

云原生时代的研发效能

演讲人：黄国峰 腾讯

目录

CONTENTS

1 研发效能的前世今生

2 研发效能的常规思路

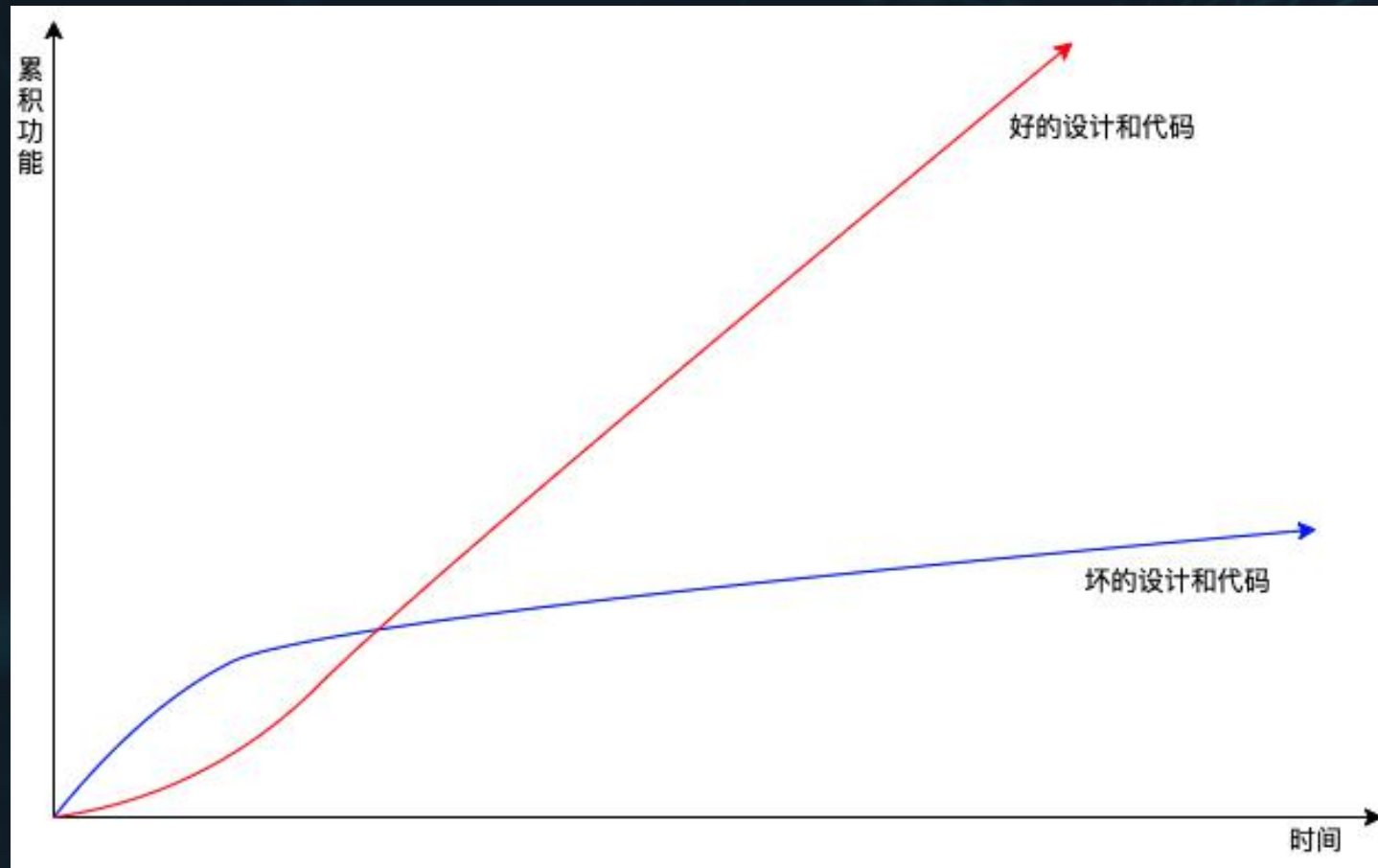
3 云原生时代的新思路

- 影响代码质量的常见问题
- 单体代码仓库
- 主干开发
- 代码评审

4 总结与未来方向

研发效能的前世今生

- 有效性 (Effectiveness)
- 效率 (Efficiency)
- 可持续性 (Sustainability)



提升研发效能的常规做法

提升研发效能的常规做法

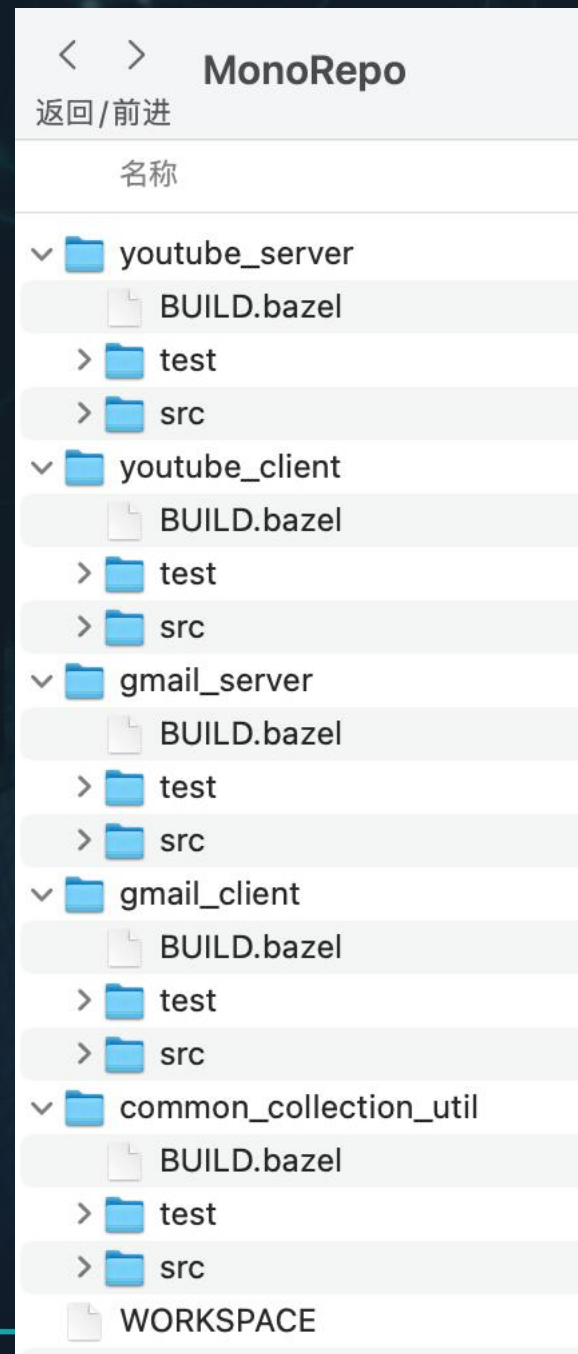
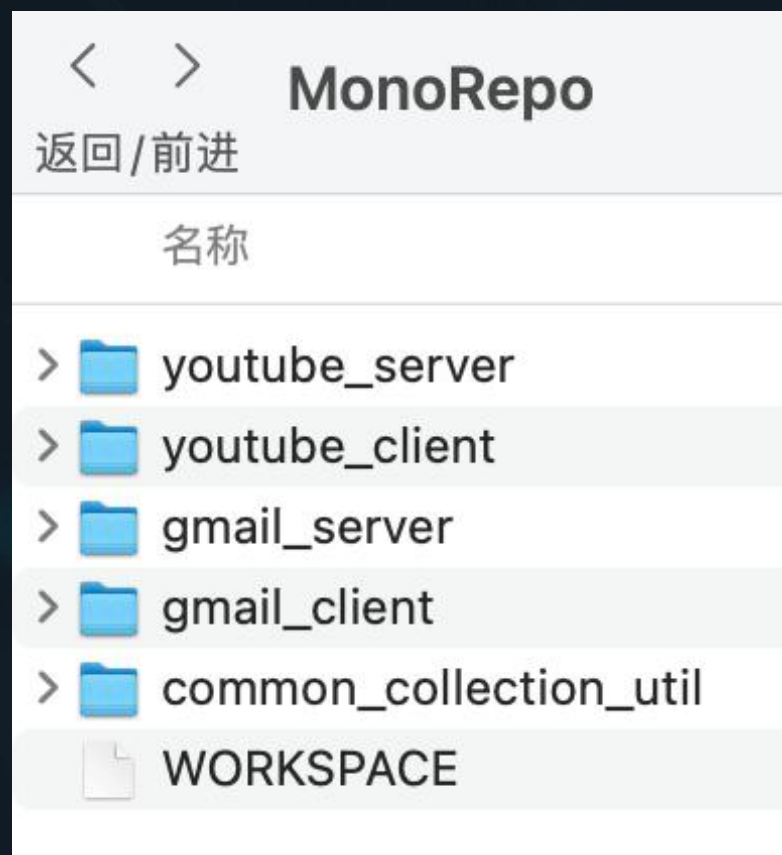
- 持续集成
- 静态代码扫描
- 自动化测试
- 持续部署

云原生时代的新思路

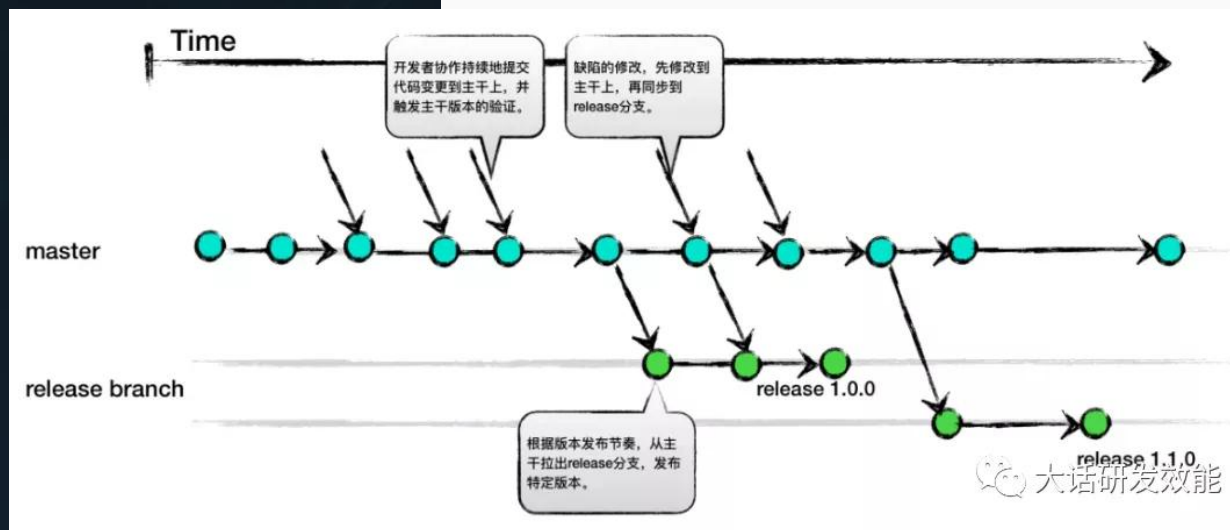
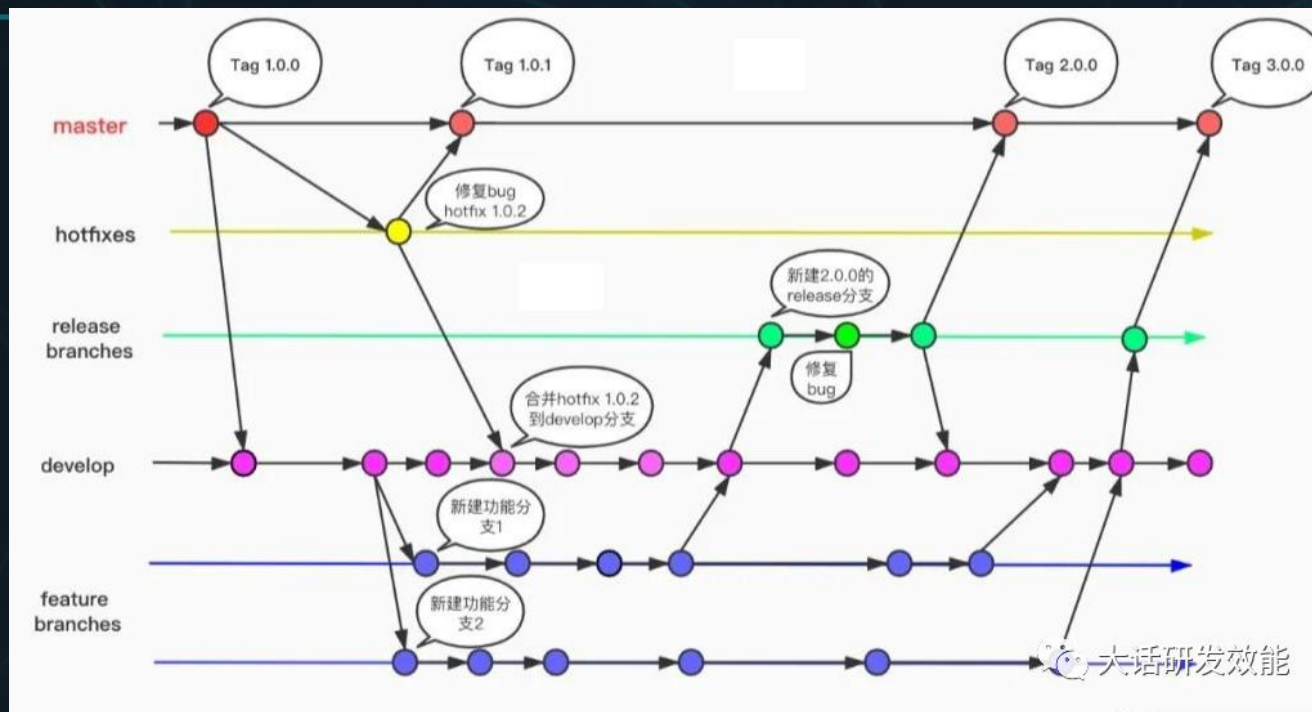
- 不规范性、不安全、不合理
 - 不一致
 - 代码复制，而不是重用
 - 功能堆砌，而不是抽象与重构
- 静态代码扫描
 - Code Review
 - 代码重复率检查、Code Review
 - 代码复杂度检查工具、Code Review

单体代码仓库

- 什么叫单体代码仓库（Monolithic Repository）？
- 有什么好处或优势？
- 什么缺点或成本？



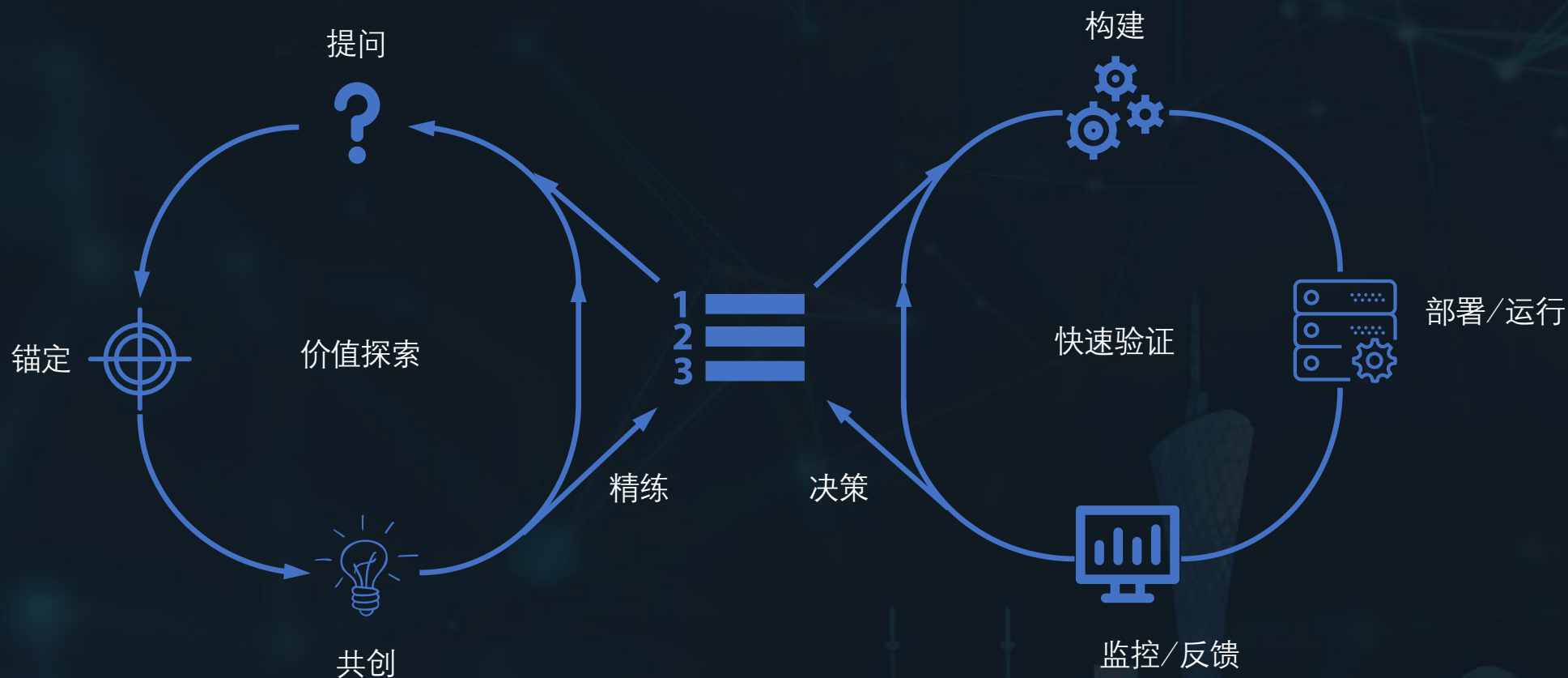
- 特性分支开发 vs. 主干开发
- 有什么优点？
- 有什么缺点或成本？



- 提升代码质量 vs. 互助传承？
- 代码评审由谁来做？
- 代码评审怎么做？
- 好的代码评审是怎样的？

总结与未来方向

总结和未来方向



持续交付“8”字环（来源：乔梁著《持续交付2.0》）

Question?

