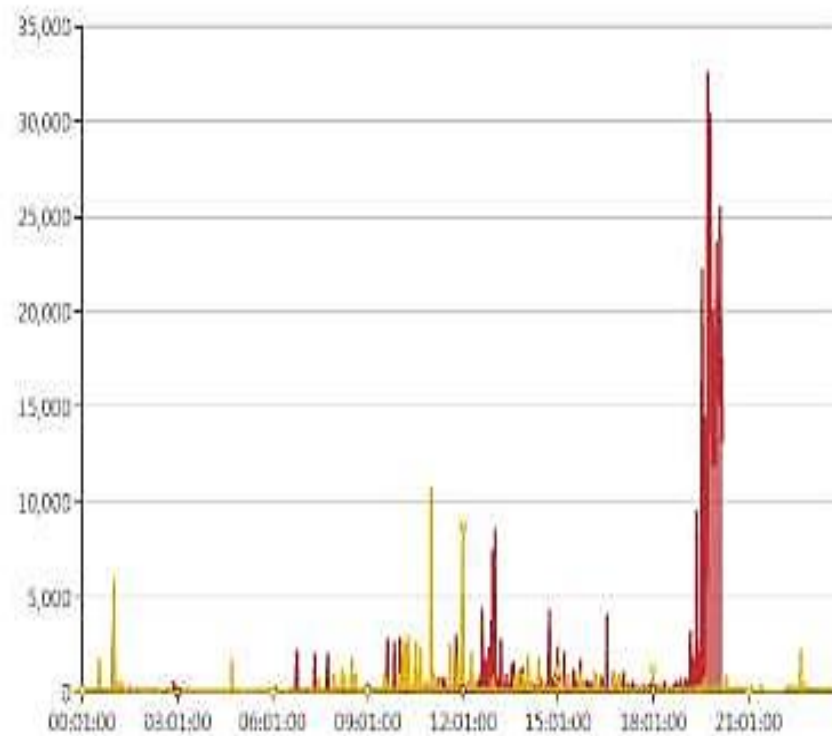
高可用

微服务实时监控

请求平均耗时



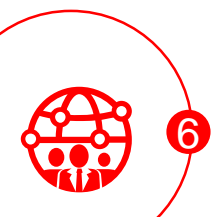
请求异常条数



高可用

高可用评价维度

描述	通俗叫法	可用性级别	年度停机时间
基本可用性	2个9	99%	87.6小时
较高可用性	3个9	99.9%	8.8小时
具有故障自动恢复能力的可用性	4个9	99.99%	53分钟
极高可用性	5个9	99.999%	5分钟



高可用

微服务分级&考核评价维度



目标：微服务出现故障，如何评估事物影响面



高可用

微服务分级&考核评价维度

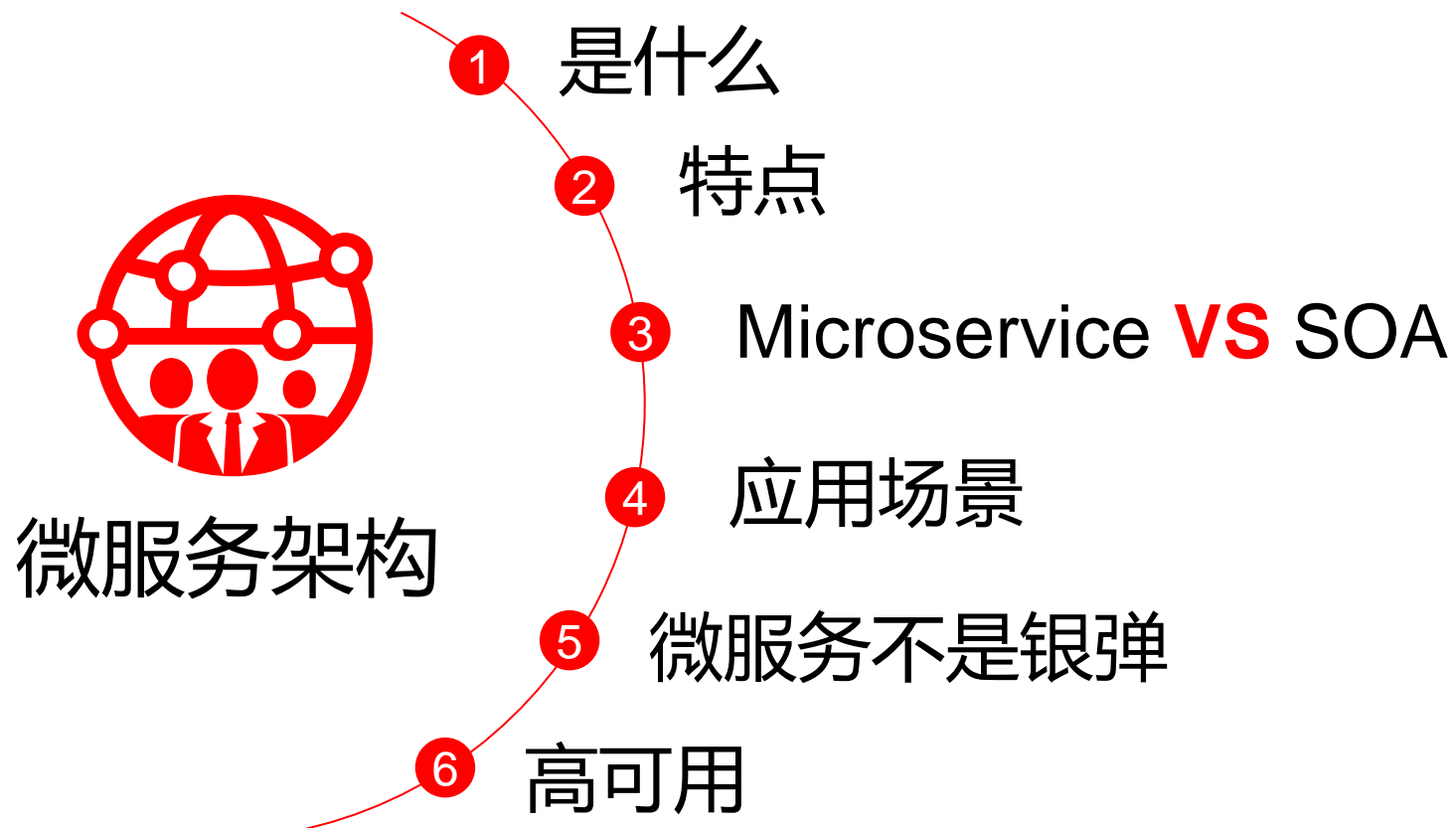
服务级别	定义标准
一级服务	1. 服务每天 pv 达到 5000w 以上 2. 收入达到公司在线收入的 1/10 3. 核系统 4. 后端基础服务至少为 1 个一级服务提供主要服务
二级服务	1. 服务每天 pv 达到 1000w 以上 2. 收入达到公司在线收入的 1/20 4. 重要商务系统 3. 后端基础服务至少为 1 个二级服务提供主要服务 4. 公司内部核心信息系统
三级服务	其他

事故定级	影响程度低	影响程度中	影响程度高
一级服务	严重	重大	特大
二级服务	一般	严重	重大
三级服务	N/A	一般	严重

服务级别	服务名称
一级服务	列表系统, 详情页系统, 发布系统, 招聘业务服务, 计费系统, 支付系统, 订单系统 IDC 网络, DNS
二级服务	消息系统, 个人中心, 呼转服务, 交友系统, CRM, OA, 权限系统, BI 系统 办公网络, VPN
三级服务	其他

影响程度定级	影响程度低	影响程度中	影响程度高
对外完全停止服务时间	2-7 分钟	7-30 分钟	30 分钟以上
系统对用户产生拒绝, 占当日预期流量的比例	0.1%-1%	1%-6%	6%以上
系统返回结果不符合预期占当日总流量的比例	0.5%-2%	3%-15%	15%以上
受影响的用户占系统用户比例	3%-20%	20%-40%	40%以上
影响数据程度	部分数据丢失但能恢复	部分数据丢失不能恢复	所有数据丢失但不能恢复
收入损失: 占日平均营收(线上+线下)比例	2%-5%	5%-30%	>30%
网络丢包率	5%-10%	10%-30%	>30%

要点回顾

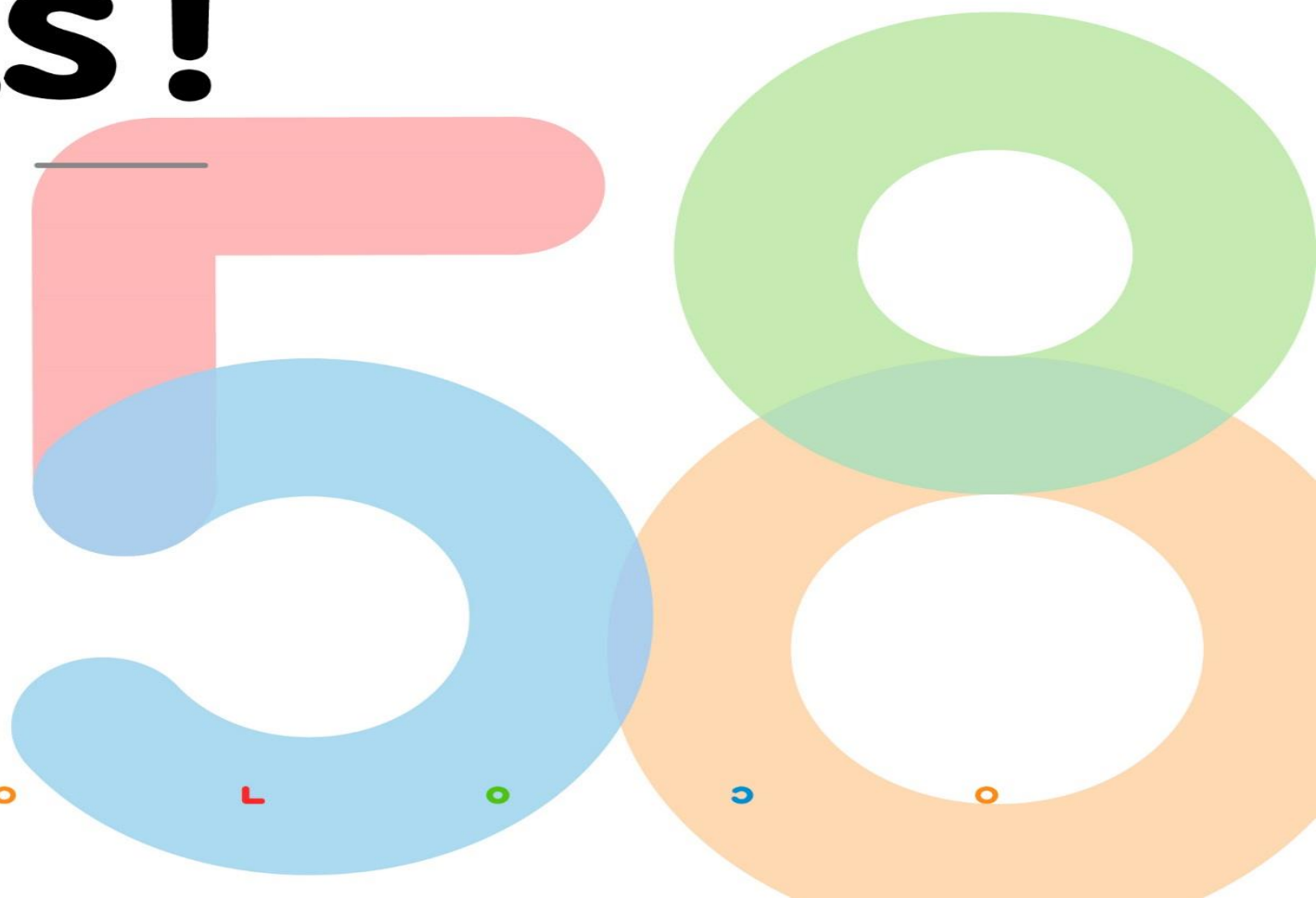


欢迎关注本人公众号 “架构之美”



Thanks!

让生活更简单



L

O

S

O

L

O

S

O