

SRE之道下金融行业的 落地实践过程

北京数人科技有限公司
2017年



01 SRE核心理念

02 金融行业ITSM特性

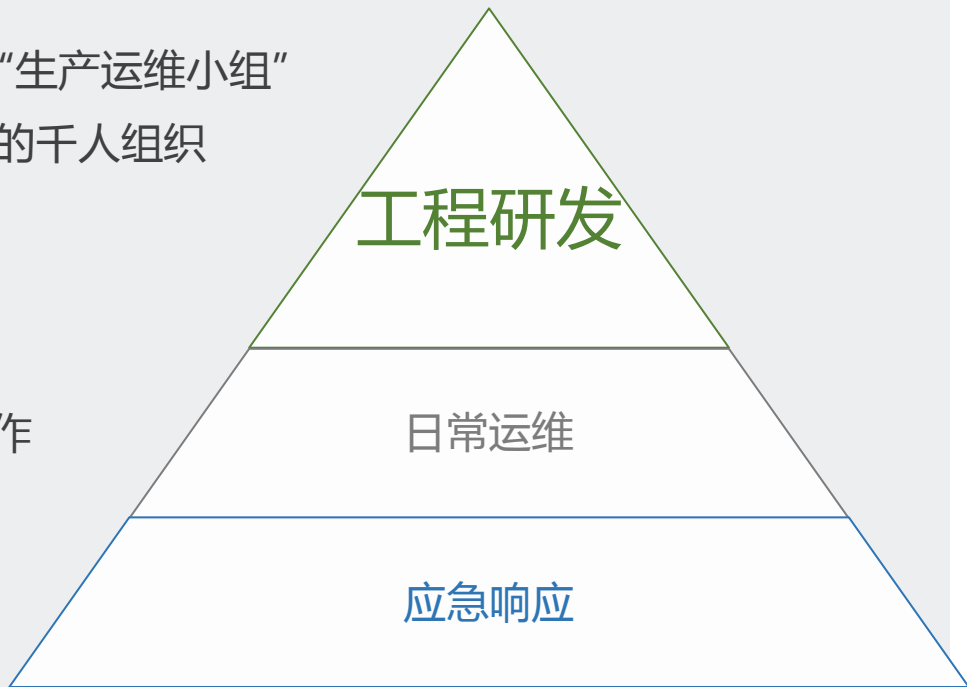
03 SRE在金融行业的实践和困境

01 SRE核心理念

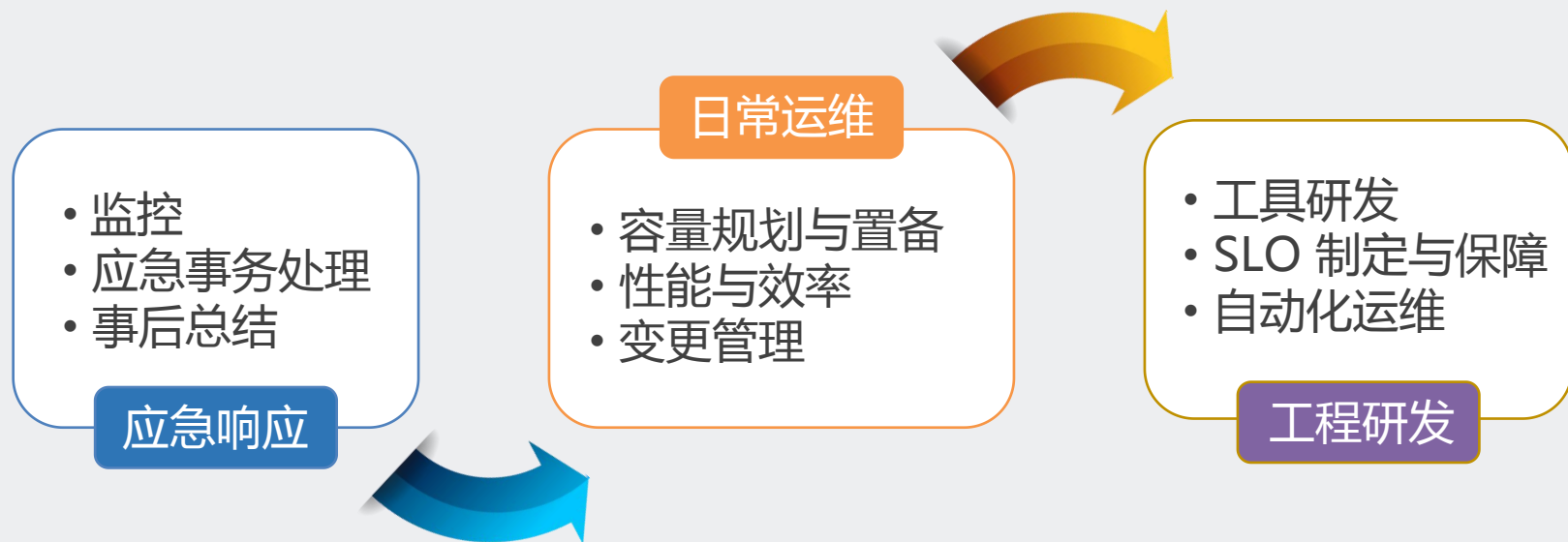
SRE 诞生

Google 在运维模式上的全新探索

- Ben Treynor Sloss
 - 资深软件开发经理
 - 2003 年加入 Google 组建 7 人 “生产运维小组”
 - 成长为覆盖 Google 所有产品线的千人组织
- 由开发工程师执行运维工作
 - 从内心里抵触重复性工作
 - 有能力进行自动化研发
 - 主要使用自动化工具完成运维工作
- SRE 是 DevOps 思想在运维方面的具体实践



SRE — 工作职责





SRE 原则

- 拥抱变化
- 服务等级目标
- 减少琐事
- 自动化
- 简单化

02 金融行业ITSM特性

金融行业ITSM特性

- 运维模式-运维与开发隔离
- 运维管理-团队规模一般随着服务部署规模增长而增长
- 运维职责-负责应急事件处理与常规更新
- 有银监会等合规要求

传统模式 vs SRE

- 大量经验可借鉴
- 易招聘
- 大量现成开发和运维工具

- 对运维人员技能的技能依赖性强
- 运维人员规模与系统规模线性相关
- 各个系统相对独立，需要人工切换
- 依赖脚本进行配置管理

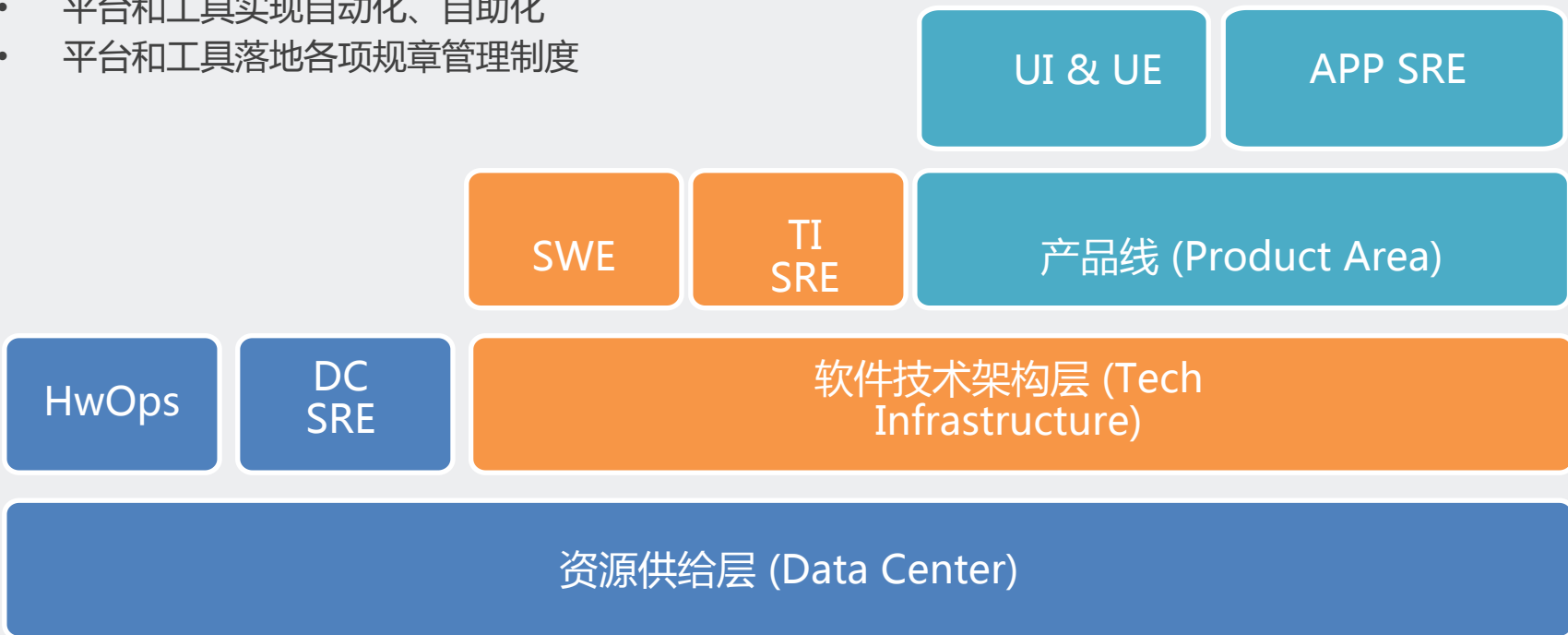
VS

- 强调自动化，通过开发运维工具实现
- 强调可编程性，固化运维的常见场景
- 强调效率，团队高效 人员成本低

- 招人难
- 团队搭建和管理没有太多业界参考
- 系统管理方式需要管理层支持

实践一平台架构模型

- 平台和工具实现自动化、自助化
- 平台和工具落地各项规章制度



03

SRE实践—发布协调

实践-发布协调

- 金融行业系统多，系统业务逻辑复杂，开发测试环境和生产环境物理隔离。
- 多个外包团队协同处理。
- 规模大的系统发布上线周期长。

以上问题，运用SRE思想如何解决？

建立发布协调团队

发布协调团队

发布协调团队
需对产品性能
指标达成共识



PM

架构师



运维
攻城狮

开发
攻城狮



发布协调团队职责

- 审核新产品和内部服务，确保产品预期的性能和非性能指标与最佳实践一致
- 跟进发布任务的进度，负责发布过程中技术相关问题
- 协调多团队之间
- 发布过程中的守门人，决定某项发布是否安全
- 整个服务生命周期的不同阶段进行审核

发布检查列表

根据发布流程，制定发布检查列表

- 架构与依赖
 - 检查逻辑架构、部署架构、检查请求流的顺序、检查负载均衡、日志规范、对第三方请求是否做超时处理。
- 容量规划
 - 依据压测报告、依据预估请求数等指标做容量规划，保障资源合理使用。
- 发布计划
 - 制定详细发布计划保障发布的成功率。

一切规范流程工具化

首先工具化是个渐进的过程。这也是SRE解决问题的思维模式，用工具解决问题。因此发布平台应运而生。



pipeline



Jenkins

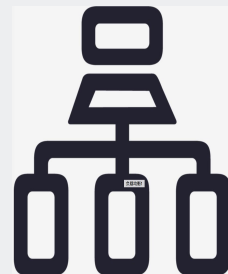
发布平台

变更管理

发布管理

发布流程模板

发布过程监控



F5/DNS

配置中心

.....

容器云平台

一切规范流程工具化

LOC

- 运营大盘
- 项目列表
- 流水线列表

▼ qlplat-core 后台服务 · 添加接口

项目: [图标]

项: [图标]

41 [图标]

60 [图标]

30 [图标]

15 [图标]

0 [图标]

任: [图标]

1 修改F5策略

```
Started by timer
Building in workspace /root/.jenkins/workspace/perf-pipeline
[perf-pipeline] $ /bin/sh -xe /data/apache-tomcat-7.0.78/temp/jenkins6233914811859915225.sh
e echo 'hello build'
hello build
```

2 人工检查

```
Started by timer
Building in workspace /root/.jenkins/workspace/perf-pipeline
[perf-pipeline] $ /bin/sh -xe /data/apache-tomcat-7.0.78/temp/jenkins6233914811859915225.sh
hello build
```

3 上传程序包

```
Started by timer
Building in workspace /root/.jenkins/workspace/perf-pipeline
[perf-pipeline] $ /bin/sh -xe /data/apache-tomcat-7.0.78/temp/jenkins6233914811859915225.sh
hello build
```

4 数据库脚本

```
Started by timer
Building in workspace /root/.jenkins/workspace/perf-pipeline
[perf-pipeline] $ /bin/sh -xe /data/apache-tomcat-7.0.78/temp/jenkins6233914811859915225.sh
hello build
```

5 刷新缓存

```
Started by timer
Building in workspace /root/.jenkins/workspace/perf-pipeline
[perf-pipeline] $ /bin/sh -xe /data/apache-tomcat-7.0.78/temp/jenkins6233914811859915225.sh
hello build
```

6 重启WebLogic

```
Started by timer
Building in workspace /root/.jenkins/workspace/perf-pipeline
[perf-pipeline] $ /bin/sh -xe /data/apache-tomcat-7.0.78/temp/jenkins6233914811859915225.sh
hello build
```

7 人工检查

```
Started by timer
Building in workspace /root/.jenkins/workspace/perf-pipeline
[perf-pipeline] $ /bin/sh -xe /data/apache-tomcat-7.0.78/temp/jenkins6233914811859915225.sh
hello build
```

8 修改F5策略

```
Started by timer
Building in workspace /root/.jenkins/workspace/perf-pipeline
[perf-pipeline] $ /bin/sh -xe /data/apache-tomcat-7.0.78/temp/jenkins6233914811859915225.sh
hello build
```

◎ 帮助 ● 小晨

日期: 2017-06-01

项目流水线 61

更多

3

占比 百分比

5%	10%
5%	10%
5%	10%
5%	10%

1)

应用0 招财童子 小芝麻

> qlplat-cms 后台管理系统 · 添加订单导出功能

> qlplat-web web前端 · 修改样式

> qlplat-img 图片服务器 · 修改图片

审批意见

请输入审批意见

同意 驳回 取消

Copyright © 2017 北京数人科技有限公司

发布协调团队收益

- 做好发布上线的守门员
- 提升自动化程度
- 提高发布效率
- 减少人肉执行动作，减少犯错误的可能

03 SRE实践—监控告警

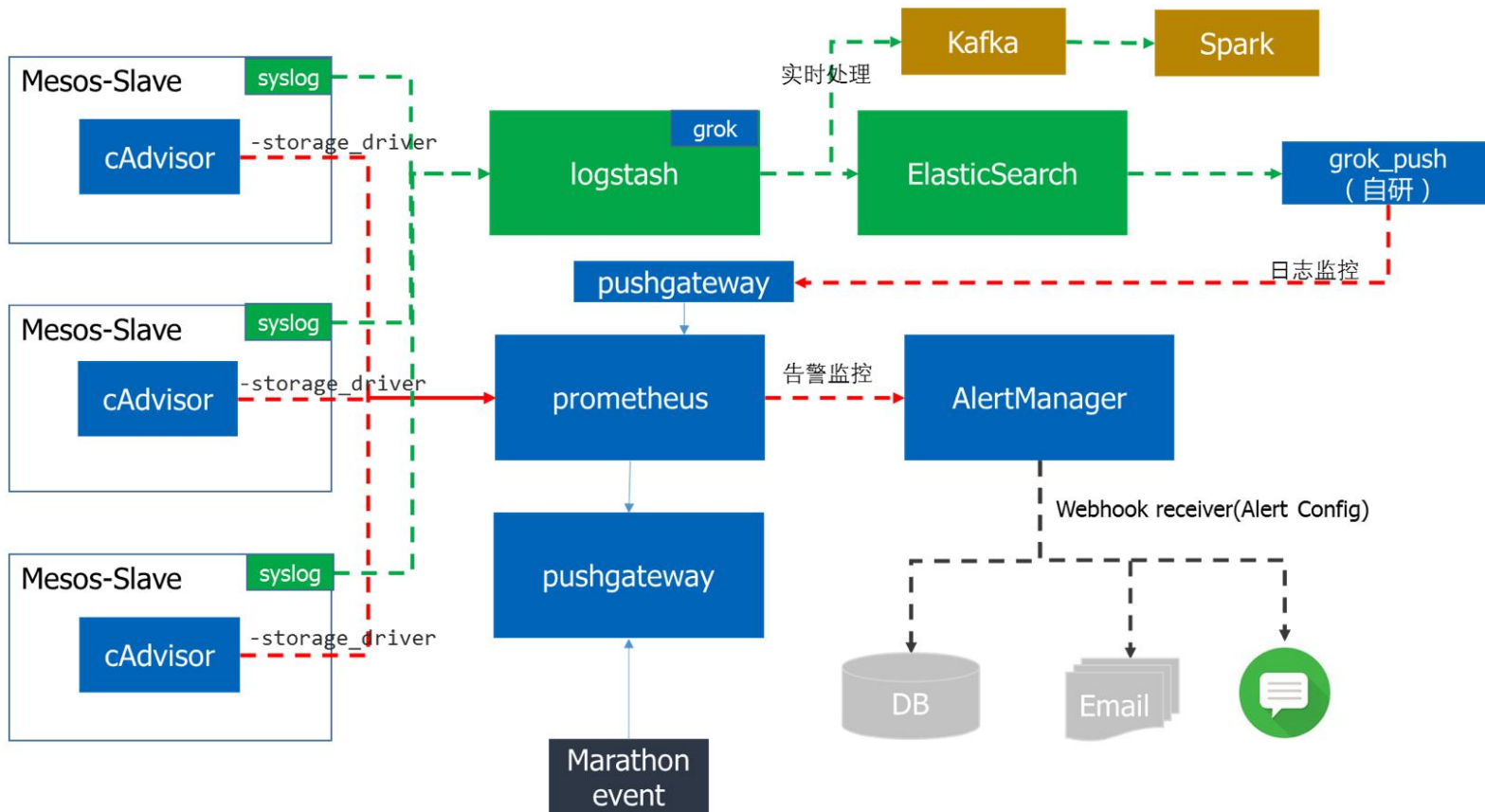
实践—监控

- 监控的重要性毋庸置疑
- SRE团队中应有1-2名监控专员，全职对监控系统进行构建和维护

监控解决的问题：

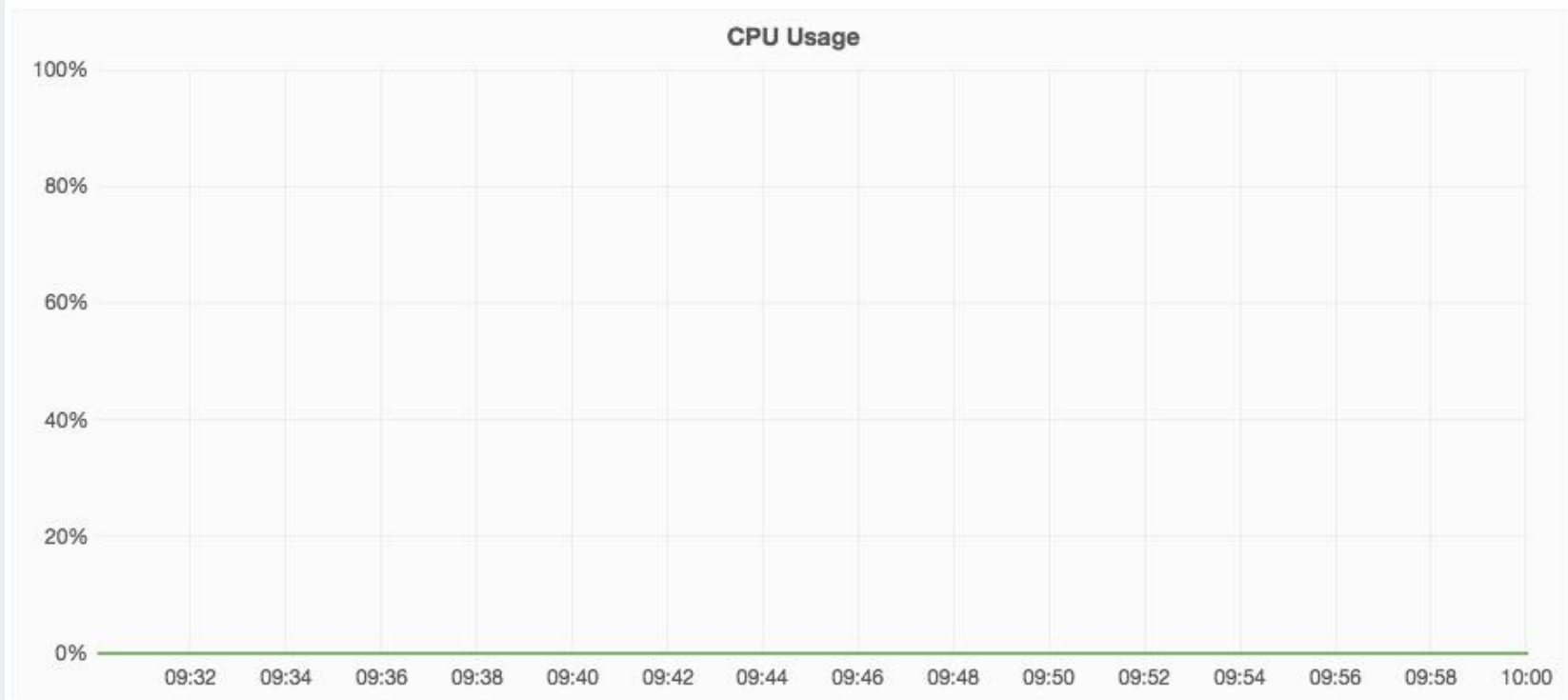
- **什么东西出故障 -- 建立监控系统**
- **为什么出故障 -- 事故总结，故障跟踪**

基于实践序列数据库的监控系统



基于实践序列数据库的监控系统

← → ↻ 127.0.0.1:9095/graph?appid=nginx-stress&metrics=cpuusage&output=graph



监控指标选择

监控系统四大黄金指标

- 延迟
 - 监控服务API, url的返回时间
- 流量
 - 监控F5等的pool member的请求数
- 错误
 - 监控HTTP请求返回码
 - 监控日志中异常关键字
- 饱和度
 - 根据不同系统, 内存资源型系统, 监控内存使用率; IO资源型系统, 监控Io等等。

指标的选择有效保障监控的时效性。

事故总结

- 事故总结的主要目标是为了找到为什么出故障
- 从失败中学习，为事故保留记录，最好能建立知识库，方便检索
- 总结好的方面
- 建立反馈机制，对监控系统的反馈，对产品及时建言
- 鼓励主动测试，甚至是不可能的问题进行模拟测试

故障跟踪

- 任何需要人工操作的事情，都只会延长恢复时间
- 通过事先预案并且将最佳方法记录在“运维手册”上通常会使 MTTR 降低 3 倍以上！
- 一个报警一个人处理，明确升级策略
- 解决线上问题为主，处女座强迫症为辅
- 做好总结：root cause、自动化

目标定位

- 基于现实，拥抱变化
- 坚持推演，演练
- 自主决策权，产品建言权

Q & A

数人云轻量级PaaS 落地SRE应用管理理念

