



Gdevops

全球敏捷运维峰会



解密百度持续交付方法与实践

演讲人：张乐

自我介绍



张乐

- 百度工程效率部 – 资深敏捷教练、架构师
- 百度内先进软件工程方法和生产力的践行者、布道者
- 十三年软件行业工作经验
 - 敏捷、精益
 - 项目管理
 - 持续交付、DevOps
- 『百度方法+』持续交付专题负责人



互联网时代对软件交付的诉求

我们处在一个VUCA的时代

软件交付面临易变性、不确定性、复杂性、模糊性

- 专注、极致、口碑、快
- 快速迭代
- 快速转型
- 只有第一，没有第二



- 用户体验至上
- 细节决定成败
- 缺陷->差评->卸载
- 服务故障->百倍赔付

你做的怎么样？

在你的公司里，仅涉及一行代码的改动需要花多长时间才能部署上线？

你是以一种可重复且可靠的方式来做这件事的吗？

引用：《Implementing Lean Software Development》,Page. 59



软件交付的困境



如何摆脱这些困境

我们需要一种方法和技术：

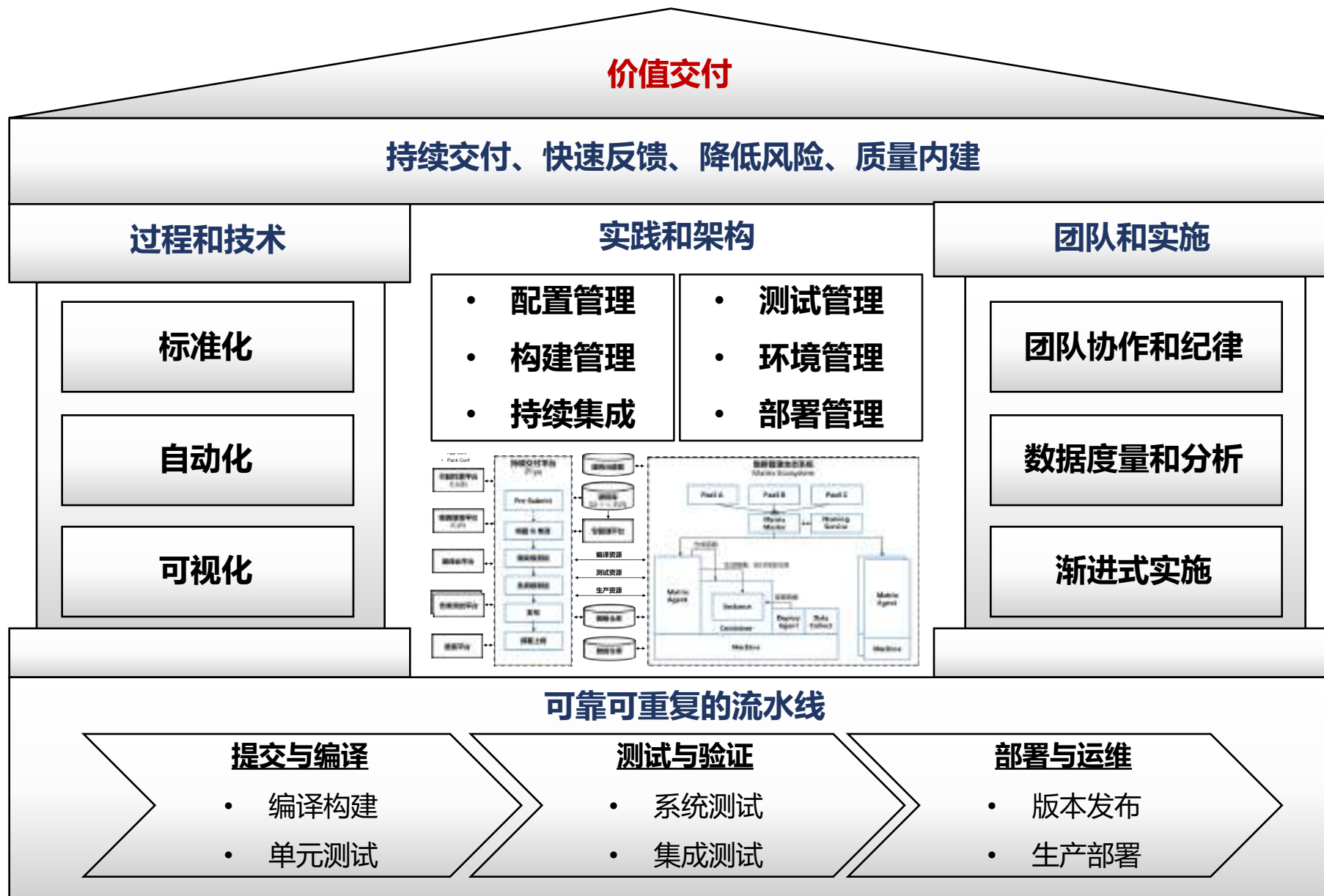
- 能够有效缩短提交代码到正式部署上线时间，降低风险
- 能够自动的、快速的提供反馈，以便及时发现和修复缺陷
- 能够让软件在整个生命周期处于可部署的状态
- 能够按一下按钮，就能将任意版本、按需部署在任意环境中
- 能够让整个交付过程变成一种可靠、可预期、可视化的过程

我们需要『持续交付』

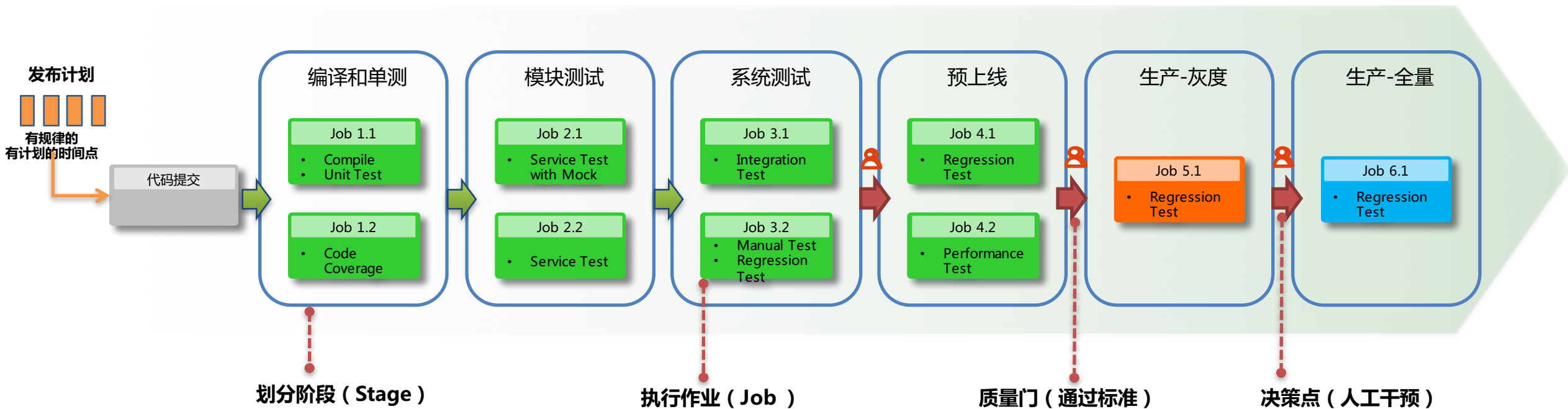
Our highest priority is to satisfy the customer
through early and continuous delivery
of valuable software

1st principle of the Agile Manifesto

百度持续交付实践框架



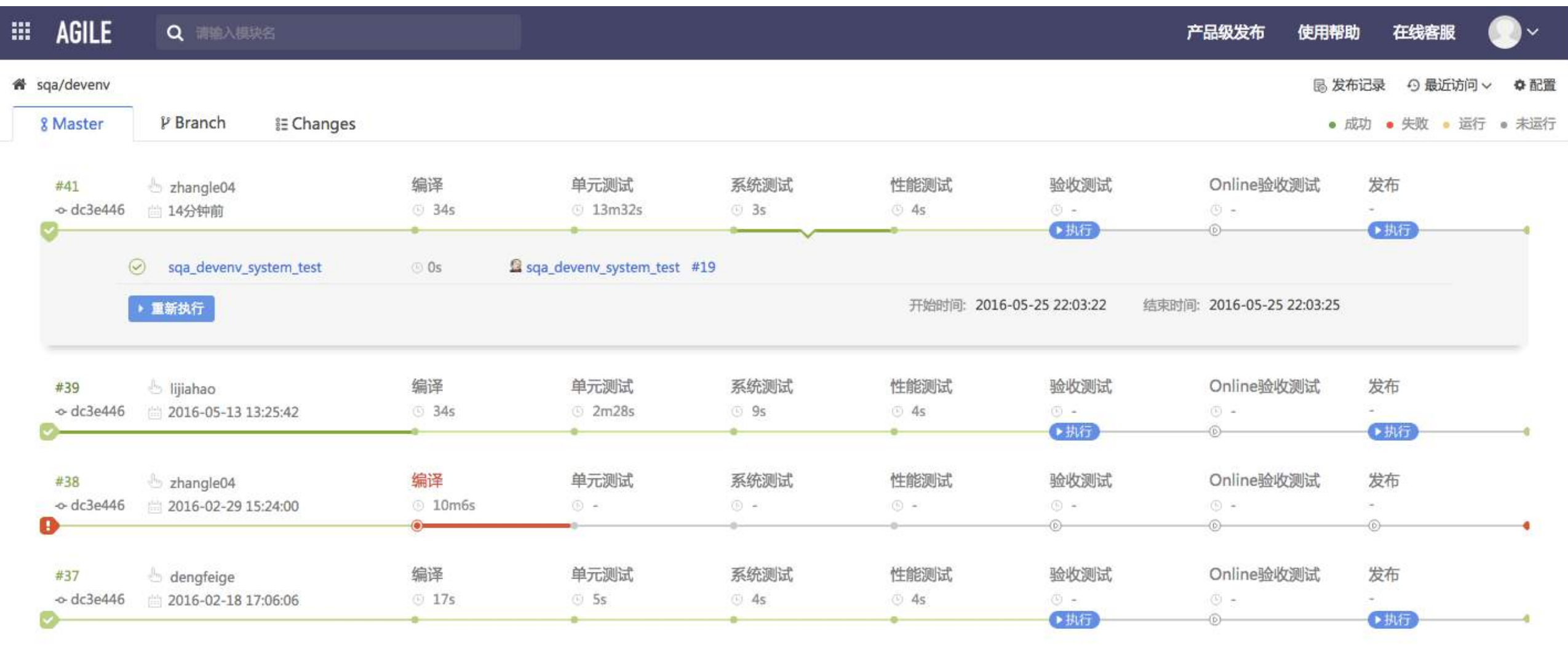
可靠可重复的流水线



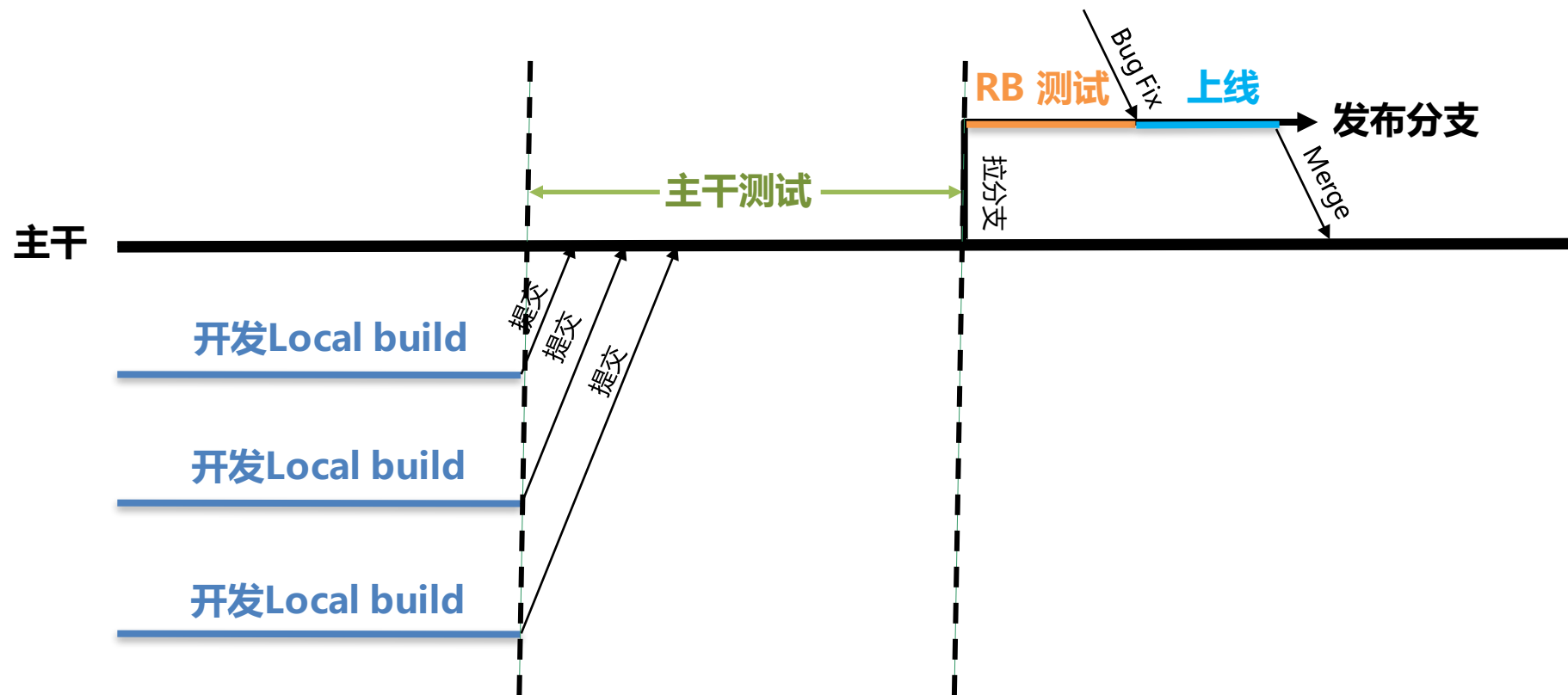
精益思想与七种浪费



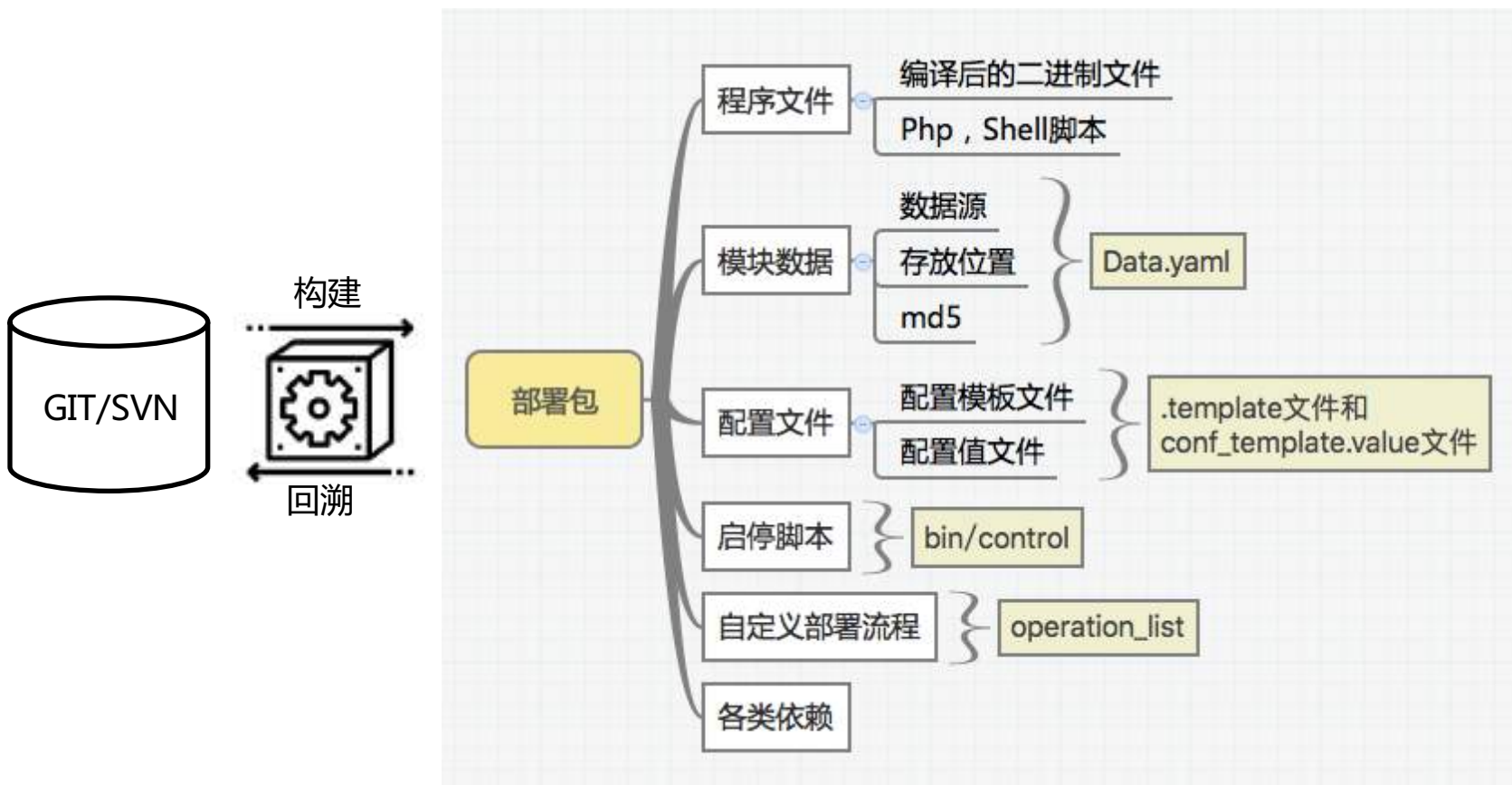
交付流水线实现



配置管理 - 分支策略



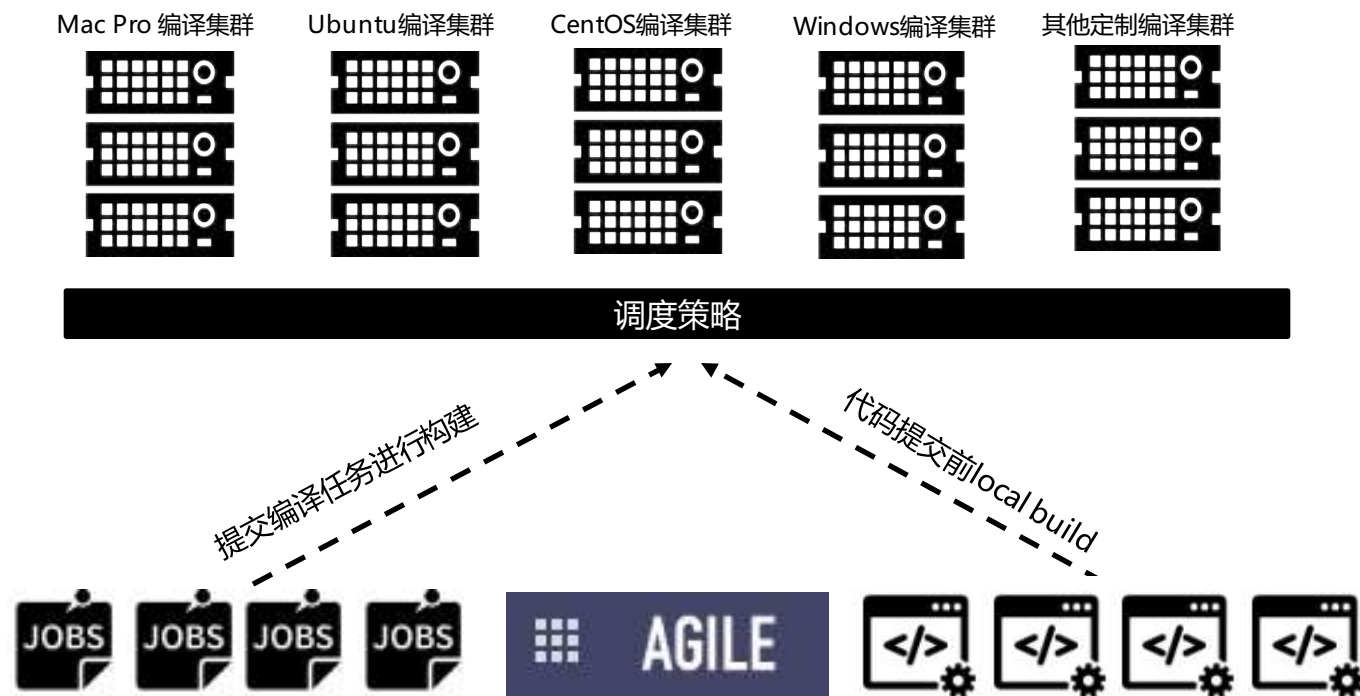
配置管理 - 部署包规范



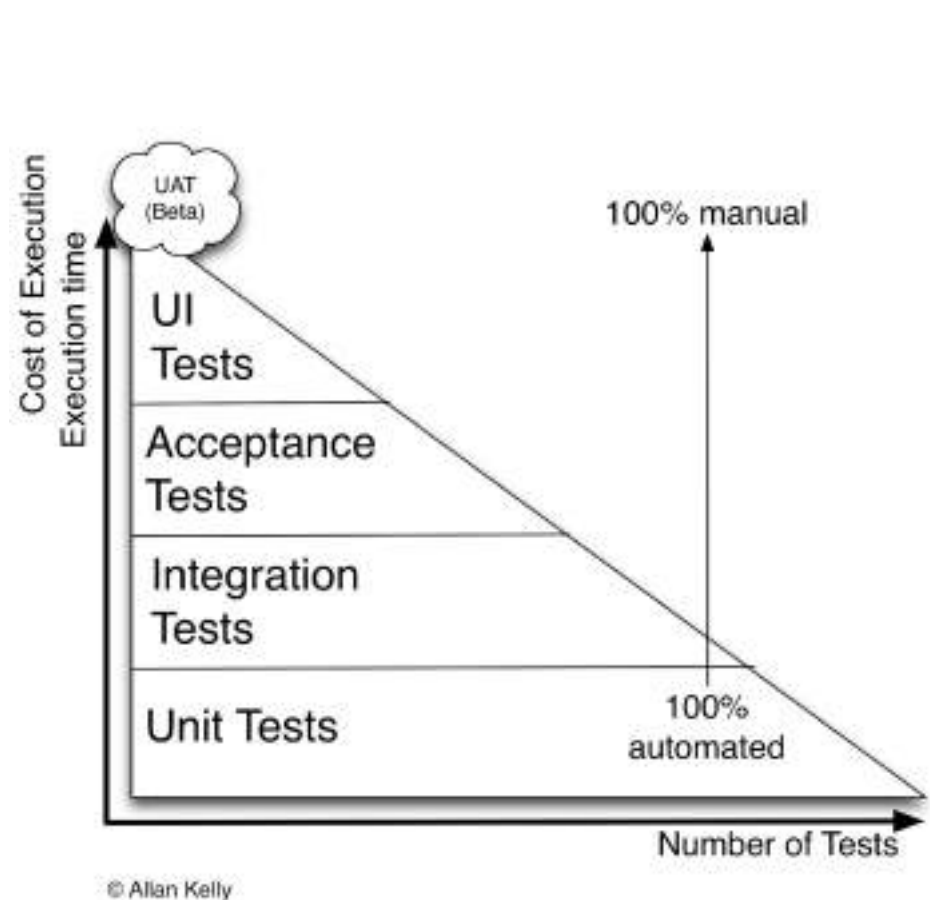
- **全量部署包**
- **提供稳定的控制接口**
- **差异化部署使用配置派生**
- **自定义上线流程**

构建管理

- 标准化编译配置文件，代码提交自动编译
- 集群加速，提高并发场景吞吐量
- 机器针对编译任务硬件优化
- 模块间依赖采用二进制构建产物
- 使用增量编译
- 编译结果推送至『制品仓库』管理



测试管理

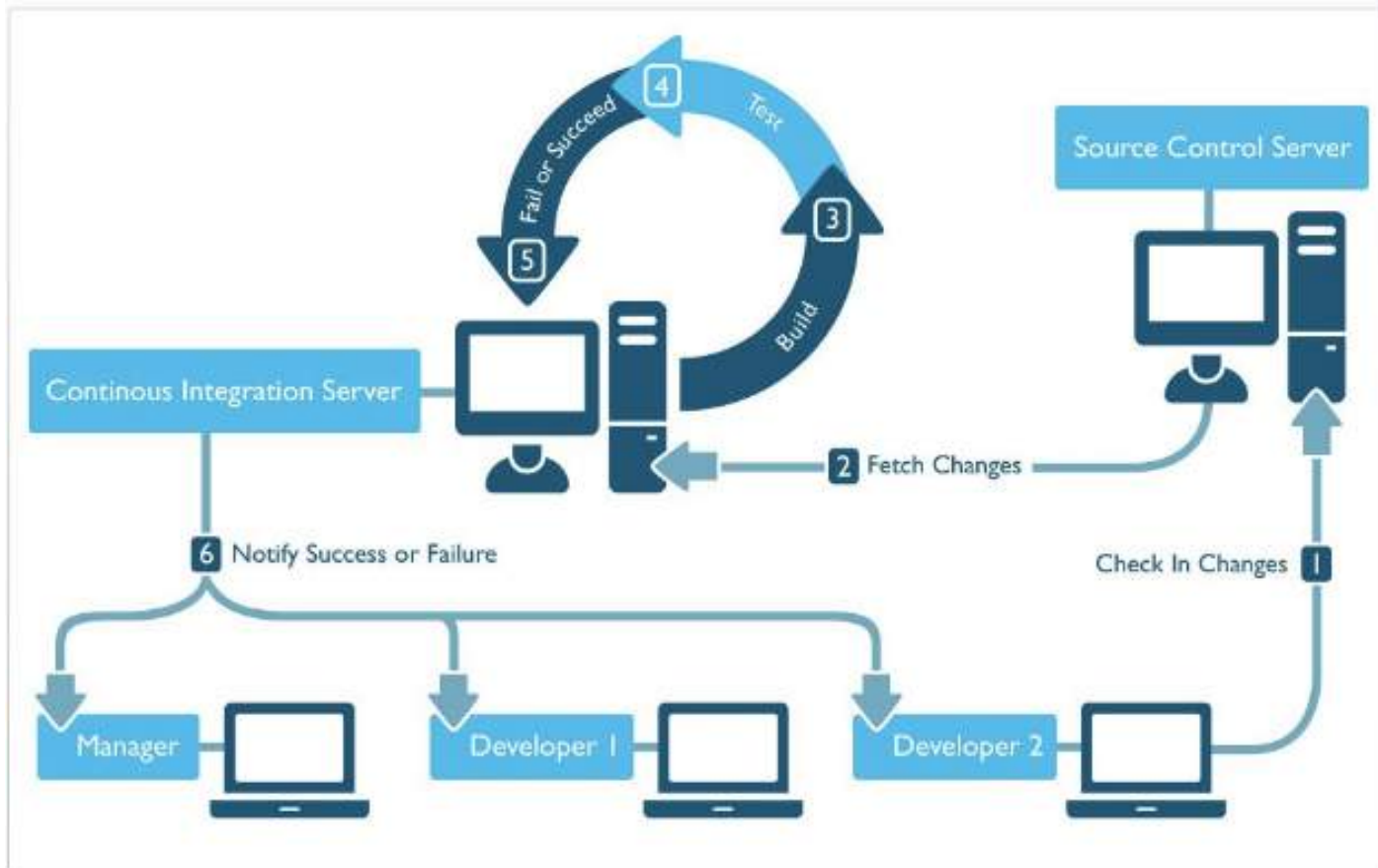


建立分级测试体系，从多个层次和多个验证角度实现质量防护网

持续集成

坚持的原则：

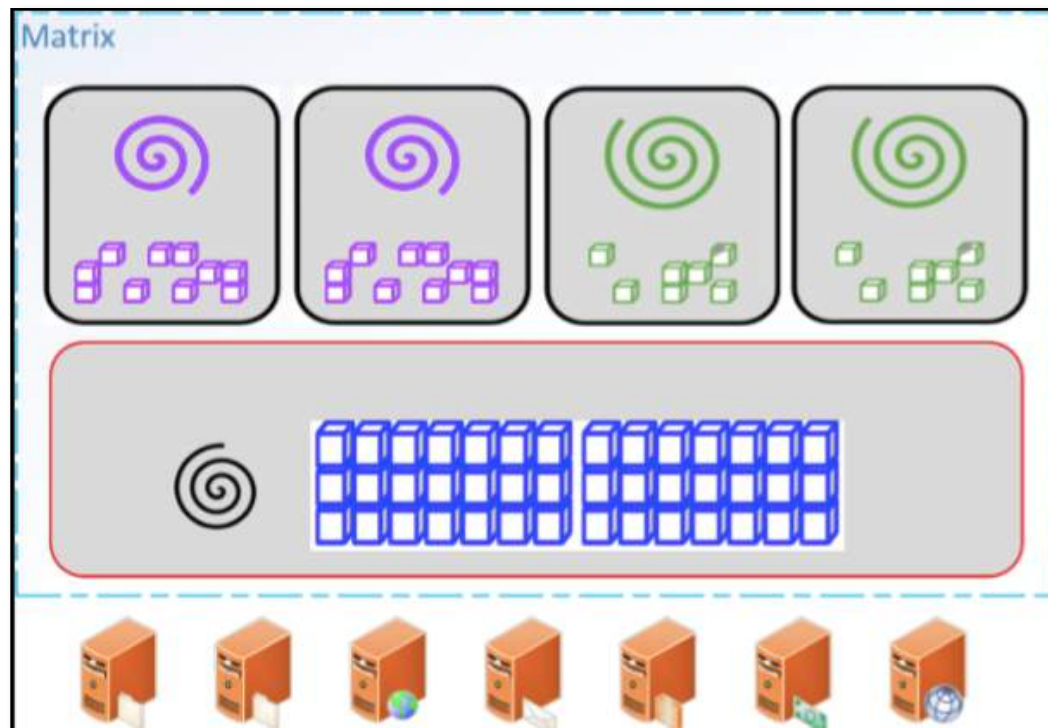
- 小改动，逐步构建
- 每人每天提交代码
- 在主干上持续集成
- 至少每天进行集成
- 使用持续集成工具
- 自动化构建和测试
- 分级测试快速反馈
- 红灯需要立即修复



环境管理 – 容器化

使用Matrix作为公司数据中心的集群管理系统，为所有的产品线提供规范的底层架构

- 资源管理：资源调度，提供虚拟化容器资源
- 服务管理：服务部署，状态监测，故障迁移，动态扩容、缩容



环境管理 – 集群操作系统

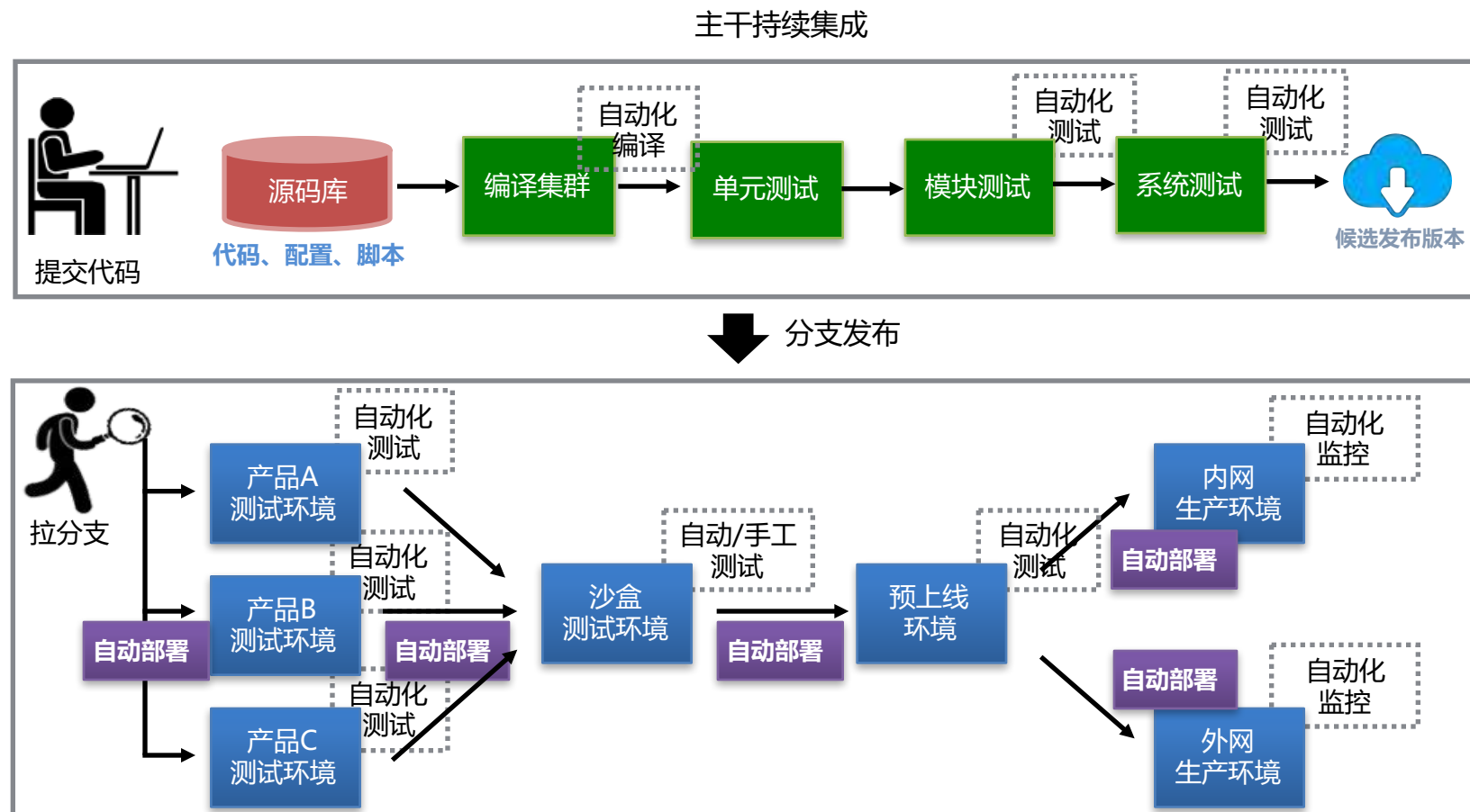
- 资源隔离与混部，公司内所有机器在统一资源管理下实现高效、充分利用
- 与产品线合作构建服务托管平台，提升服务的自动化运维效率
- 部署机器占公司机器总数50%以上



部署管理

部署管理实践：

- 部署包来自制品仓库
- 各环境相同部署方式
- 各环境相同部署脚本
- 运维人员参与部署过程创建
- 部署后执行冒烟测试
- 只有流水线才能修改生产环境
- 生产环境部署方式：灰度发布



部署管理 - 灰度发布

任务552088详情

[取消](#) [暂停](#) [继续](#) [清空失败计数](#) [重做失败实例](#) [刷新](#)

服务名字: demo.s

集群列表: yf01 cq

操作实例: 所有实例

统计数据: 1/0/0/0/3 (成功/失败/忽略/执行中/未执行数目)

已完成进度: 1/4 25%

任务id	服务描述文件id	部署策略			开始时间	完成时间	任务状态	正在处理的分组
		并发度	暂停点	容错策略				
552088		1	无暂停点	1	2016-01-27 11:21:21	无	DOING	无

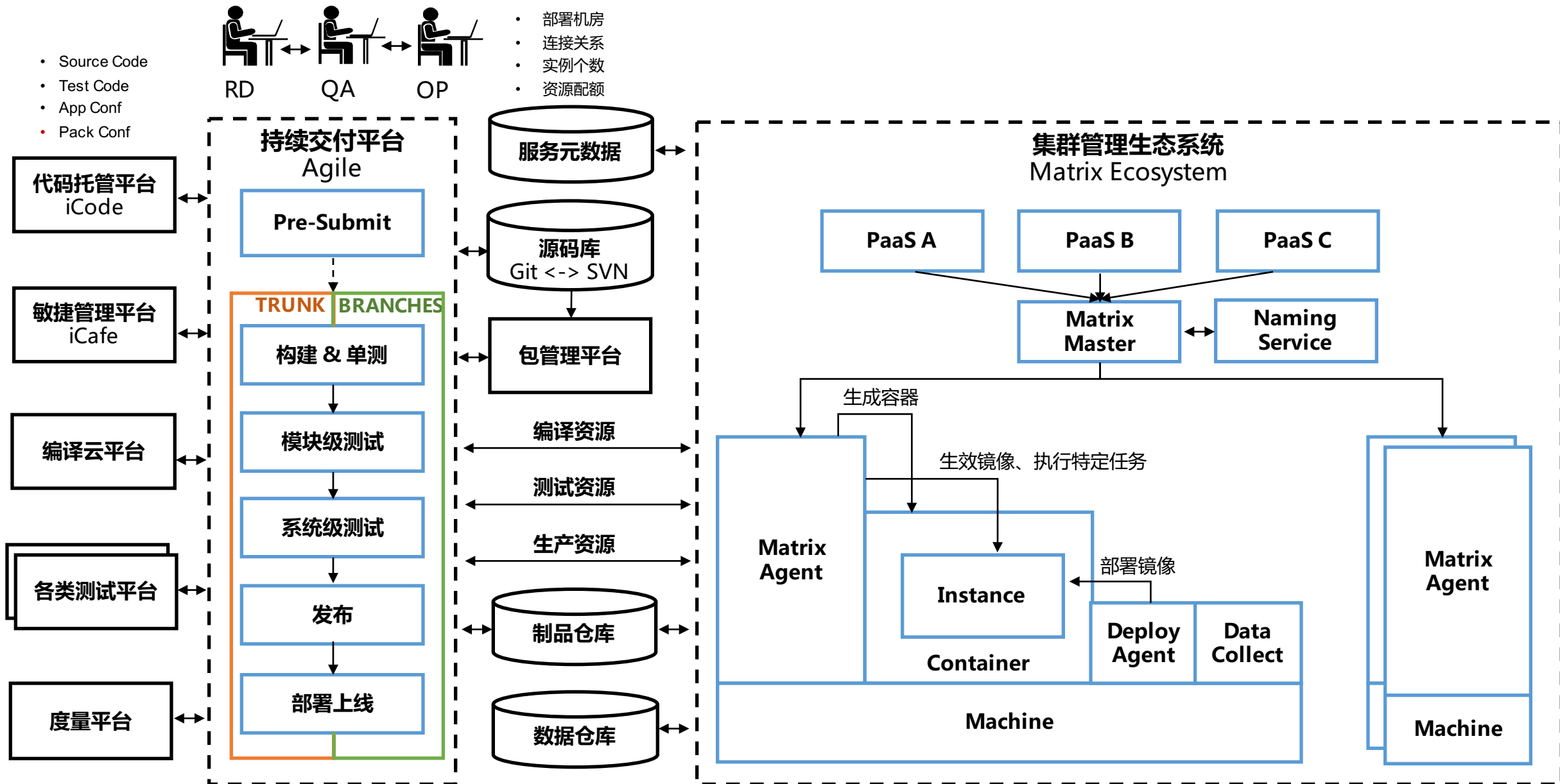
分组名称	部署策略			开始时间	完成时间	分组状态	操作
	并发度	暂停点	容错策略				
group1	1	1	1	2016-01-27 11:21:21	无	PENDING	暂停 继续 清空失败计数

统计数据: 1/0/0/0/3 (成功/失败/忽略/执行中/未执行数目)

[批量安装](#)

失败实例	实例id	集群	实例操作	失败被忽略	操作时间		操作信息		Host信息	操作
					开始时间	完成时间	状态	操作详细信息		
<input type="checkbox"/>	0.demo.s	cq	ADD	否	2016-01-27 11:21:21	无	SUCCESS	INSTANCE_OK 详细错误信息	10.48.65.55 cq01-matrix-storage0309.cq01.baidu.com	重做
<input type="checkbox"/>	1.demo.s	cq	ADD	否	无	无	CREATED	INSTANCE_OK 详细错误信息		重做
<input type="checkbox"/>	2.demo.s	yf01	ADD	否	无	无	CREATED	INSTANCE_OK 详细错误信息		重做

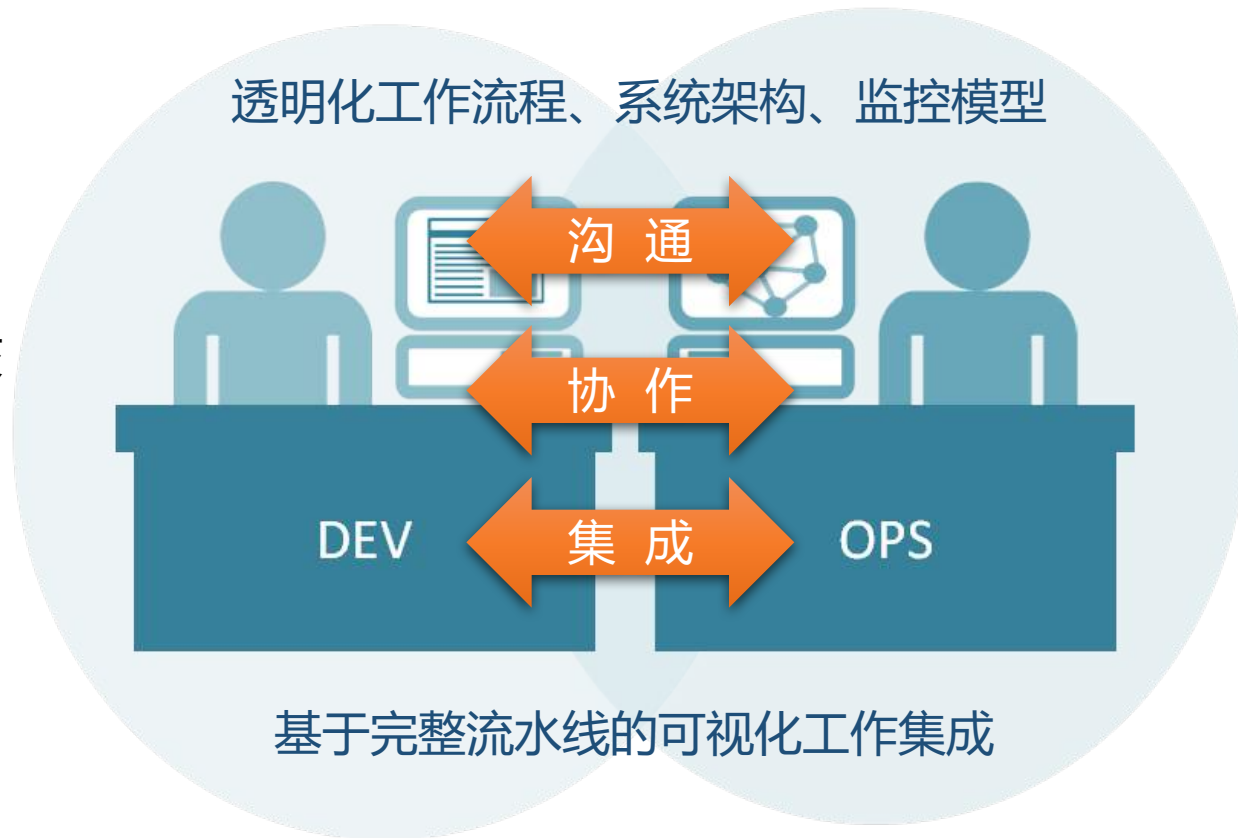
持续交付技术栈



团队协作和纪律

DEV

- 避免长周期分支
- 每日代码提交
- 代码和测试同步提交
- Local Build
- Check In Dance
- 红灯需要立即修复



OPS

- 标准化部署过程
- 自动化部署脚本
- 服务能力转换为平台能力
- 减少对人员和经验的依赖
- 上线执行者转变为审核者
- 面向事务转变为面向价值

数据度量和分析

1 流水线构建总览、分团队总览



3 核心数据汇总，环比变化趋势

构建级别	L1					
产品名称	CI数	CI数环比	构建数	构建数环比	构建成功率	构建成功率
Ol	10	+150%	15	+650%	40%	-60%
Cl	8	-64.81%	16	+250%	100%	+0%
M B	30	+143.45%	62	+147.92%	64.58%	-20.83%
Ni ndy	31	+190%	23	+144.64%	100%	+0%
IA	3	-50%	16	+100%	100%	+0%
Q	2	-33.33%	2	+100%	100%	+0%
Fil e	36	+254.17%	27	+87.5%	99.02%	+51.26%
Bl	17	+41.67%	23	+76.92%	100%	+18.18%

2 数据筛选和下钻，各团队数据

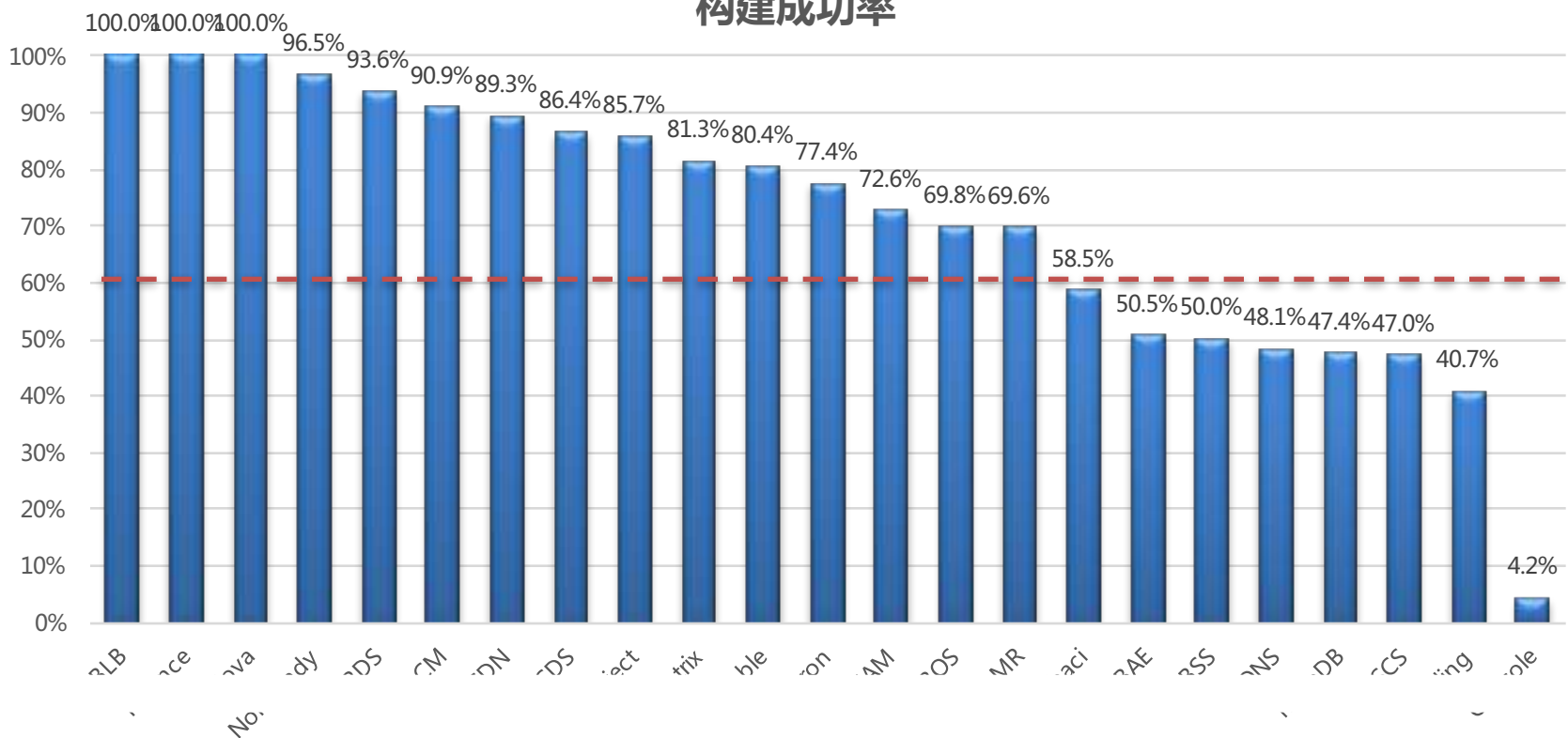


4 自动分析和异常报表推送邮件

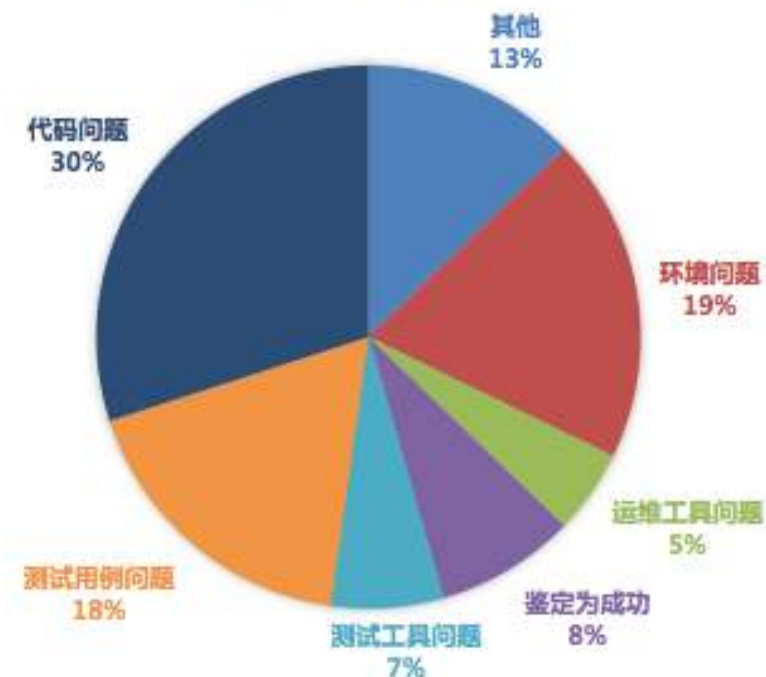


从度量中找到问题，从度量中看到进步

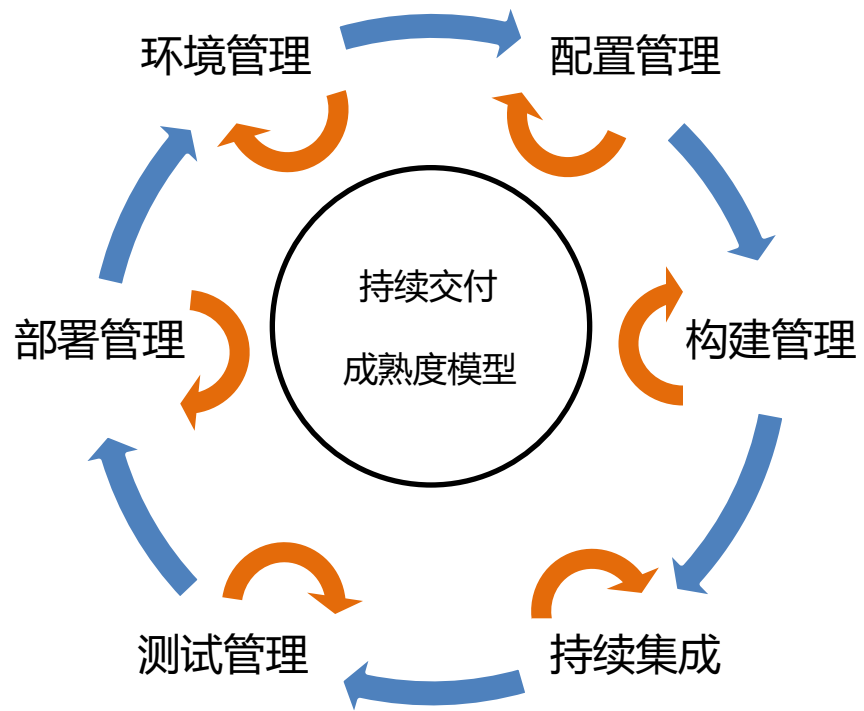
构建成功率



构建失败原因分布

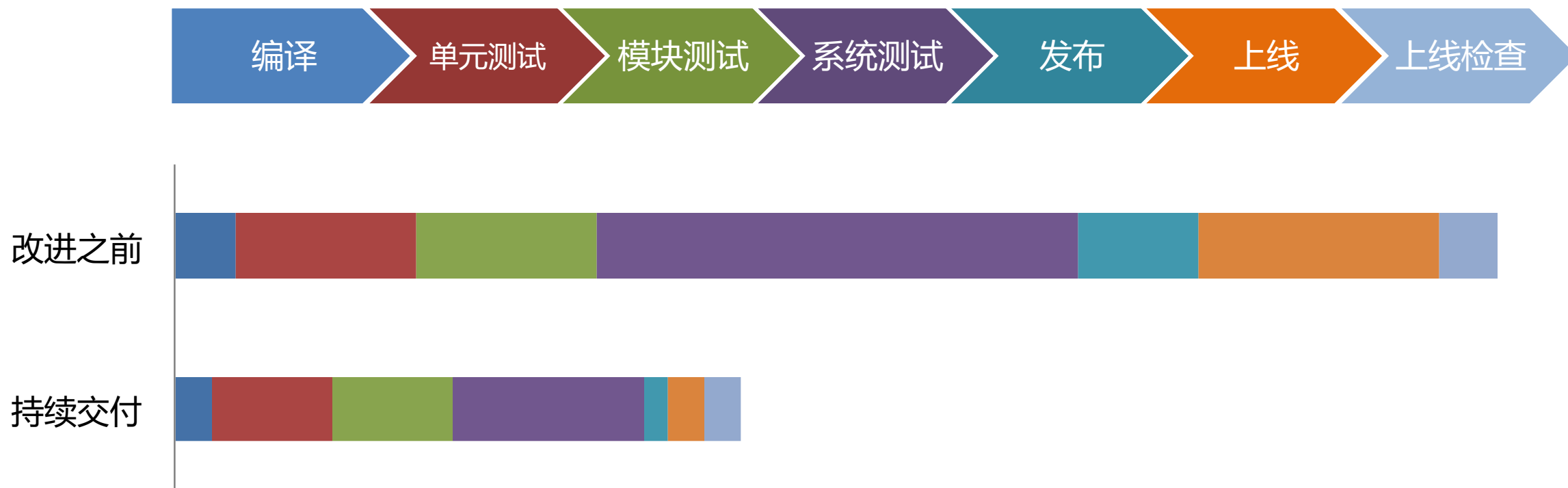


渐进式实施



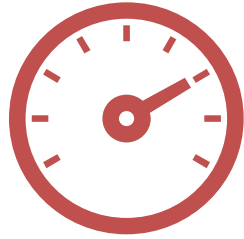
Ad-Hoc	Repeatable	Consistent	Optimised	Leading
Ad-Hoc	Repeatable	Consistent	Optimised	Leading
Ad-Hoc	Repeatable	Consistent	Optimised	Leading
Ad-Hoc	Repeatable	Consistent	Optimised	Leading
Ad-Hoc	Repeatable	Consistent	Optimised	Leading
Ad-Hoc	Repeatable	Consistent	Optimised	Leading
Ad-Hoc	Repeatable	Consistent	Optimised	Leading

持续交付效果案例



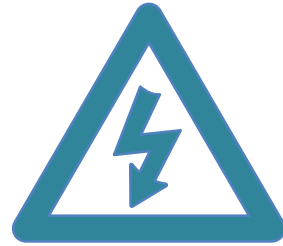
- 编译、开发、测试、部署上线全面加速，整体交付周期明显缩短
- 各角色可以基于统一的交付流水线紧密协作，产品交付过程可视化、可控制
- 每日多次发布能力、故障快速回滚的能力

持续交付的价值



快速交付

从需求到交付，
端到端周期缩短



保证质量

效率提升的同时，
保证质量稳定



降低风险

稳定节奏的交付，
增强可预测性

Q & A

