



DevOpsDays

Shanghai

— 2017.8.18-8.19 —

上海龙之梦酒店（长宁区延安西路1116号）

主办单位：



高效运维社区
GreatOPS Community



Best Practice
最佳实践



DevOps下的质保测试方法

Allan Dong
世界质量报告中国区撰稿人
凯捷测试总监

目录



1

Client Challenges 面临的挑战

2

DevOps QA/Test Solution 质保测试方案

3

Business Benefits 业务效益

4

Case Story 参考案例

5

Point of View 质量观点

6

Solution Detail 方案细节

谁是我们的客户，他们的挑战是什么？



我们大部分的客户都是行业的品牌标杆。作为企业的掌管和运作者，CXO们会担心DevOps对与交付速度的要求和DevOps云化转型可能会导致更多的IT故障，从而危及企业形象，客户满意度或业务运营效率。

- ❑ 从软件故障中保护企业形象，客户满意度和业务运营
- ❑ 从DevOps计划中确定业务成果（收益/回报率）
- ❑ 达成更快的上市时间并提供匹配目标的IT解决方案
- ❑ 实现最大程度的自动化质量检查和测试运行
- ❑ 确保正确的质量验证检查与适当的覆盖率
- ❑ 确保每个人都有质量责任，并执行相关质量检查
- ❑ 高效快速的使用正确的工具和环境以及测试集来支持团队
- ❑ 通过持续的质量监控实现预测分析



DevOps环境中IT质量的挑战



一些真实发生的典型软件故障：

- 由于软件故障，3,200名美国囚犯提前释放
- 汇丰中断：数百万客户无法访问在线银行帐户2天
- 由于安全气囊软件故障，1,000,000日产汽车召回
- 由于软件注册失灵，60%的美国和加拿大星巴克商店被迫关闭
- 苹果被迫将iOS 8更新
- NEST (Google) 智能恒温器使用户感冒
- 彭博交易终端下跌了两个小时
- 合作食品客户被收费两次
- 由于软件缺陷，英国空域关闭
- 失踪60,000瑞士法郎
- 911紧急号码在美国七个州下线了六个小时
- 印度航空迫使波音787航班转机
- 信用卡故障阻止以色列主要交易渠道
- 不伦瑞克电气客户发现由于软件错误而自动切断电源
- 微软在Azure云平台体验中断
- Vodacom移动网络维修软件漏洞导致用户电话号码泄漏
- Claremont Electric不正确地向客户收取软件错误的结果
- Dropbox应用程序把Dropbox删除文件作为Bug
- 奥迪召回超过10万辆A4汽车由于安全气囊软件问题
- 电子产品ebay Store软件Bug导致所有电子产品的价格不正确
- Windows发布第二个Windows补丁在修复首次问题后
- 安大略省政府软件中的Bug几乎导致2000万美元的超额支付
- 加拿大本地选举由制表软件中的错误而导致延迟
- 由于软件故障导致的交通门票无效
- 软件错误每年花费美国经济60亿美元

65%的组织使用DevOps原则来提高交付速度

软件故障越来越多的影响业务绩效和企业形象

目的是为客户和业务
提供更快软件和
最大限度地减少潜在的失败影响

采用快速试错和失败前移的原则

尽早测试

最优测试

快速测试

对质量持续监控

目录

- 1 Client Challenges 面临的挑战案
- ➔ 2 DevOps QA/Test Solution 质保测试方案
- 3 Business Benefits 业务效益
- 4 Case Story 参考案例
- 5 Point of View 质量观点
- 6 Solution Detail 方案细节

DevOps质量检测及测试解决方案和服务



DevOps质量咨询

- DevOps质量蓝图



DevOps质量服务和解决方案

- DevOps质量工程团队
- 持续集成自动化
- 测试环境与服务虚拟化
- 测试数据自动化
- 配置和环境管理
- 持续质量监测



过程资产库

- DevOps QA过程工具包 | DevOps QBP | TMap Suite ® in Scrum | BDD (行为驱动开发) | TDD (测试驱动开发)
- CAFÉ | 持续集成自动化框架 | DevOps QA Workbench
- 测试虚拟框架 | OneShare
- TDM 中心 | 测试数据生成器
- LIVE

DevOps QA 能力



> 2000 DevOps QA 工程师

DevOps QA 案例企业

Morgan Stanley



目录

- 1 Client Challenges 面临的挑战案
- 2 DevOps QA/Test Solution 质保测试方案
- ➔ 3 Business Benefits 业务效益
- 4 Case Story 参考案例
- 5 Point of View 质量观点
- 6 Solution Detail 方案细节

我们承诺的收益和成果



快速交付客户及业务需求且最小化潜在的故障影响

>30%

投资回报

>30%

上市时间

>98%

部署预测

>30%

产能提升

成功杠杆

基础

中级

高级

精益方法

敏捷, 看板, 极限编程, SAFe

60%

80%

生命周期质量保证自动化

持续集成持续部署的程度

60%

75%

适岸方法

敏捷划分程度

50%

75%

API 与灰盒测试

服务虚拟化, 打桩, 分层

30%

50%

测试覆盖率最优化

基于风险分析的测试方法

40%

50%

测试与质量保证标准化

工作平台及环境打包一体化

40%

60%

测试环境

环境及数据最优化

40%

60%

QE 或 SDET 方法

测试驱动, 业务驱动, 单元测试自动化

40%

60%

*可定制化的杠杆百分比等级指示:

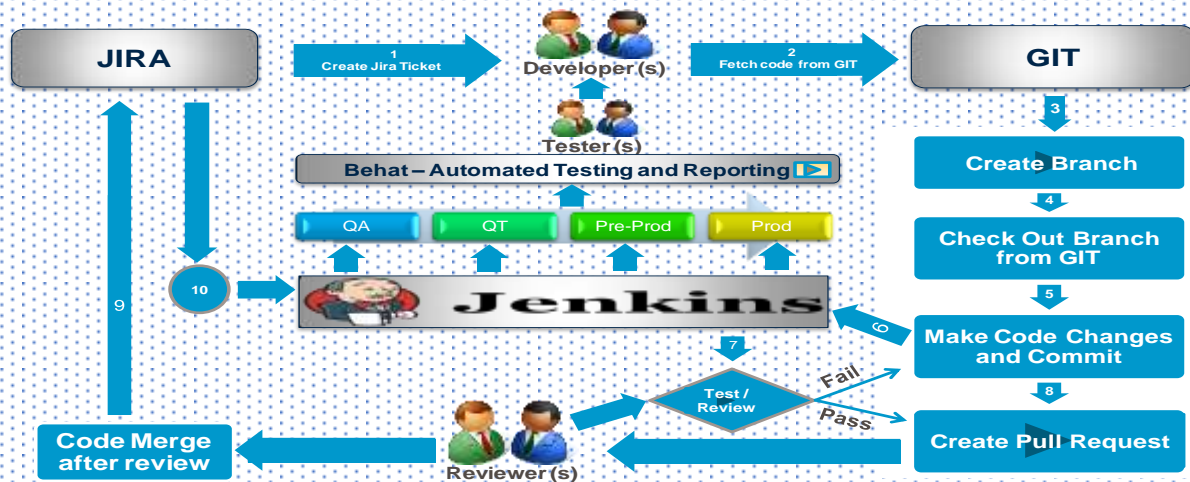
目录

- 1 Client Challenges 面临的挑战案
- 2 DevOps QA/Test Solution 质保测试方案
- 3 Business Benefits 业务效益
- ➔ 4 Case Story 参考案例
- 5 Point of View 质量观点
- 6 Solution Detail 方案细节

DevOps下的质量保证 – 欧洲某大型邮政企业



从2012年走向敏捷和DevOps流程及环境转型



基于Jenkins, Jira, Gherkin, Selenium, GIT, SoapUI 和 JMeter的CI/CT DevOps自动化框架

- 支撑86个应用程序, 7M用户, 200个接口, 18个服务合作伙伴
- 使用BDD定制的基于Selenium的测试自动化框架
- 开发与质量保证环境搭建时间从4天缩短为到4小时
- 环境故障停机从每月8小时减少到4小时
- 一键完成环境配置
- 发布周期从三周缩短到一周
- 需求处理能力提升50%
- 开发成本降低30%

某欧洲银行的DevOps转型



瀑布方法 – 预见需求，双重测试

开始采用敏捷（Scrum）方法

持续交付，工具知识传授

开发和运营一体化团队

DevOps 交付

初始情况

- ❑ 凯捷完成瀑布式开发
- ❑ 客户负责测试和运维

面临的典型问题:

- 高治理成本
- 项目准备时间长
- 部门墙繁多
- 团队不稳定

我们的解决方案

- ❑ 提供DevOps咨询服务，分阶段渐进式推进DevOps原则
 - 首先, 客户和Capgemini的团队成员都接受了Scrum的指导，并且增加了对开发环境的访问权限
 - 团队充分采用敏捷方法论的scrum原则
 - 然后, 团队开始实施持续交付，同时实施了严格的培训计划和工具部署
 - 渐进式推进, 排除所有障碍，团队协助，并且完全采用DevOps.

收益

- ❑ 上市周期缩短
 - 高效的测试流程
 - 变更过程无时间损失
- ❑ 运维稳定
 - 大量的自动化测试
 - 开发团队同时负责测试及维护
- ❑ 业务高效
 - 明确重点（做正确的事）
 - 及早反馈
 - 责任明确

DevOps实现 - 瀑布场景



阶段 1

加快串行执行速度

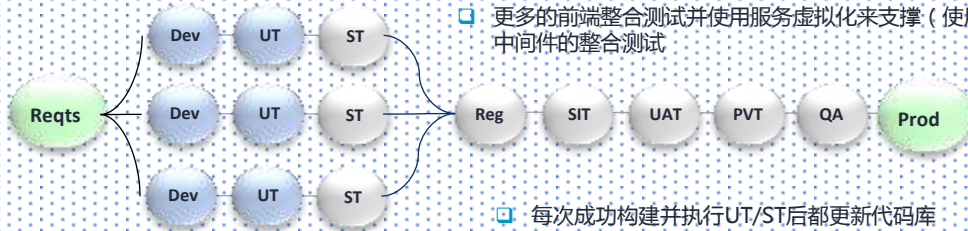
- ❑ 使用SVN, GIT或 Artifactory 来管理代码库
- ❑ 通过Jenkins/Bamboo实现构建自动化, 包括单元/回归测试自动化
- ❑ 提高从ST到QA的测试自动化程度



- ❑ “一个” 环境团队 (虚拟或委托)
- ❑ 构建自动化: 使用Puppet或Chef进行服务器配置, 并使用Jenkins或Bamboo进行部署
- ❑ 运维监控可视化

阶段2

划分可能并行的开发及测试以进一步加快速度

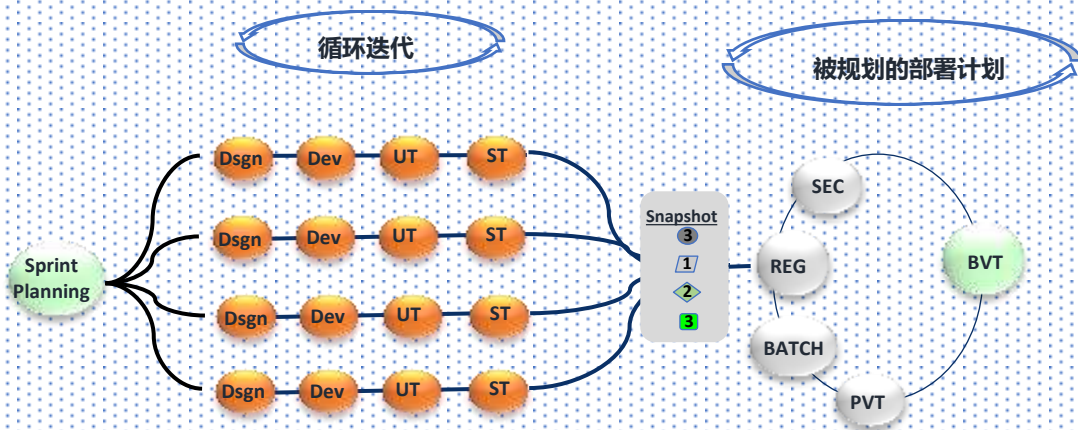


- ❑ 并行的开发及前端模式并使用通用代码库管理
- ❑ 更多的前端整合测试并使用服务虚拟化来支撑 (使用CA Lisa) 后端或中间件的整合测试

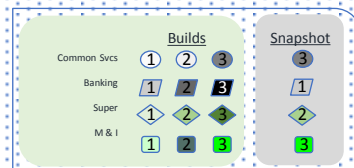
- ❑ 每次成功构建并执行UT/ST后都更新代码库
- ❑ 使用SVN/GIT或Artifactory 对每次构建的包进行仓库化管
- ❑ 使用HP Codar, Jenkins或Bamboo进行部署的自动发布管理

推荐的DevOps实现 - 敏捷场景

实现持续整合，交付，部署



- ❑ Involve operations in sprints 迭代内包含运维
- ❑ Enhanced UT (JUnit, code analyzers) 提升单元测试 (JUnit, code analyzers)
- ❑ 服务虚拟化 (CA Lisa) 及模拟 (mobile) 支撑的并行化集成测试下的持续集成 (Bamboo)
- ❑ 使用GIT或Artifactory 对每次构建进行仓库化管 (源代码, 构建包)
- ❑ 将中间件和集成部分转化为微服务, 并将其纳入Dev & Test环境中
- ❑ 使用Pivotal Cloud Foundry 和 HP Codar 支撑持续交付, 并使用其他工具协同构建, 部署
- ❑ 反馈收集部署: HockeyApp, Appaloosa, TestFairy, Play Store, Apphanc



目录

- 1 Client Challenges 面临的挑战案
- 2 DevOps QA/Test Solution 质保测试方案
- 3 Business Benefits 业务效益
- 4 Case Story 参考案例
- ➔ 5 Point of View 质量观点
- 6 Solution Detail 方案细节

DevOps质量的关键点



DevOps 综合质量方法

DevOps 质量工程团队

行为驱动开发
&
测试驱动开发

零介入
持续化 自动化
测试

测试环境虚拟化
&
测试数据自动化

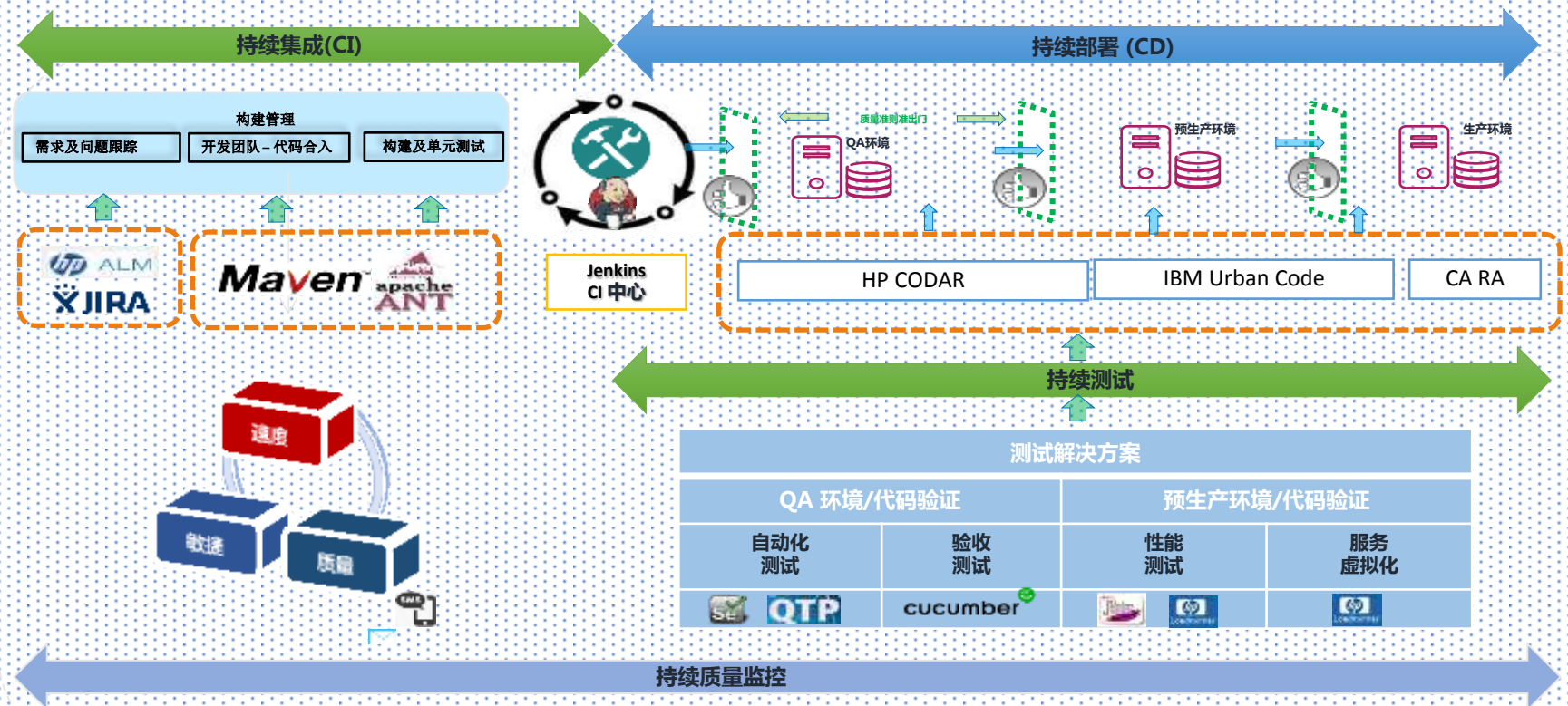
DevOps 质量度量模型

持续质量监控

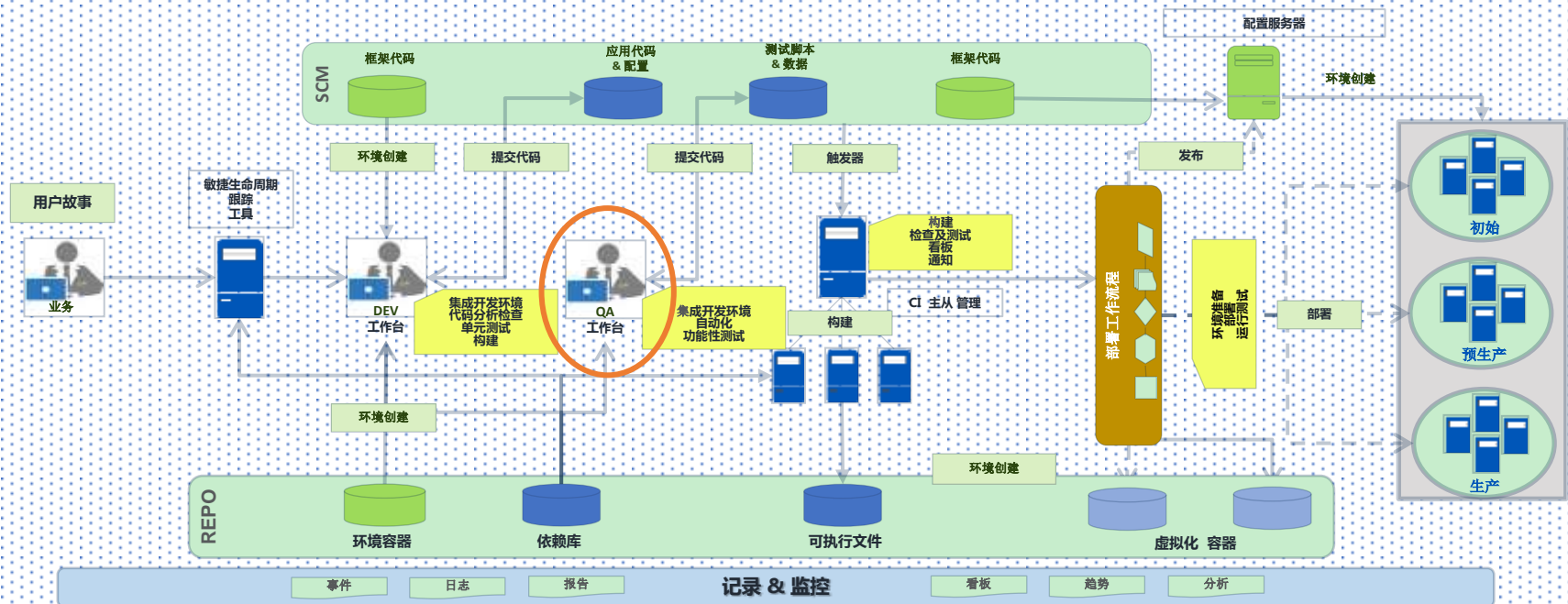
目录

- 1 Client Challenges 面临的挑战案
- 2 DevOps QA/Test Solution 质保测试方案
- 3 Business Benefits 业务效益
- 4 Case Story 参考案例
- 5 Point of View 质量观点
- ➔ 6 Solution Detail 方案细节

DevOps的综合质量方法



DevOps质量保证流程

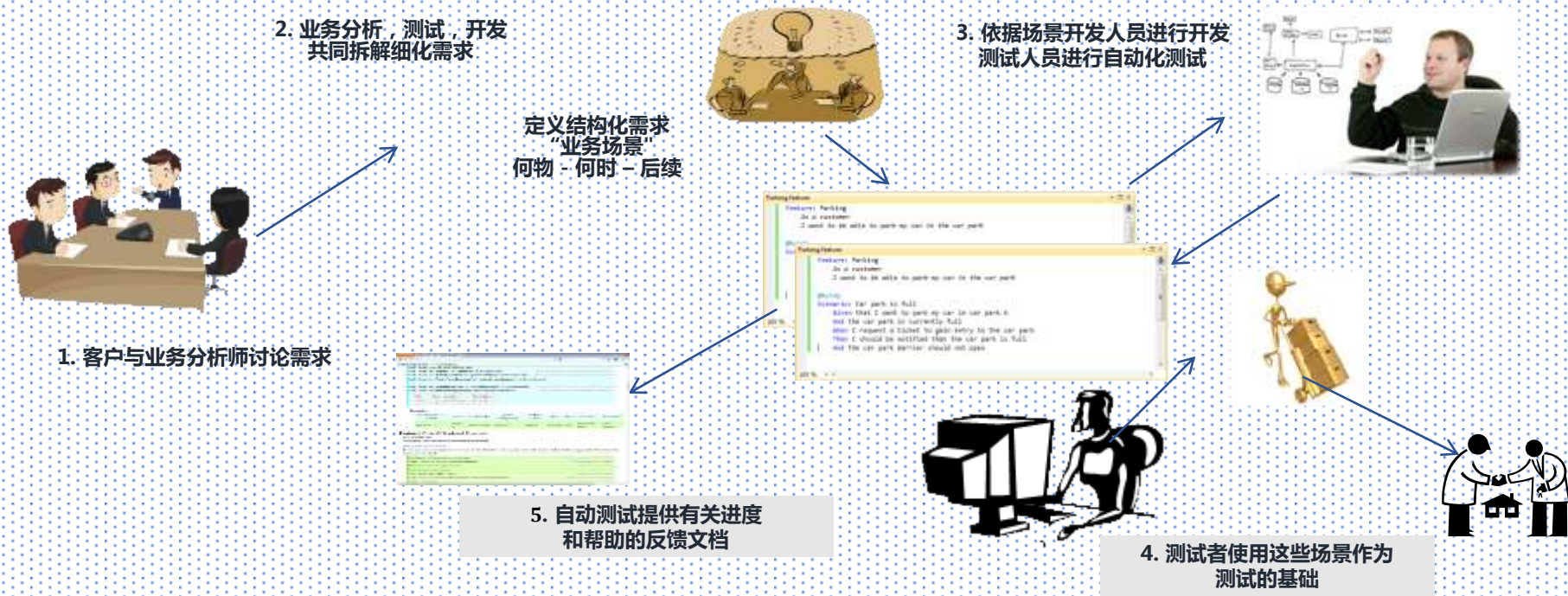


DevOps QA 工程师提供



- ❑ 所有流程和所有团队成员的质量保证指南
- ❑ 共有主要QA自动化平台
- ❑ 灵活的QA环境，具有服务虚拟化和基于云的环境
- ❑ 每个应用程序进行端到端QA验证检查
- ❑ 测试覆盖率检查
- ❑ 应用程序的整体质量级监控

行为驱动开发流程 (BDD)



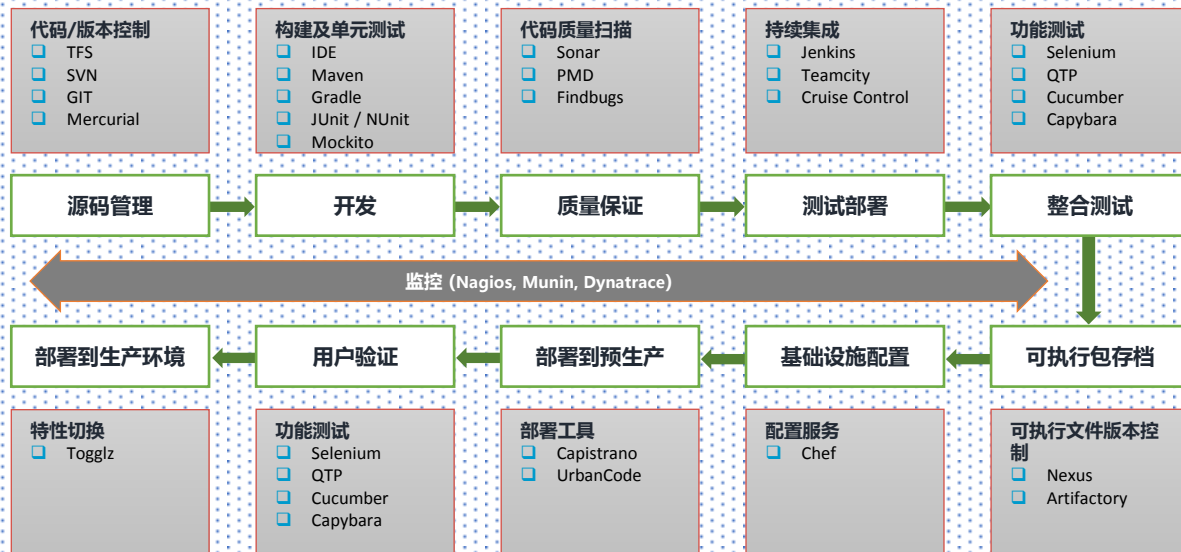
- ❑ BDD 需要在实施软件开发之前编写实例 (或场景) , 并与非技术利益相关者合作设计安排迭代计划
- ❑ BDD 需要自动验收测试支撑
- ❑ BDD 是一个开发过程, 其中代码的测试是在编写实现代码之前编写的

零介入质量自动化核心



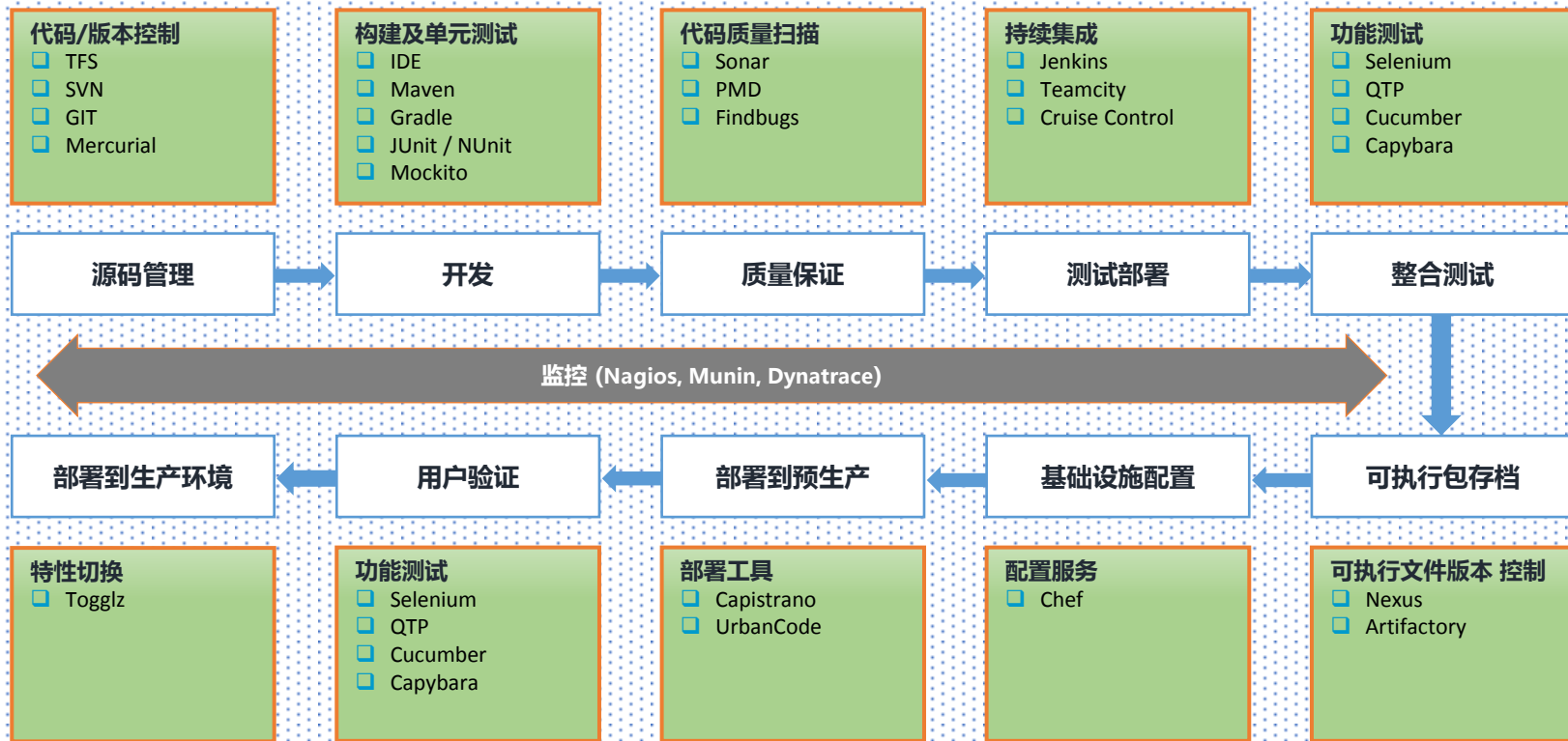
DevOps质量自动化平台核心：

- ❑ BDD & TDD 赋能工具
- ❑ 配置管理
- ❑ 持续集成 Build Automation
- ❑ 部署
- ❑ 缺陷管理
- ❑ 功能测试
- ❑ 性能测试
- ❑ 服务虚拟化
- ❑ 环境虚拟化
- ❑ 测试数据管理
- ❑ 持续质量监控Continuous Quality
- ❑ 预测分析



特点：集成企业标准工具集，可用于所有活动，开放的，可以快速并简洁的整合新兴技术解决方案的DevOps平台

零介入质量自动化平台工具



环境与数据虚拟化



基于云的 测试环境

- 支撑及时和可定制化的QA环境
- 有效利用硬件基础设施



服务 虚拟化

- 模拟服务进行集成测试
- 左移集成测试



网络 虚拟化

- 虚拟化网络架构和依赖
- 左移真实测试并模拟真实环境



测试 数据虚拟化

- 快速可靠的创建数据并用于测试
- 可重复使用的测试自动化数据集

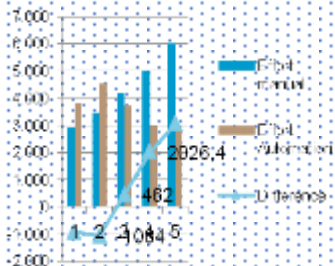


DevOps质量指标和KPI模型

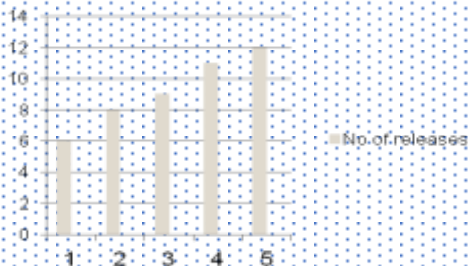


看板示例

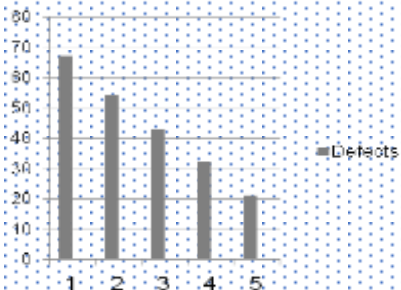
工作 手动 VS 自动



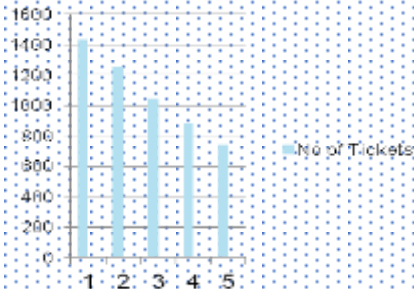
发布次数 YoY



迭代内缺陷泄漏量



问题单数量 YoY





Thanks

荣誉出品

高效运维社区

国际最佳实践管理联盟



想第一时间看到
高效运维社区公众号
的好文章吗？

请打开高效运维社区公众号，点击右上角小人，如右侧所示设置就好



DevOps的关键质量挑战



- ❑ 由于测试自动化程度不足导致测试速度不足 (39%)
- ❑ 难以确定测试应该关注的正确领域 (33%)
- ❑ 缺乏适当的测试环境和数据 (31%)
- ❑ 团队缺乏专业的测试和质量专长(31%)
- ❑ 缺乏适合DevOps原则的良好测试方法(29%)
- ❑ 在迭代内测试用例及数据难以重复使用 (29%)

世界质量报告2015

解决方案关键组件

DevOps 质量
方案

DevOps 质量
自动化框架

DevOps 质量
工程师

DevOps 质量
环境

DevOps 质量
度量模型



- TMAP (测试管理方案)始于1995
- TMAP NEXT强化版于2006
- TMAP 套件于 2014:
 - ☐ TMAP HD - Human Driven (Modern Agile)
 - ☐ TMAP.NET for building blocks
 - ☐ TMAP NEXT