

QCon 全球软件开发大会 【北京站】2016

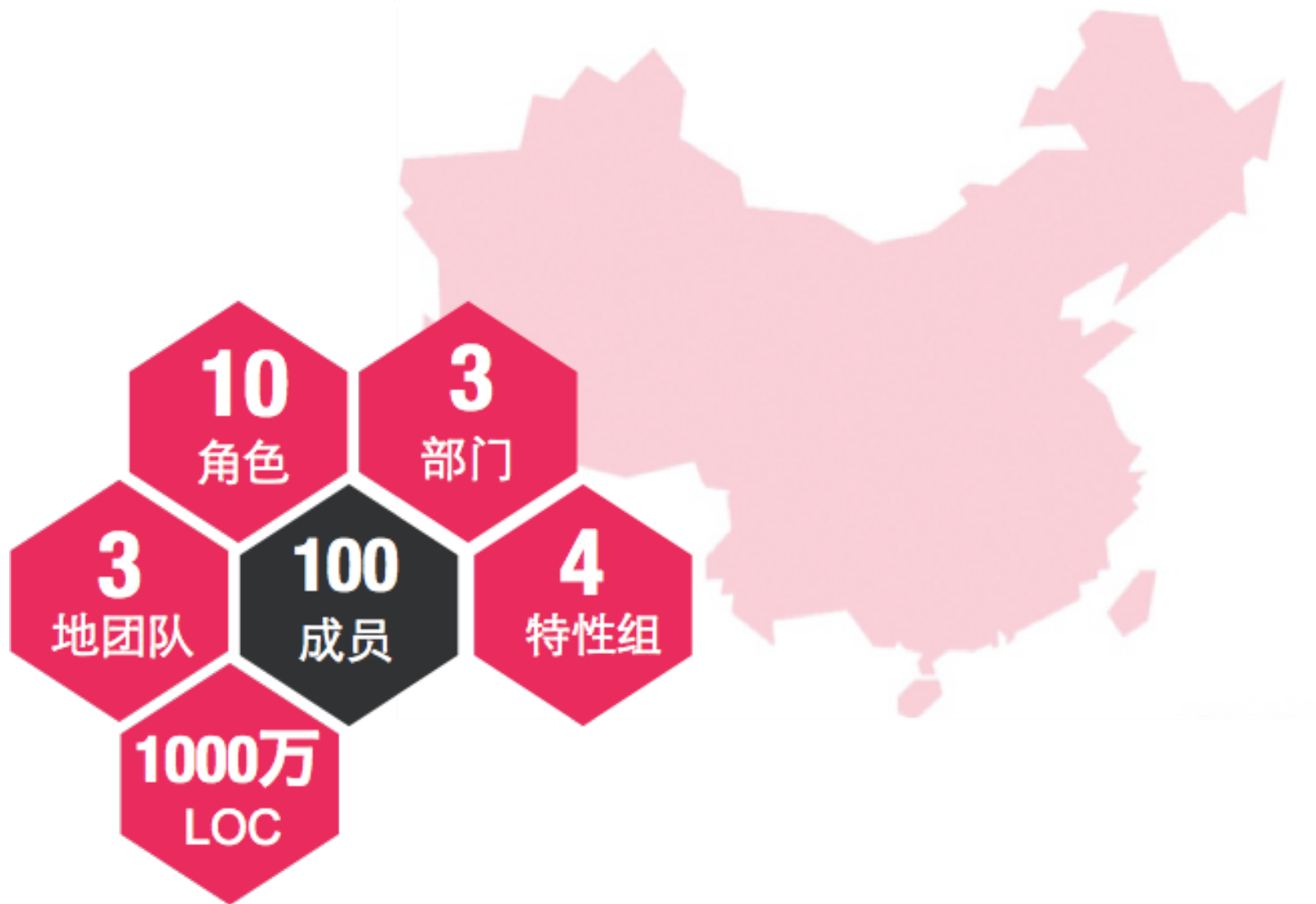
大型分布式团队中的SCM

工程效率提升/ 刘冉 ThoughtWorks

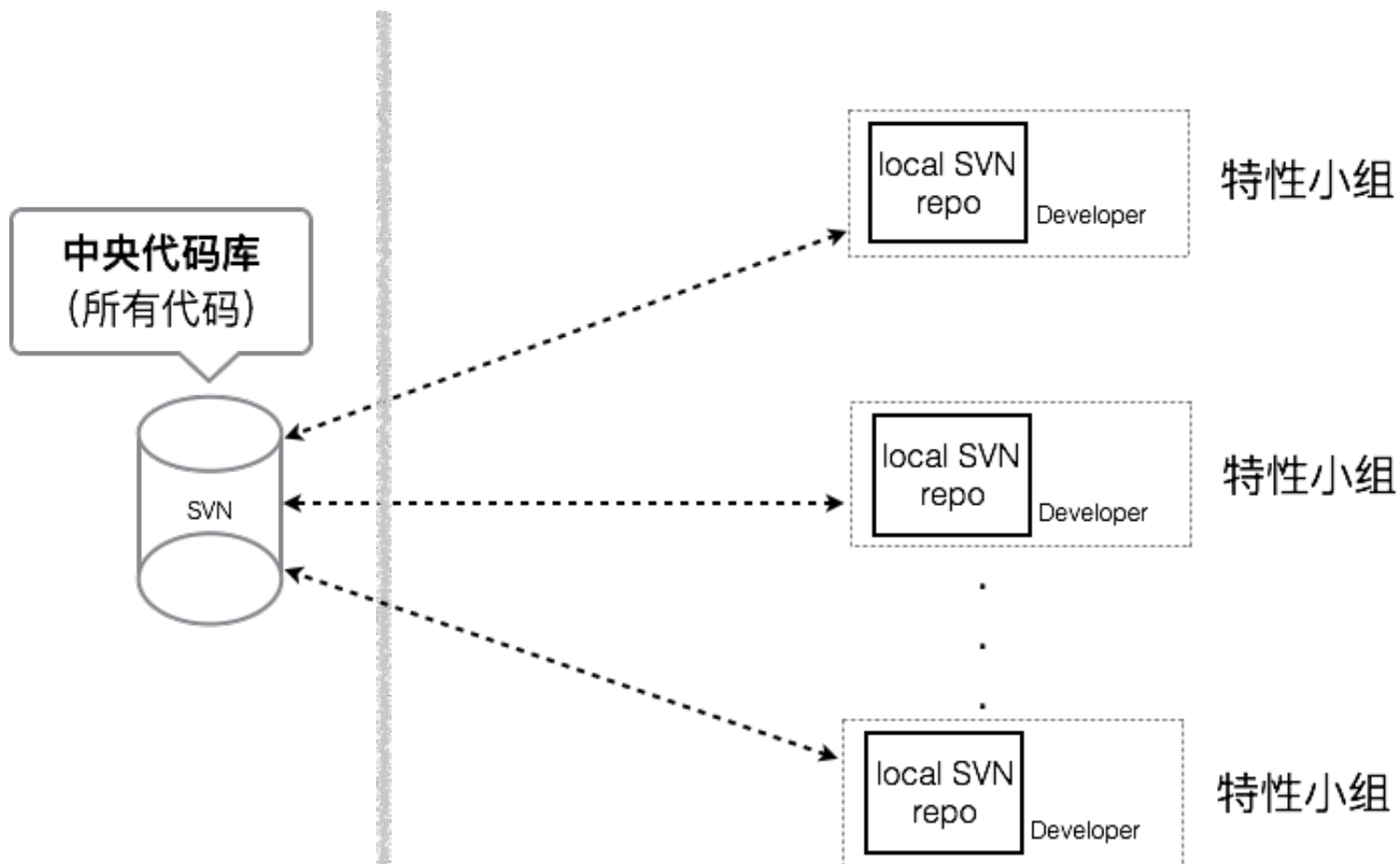
议程

- 曾经有这样一个团队
- 分布式的SCM方案
- SVN到GIT的迁移
- Android-评审和大量模块
- OpenStack-大量提交
- 通信厂商-大量产品线

曾经有这样一个团队



他们的SCM方法



他们的梦想

- 希望自己的团队只用管理自己代码库
- 希望可以快速的建立开发分支
- 希望没有网络的时候可以继续查看代码日志
- 希望能快速查看日志
- 希望不用停机备份代码库

问题和痛点

分布式团队

- 基于团队的代码模块分离困难
- 需要中心代码集成功能

服务器不可用时

- 不能查看提交记录
- 不能比较文件
- 不能提交代码

代码分支

- 分支创建速度慢
- 多分支管理困难

提交流程

- 集成Code Review
- 集成CI Review

代码庞大

- 查看日志慢
- 服务性能要求高

备份问题

- 需要停机备份
- 备份成本高

SCM的历史

1 本地式

提供本地代码版本控制，比如SCCS(1972), PVCS(1985)等

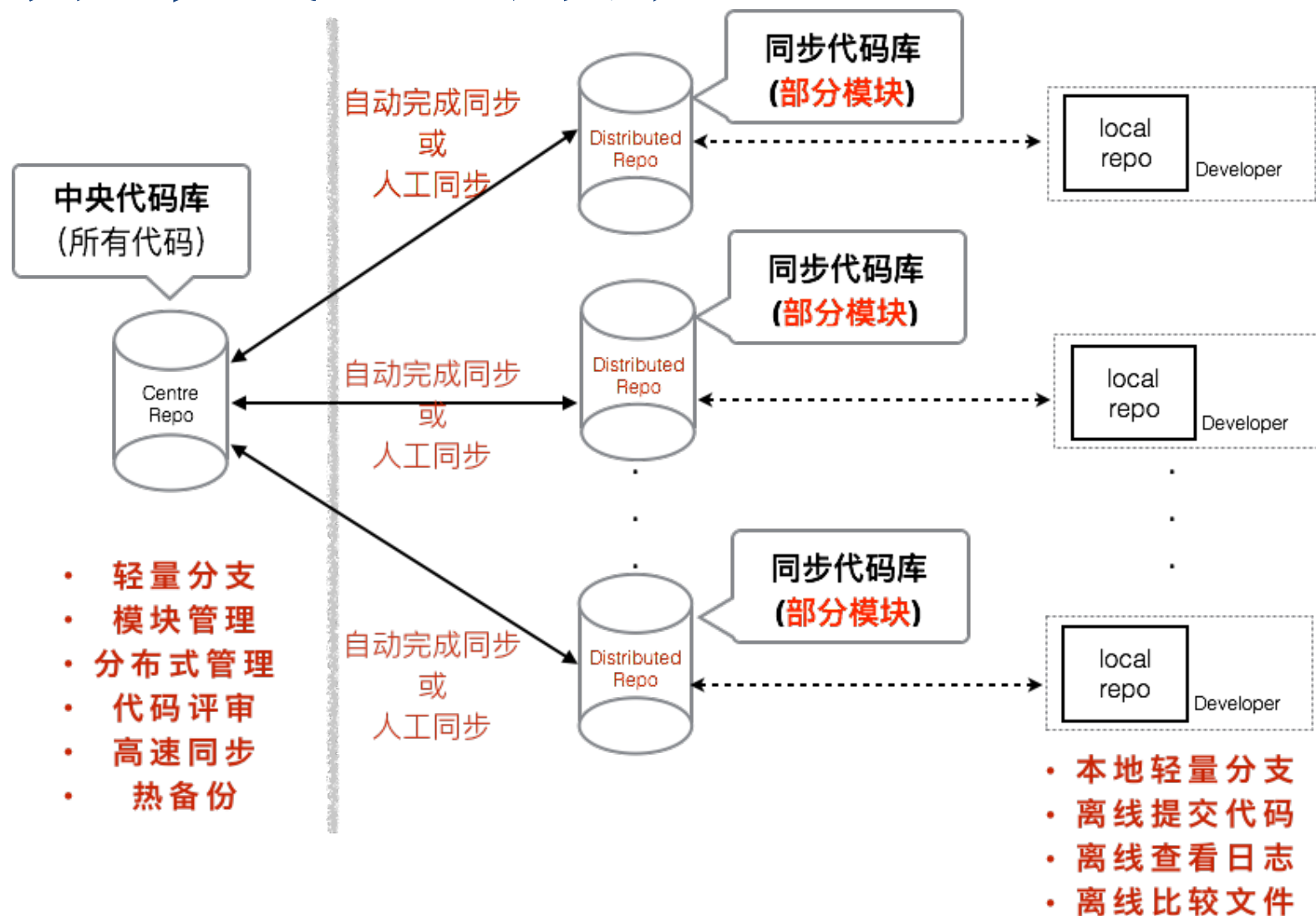
2 集中式

提供集中式服务器端代码版本控制,比如 CVS(1986), ClearCase(1992), Perforce(1995), Subversion(2000) 等

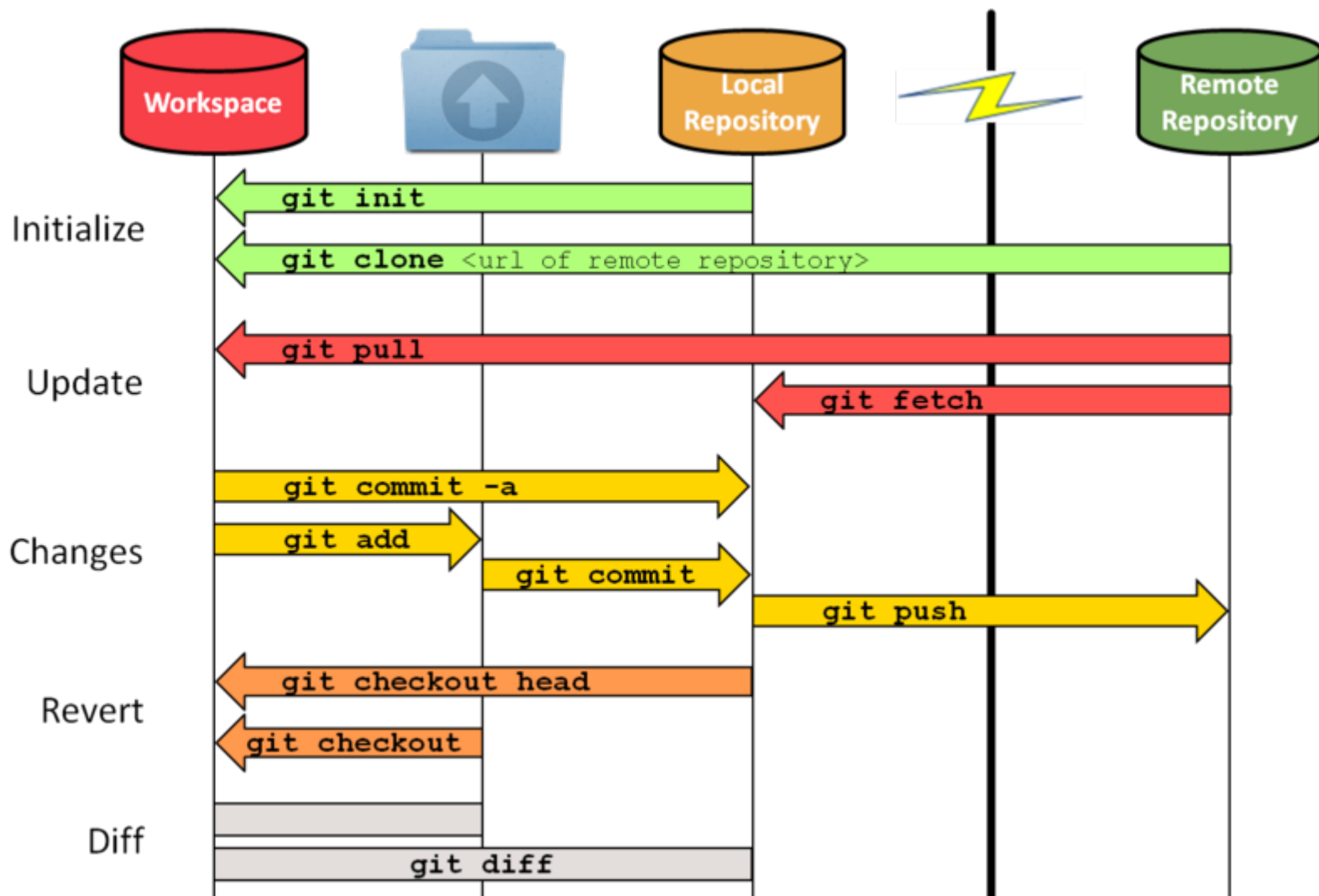
3 分布式

提供分布式代码版本控制,比如Git(2005), Mercurial(2005)等

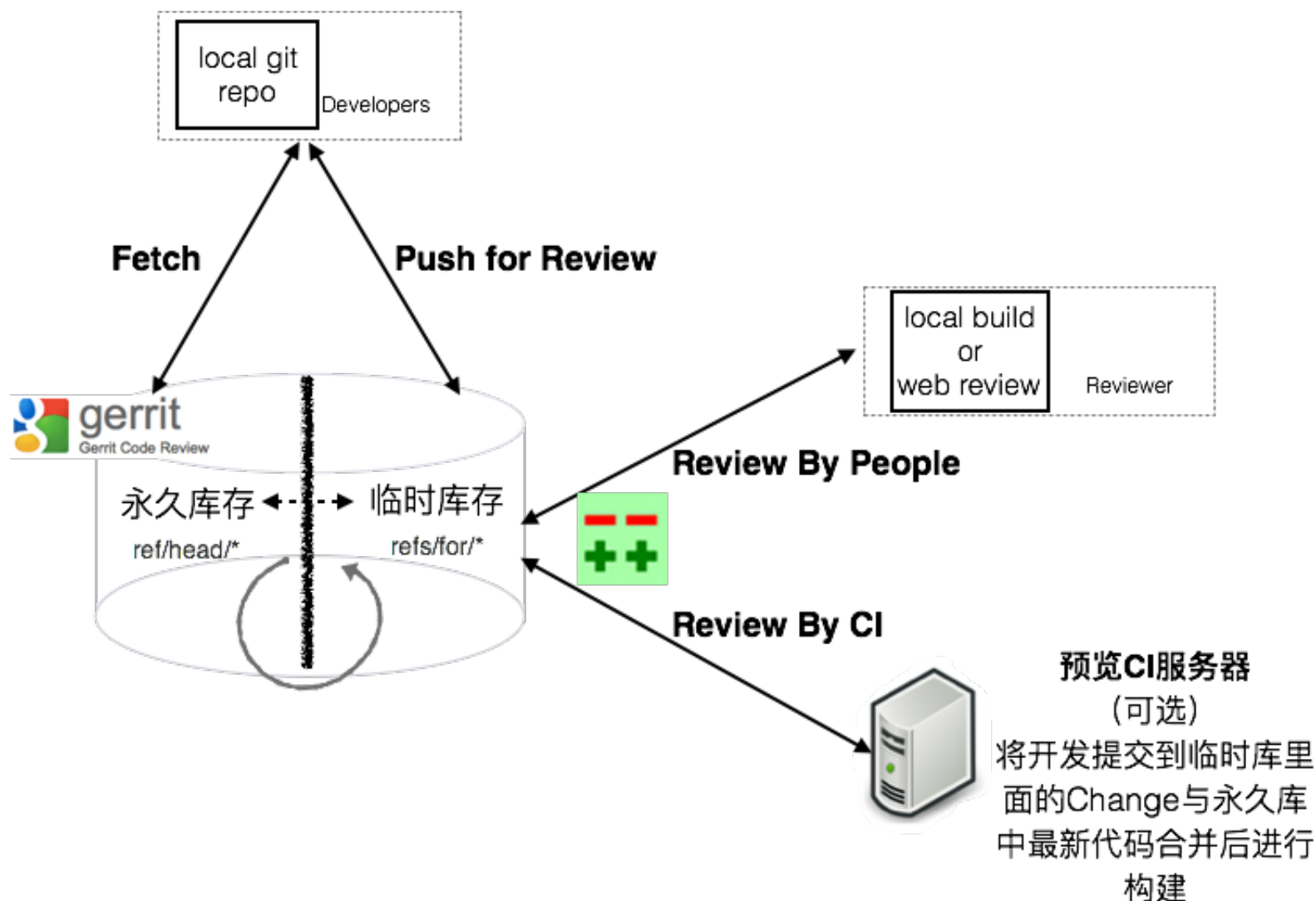
分布式SCM方案



GIT介绍



GERRIT介绍



使用GERRIT的项目



<https://android-review.googlesource.com>



<https://review.openstack.org>



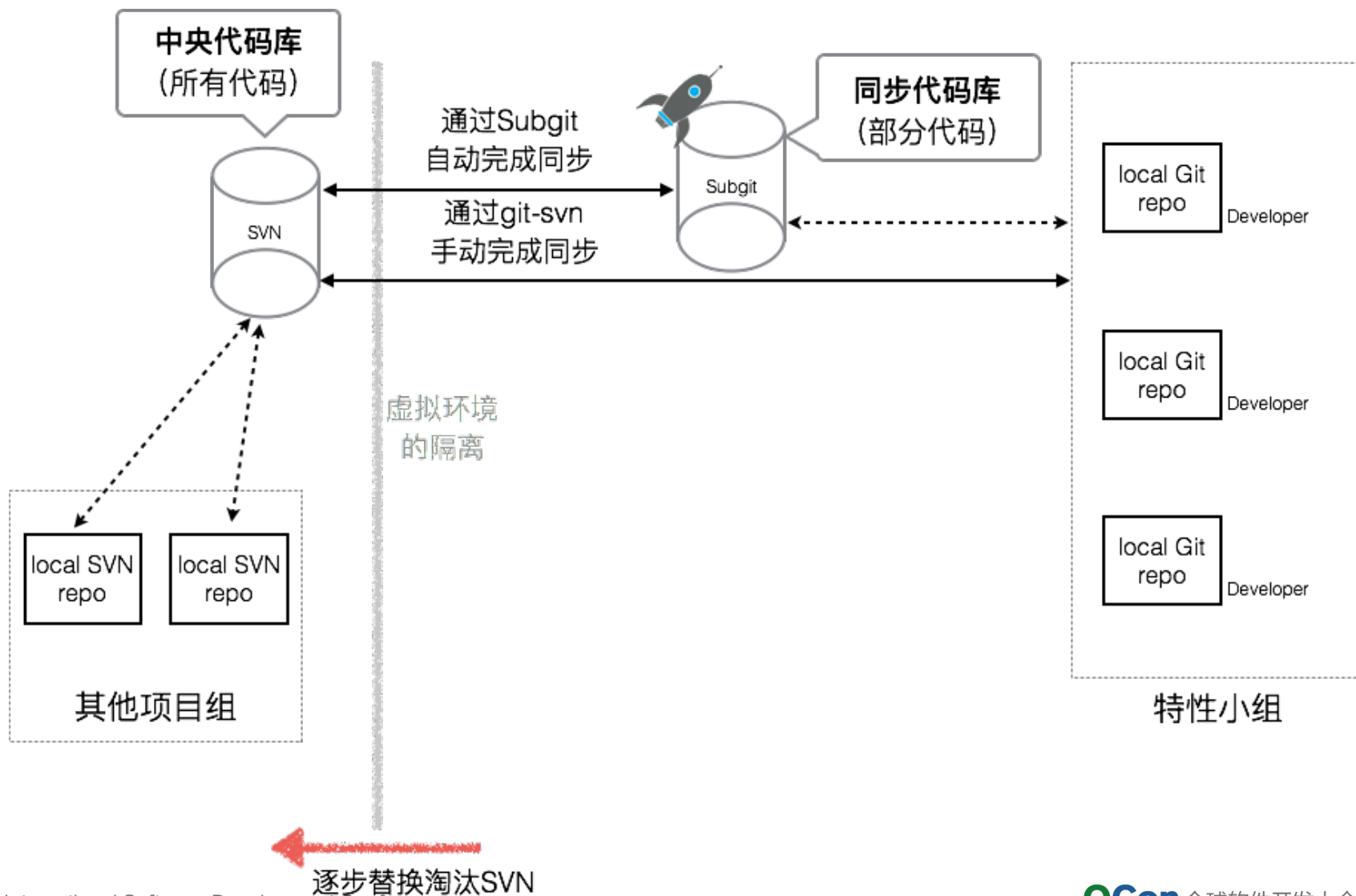
<https://codereview.qt-project.org>

SVN迁移到GIT

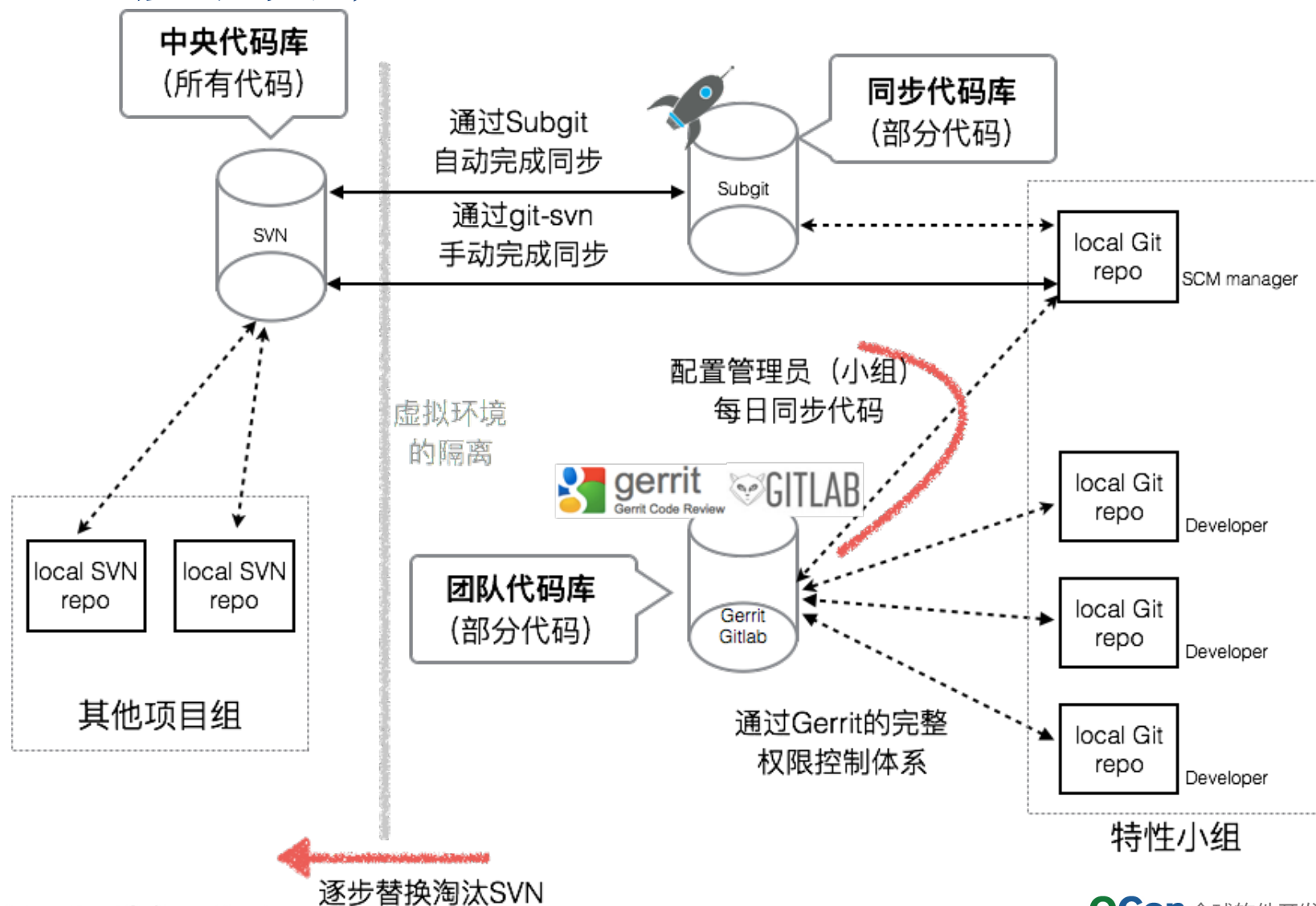


团队太多，如何逐一迁移？

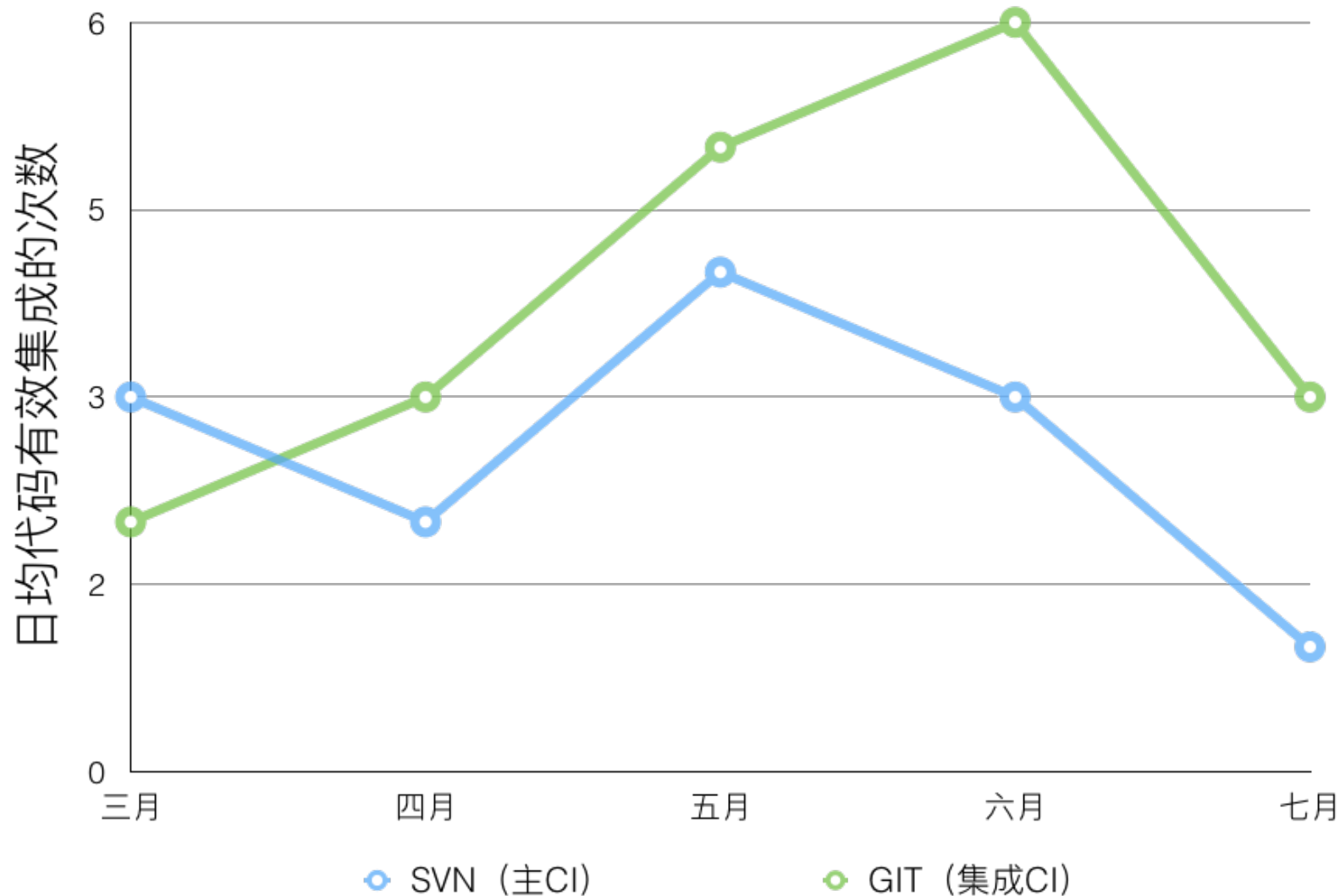
迁移方案-SVN TO GIT



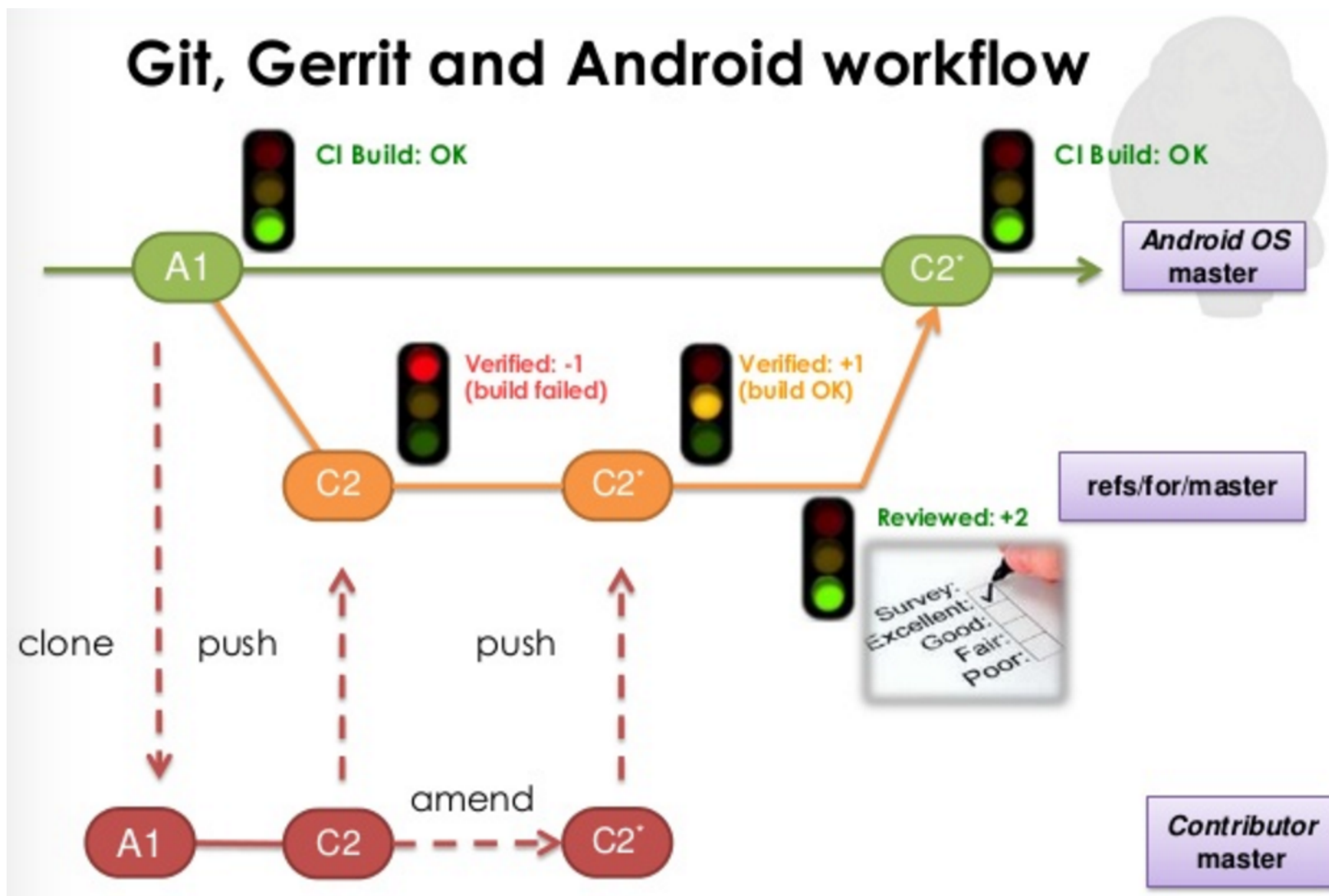
迁移方案-SVN TO GERRIT/GITLAB



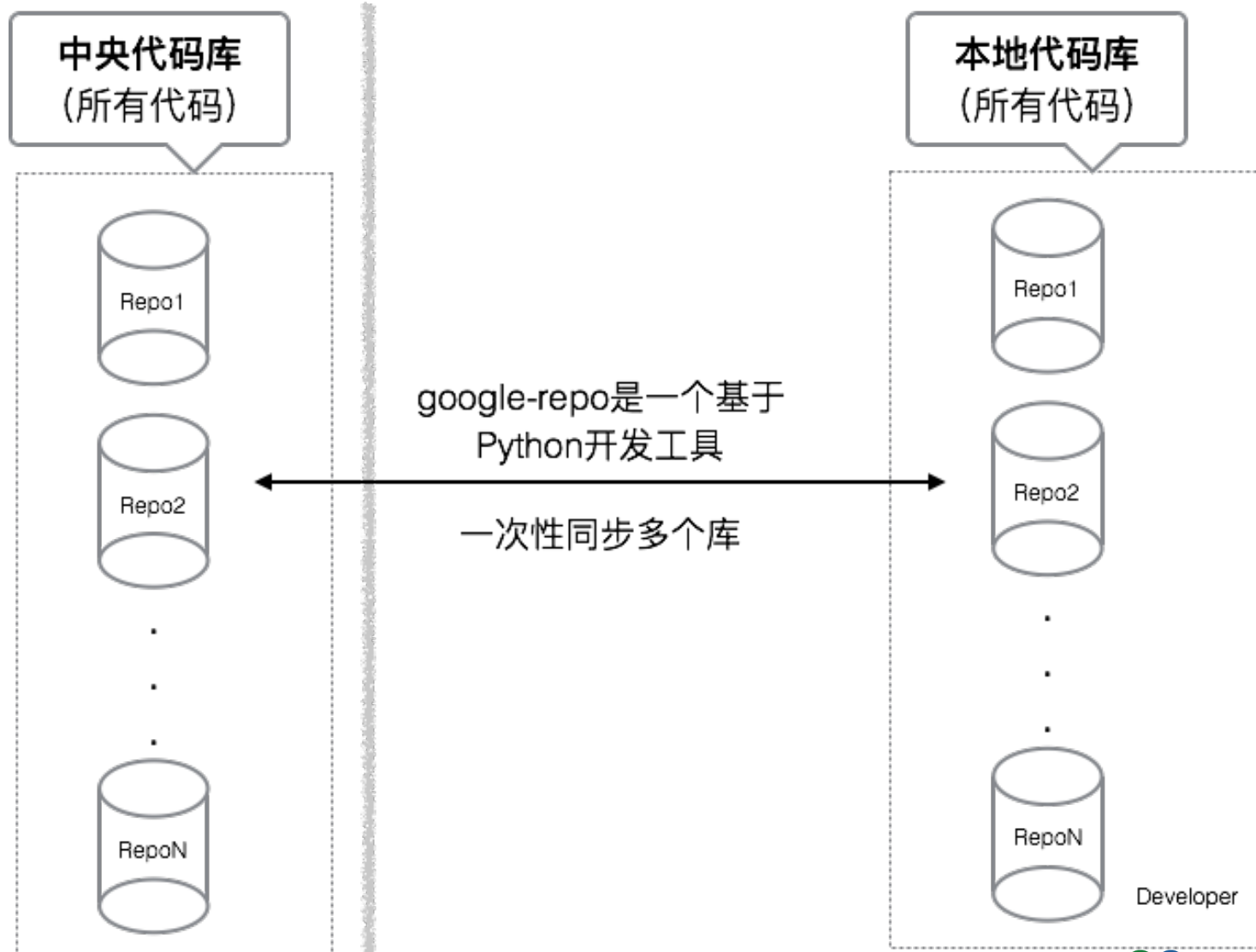
效率提升



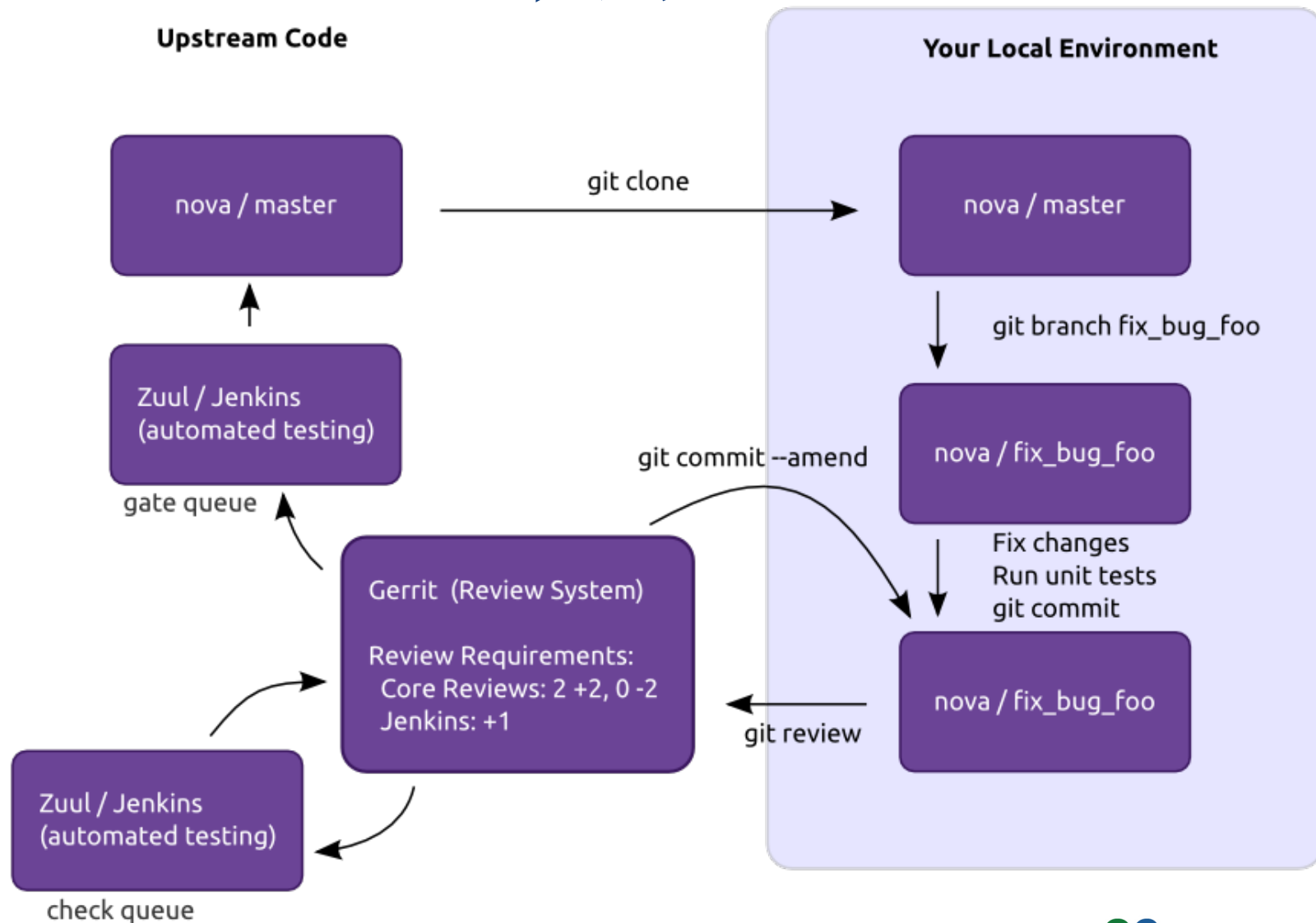
ANDROID中的GERRIT



GOOGLE-REPO介绍



OPENSTACK中的GERRIT



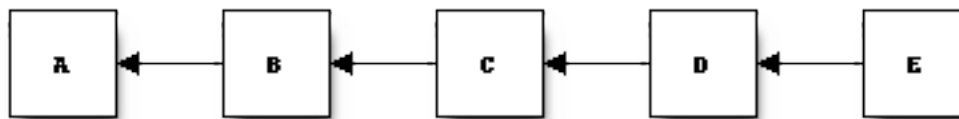
OPENSTACK-大量提交



提交和合并的时间间隔太长。

OPNESTACK-ZUUL

一个核心开发者通过了5个提交，产生合并队列：

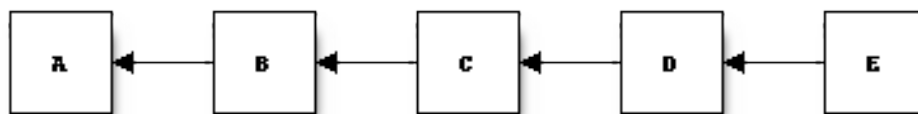


同时启动5个 *Job*，分别测试：

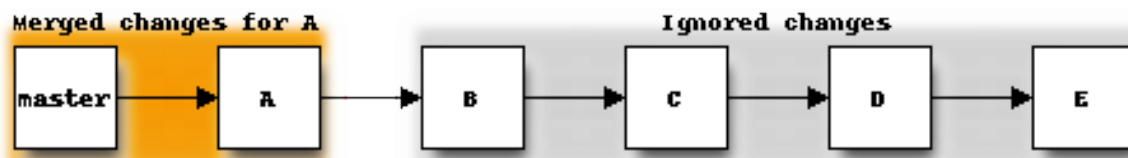
1. *A*;
2. *A + B*;
3. *A + B + C*;
4. *A + B + C + D*;
5. *A + B + C + D + E*;

OPNESTACK-ZUUL

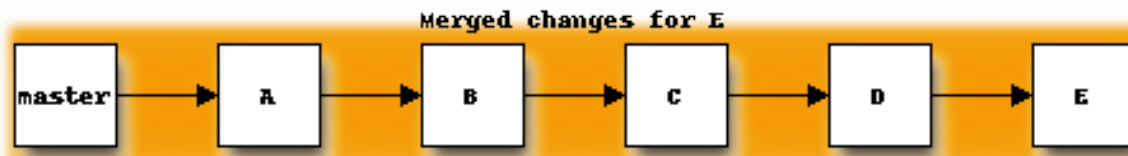
一个核心开发者通过了5个提交，产生合并队列：



测 *A* 的 *Job*，只测 *A*，不管 *B*、*C*、*D*、*E*：



测 *E* 的 *Job*，包含整个依赖链，即 *A*、*B*、*C*、*D*、*E*：

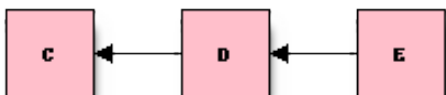


OPNESTACK-ZUUL

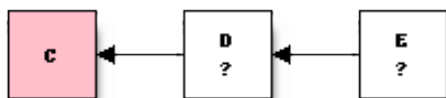
如果只有 *A*、*B* 通过测试：



队列变成：



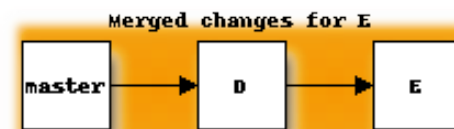
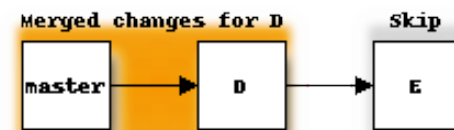
C 挂了，但 *D* 可能 *D* 是好的：



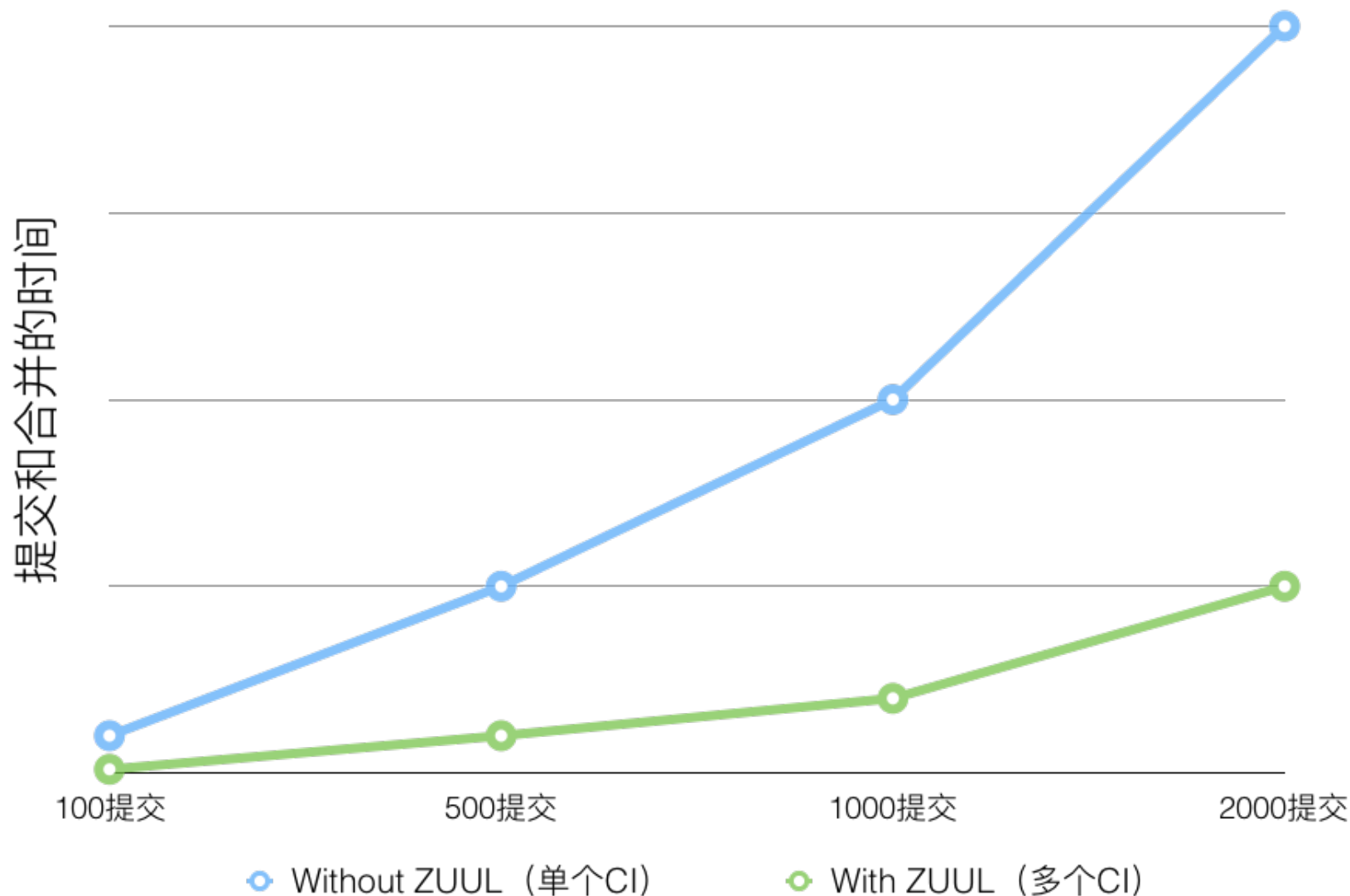
移除 *C*，并报告失败：



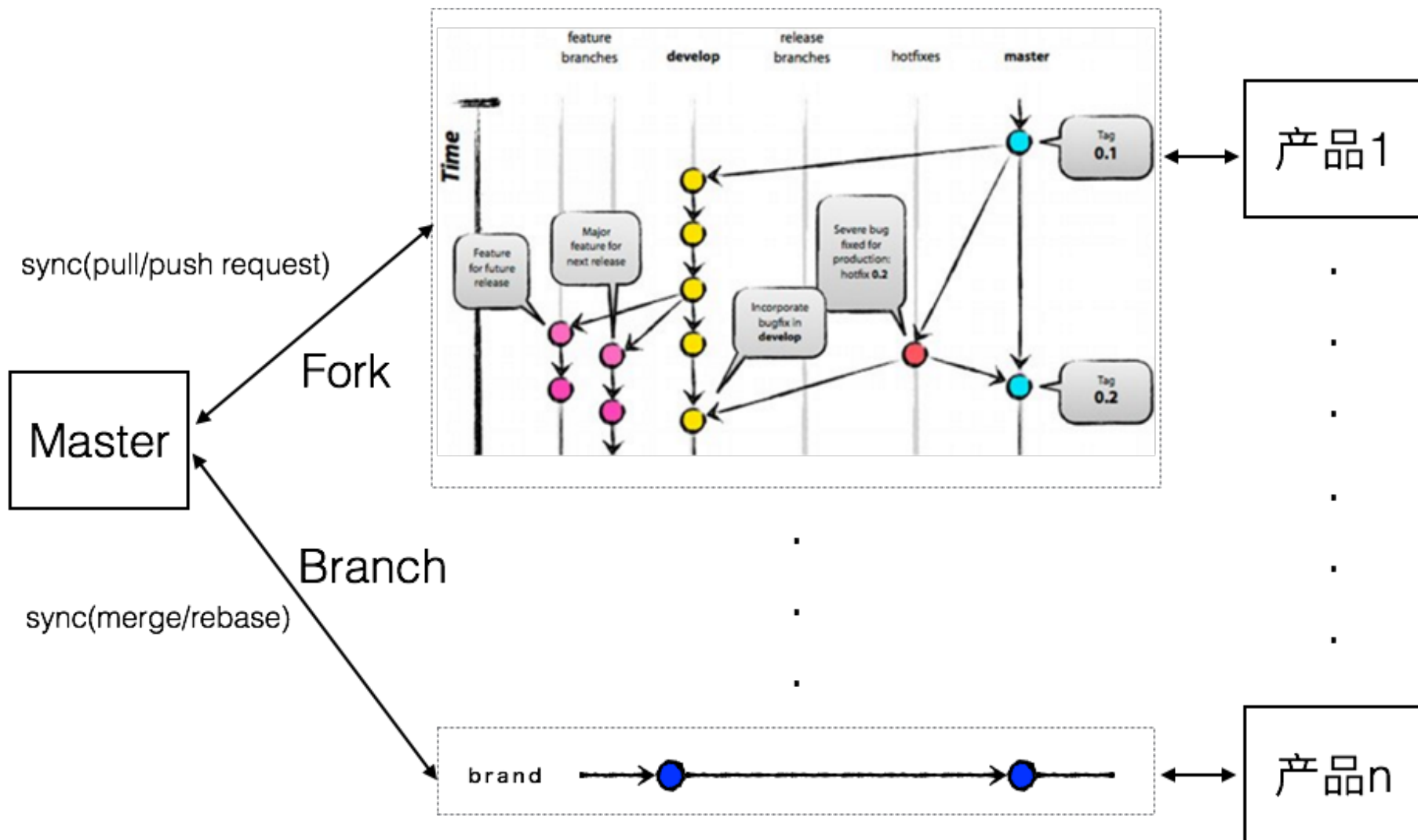
重新对队列跑测试：



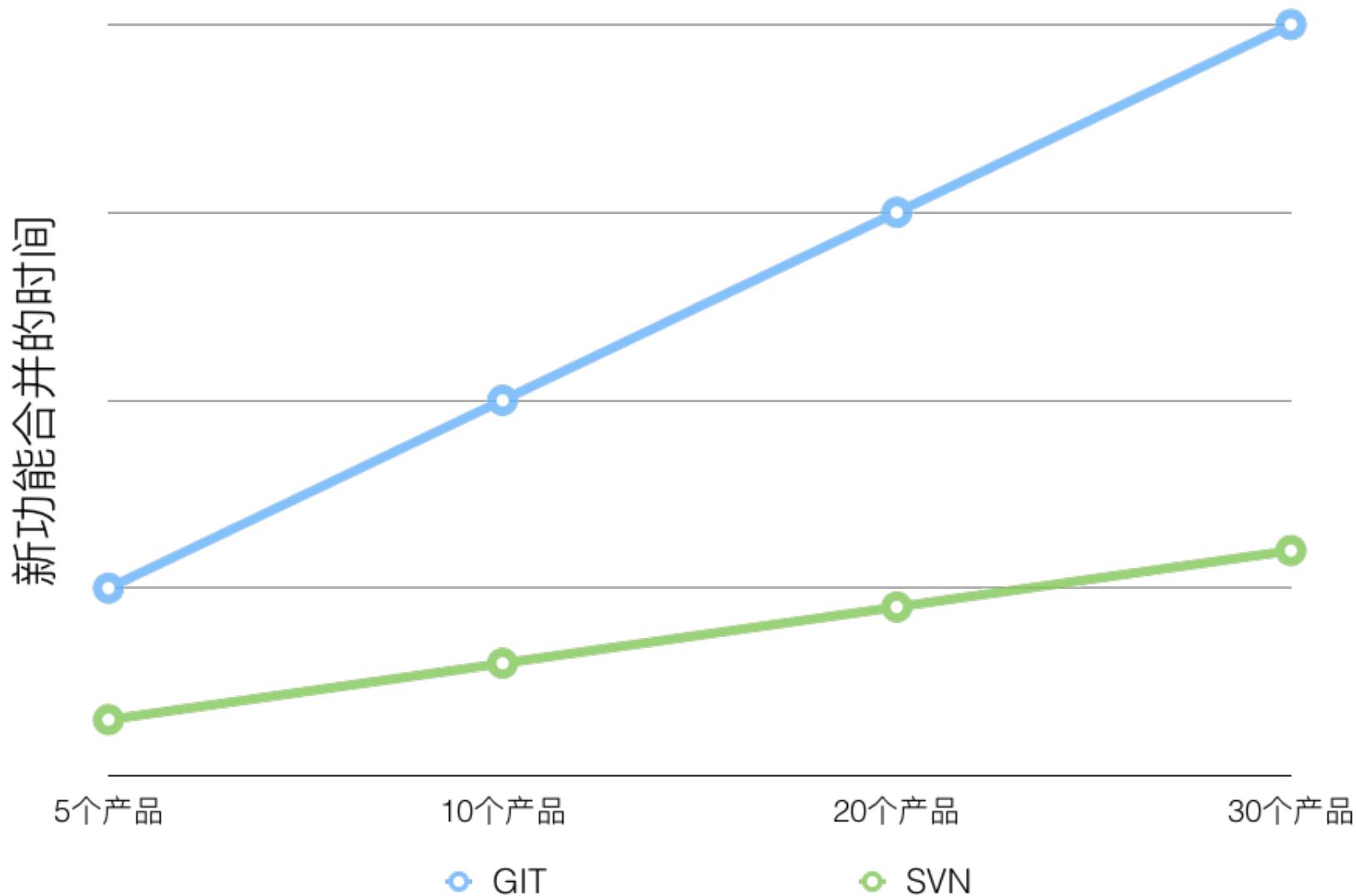
效率提升



多产品线



效率提升





THANKS!

Mail: rliu@thoughtworks.com

WeChat: liuranman

