



**QCon** 全球软件开发大会  
INTERNATIONAL SOFTWARE  
DEVELOPMENT CONFERENCE

BEIJING 2017

# 业务端技术团队真的痛？

58到家 沈剑

# 关于-我

- 前百度 - 高级工程师
- 58同城 - 高级架构师，技术委员会主席，技术学院优秀讲师
- 58到家 - 高级技术总监，技术委员会主席
- “架构师之路” 作者，深夜写写技术文章
- 本质：**技术人一枚**



# 关于-带队

- 58同城
  - 负责过即时通讯业务技术团队（后端）
  - 在支付、推荐、APP团队带过项目（超过1000人日的项目）
  - 负责过二手、心宠、优品等业务技术团队
- 58到家
  - 负责过测试平台、PMO、DBA、OP、信息系统团队
  - 负责过架构、基础服务、后端平台、中台业务等后端团队
  - 现负责支付平台、营销平台、客户平台、企业平台等业务技术团队

# 目录

- 目录

- (1) RD和PM相处之道

- (2) RD和QA相处之道

- (3) 加班怎么破

- (4) BUG怎么破

- (5) 士气怎么破

业务技术团队常见难题？

# 常见痛点

- 和PM为何不能愉快的玩耍？
- 和QA为何不能愉快的玩耍？
- 压力大、项目做不完、永远加班做业务怎么破？
- bug多、质量低、永远加班改bug怎么破？
- 人员流动率高，真的是因为没有技术含量么？

# 一、PM相处篇

# PM相处 ( 1 )

- 存在现象：目标不一致
- 案例

( 1 ) PM的绩效目标：PV增长xxx，UV增长xxx，DAU增长xxx

( 2 ) RD的绩效目标：稳定性达到4个9，千行bug率低于xxx

=>

结果是？



找PMM-leader聊一聊

达成一致的目标

# PM相处（1）

- 存在现象：目标不一致
- 解决方案：一损俱损，一荣俱荣

（1）以业务指标为第一目标

（2）统一目标，统一KPI

# PM相处 ( 2 )

- 存在现象：想做技术改进项目，PM不理解
- 案例
  - ( 1 ) 重构APP，分层抽象，沉淀公共组件，提高未来编码效率
  - ( 2 ) 重构数据访问，加入缓存，使得架构结构更合理
  - ( 3 ) 重构passport，统一第三方联合登录

=>

结果是？

目标一致，数据说话，以理服人

# PM相处 ( 2 )

- 存在现象：想做技术改进项目，PM不理解
- 解决方案：目标一致，数字说话，以理服人
  - ( 1 ) 降低xx%APP安装包大小，提高下载时间
  - ( 2 ) 降低yy%访问速度，提升用户体验
  - ( 3 ) 之前每次项目都有5人日的passport开发，一次解决，一劳永逸

# PM相处 ( 3 )

- 存在现象：研发资源永远不够，需求永远做不完
- 案例：这还需要案例？不是每个公司都这样么？

=>

结果是？

你以为加人能解决？  
相信我，人加一倍，需求照样做不完

效果说话，增强信任

# PM相处 ( 3 )

功能、项目要有收益总结，提升需求质量，建立信任关系  
预期收益可以不准，总会越来越准

- 存在现象：研发资源永远不够，需求永远做不完
- 解决方案：效果说话，增强信任
  - ( 1 ) 合适的pm与rd比例：信息系统1:15，用户产品呢？
  - ( 2 ) 提升pm需求质量：看看你们都做了一些什么，数据说话
  - ( 3 ) 需求必须有预期收益，需求有优先级：投入产出比衡量优先级
  - ( 4 ) 需求来源收口，统一对接人：产品，运营，兄弟部门产品，兄弟部门rd都来对接，压力大



## 二、QA相处篇

# QA相处 ( 1 )

- 存在现象：目标不一致
- 案例
  - ( 1 ) RD的绩效目标：千行bug率低于xxx
  - ( 2 ) QA的绩效目标：千行bug测出率高于xxx

=>

结果是？

建议：业务团队统一目标

PM、RD、QA、FE、UI

# QA相处 ( 2 )

- 存在现象：提测质量低，可测性差，bug研发搞不定，性能低下
- 常听见的问题
  - ( 1 ) 自测了么？准入通过了么？
  - ( 2 ) 没有接口说明怎么做接口测试？
  - ( 3 ) bug？我这里好的，你重启试试
  - ( 4 ) 我知道性能不达标，哪里不达标，我改

=>

结果是？

目标一致，分工明确，数字说话

# QA相处 ( 2 )

- 存在现象：提测质量低，可测性差，bug研发搞不定，性能低下
- 能达成一致么？RD为质量负责，QA辅助
- 常听见的问题
  - ( 1 ) 自测了么？准入通过了么？ -> 达成一致，必须通过准入
  - ( 2 ) 没有接口说明怎么做接口测试？ -> 需要引入设计，后面说
  - ( 3 ) bug？我这里好的，你重启试试 -> 复现说明，RD解决
  - ( 4 ) 我知道性能不达标，哪里不达标，我改 -> RD为性能负责
- 质量月报，有理有据

# QA相处 (2)

业务线	需求						
	产品需求	技术改进	BUG修复	上线需求总数	需求总数	需求完成率	测试超5人天需求数
家政	34	8	0	42	42	100.00%	4
丽人	17	4	0	21	21	100.00%	6
中台	10	4	9	23	23	100.00%	5
速运	28	10	1	39	40	97.50%	2
到家app	17	1	0	18	18	100.00%	1
第三方平台	15	1	1	17	17	100.00%	3
新业务	10	5	0	15	15	100.00%	3
后台	15	2	1	18	18	100.00%	2
合计	146	35	12	193	194	99.48%	26

- 数据说话，有理有据

业务线	上线项目数	线下Bug数	每个项目产生线下Bug数
家政	42	304	7.24
丽人	21	173	8.24
中台	23	14	0.61
速运	39	103	2.64
到家app	18	170	9.44
第三方平台	17	279	16.41
新业务	15	143	9.53
后台	18	342	19.00
合计	193	1528	7.92

业务线	线上bug数
家政	0
丽人	0
中台	9
速运	0
到家app	1
第三方平台	1
新业务	0
后台	1
总计	12

找到主要矛盾，针对改进



# 三、加班篇

业务压力大  
项目完不成  
加班严重

# 项目为什么延期？

- 实际项目是怎么执行的

- (1) 需求评审 -> PM找RD要排期

- (2) RD根据经验拍排期，PM不信任argue，讨价还价后确定排期

- (3) RD按照排期执行，完不成

- (4) RD反馈，不能如期完成，PM说“排期是你排的”

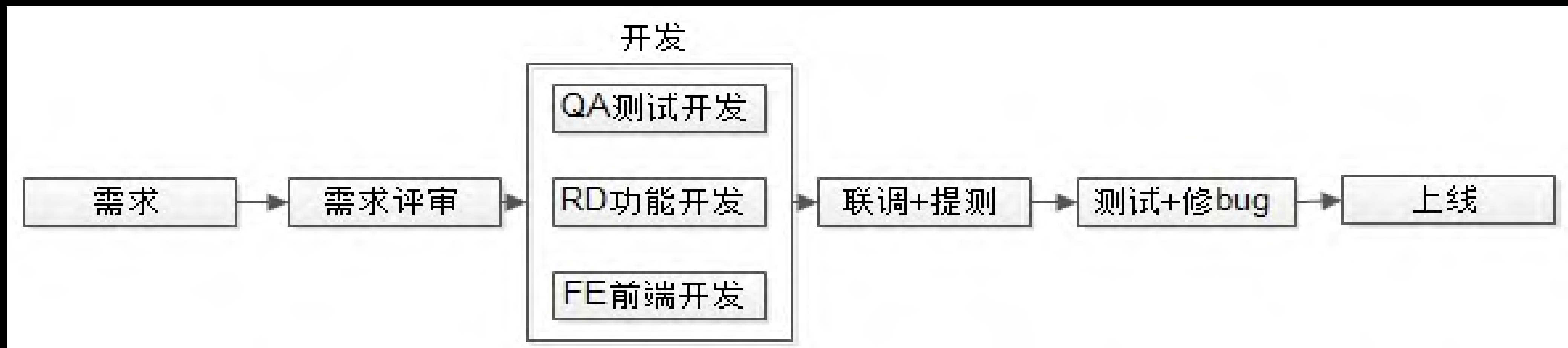
- (5) PM向老大反馈，RD不给力

- (6) RD加班

- ...

- (N) 多轮过后，离职

# 存在什么问题？



根据经验拍的排期真的准么？

更可怕的是

# 实际流程

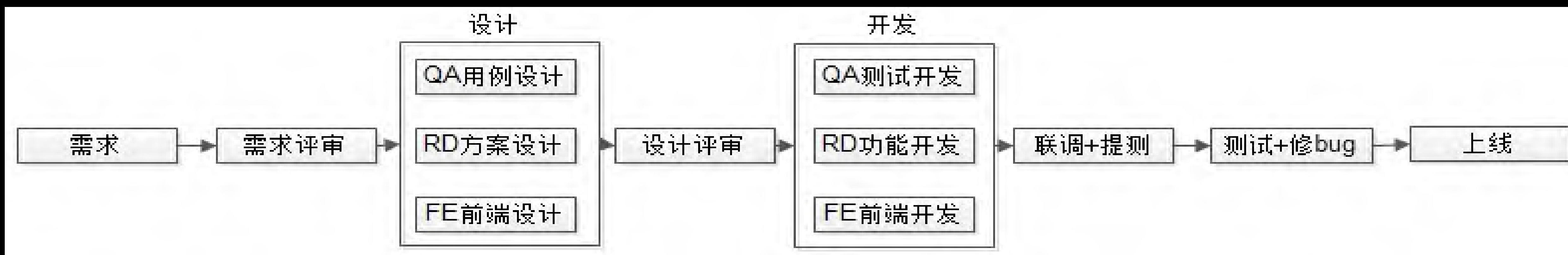


# 项目周期定理

- 变化出现在项目流程的越上游，项目风险越小，项目越可控
- 需求变化 -> 变化出现在项目中的第一个步骤
- 方案变换 -> 变化出现在项目中的第二个步骤
- 改进：
  - (1) 控制需求变化频率（不是不变，变要有代价）
  - (2) 加入研发设计+评审环节



# 改进



# 四、质量篇

线上bug多

投入大量时间改bug

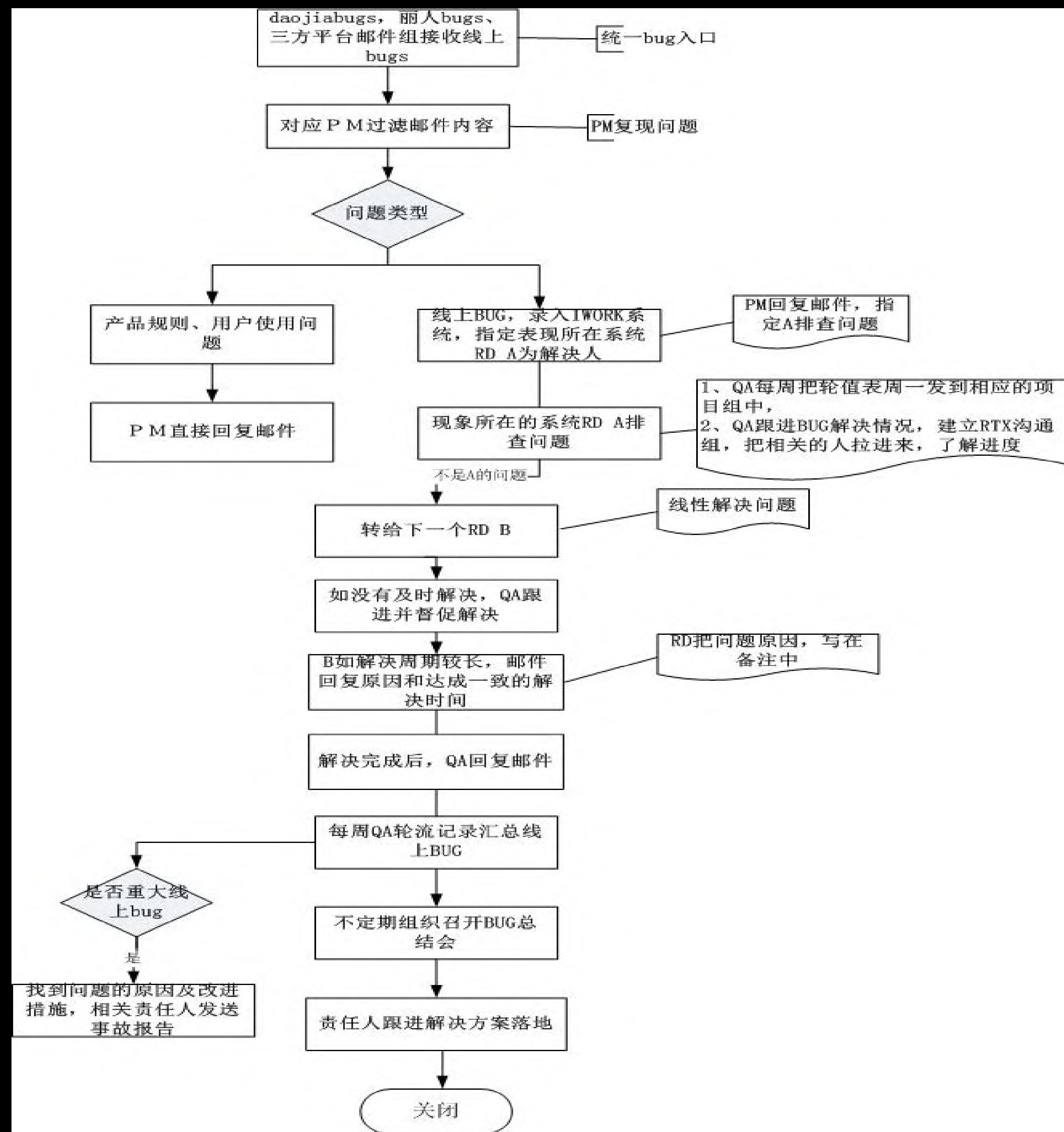
项目进一步加班

# Bug过多，占用时间

- 数据说话，bug的收集与分类
- 不怕一次性bug，怕一类bug反复出现
  - (1) 运营、pm配置错误，引发rd修复数据 -> 犯错需要代价
  - (2) 功能，由于低频需要rd手动修复数据 -> 做成功能
  - (3) bug不分优先级，总让rd第一时间介入修复 -> 规范流程

# Bug处理流程

- Bug处理流程
- Bug分级



# 五、技术含量篇

士气低落

自认为做的事情没有技术含量

人员流动率高

# 是没有含量，还是想得不够？

- 真的做到极致了么？

案例：一个“微信钱包”静态落地页的运营需求

- 真的抽象的够好了么？了解我们所负责系统的现状么？

案例：

(1) 系统监控

(2) 性能数据统计，业务数据统计

(3) 查找bug效率高么？



定期技术分享，营造学习氛围

技术培训夜校，帮助大家提高

# 总结

## (1) PM篇

- 目标一致
- 数据说话，以理服人
- 项目总结，效果说话，增强信任

## (2) QA篇

- 目标一致
- 分工明确，数字说话
- 找主要矛盾，针对改进

## (3) 加班篇

- 加入技术设计+评审环节
- 控制需求变更

## (4) 质量篇

- 数据说话，总结，针对性修复
- bug修复流程

## (5) 技术含量篇

- 多思考，把技术做到极致
- 多抽象，共性问题下沉解决
- 定期技术分享，营造学习氛围
- 技术培训夜校，帮助大家提高

# Q&A

“架构师之路” 公众号





关注QCon微信公众号，  
获得更多干货！

# Thanks!



主办方 **Geekbang** > **InfoQ**  
极客邦科技