

LinkedIn 应用 SRE 团队建设与发展

李 琥 (Li Xiao)

Sr. Manager, Site Reliability Engineer @ LinkedIn



促进软件开发领域知识与创新的传播



关注InfoQ官方微信
及时获取ArchSummit
大会演讲视频信息



全球软件开发大会 [北京站]

2017年4月16-18日 北京·国家会议中心

咨询热线: 010-64738142



全球架构师峰会 2016 [深圳站]

2017年7月7-8日 深圳·华侨城洲际酒店

咨询热线: 010-89880682

大纲

LinkedIn 简介

SRE 雏形 – 生存为主

SRE 发展 – 谋求发展

SRE 未来 – 创造价值

经验 教训 – 掉过的坑

LinkedIn 简介

1. 450M (4.5亿) 用户
2. ~4000 开发人员
3. 产品开发以Java为主，SRE以Python为主
4. 2012年 SRE：20人
5. 2016年 SRE：350人，Embedded SRE: 80+
6. 5个数据中心，50000+ 服务器

关于我

1. Before LinkedIn: Developer
2. 2012 至今 : LinkedIn SRE
3. Sr. Manager, Site Reliability Engineer @ LinkedIn
4. 负责三个应用运维团队
 1. 支付, 高级会员, 企业平台
 2. 销售解决方案, 公共API
 3. 内容抓取, 视频录制和发布
 4. 130/500+ 服务

应用 SRE

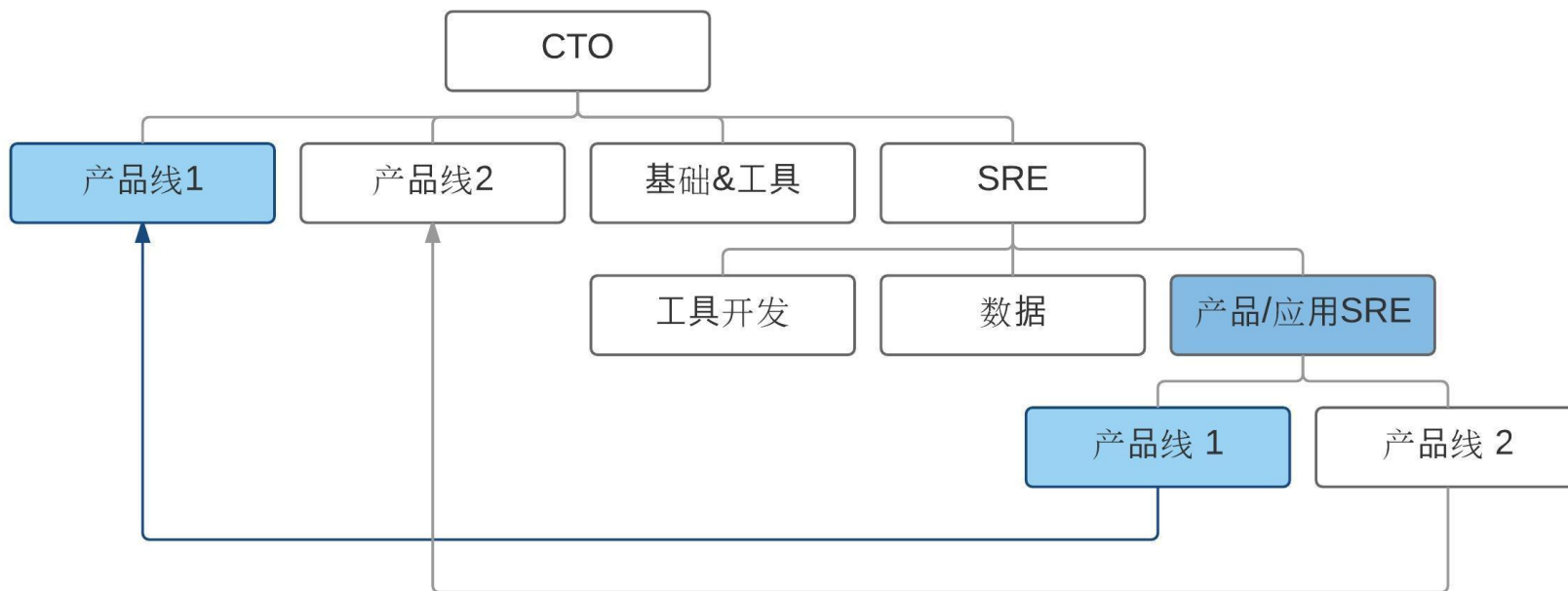
1. 什么是应用 SRE

1. Embedded/Product/Application SRE/PE
2. 与 Infrastructure / Tooling 的区别

2. 为什么需要应用SRE

- 开发与SRE直接目标、语言、文化不同
- 系统复杂度

应用 SRE



大纲

LinkedIn 简介

SRE 雏形 – 生存为主

SRE 发展 – 寻求发展

SRE 未来 – 创造价值

经验 教训 – 掉过的坑

第一阶段 生存

主要工作内容

1. 100% Operation
2. NOC 主导
3. 救火队
4. 手工操作



第一阶段 生存

几个小故事

1. 发布 (release)
2. on-call支持

第一阶段 生存

与开发团队的关系

1. 敌对
2. 不信任



第一阶段 生存



大纲

LinkedIn 简介

SRE 雏形 – 生存为主

SRE 发展 – 谋求发展

SRE 未来 – 创造价值

经验 教训 – 掉过的坑

第二阶段 发展

主要工作内容 (1)

1. 自动化尽可能多的人工操作

- Track-CLI / log parser etc.
- Auto-remediation

2. 切入产品设计开发阶段

1. 特殊用户请求
2. 支付网关

3. 事故复盘 Post-mortem

第二阶段 发展

主要工作内容 (2)

1. 反馈需求给Tools和Infra开发团队
 - SRE ♥ Tools
 - Oncall validation
2. 更多双向交流和培训
3. On-call

第二阶段 发展

和开发团队的关系

1. 理解对方立场
2. 有保留的相互信任 (trust&verify)
 - 变更需要SRE 审核
 - `Class com.linkedin.xxx.InternalError`

大纲

LinkedIn 简介

SRE 雏形 – 生存为主

SRE 发展 – 谋求发展

SRE 未来 – 创造价值

经验 教训 – 掉过的坑

第三阶段 创新未来

主要工作内容

1. 解决疑难杂症
2. 基础、框架层面优化
3. 设计和实现应用层面的特殊需求
4. 全局观念，做决定
5. 事故预演和预防：Pre-mortem
6. On-call

第三阶段 创新未来

主要工作内容 – 疑难杂症

1. 语言包更新造成GC
2. 30ms 跨数据中心延迟 导致的请求丢失
3. 把经验转换成工具转换成系统

第三阶段 创新未来

主要工作内容 – 框架层面创新和优化

1. 协议层超时和重试
2. 协议层插错机制

第三阶段 创新未来

主要工作内容 – 设计实现需求

- FTP功能的设计实现
 - 安全和可管理
- Multi-master 多主 快速切换
 - 可运维

第三阶段 创新未来

全局产品观念 辅助做决定

1. 推迟新品上线时间
2. 权衡利弊做选择

第三阶段 创新未来

与开发团队的关系

1. 信任与支持
 1. 全局视角
 2. 技术和政治上互相支持
2. 赋予开发团队更多的权利和义务
3. 提供“安全带” 避免开发过程中犯错

大纲

LinkedIn 简介

SRE 雏形 – 生存为主

SRE 发展 – 谋求发展

SRE 未来 – 创造价值

经验 教训 – 掉过的坑

经验教训 (1)

SRE 团队建设

1. 正确的时间，选择正确的人
2. 不断更新知识
3. 更新队伍
4. 消除团队里的单点
5. 避免个人英雄主义
6. 明确团队成长方向

经验教训 (2)

其他团队 (开发、产品)

1. 自上到下的文化改变
 - 可以拒绝变更 (Pushback)
 - 可运维性也是需求的一部分
2. 产品经理很重要
3. 沟通 和 理解 (轮岗)
4. 和开发团队同步成长
 - LinkedIn
 - Netflix/FB

经验教训 (3)

50/50

1. 保证团队不超过50%时间在做“维护”
2. 剩下50%的时间
 1. 开发
 2. 改善流程
 3. 学习、培训

经验教训 (4)

On-Call

1. $7*24?$ $7*12?$ $(3+4)*24?$
2. 开发团队共担责任
3. 数据驱动提高和进步

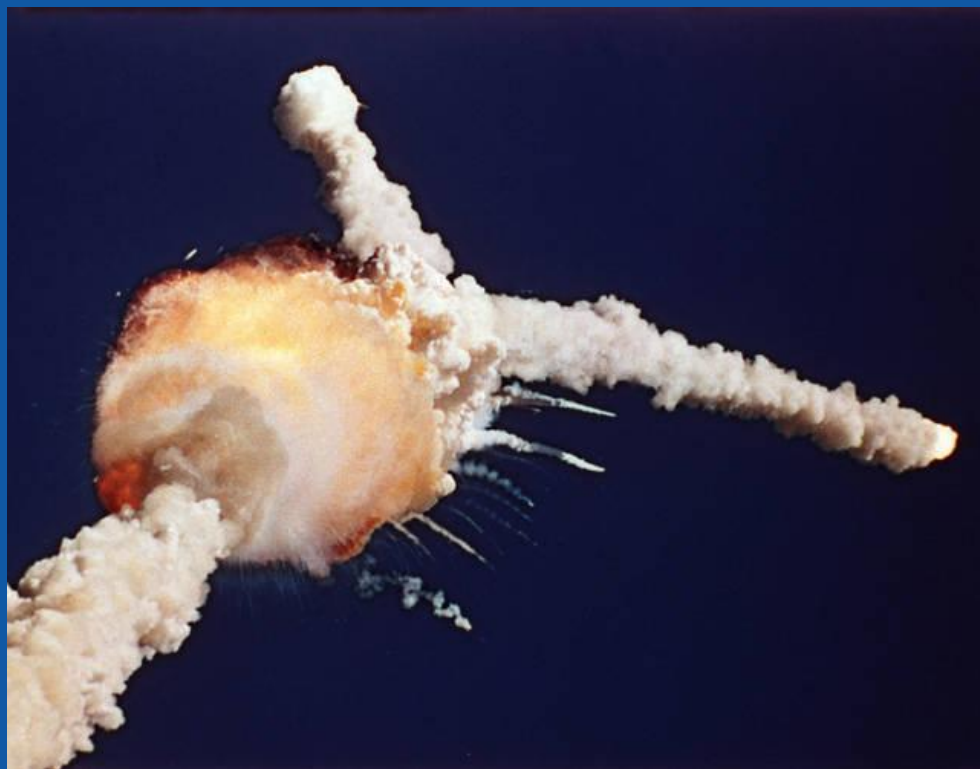
经验教训 (4) Oncall

数据驱动改变



经验教训(5) Post/Pre-Mortem

没完没了的事故复盘



总结

1. 应用SRE团队的发展和价值

- 业务需求是主要驱动力
- 开发、运维团队成熟度共同发展
- 运维团队开发能力如何提高

2. 过程是痛苦的

- 团队更新
- 数据驱动改变

3. 结果是美好的

- 开发团队的认可
- SRE有长远职业发展 (业务专家，深入开发，流程改进)

THANKS



[北京站]

主办方 **Geekbang** > **InfoQ**
极客邦科技