

DevOps 实战：百度持续交付体系与最佳实践大解密

摘自：<http://blog.csdn.net/xuguokun1986/article/details/54911281>

“互联网+”时代，软件产品要想满足快速增长的用户需求，高效、快速的迭代转型必不可少，面对时刻发生改变的互联网及业务模式需求，搭建高效的交付流水线更是势在必行。那么，如何构建一套能快速交付、保质又少风险的持续交付系统呢？

在 Gdevops 全球敏捷运维峰会北京站的讲台上，百度资深敏捷教练张乐便以持续交付为题，给现场带来了《解密百度持续交付方法与实践》的精彩演讲，独家分享百度在解决这方面问题和挑战时的最佳实践经验。



（点击[“这里”](#)听张乐演讲完整录音）

常言说，“工欲善其事，必先利其器”。百度持续交付体系如此高效的秘诀在于他们构建了一套符合业务发展的持续交付系统：通过建立一套持续交付实践框架和一条可靠可重复的流水线，辅以 7 种消除浪费与精益思想，在配置管理、构建管理、测试管理、持续集成、环境管理和部署管理六大核心实践的合力下，让整个交付变成一种标准化、自动化、可视化的过程。

如果说把敏捷作为一种精益思想是在需求研发阶段的一个实践，那 DevOps 就是精益在发布和运维阶段的一个实现。通过 DevOps 落地，百度让开发和运维紧密合作，形成合力，共同促进了价值的持续交付。



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？

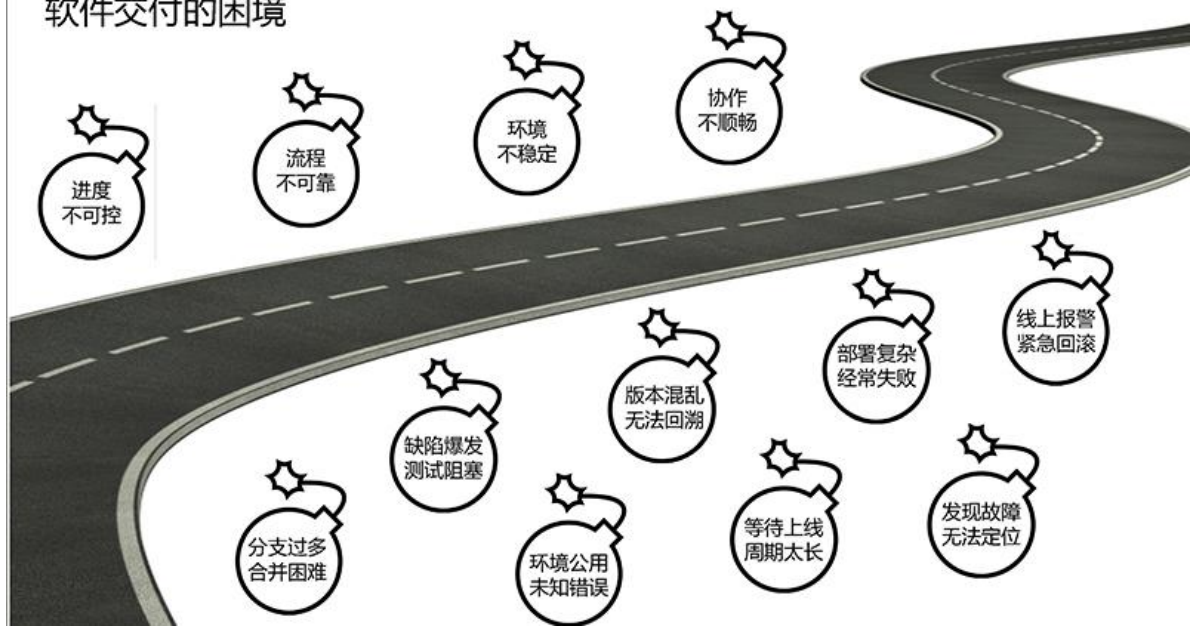
在你的公司里，仅涉及一行代码的改动需要花多长时间才能部署上线？

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

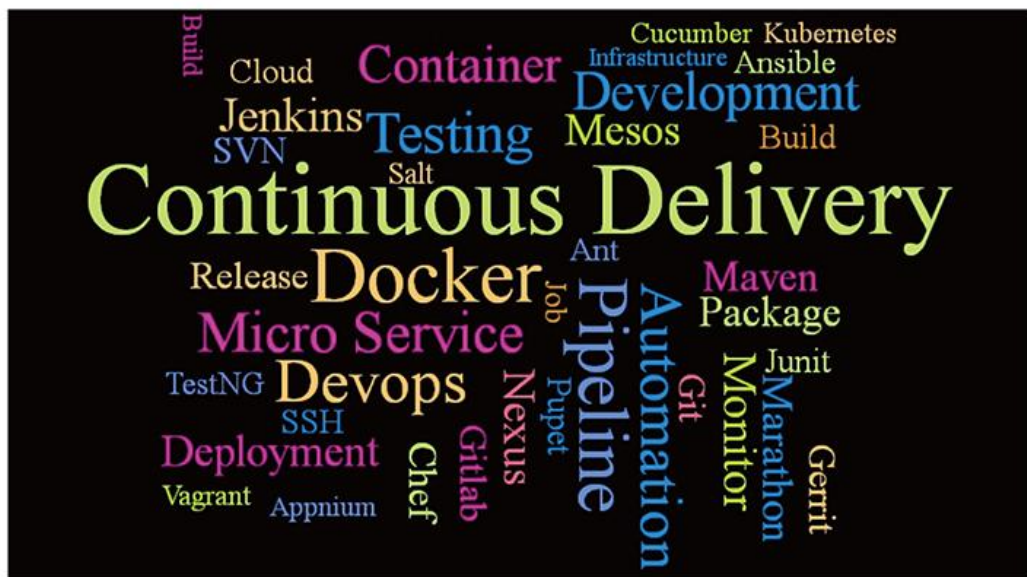
我们需要一种方法和技术：

- 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

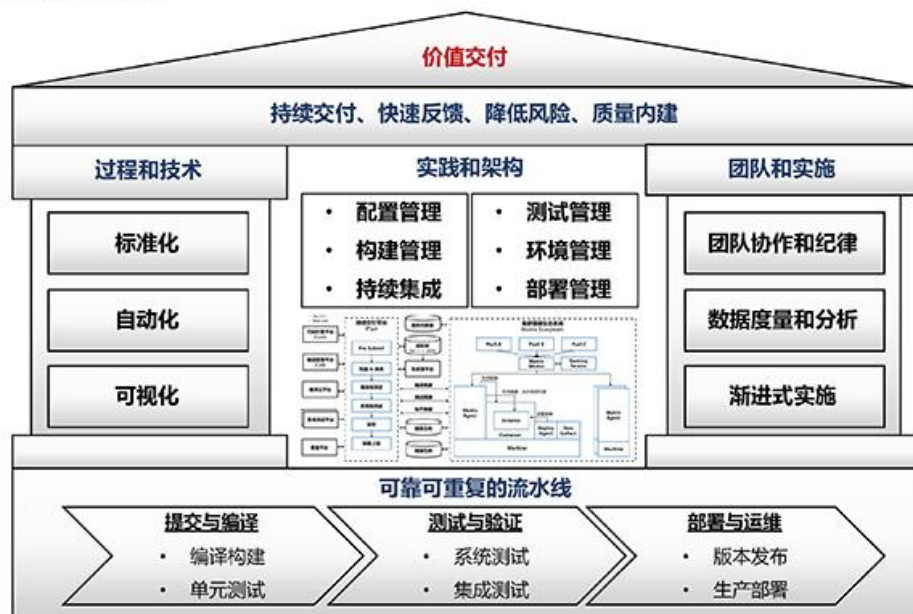
我们需要『持续交付』



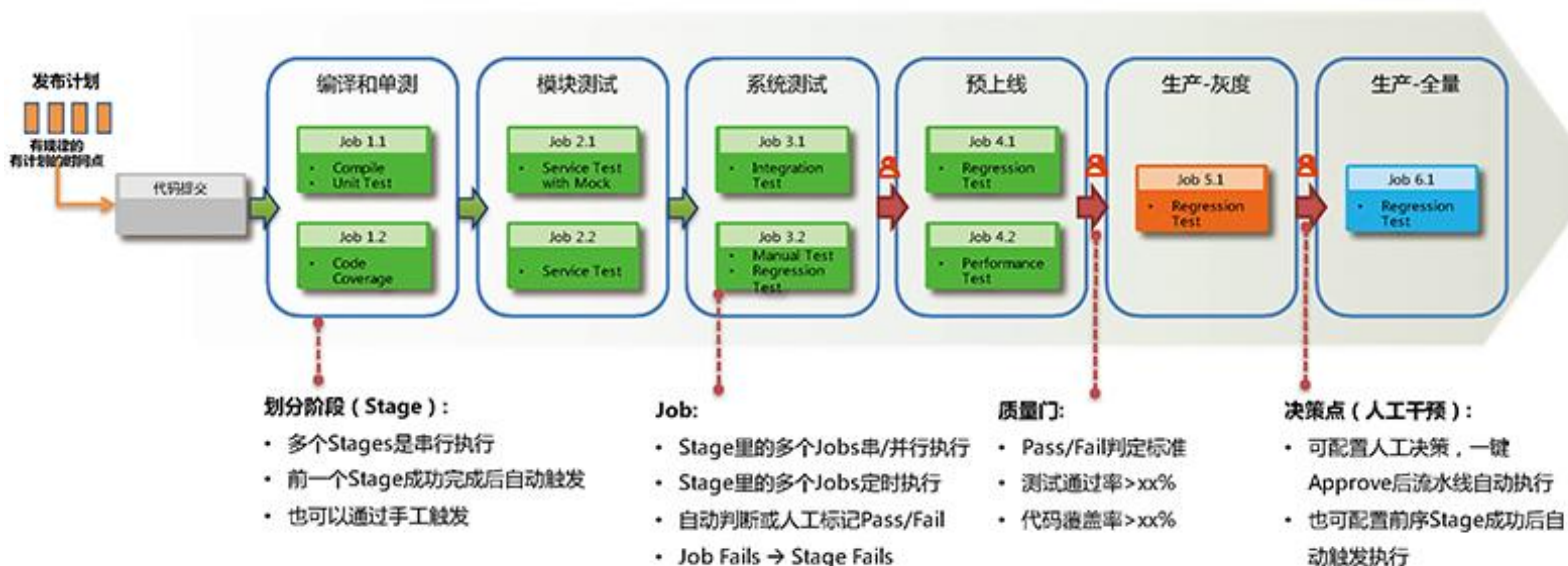
如何实现持续交付？



持续交付实践框架



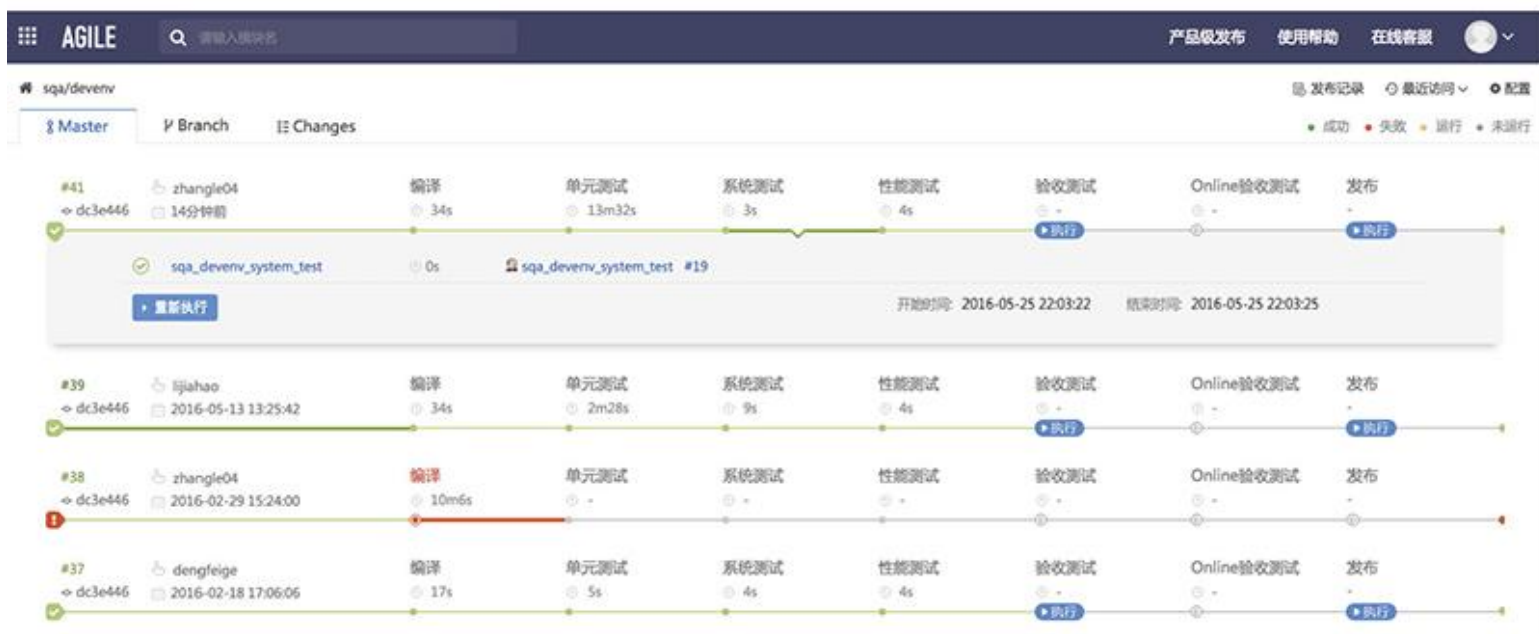
可靠可重复的流水线



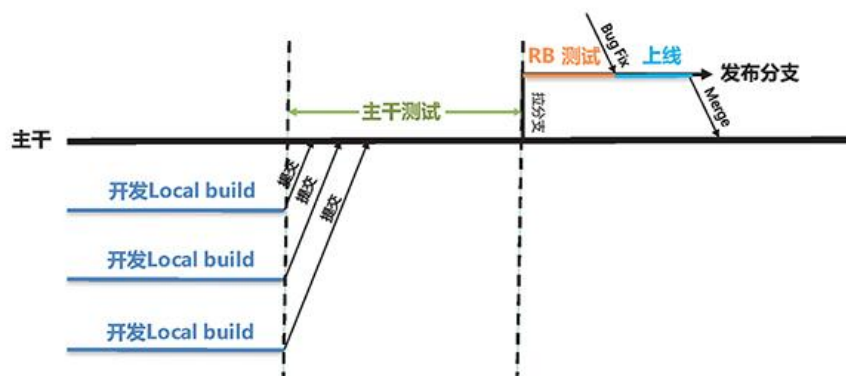
精益思想与七种浪费



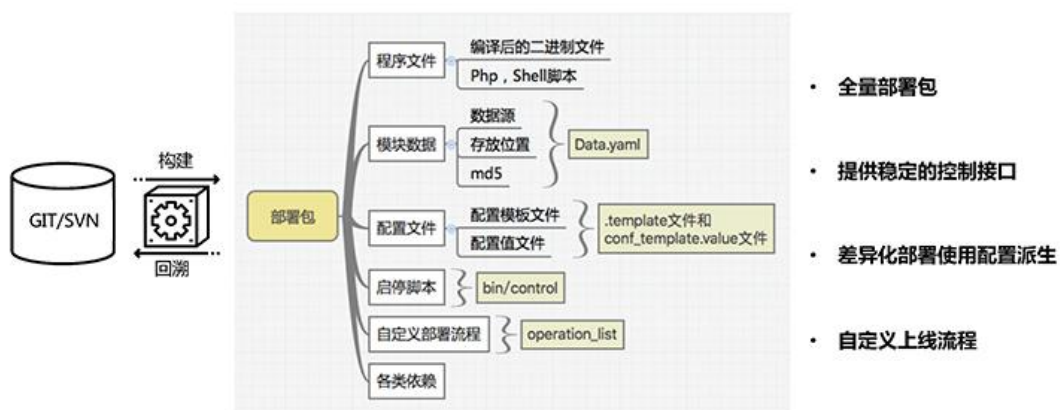
交付流水线实现



配置管理 – 分支策略

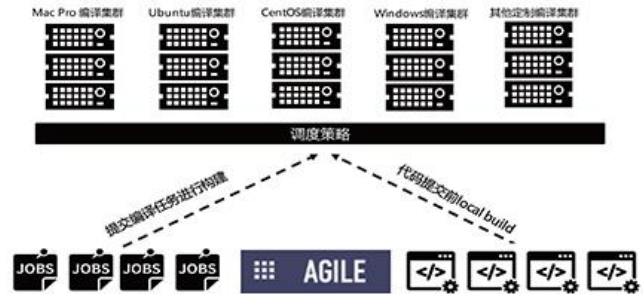


配置管理 – 部署包规范

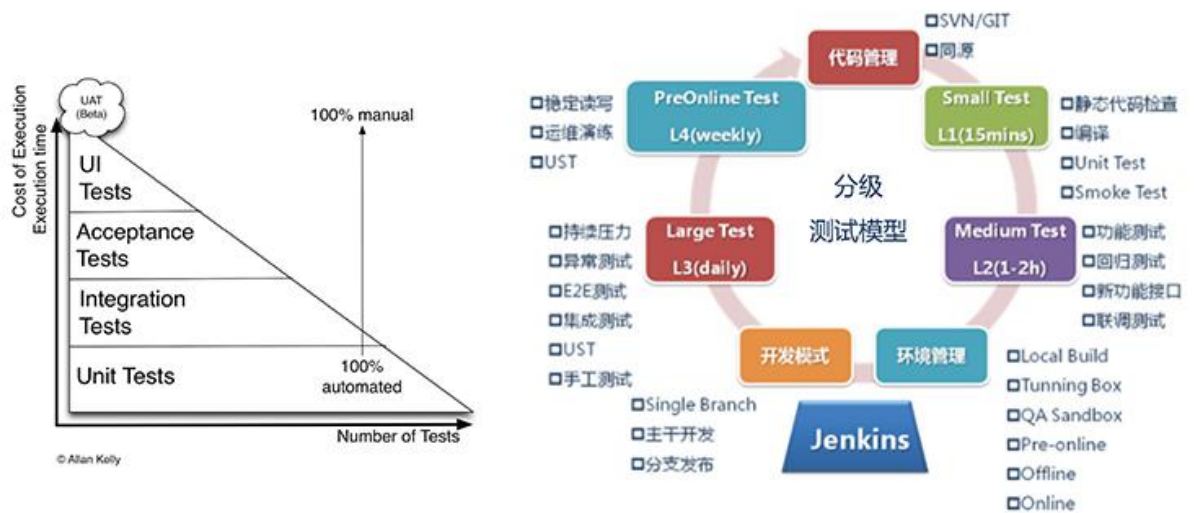


构建管理

- 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

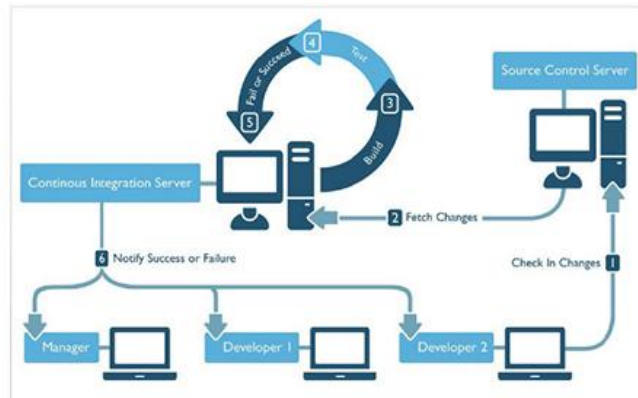


建立分级测试体系，从多个层次和多个验证角度实现质量防护网

持续集成

坚持的原则：

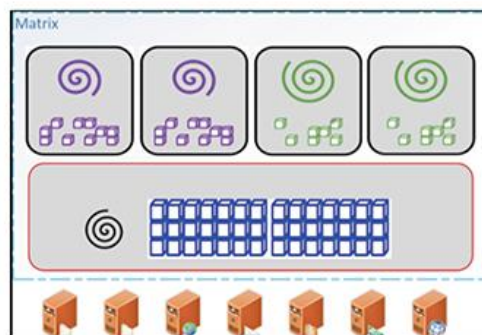
- 小改动，逐步构建
- 每人每天提交代码
- 在主干上持续集成
- 至少每天进行集成
- 使用持续集成工具
- 自动化构建和测试
- 分级测试快速反馈
- 红灯需要立即修复



环境管理 – 容器化

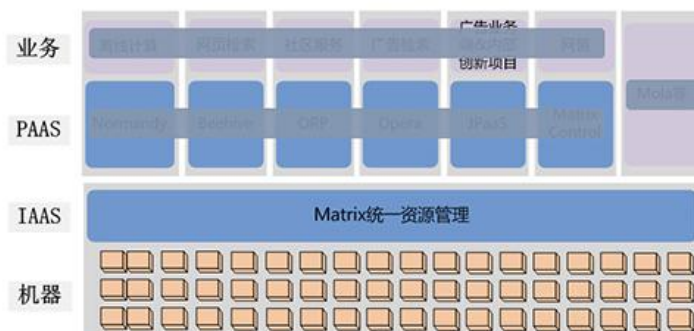
使用Matrix作为公司数据中心的集群管理系统，为所有的产品线提供规范的底层架构

- 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

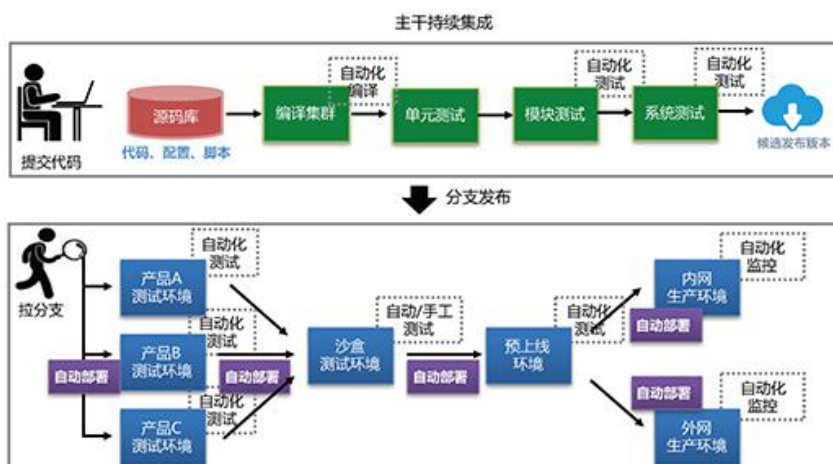
- 资源隔离与混部，公司内所有机器在统一资源管理下实现高效、充分利用
- 与产品线合作构建服务托管平台，提升服务的自动化运维效率
- 部署机器占公司机器总数50%以上



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

部署

部署

部署

清空失败计数

重置失败时间

刷新

服务名字: demo.s
集群列表: y01 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/未执行数量)
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
		开发度	暂停点	管理策略				
552088		1	无暂停点	1	2016-01-27 11:21:21	无	DOING	无

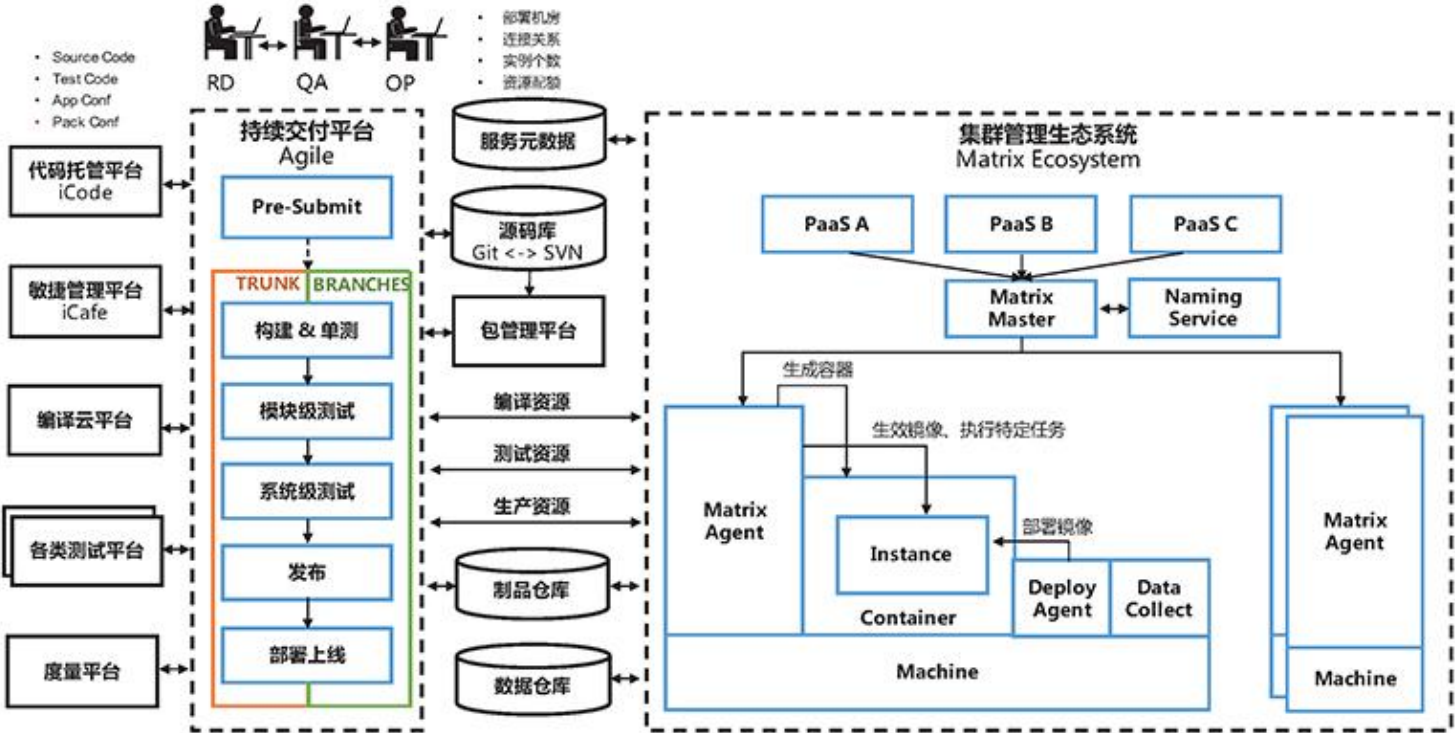
分组名称		部署策略			开始时间	完成时间	分组状态	操作
		开发度	暂停点	管理策略				
group1		1	1	1	2016-01-27 11:21:21	无	PENDING	<div>暂停 继续 清空失败计数</div>

统计数据: 1/0/0/0/3 (成功/失败/忽略/执行中/未执行数量)

批量安全

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	操作详细结果		
<input type="checkbox"/>	0 demo.s	cq	ADO	否	2016-01-27 11:21:21	无	SUCCESS	INSTANCE_OK 详细结果信息	10:48:05.55 cq01-matrix-storage0309.cq01.baidu.com	查看
<input type="checkbox"/>	1 demo.s	cq	ADO	否	无	无	CREATED	INSTANCE_OK 详细结果信息		查看
<input type="checkbox"/>	2 demo.s	y01	ADO	否	无	无	CREATED	INSTANCE_OK 详细结果信息		查看

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



从度量中找到问题，从度量中看到进步

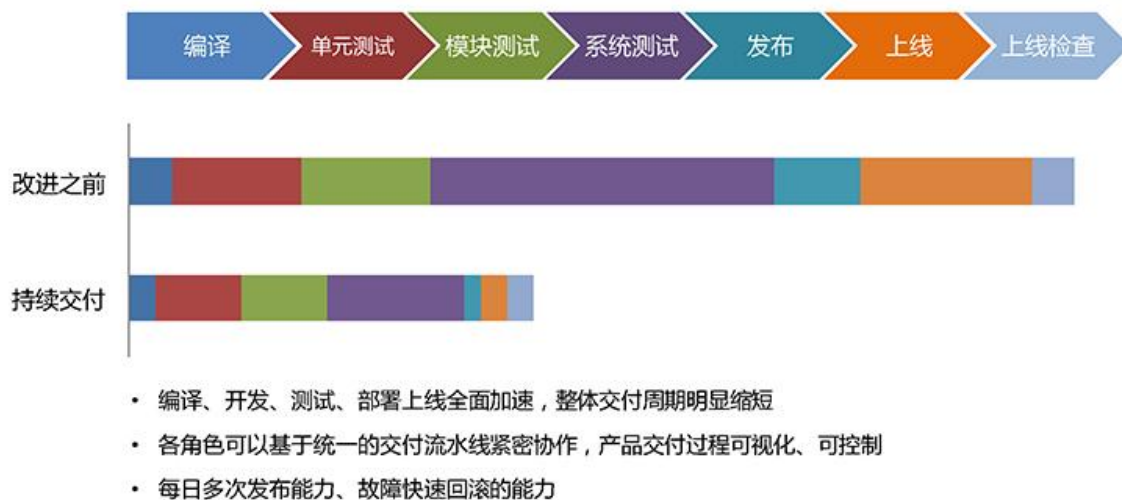


渐进式实施



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

持续交付效果案例



持续交付的价值



flying 飞翔 收集整理
QQ: 715031064
广州 linux 运维: 478477301
linux 运维菜鸟: 313184229
2018.3.9

