
微服务架构案例

使用微服务架构改造企业核心业务系统的实践

一些数据

30%

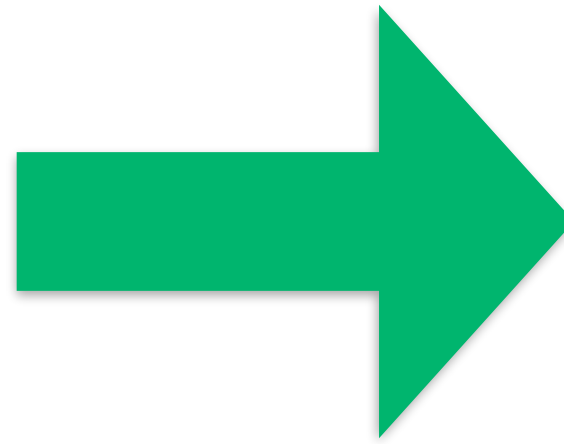
3x

10x

一年前.....

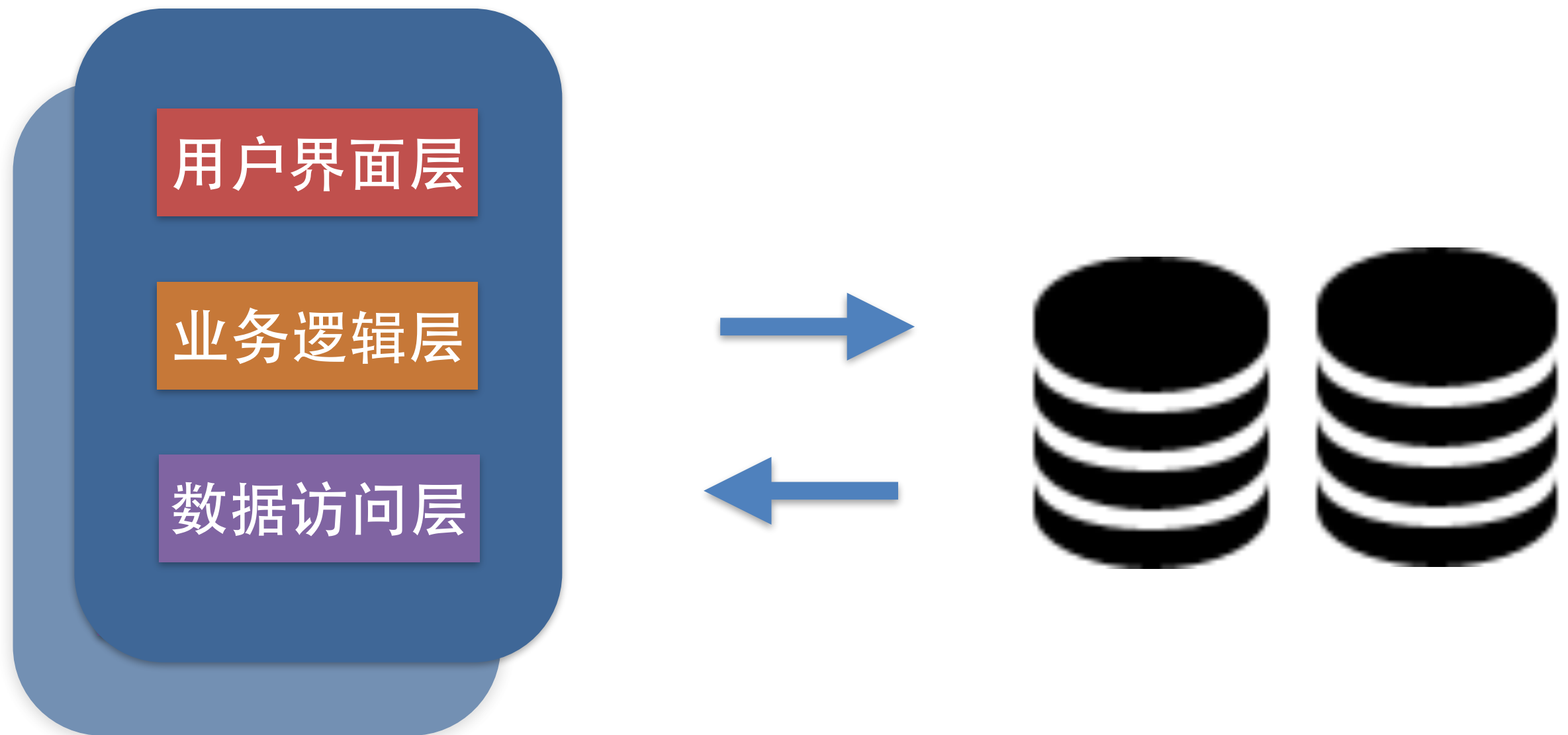


10万元合同
5亿元
5亿元



3.5人天
?????
66人年

CRM



CRM

400K_{Loc}

10_{+ members}

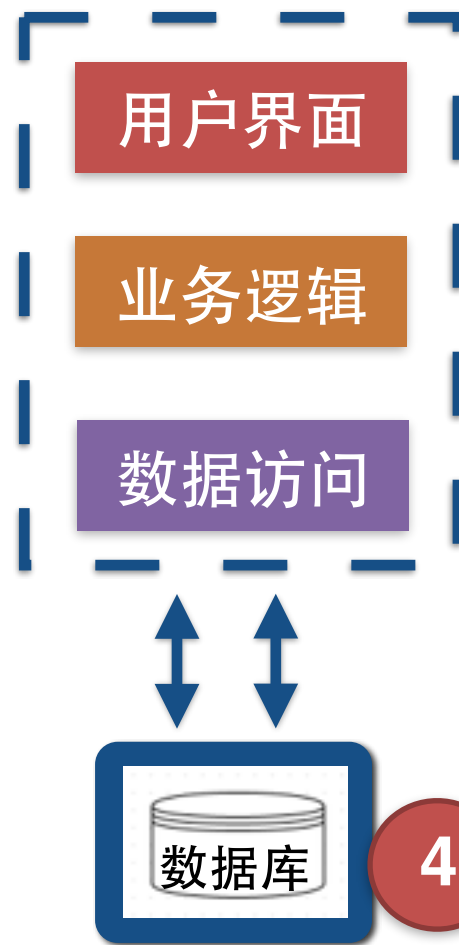
30%

50_{mins}

3_{Day}

改造策略

1 最小修改



4 数据同步

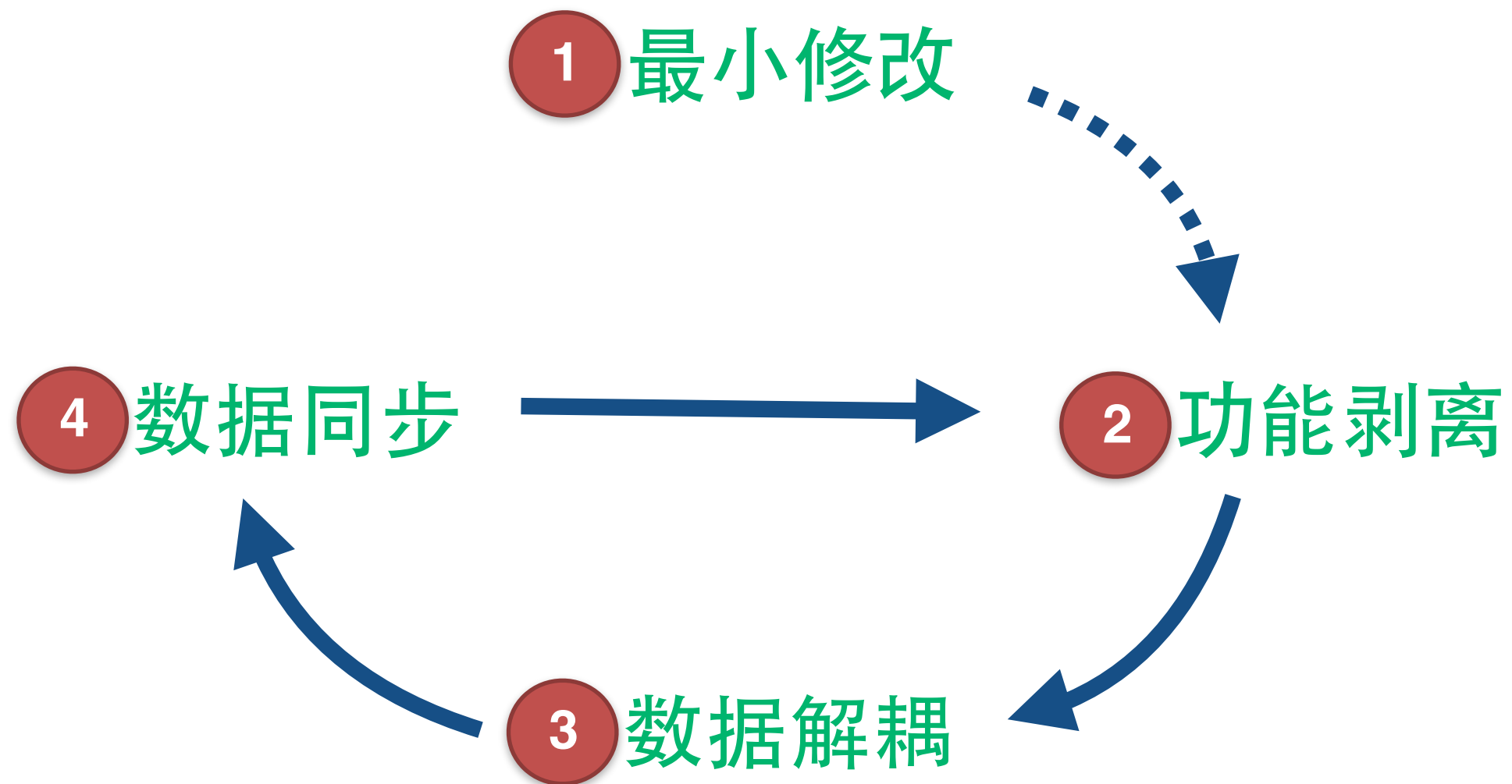


2 功能剥离

3 数据解耦



改造策略



每10万元合同

-71%



CRM IT团队成员

-30%



持续集成+构建

-64%

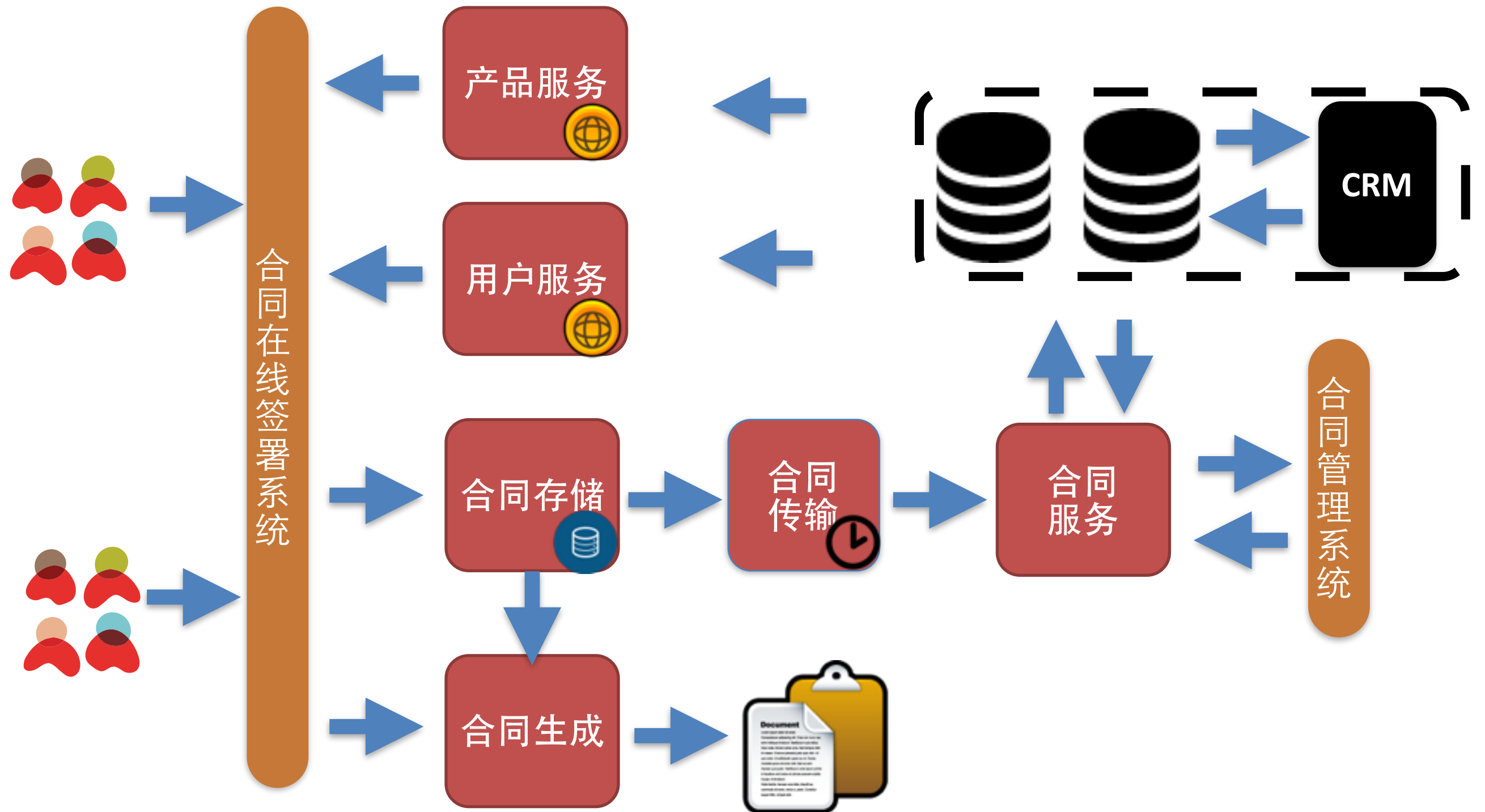


部署周期

-91%



目前的系统



技术栈与核心工具

产品服务



从现有的数据库中读取产品数据, 使用Redis进行缓存

客户服务



从现有的数据库中读取客户数据, 使用Redis进行缓存

合同存储



提供接口, 存储签约的合同. PostgreSQL(JSON type)

合同传输



使用Resque作为定时任务(Redis作为存储)

合同服务

提供接口获取/存储合同

合同生成

获取合同信息并生成PDF

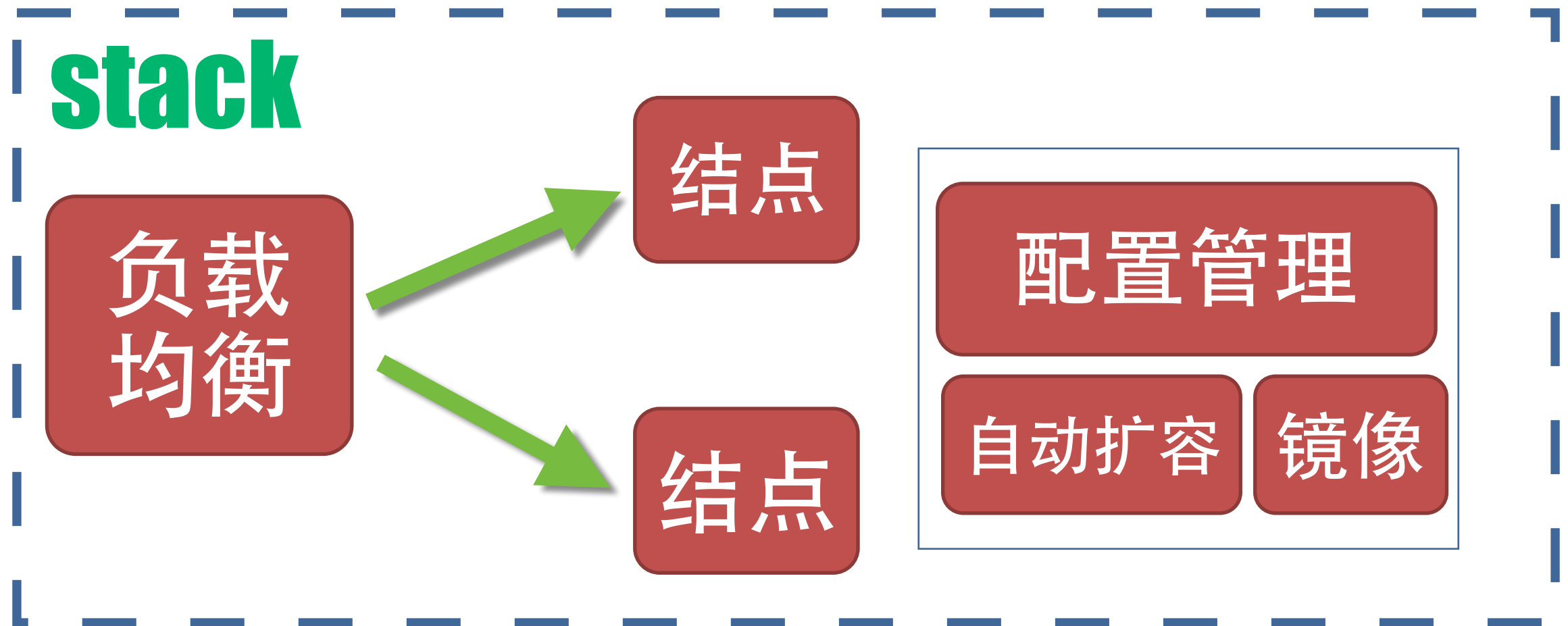
合同管理系统

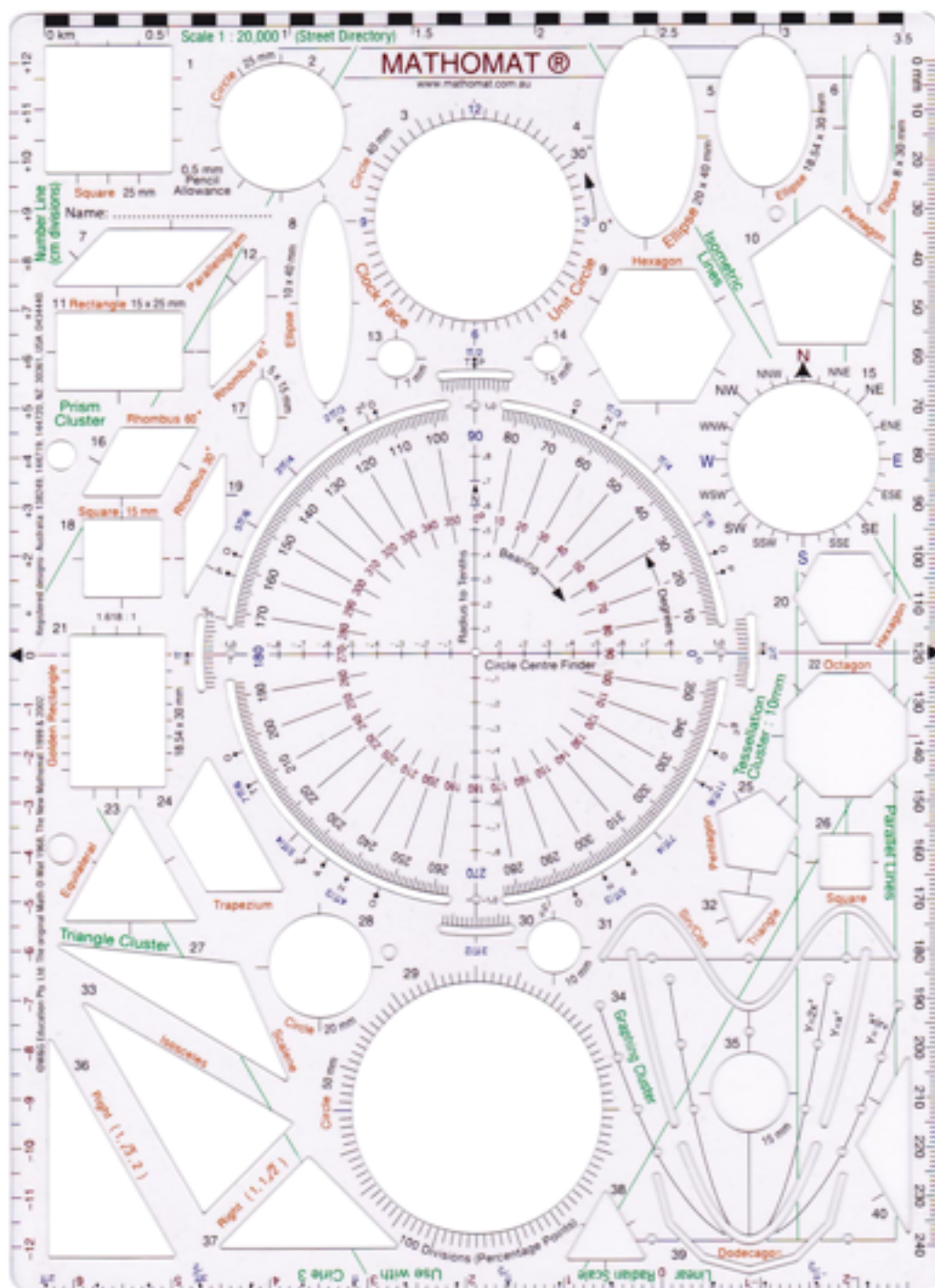
合同签署系统

使用AngularJS实现的瘦客户端



基础设施 - AWS



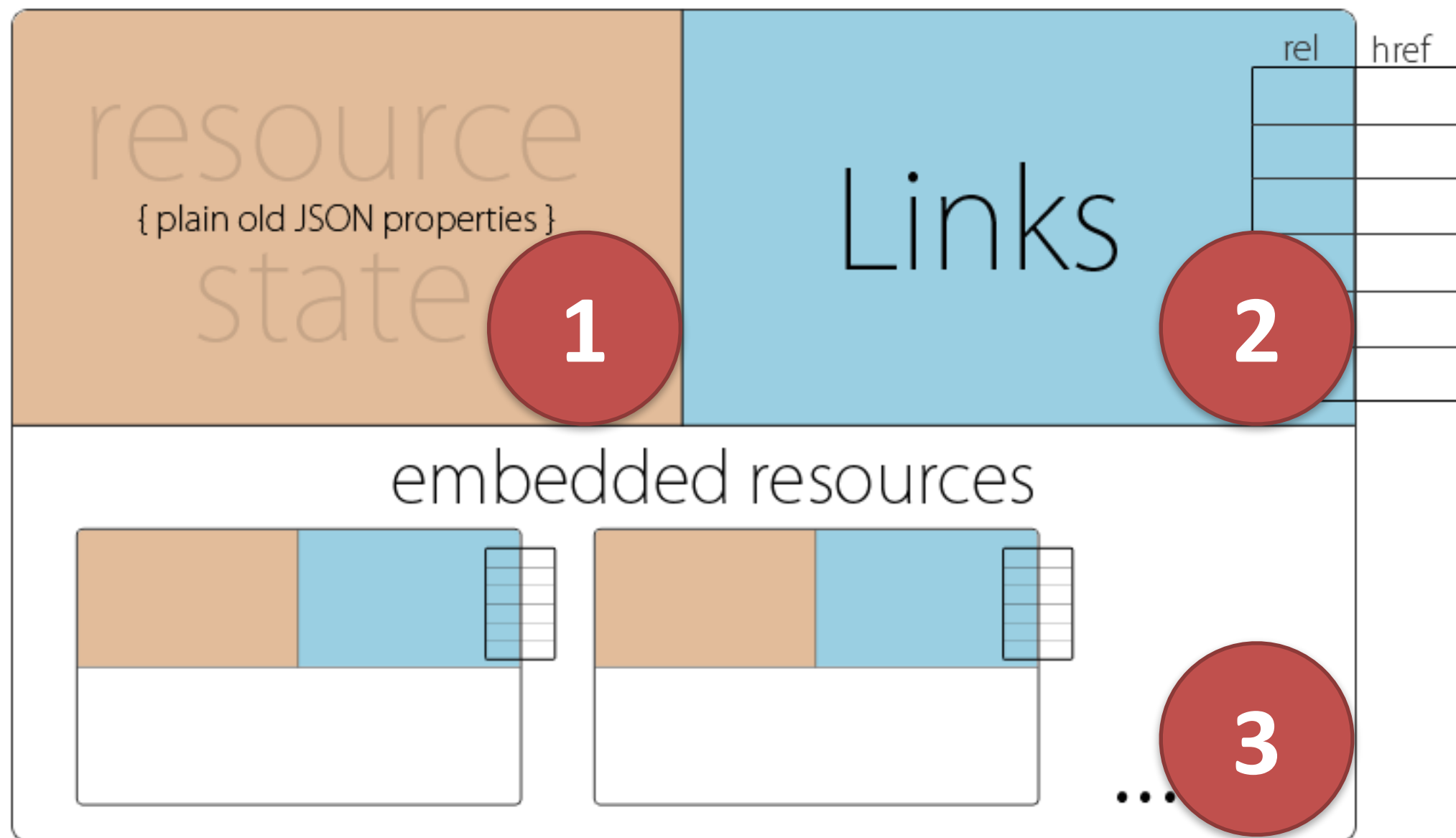


- 代码框架
- 部署脚本
- 容器配置
- 监控、告警配置
- 日志聚合配置
- 命令行接口

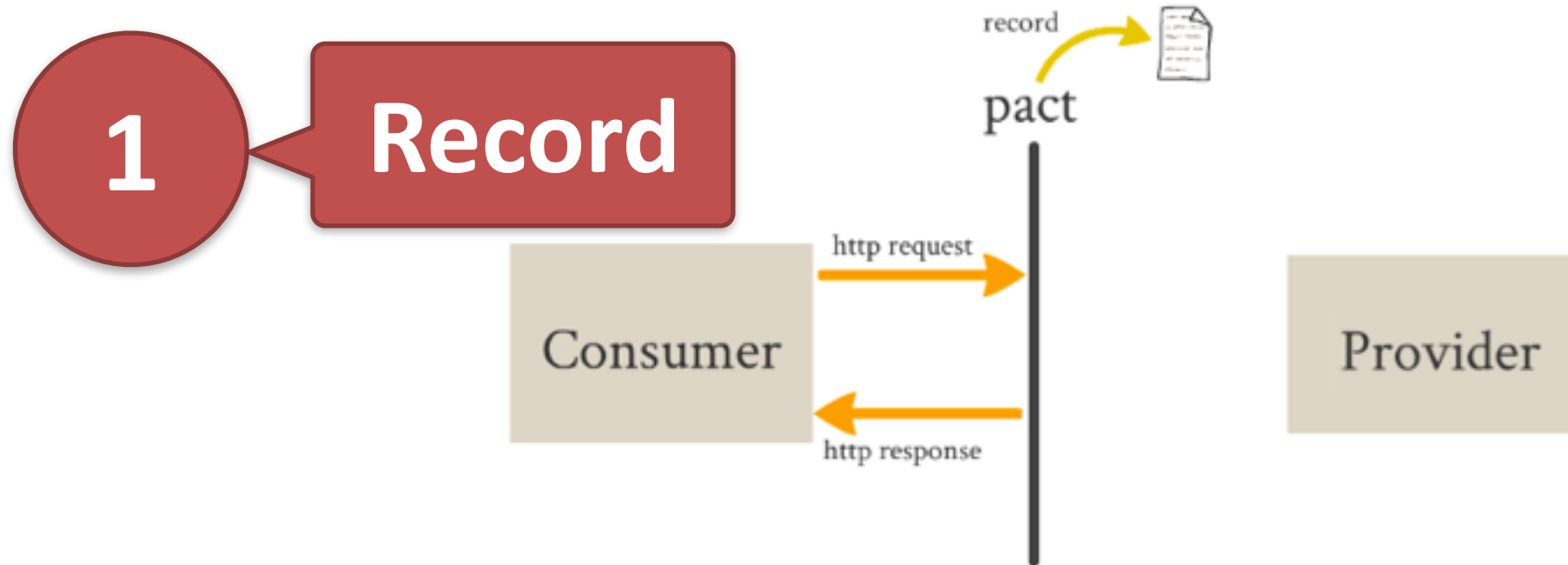
实践2 'Stencil' (微服务模板)

HYPERTEXT APPLICATION LANGUAGE

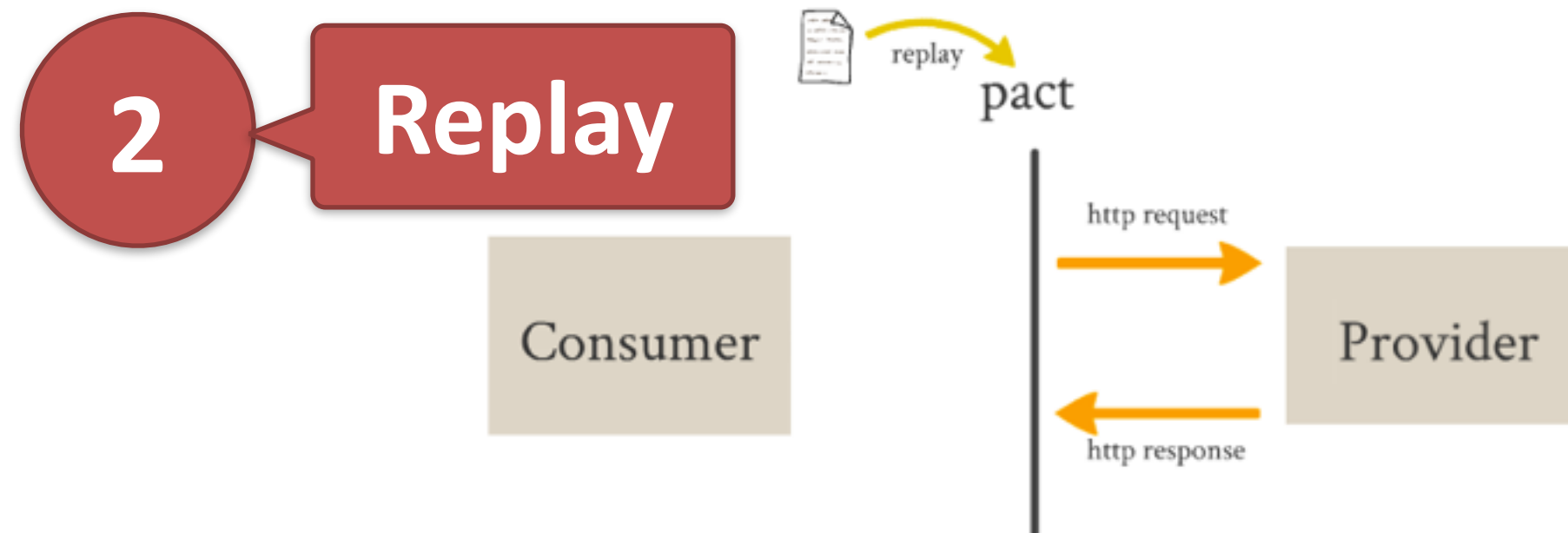
Resource



Step 1 - Define Consumer expectations



Step 2 - Verify expectations on Provider



实践4 - 消费者驱动契约测试(PACT)

实践5 - 基础设施自我管理-TMI

团队基础设施自我管理，更高效的开发和维护服务。



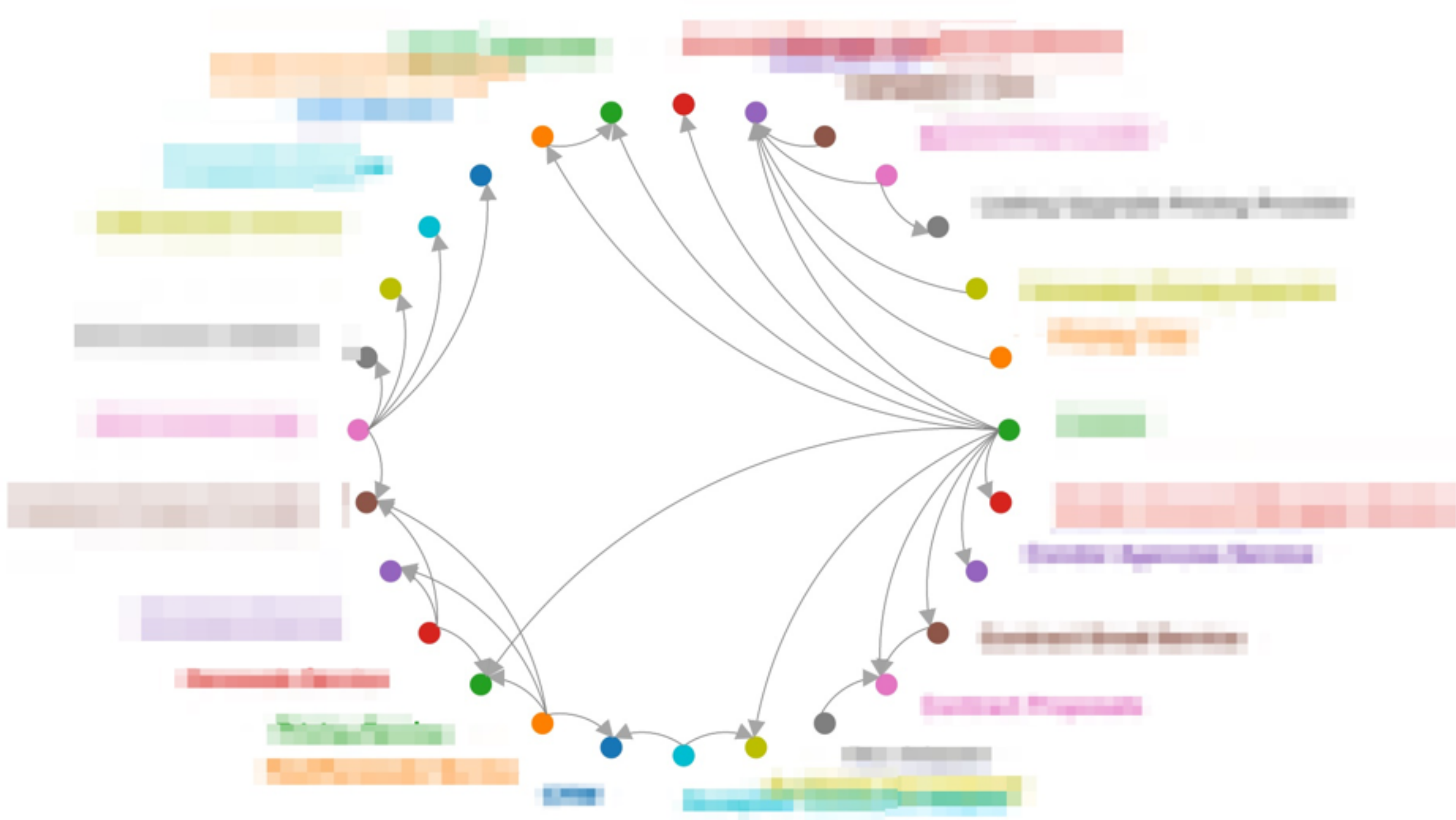
基础设施自我管理-TMI

- 持续集成 -> 打包
 - 提交请求 -> 审批
 - Ops部署
- 基础设施搭建
 - 自动化配置
 - 自动化部署

-> deploy *[APP]* *[VERSION]*



实践6 - “藏宝图”



实践6 - “藏宝图”

独立的环境

Isolated Environment

演进的开发模板

Stencil

轻量级的通信协议

HAL

基于消费者的契约测试

PACT

基础设施自管理

TMI

微服务藏宝图

MAP

总结