配置管理与持续集成

清华大学软件学院 刘强

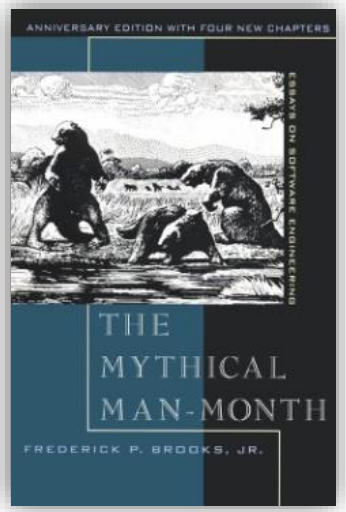
教学提纲

1. 软件配置管理
2. 持续集成与交付
3. 实验：微信抢票应用开发

软件配置管理

There is nothing in this world constant but inconstancy.

—— Franklin D. Roosevelt



开发人员交付的是用户满意程度，而不仅仅是实际的产品。用户的实际需要和用户感觉会随着程序的构建、测试和使用而变化。

软件配置管理

- 找不到某个文件的历史版本
- 开发人员使用错误的版本修改程序
- 开发人员未经授权修改代码或文档
- 人员流动，交接工作不彻底
- 无法重新编译某个历史版本
- 因为协同开发或异地开发，版本变更混乱

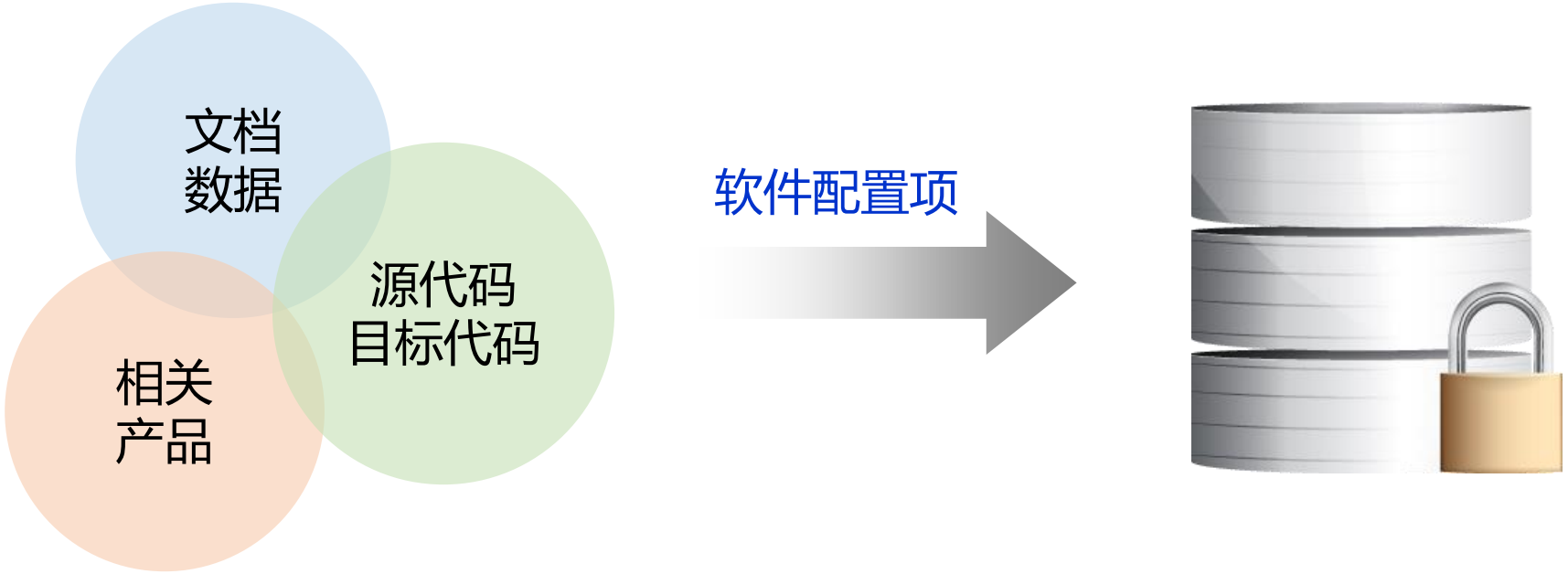
软件配置管理

软件配置管理是一种标识、组织和控制修改的技术，它作用于整个软件生命周期，其目的是使错误达到最小并最有效地提高生产率。

- 记录软件产品的演化过程
- 确保开发人员在软件生命周期的每一个阶段都可以获得精确的产品配置
- 保证软件产品的完整性、一致性和可追溯性

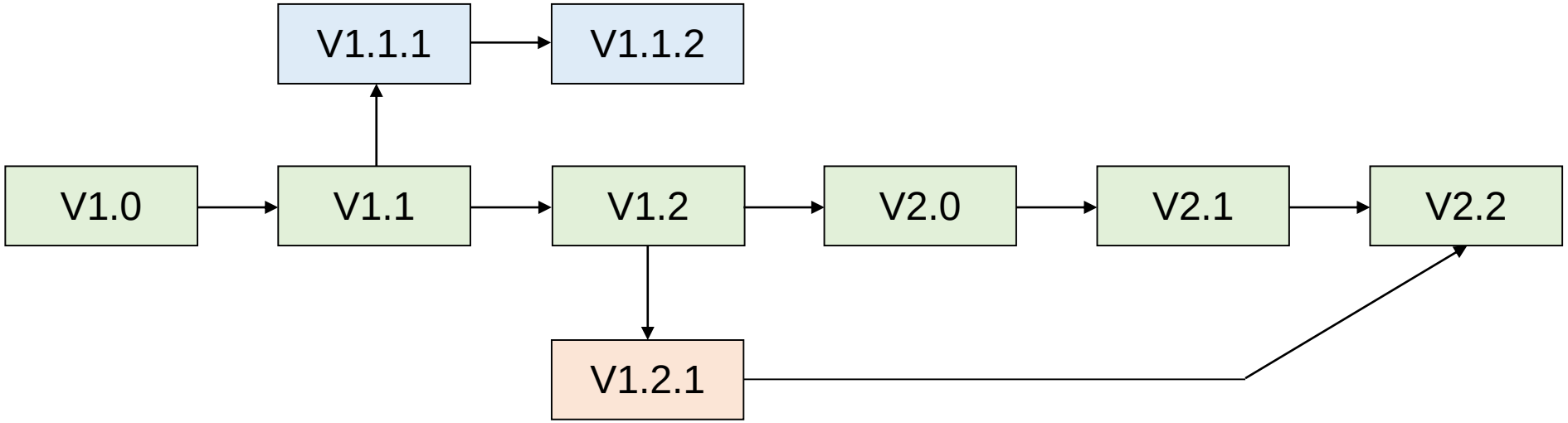
术语：软件配置项

软件配置项（Software Configuration Item，简称SCI）是为了配置管理而作为单独实体处理的一个工作产品或软件。



术语：版本

版本（Version）是在明确定义的时间点上某个配置项状态；随着功能的增加，修改或删除，配置项的版本随之演变。



术语：版本

版本以版本号进行标识，即为了简略表达特定版本的目的和意义，方便区分不同的版本，我们使用版本号这个概念。

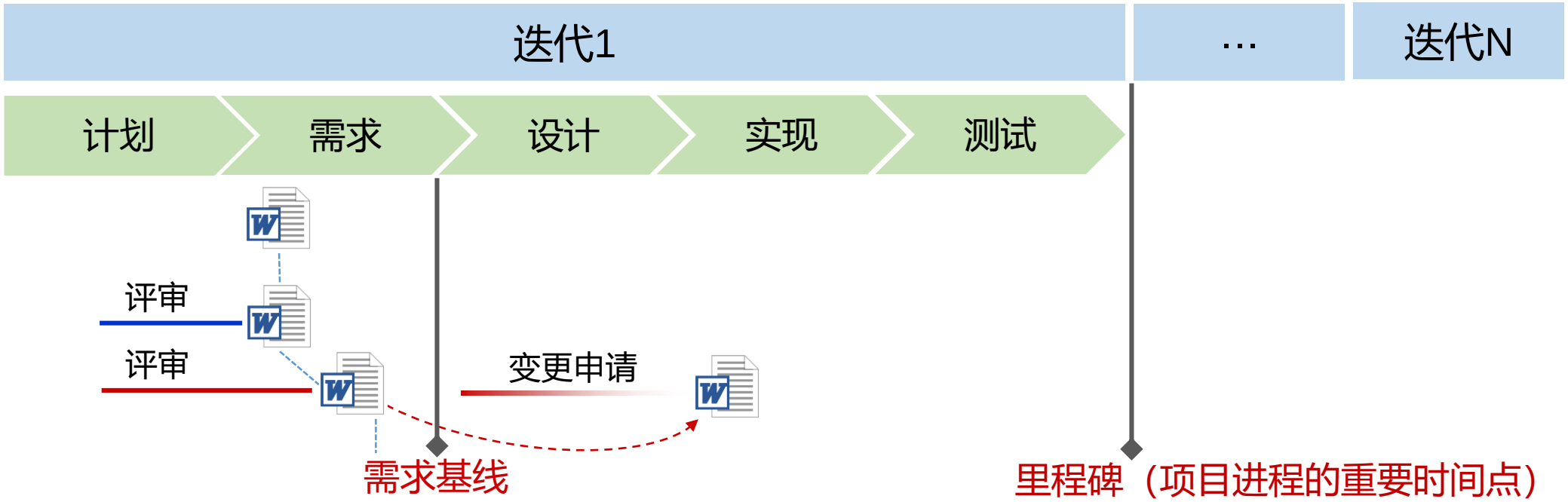
版本号格式：X.Y.Z

微信 WeChat 6.7.2

- X是主版本，当配置项版本升级幅度比较大，或者有重大架构变更时，允许增大X值。
- Y是次版本，如果配置项的版本升级幅度比较小，有一些功能增加或变化，一般只增大Y值， X值保持不变。
- Z是增量版本，如果只是配置项上的修改，主要是修复一些缺陷，一般只增大Z值，X.Y值保持不变。

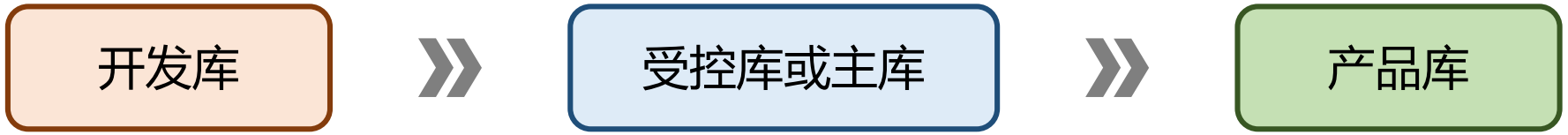
术语：基线

基线（Baseline）是软件配置项的一个稳定版本，它是进一步开发的基础，只有通过正式的变更控制过程才能改变。



术语：配置库

配置库（SCM Repository）存储配置管理信息以及软件配置项的版本信息。



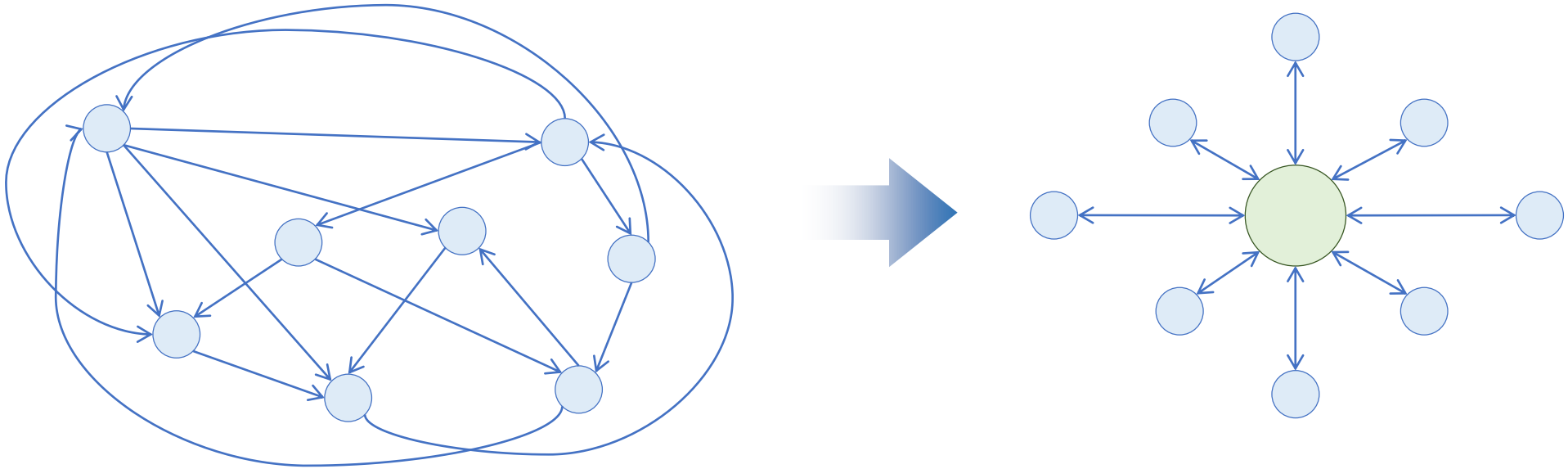
- 存放项目期间处于开发状态的相关文档和代码
- 存放项目组工作期间的相关沟通记录等
- 软件开发人员的工作区，由开发人员控制

- 管理当前基线和控制对基线的变更
- 开发人员在源代码完成并通过自测后将其并入受控库
- 一些开发计划、周报等文档，由专人维护其变更

- 存档各种已分发使用的发布基线
- 在开发产品完成系统测试之后，作为最终产品存入产品库内，等待交付用户或现场安装

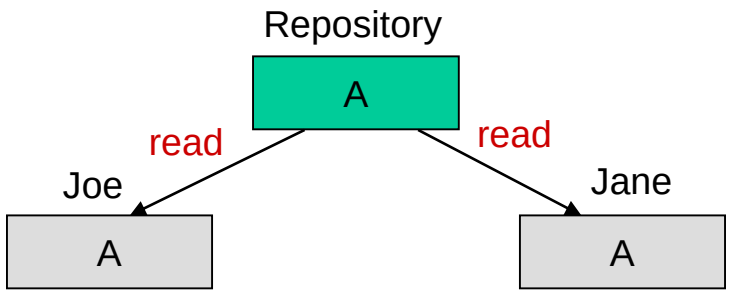
版本控制

场景1：每个程序员各自负责不同的专门模块，没有出现两个程序员修改同一个代码文件的问题。

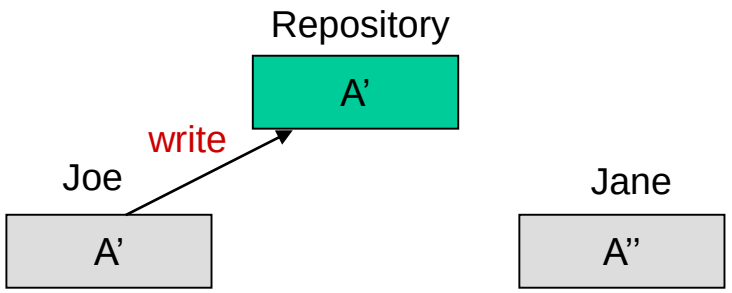


版本控制

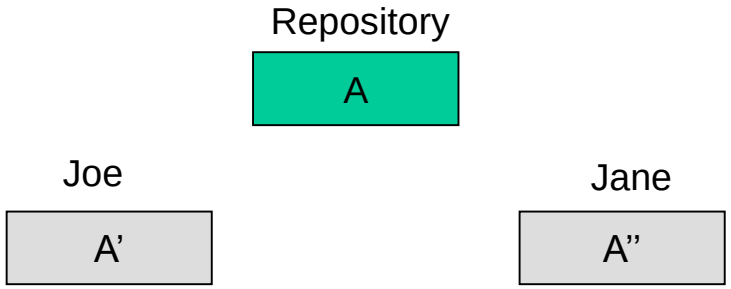
场景2：假设两个程序员同时修改同一个代码文件，就会出现代码覆盖问题。



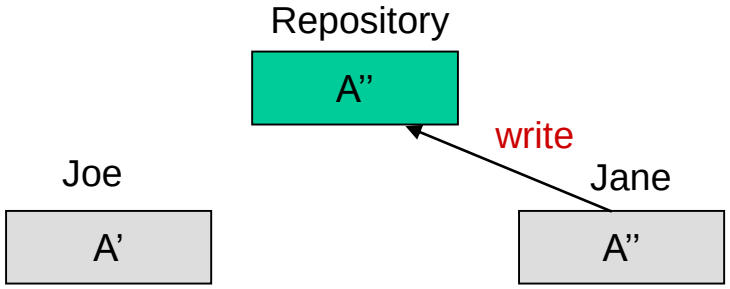
1. 两个程序员读取同一个文件



3. Joe首先发布其版本



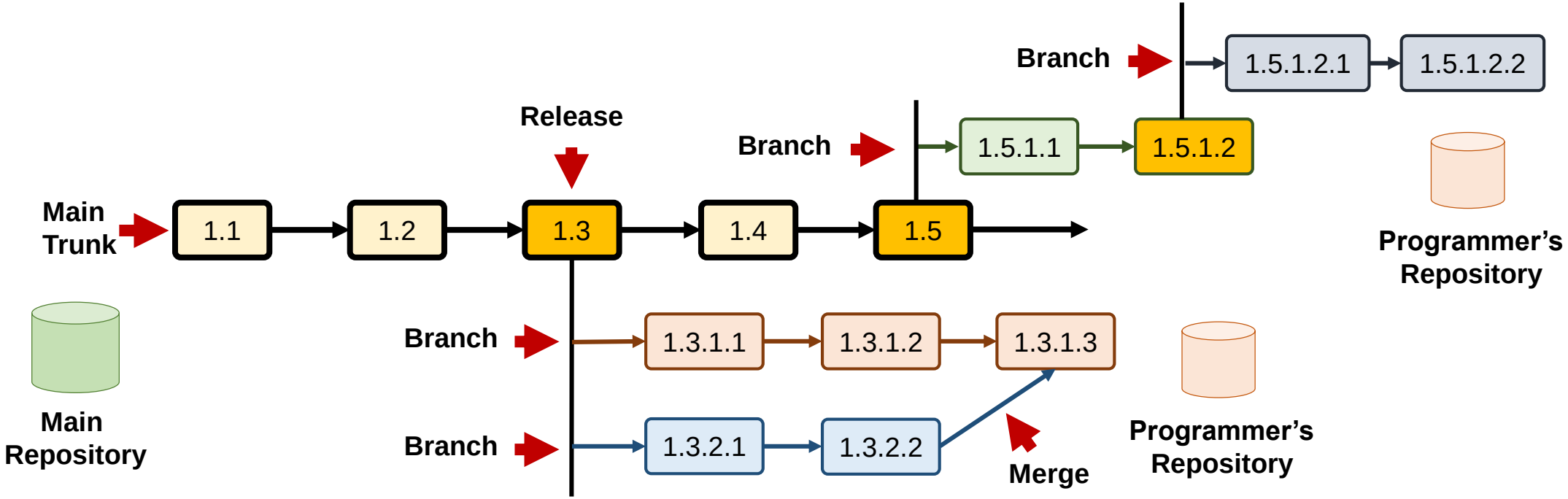
2. 两个程序员开始编辑文件副本



4. Jane意外地覆盖了Joe的版本

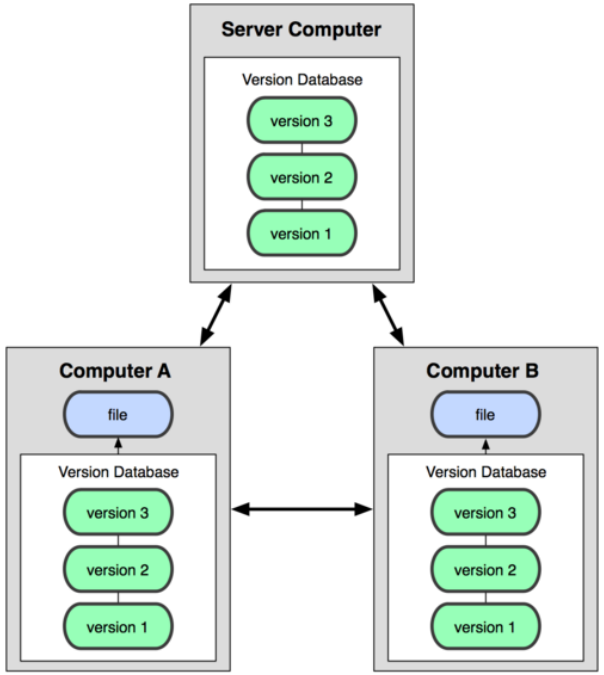
版本控制

版本控制（Version Control）是对系统不同的版本进行标识和跟踪的过程，从而保证软件技术状态的一致性。



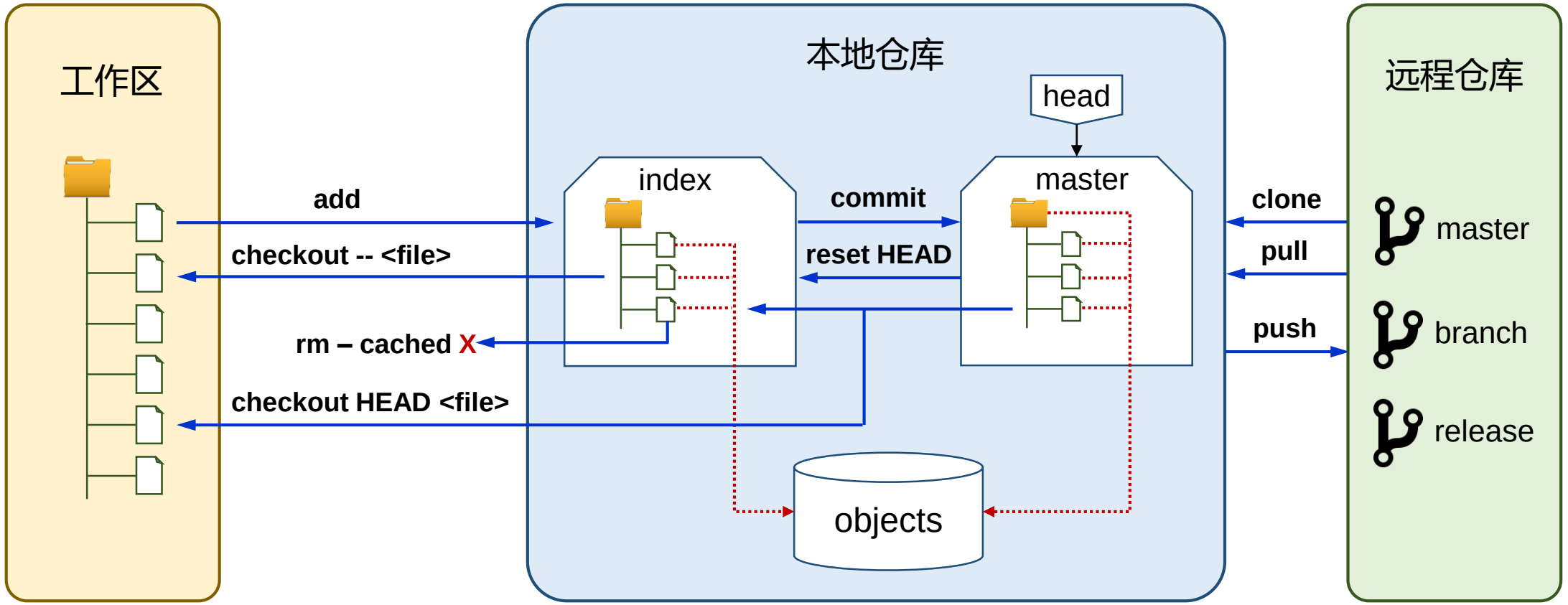
软件配置管理工具 Git

Git 是一个开源的分布式版本控制系统，它最初由 Linux Torvalds 编写，用作 Linux 内核代码的管理，后来在许多其他项目中取得很大的成功。



- 使用中央存储库，但给项目中的每一个开发人员提供完整的项目源代码副本，从而减少对中心仓库的依赖，并支持离线工作。
- 支持快速且可靠地在项目中创建不同的工作集（称为分支），可以跨分支共享更改（称为合并）。
- 可以轻松在开发人员子集之间共享分支和代码更改，这些更改无需先签入中央存储库。

Git 配置库

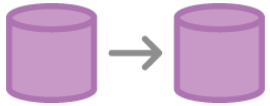

工作区
本地仓库
head
index
master
远程仓库
add
commit
clone
checkout -- <file>
reset HEAD
pull
rm – cached X
push
checkout HEAD <file>
objects
master
branch
release

Git 基本操作

建立一个 Git 仓库

git init

创建一个空的本地仓库



git clone

将远程仓库中的代码克隆到本地

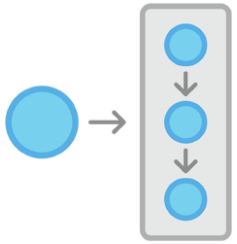
git config

配置或读取相应的工作环境变量

记录快照

git add

把工作区的所有修改提交到暂存区，为提交到本地库准备快照



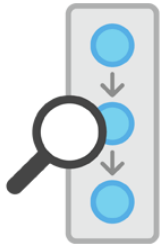
git commit

把暂存的快照提交到本地的版本库中

查看 Git 库

git status

查看工作区和暂存区文件的状态

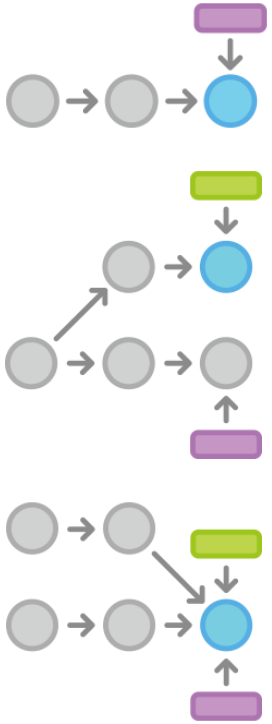


git log

查看提交历史

Git 基本操作

Git 分支



git branch

用于分支操作，包括创建一个新的分支、查看分支等

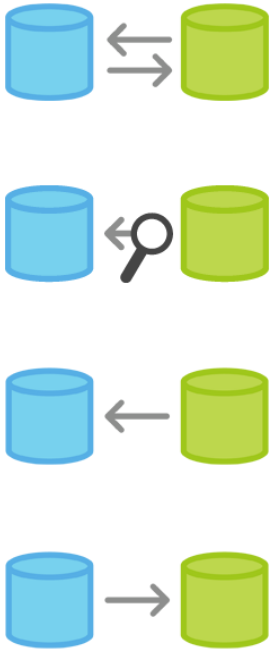
git checkout

将某个文件的历史版本检出到工作区，或者切换到指定分支

git merge

把一些分支内容合并到指定分支

远程仓库



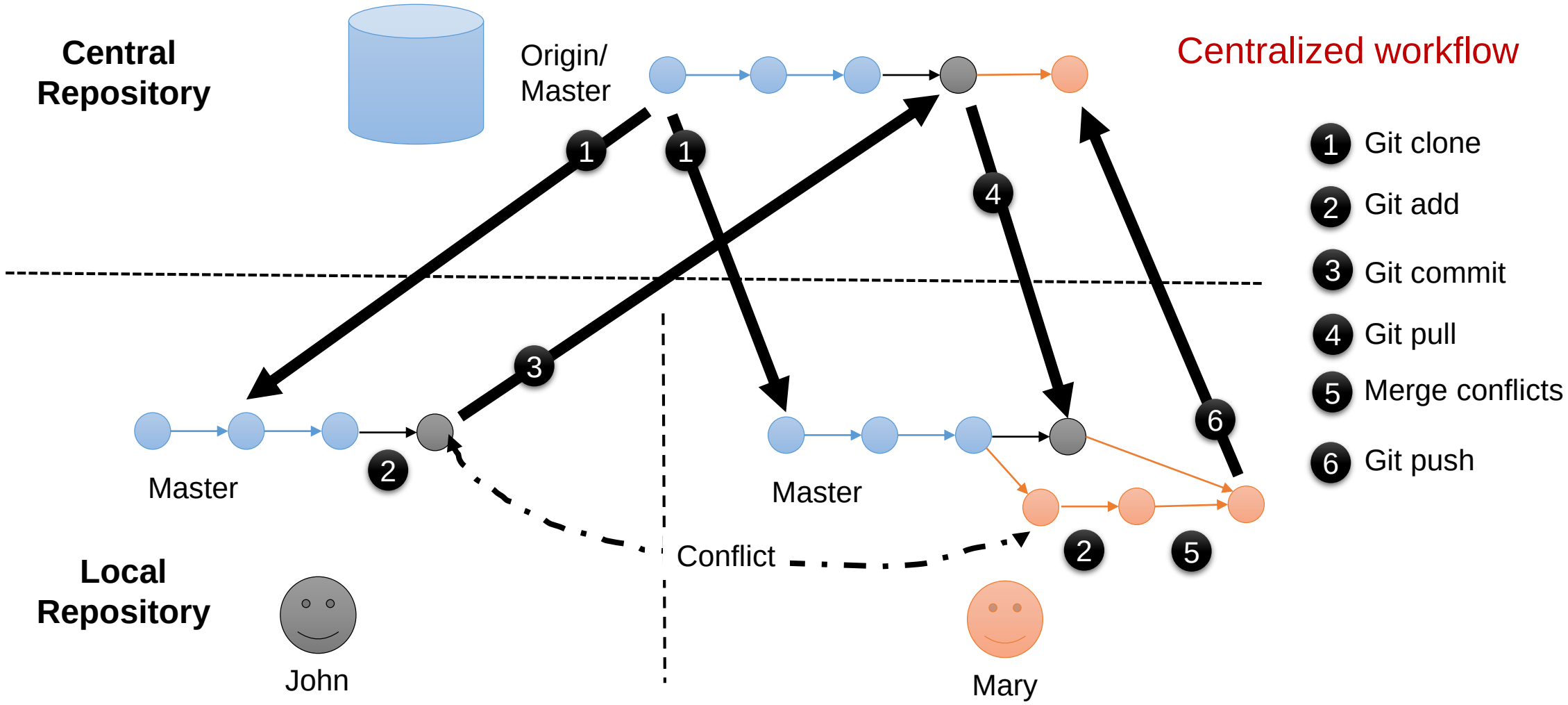
git remote

git fetch

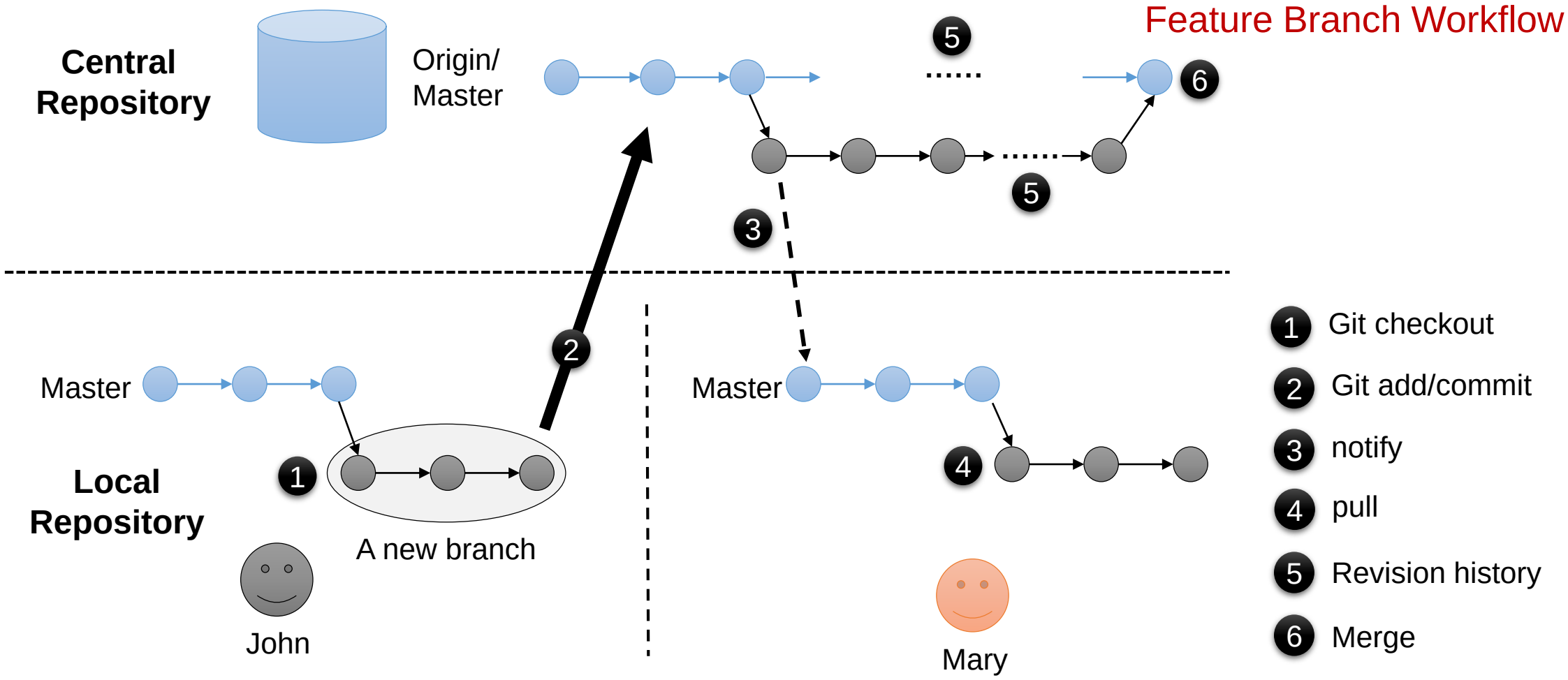
git pull

git push

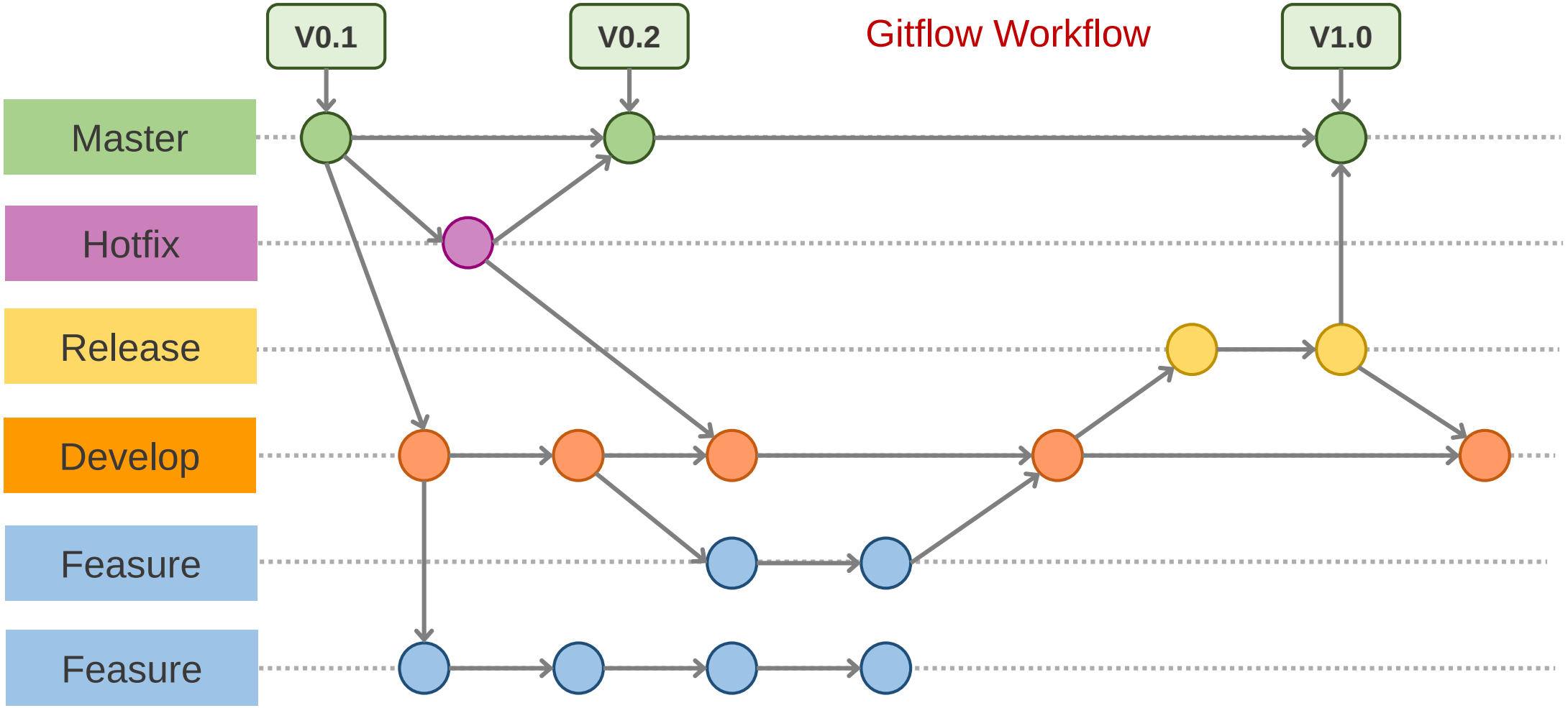
Git 团队协作模式


Central Repository
Origin/ Master
Centralized workflow
1 Git clone
2 Git add
3 Git commit
4 Git pull
5 Merge conflicts
6 Git push
Master
Master
Conflict
Local Repository
John
Mary

Git 团队协作模式


Feature Branch Workflow
Central Repository
Origin/ Master
Master
Local Repository
A new branch
John
Master
Mary
1 Git checkout
2 Git add/commit
3 notify
4 pull
5 Revision history
6 Merge

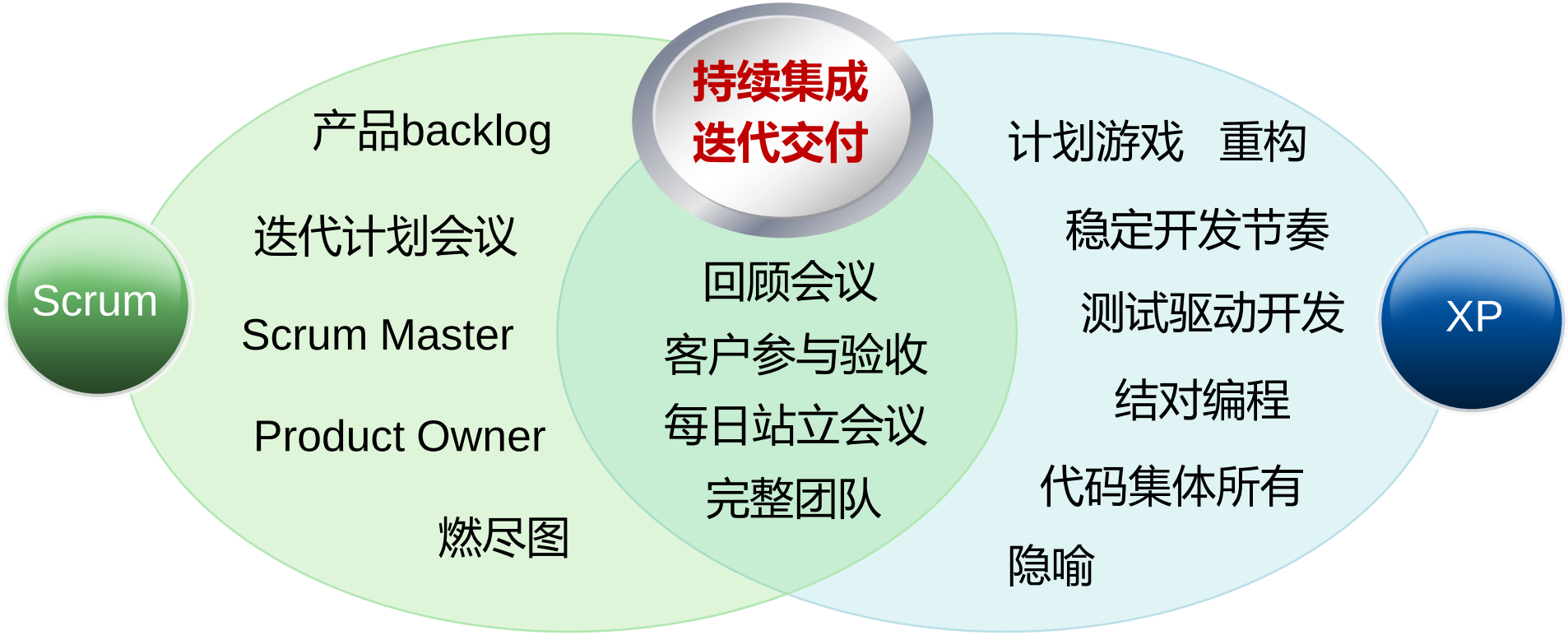
Git 团队协作模式


V0.1
V0.2
Gitflow Workflow
V1.0
Master
Hotfix
Release
Develop
Feasure
Feasure

教学提纲

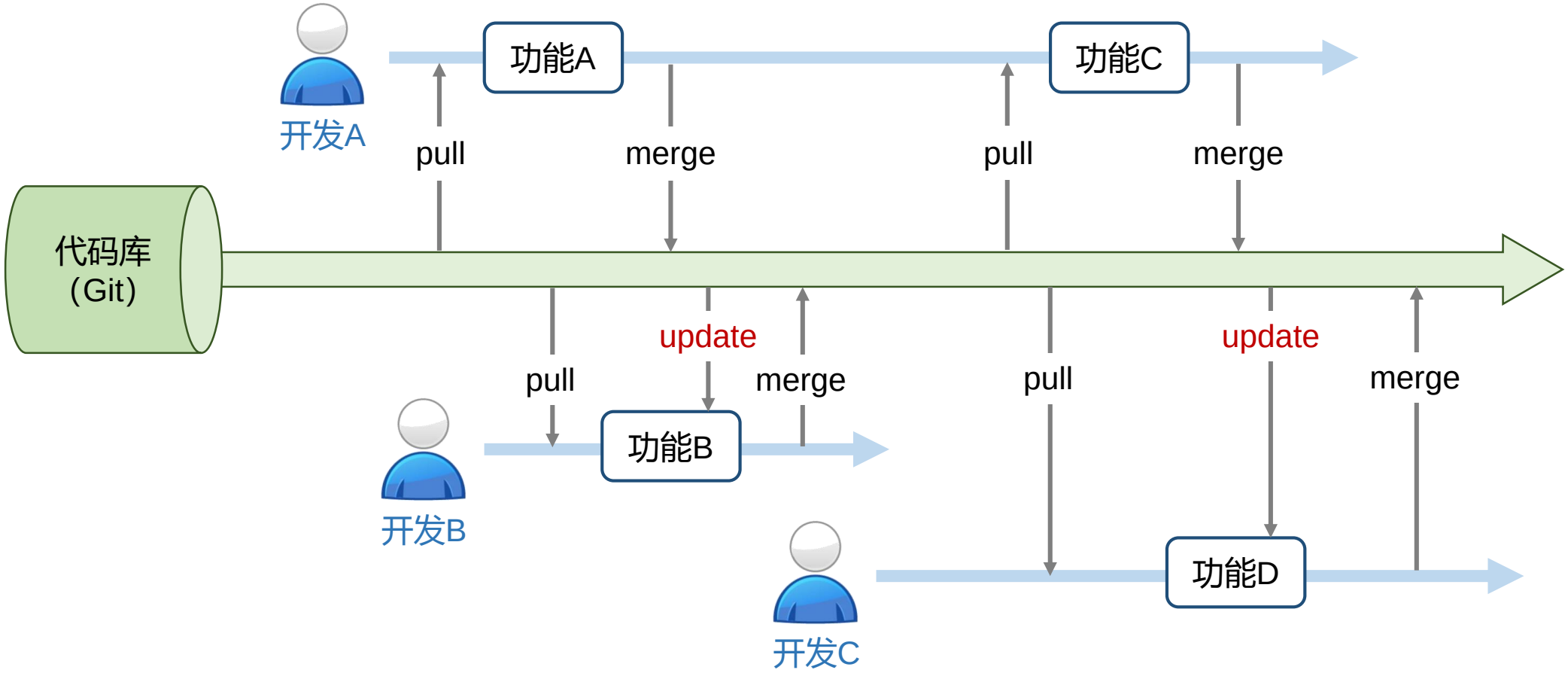
1. 软件配置管理
2. 持续集成与交付
3. 实验：微信抢票应用开发

回顾：敏捷开发方法



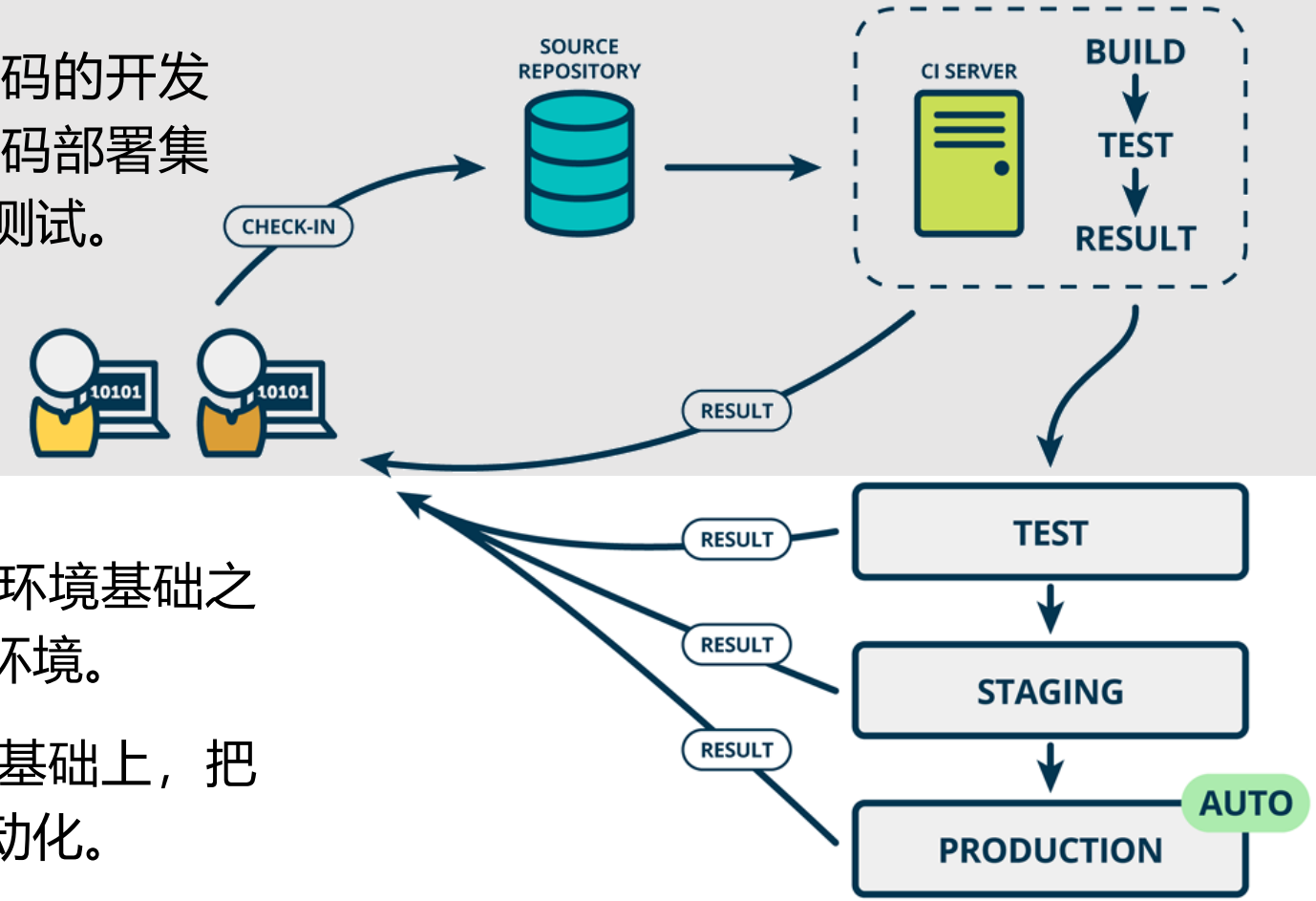
Scrum偏重项目管理　　XP偏重编程实践

持续集成的工作模式


开发A
功能A
功能C
pull
merge
pull
merge
代码库
(Git)
update
update
pull
merge
pull
merge
开发B
功能B
开发C
功能D

持续集成与交付

持续集成是指开发者在代码的开发过程中，可以频繁地将代码部署集成到主干，并进行自动化测试。



持续交付是在持续集成的环境基础之上，将代码部署到预生产环境。

持续部署是在持续交付的基础上，把部署到生产环境的过程自动化。

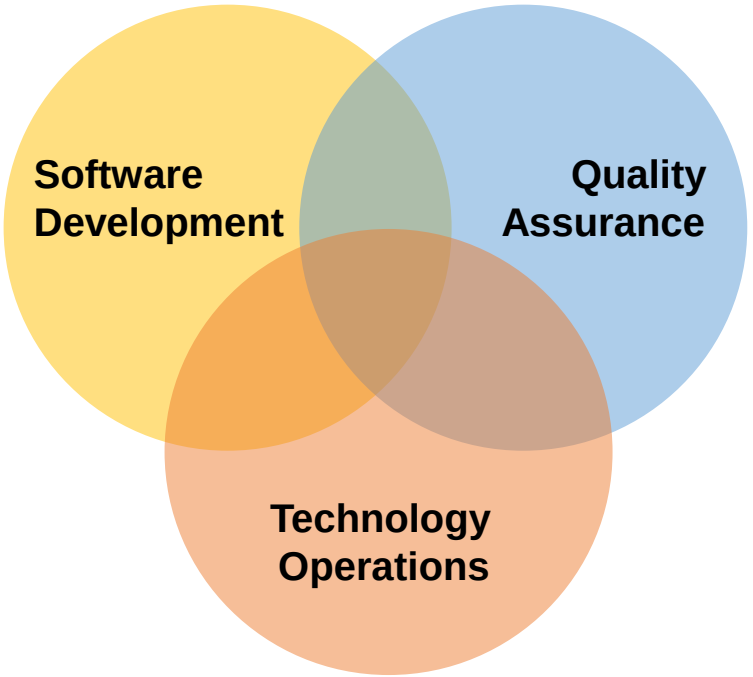
持续集成与交付

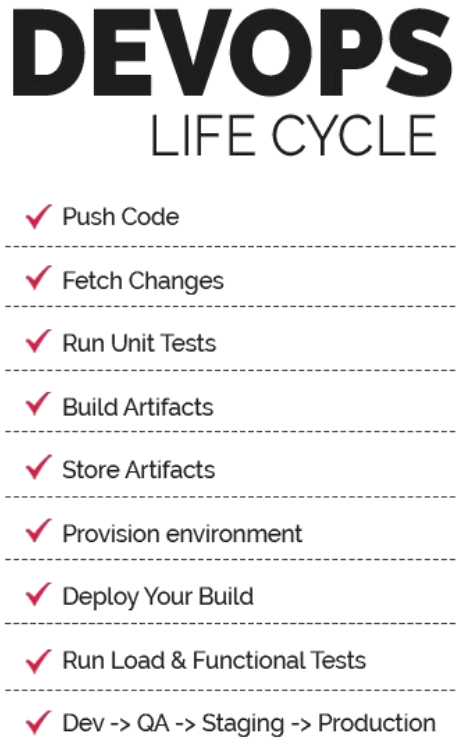
- 一周内平均部署几十次，几乎每个开发人员的每次修改就会导致一次部署。
- 这不仅仅意味着可以更快从用户那里得到使用反馈，更可以迅速对产品进行改进， 更好地适应用户的需求和市场的变化。

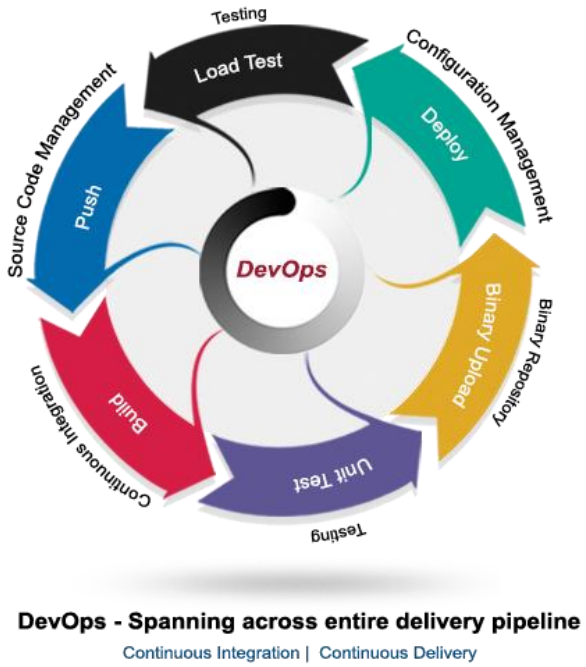
- Facebook有数千名开发和运维人员，成千上万台服务器。
- 平均来说一位运维人员负责500台服务器，他们每天部署两次。

DevOps（Development+Operation）

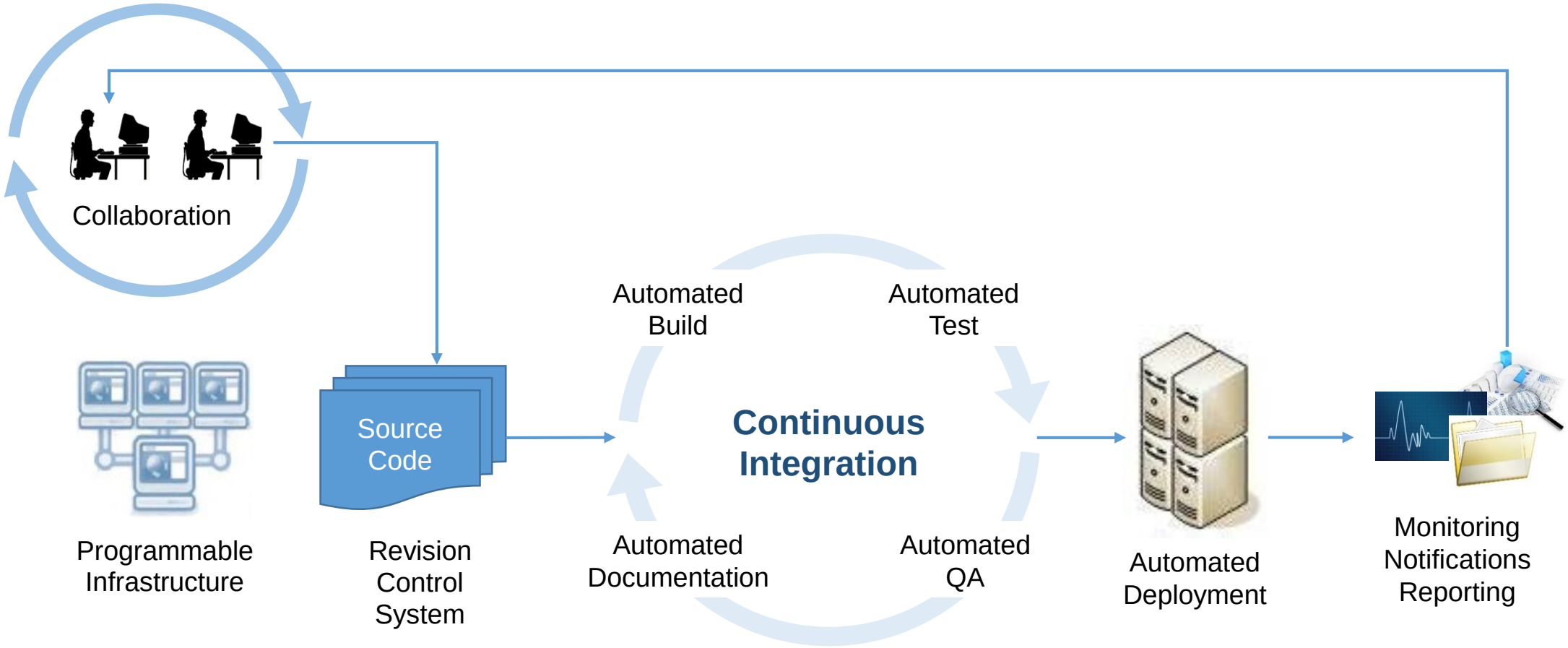
DevOps术语是Development与Operations的合并简写，描述于精简软件交付流程，强调从生产环境到开发的生命周期快速地反馈学习。





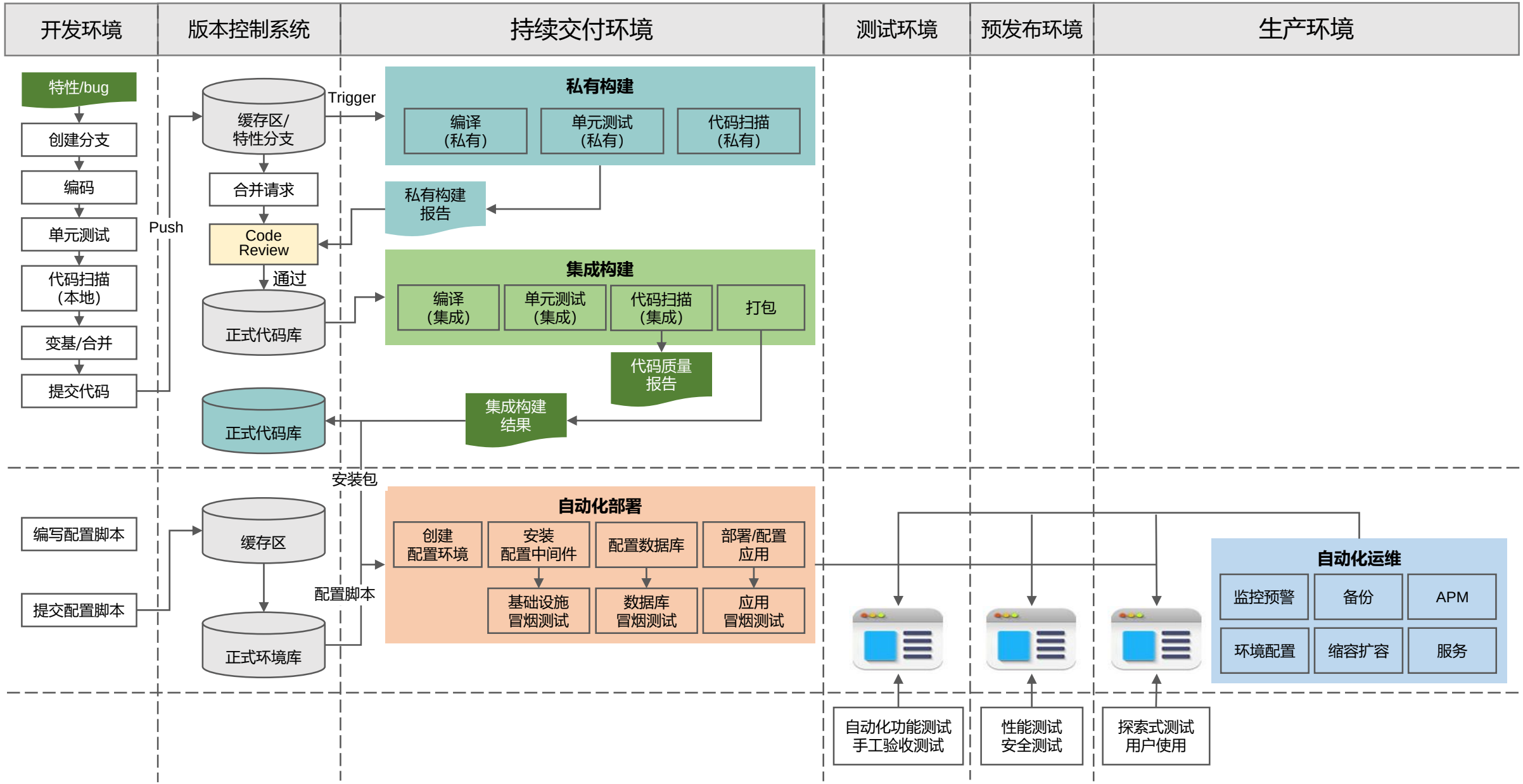


DevOps

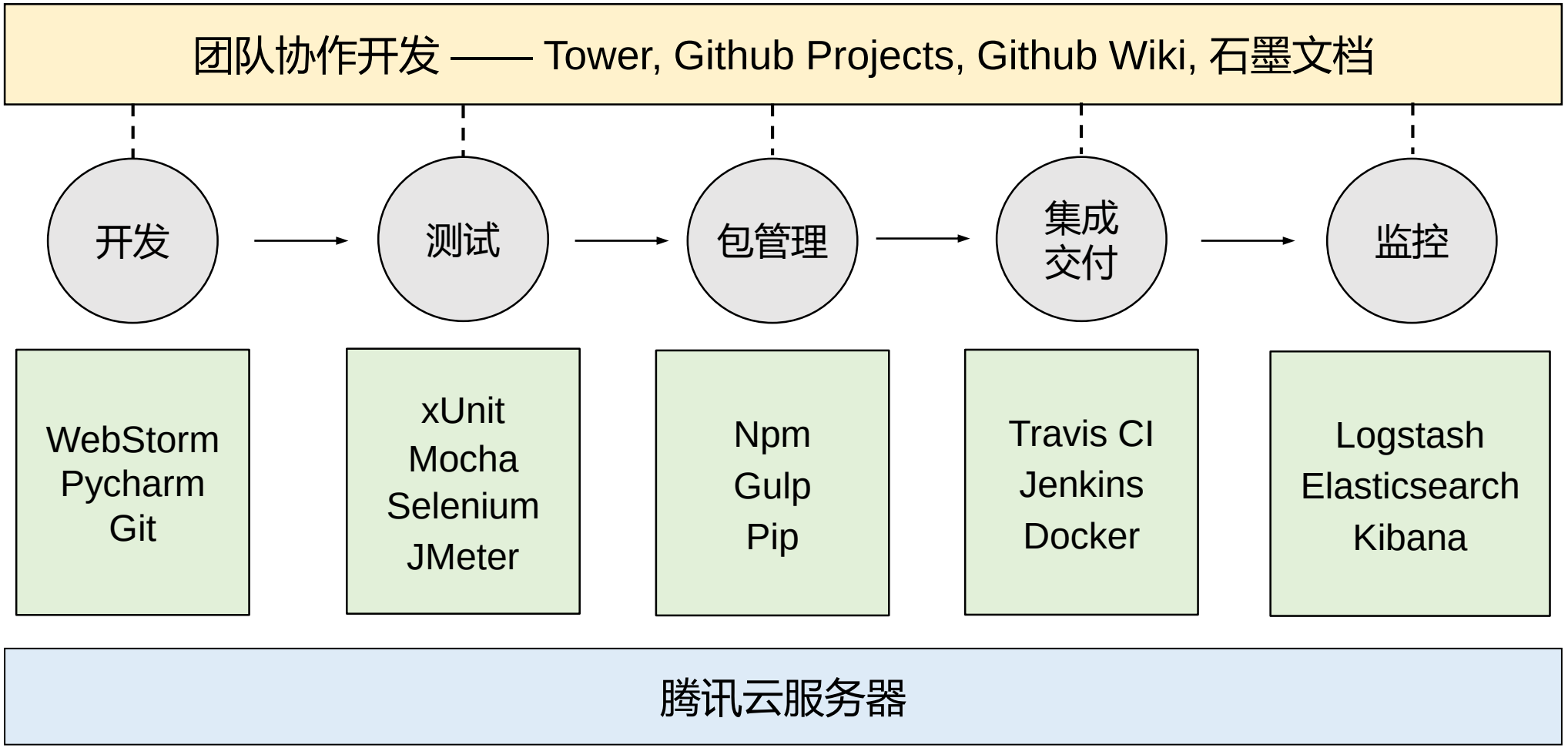

Collaboration
Programmable Infrastructure
Source Code
Revision Control System
Automated Build
Automated Test
Continuous Integration
Automated Documentation
Automated QA
Automated Deployment
Monitoring Notifications Reporting

DevOps

- **编码**：代码开发和审阅，版本控制工具、代码合并工具
- **构建**：持续集成工具、构建状态统计工具
- **测试**：通过测试和结果确定绩效的工具
- **打包**：成品仓库、应用程序部署前暂存
- **发布**：变更管理、发布审批、发布自动化
- **配置**：基础架构配置和部署，基础架构即代码工具
- **监视**：应用程序性能监视、最终用户体验


开发环境
版本控制系统
持续交付环境
测试环境
预发布环境
生产环境
特性/bug
创建分支
编码
单元测试
代码扫描
(本地)
变基/合并
提交代码
Push
缓存区/
特性分支
合并请求
Code
Review
通过
正式代码库
Trigger
私有构建
编译
(私有)
单元测试
(私有)
代码扫描
(私有)
私有构建
报告
集成构建
编译
(集成)
单元测试
(集成)
代码扫描
(集成)
打包
代码质量
报告
集成构建
结果
正式代码库
安装包
编写配置脚本
提交配置脚本
缓存区
正式环境库
配置脚本
自动化部署
创建
配置环境
安装
配置中间件
配置数据库
部署/配置
应用
基础设施
冒烟测试
数据库
冒烟测试
应用
冒烟测试
自动化运维
监控预警
备份
APM
环境配置
缩容扩容
服务
自动化功能测试
手工验收测试
性能测试
安全测试
探索式测试
用户使用

DevOps工具链


团队协作开发 —— Tower, Github Projects, Github Wiki, 石墨文档
开发
测试
包管理
集成
交付
监控
WebStorm
Pycharm
Git
xUnit
Mocha
Selenium
JMeter
Npm
Gulp
Pip
Travis CI
Jenkins
Docker
Logstash
Elasticsearch
Kibana
腾讯云服务器

Travis CI

Travis CI 是在软件开发领域中一个在线的、分布式的持续集成服务，用来构建及测试在GitHub托管的代码。

https://travis-ci.org/

- 目前大多数github项目都已经移入到Travis CI的构建队列中，据说Travis CI每天运行超过4000次完整构建
- 支持多种编程语言，包括Ruby、JavaScript、Java、Scala、PHP、Haskell和Erlang等
- 免费支持Github项目，通过yml语法驱动执行，开通Travis后只需编写.travis.yml就能完成持续集成，简洁清新而且独树一帜

Jenkins

Jenkins是一个开源的、提供友好操作界面的持续集成工具，主要用于持续和自动地构建与测试软件项目、监控外部任务的运行。

https://jenkins.io/

易安装：只有一个 java -jar jenkins.war，从官网下载该文件后直接运行，无需额外的安装，更无需安装数据库

易配置：提供友好的GUI配置界面

测试报告：以图表等形式提供详细的JUnit/TestNG测试报表

分布式构建：把集成构建等工作分发到多台计算机中完成

文件识别：跟踪每次构建生成哪些jar包以及使用哪个版本jar包

插件支持：通过第三方插件扩展，使其变得更加强大

教学提纲

1. 软件配置管理
2. 持续集成与交付
3. 实验：微信抢票应用开发

校园委活动排队领票

参加学校举办的活动和演出，学生总是需要排队买票或领票

烦！

- WebApp
- 手机App
- 微信应用

目标：微信抢票应用

爽！

微信抢票应用 V1.0


中国移动 01:15
微信
清华大学紫荆之声
01:15
“紫荆之声”使用指南
11月13日
不想错过园子里精彩的资讯？又没时间没心情到处搜罗信息？想要参加高大上的活动却不想提前数小时排队？微信“紫荆之声”帮您便捷解决这些问题！快来看看“紫荆之声”怎么使用吧！
您尚未绑定学号，回复“绑定”进行相关操作:)
阅读全文
资讯
团聚清华
服务


中国移动 上午11:05
返回
清华紫荆之声
距抢票开始还有:
00 天 00 小时 04 分 26 秒
活动预计发票:
10 张
抢票时间:
2013年12月17日 11时10分 至 12时00分
活动时间:
2013年12月18日 00时00分 至 00时00分
活动地点:
六教
取票代码:
软件工程
活动详情:

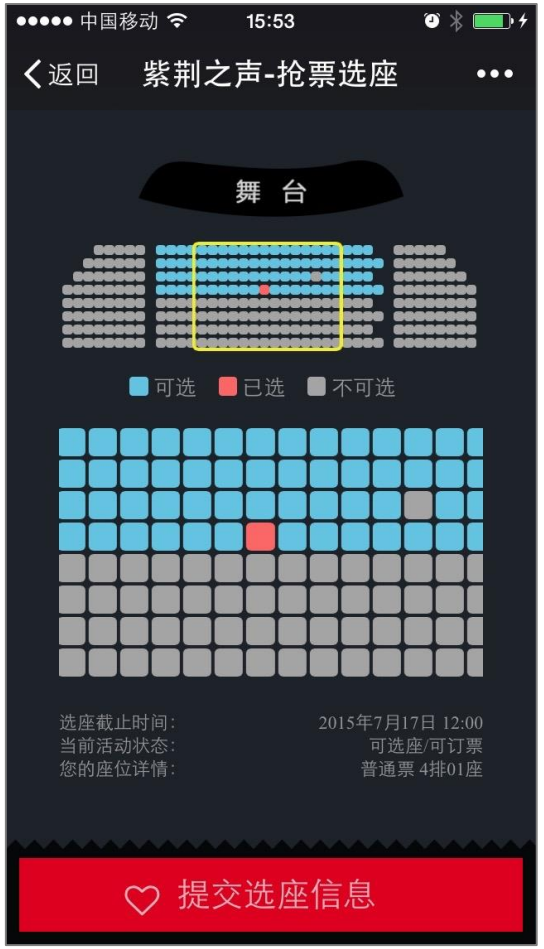

中国移动 上午11:15
微信
清华紫荆之声
测试
阅读全文
上午11:15
抢票 软件工程
对不起，尚未绑定账号，不能抢票，点此绑定信息门户账号
资讯
服务
抢票


中国移动 上午11:16
微信
清华紫荆之声
抢票 软件工程
软件工程
12月17日
活动时间：2013-12-18 00:00
活动地点：六教
回复“退票 软件工程”即可退票
阅读全文

微信抢票应用 V2.0


中国移动 15:52
微信
软件工程
芭蕾舞剧《吉赛尔》
7月9日
舞剧《吉赛尔》是浪漫主义芭蕾舞剧的代表作，有“芭蕾之冠”的美誉，由简·克拉里、朱尔·佩罗和马里乌斯·彼季帕共同创作，取材于德国诗人海涅（1787-1856）的《自然界的精灵》与法国作家维克多·雨果《东...
阅读全文
资讯
抢票
个人中心


中国移动 15:52
返回
芭蕾舞剧《吉赛尔》-...
Tsinghua University
芭蕾舞剧《吉赛尔》
距抢票结束还有:
7天20小时8分
活动时间: 7月30日 19:00 – 21:00
活动地点: 新清华学堂
剩余票数: 85 张
活动介绍
舞剧《吉赛尔》是浪漫主义芭蕾舞剧的代表
抢票方式:
7月7日 11:45 – 17日 12:00 线上抢票，先到先得!
· 在微信中直接回复 "抢票 吉赛尔"
· 或者，点击"抢票"菜单下的 "吉赛尔" 按钮


中国移动 15:53
返回
紫荆之声-抢票选座
舞 台
可选
已选
不可选
选座截止时间: 2015年7月17日 12:00
当前活动状态: 可选座/可订票
您的座位详情: 普通票 4排01座
提交选座信息

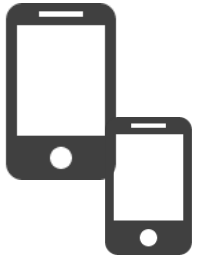

中国移动 15:53
返回
我的票夹
我的电子票
票号：2015337827182413
芭蕾舞剧《吉赛尔》
等待支付
场馆：新清华学堂
日期：2015年7月30日 19:00
座位：4排01座
退票方式：回复 '退票 吉赛尔'
票价：30元
换票方式： 凭二维码支付并换取纸质票
请到活动指定地点领票

实验要求

- 前后端开发
  - 在给定框架的基础上进行后端开发（https://github.com/ThssSE/WeChatTicket）
  - 建议遵循给定的前后端接口进行后端开发
  - 修复可能发现的前端缺陷
- 系统测试
  - 对后端代码和业务逻辑进行单元测试和功能测试（至少覆盖核心抢票功能）
  - 针对数据库设计及业务逻辑实现，对系统进行性能优化和提升（可选）
- 持续集成和交付
  - 运用敏捷开发和持续集成的方法（Github + Travis CI）
  - 所开发的微信抢票应用上线部署和交付

微信公众号应用开发

微信 | 公众平台



客户端

发送请求

回复消息

微信服务器

ToUserName 消息接收方微信号，一般为公众平台账号微信号
FromUserName 消息发送方微信号
CreateTime 消息创建时间
MsgType 消息类型；文本消息为text
Content 消息内容
MsgId 消息ID号

POST数据

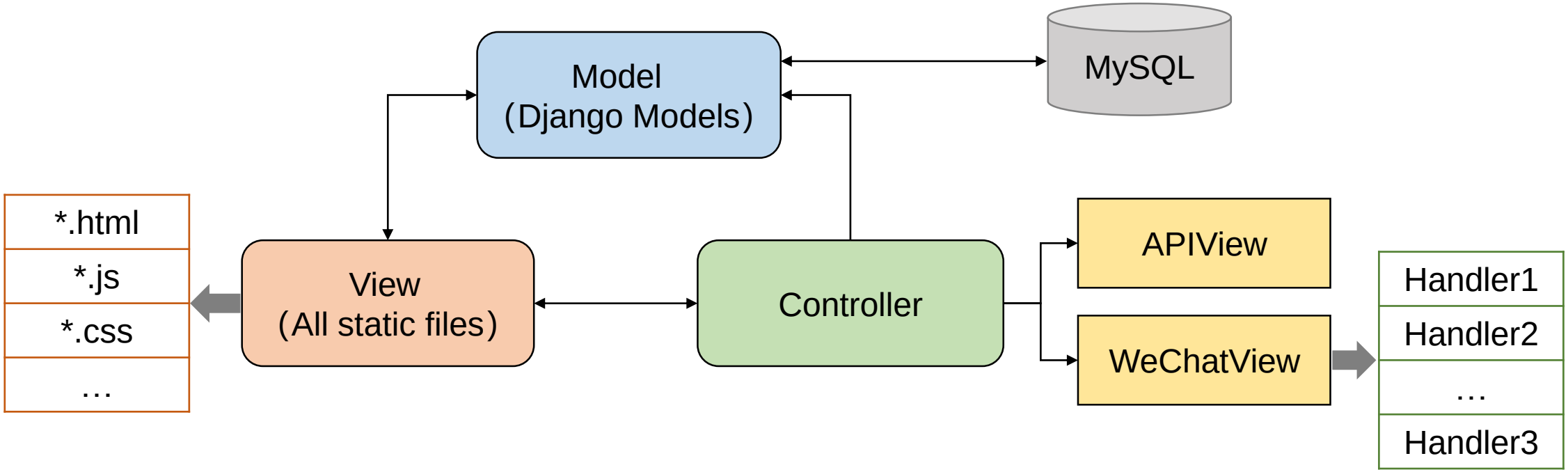
返回数据

FromUserName 消息发送方
ToUserName 消息接收方
CreateTime 消息创建时间
MsgType 消息类型，文本消息必须填写text
Content 消息内容，大小限制2048字节，字段为空为不合法
FuncFlag 星标字段

应用服务器

系统总体架构

- 基本环境：Python3、MySQL、相关Python包
- 系统采用MVC架构，整个框架基于Django1.9.x进行了二次封装，实现前后端完全分离，后端遵循 RESTful



数据库设计

票绑定在学号上，因此用户更换绑定微信号后仍能取票

User	
id	primary
open_id	unique, index
student_id	unique, index

Ticket	
id	primary
student_id	index
unique_id	unique, index
activity	foreign key
status	

Activity	
id	primary
name	
key	index
book_start	index
book_end	index
…	

优化提示：

- 外键会影响读写性能
- 索引会影响写入性能

接口说明

需要实现的两种接口：

API接口

微信接口

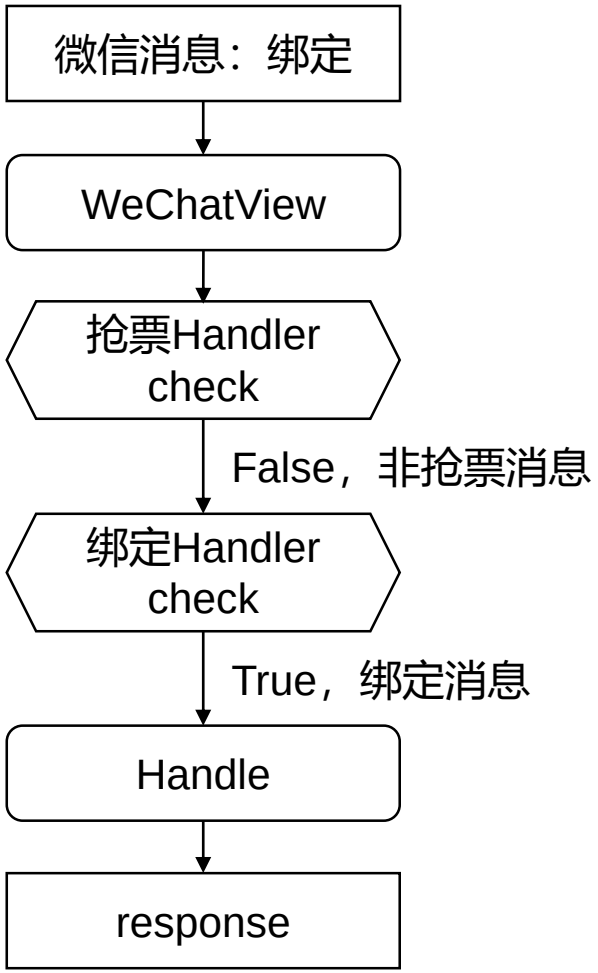
API接口

- 主要用于后台管理（活动和票的管理）、用户手机端页面与后端的交互等
- 遵循RESTful，request数据格式为JSON或form-data，response数据格式为JSON
- 具体接口的数据格式，详见《前后端接口》文档

接口说明

微信接口

- 处理微信消息，实现具体的微信交互业务逻辑
- 微信消息将由WeChatView处理，其中定义若干WeChatHandler，依次调用它们check方法，并交给第一个返回true的WeChatHandler的handle方法进行业务逻辑处理
- 开发需要增加、实现具体WeChatHandler的check和handle方法



谢谢大家！

THANKS