



架构迎接未来变化  
IAS2017 • NANJING

# DevOps监控平台实践

中兴通讯 刘通博

# Me

- 刘通博
- 2006年加入中兴通讯
- DevOps团队架构师
- 热爱技术，热爱篮球，热爱游戏

# 目录

CONTENTS

01

背景介绍

02

问题分析

03

架构方案

04

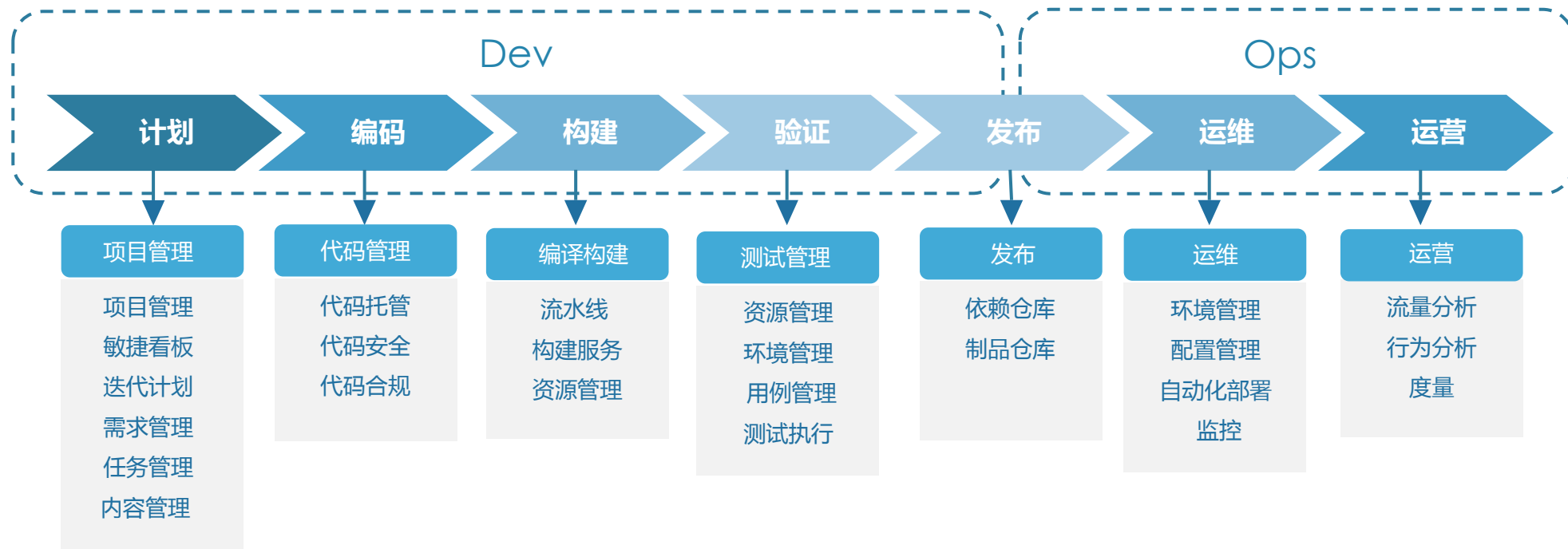
总结

# 01 背景介绍

让研发人员像使用水电气一样的使用研发资源，  
使其能聚焦高价值的研发工作。

## DevOps团队愿景

# 软件DevOps工具链



# Gerrit的发展

● JVM性能调优

2000

- 迁移了物理设备
- JVM性能调优
- Gerrit参数调优

5000

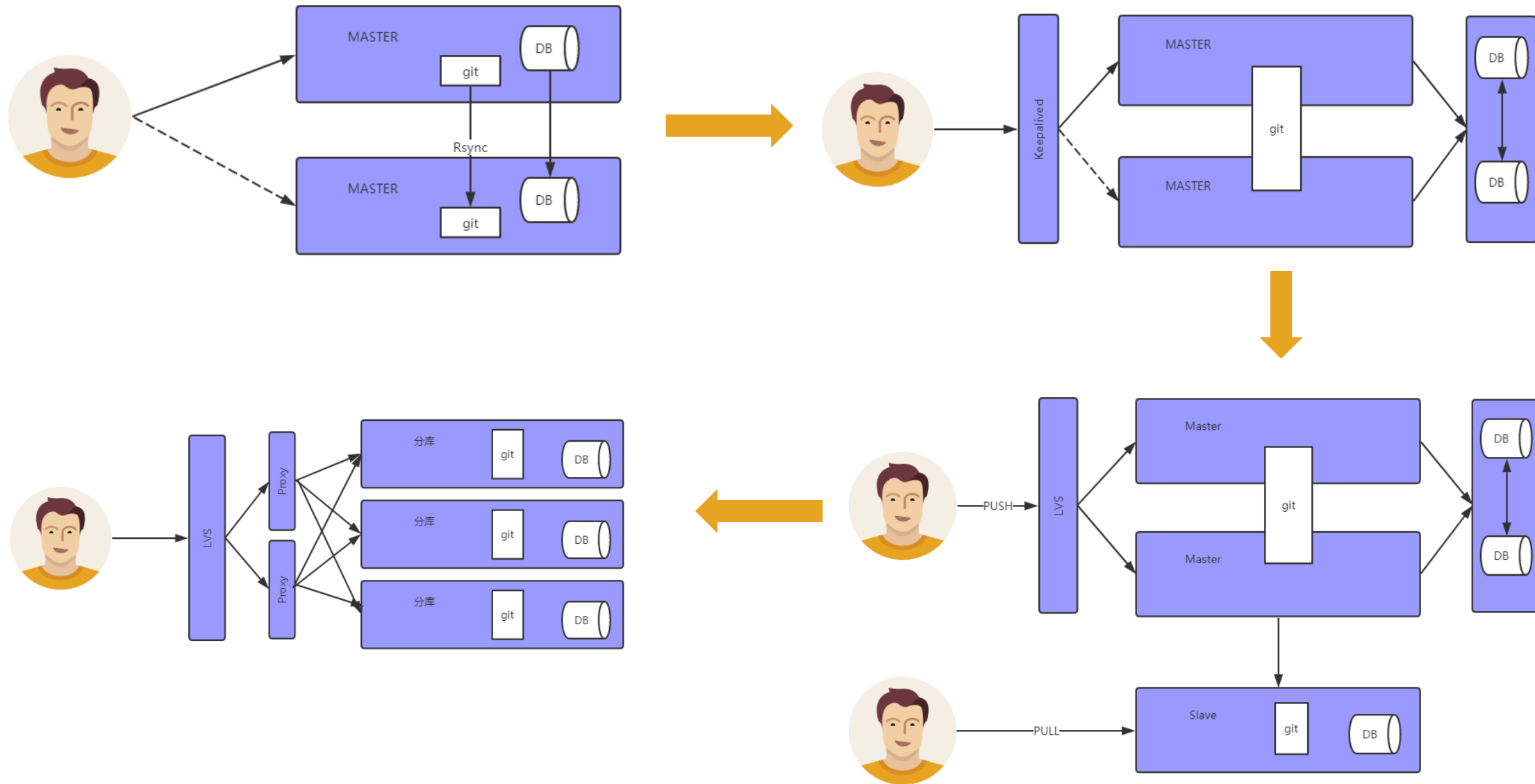
- 单主多活架构
- 使用插件共享事件流，索引，缓存，会话

10000

● 开发git代理服务

未来

# Gerrit架构的演进





# Gerrit数字化表达

7

6000+

14000+

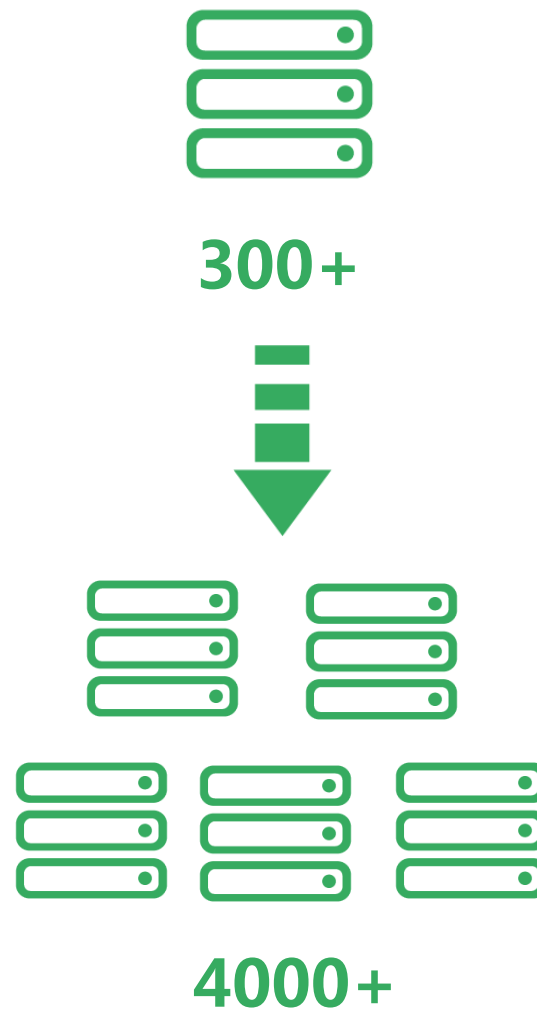
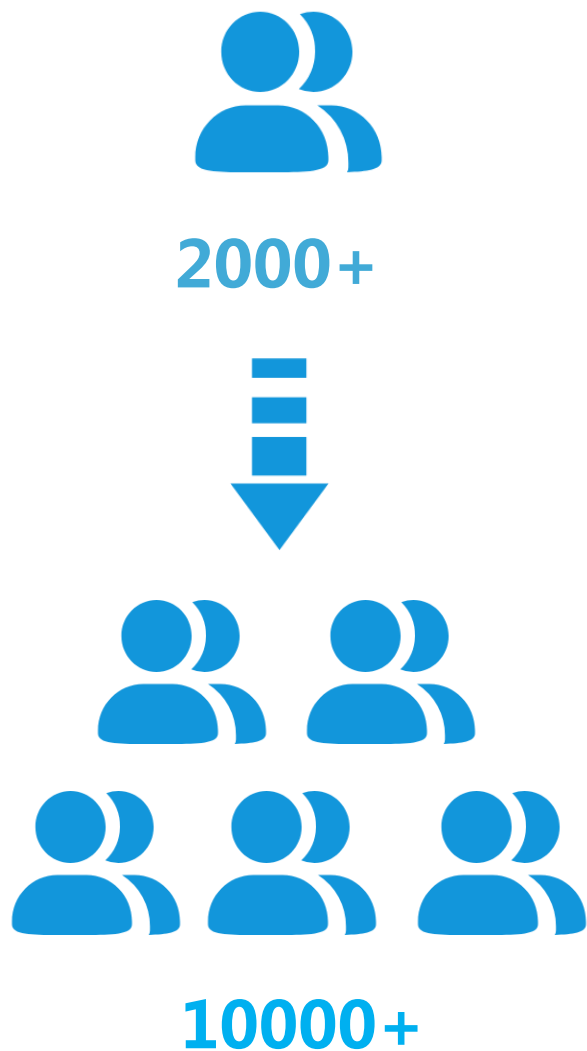
3200000+

80%

## 02 问题引入&分析

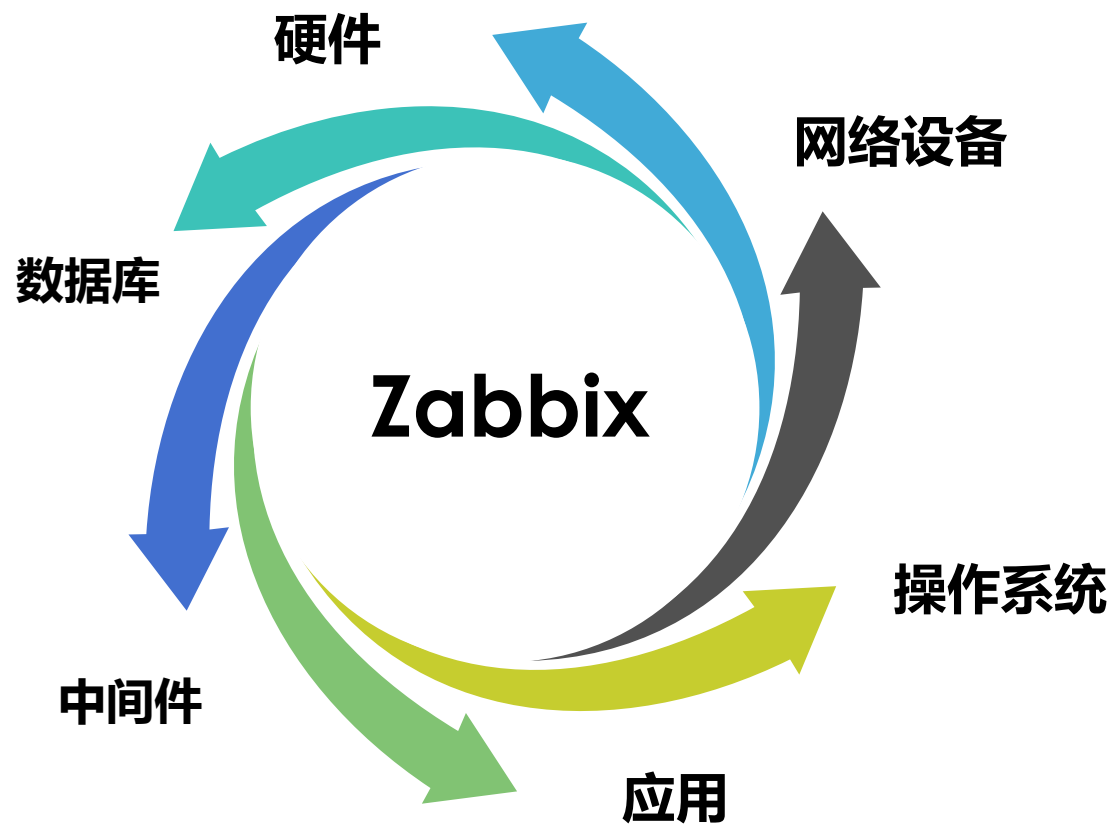


# DevOps规模扩张





- 用户：Gerrit系统下代码太慢了！
- 我们：我刚试了下，没慢啊。
  
- 领导：Gerrit系统挂了，哪里出了问题？
- 我们：还不清楚原因，正在排查。

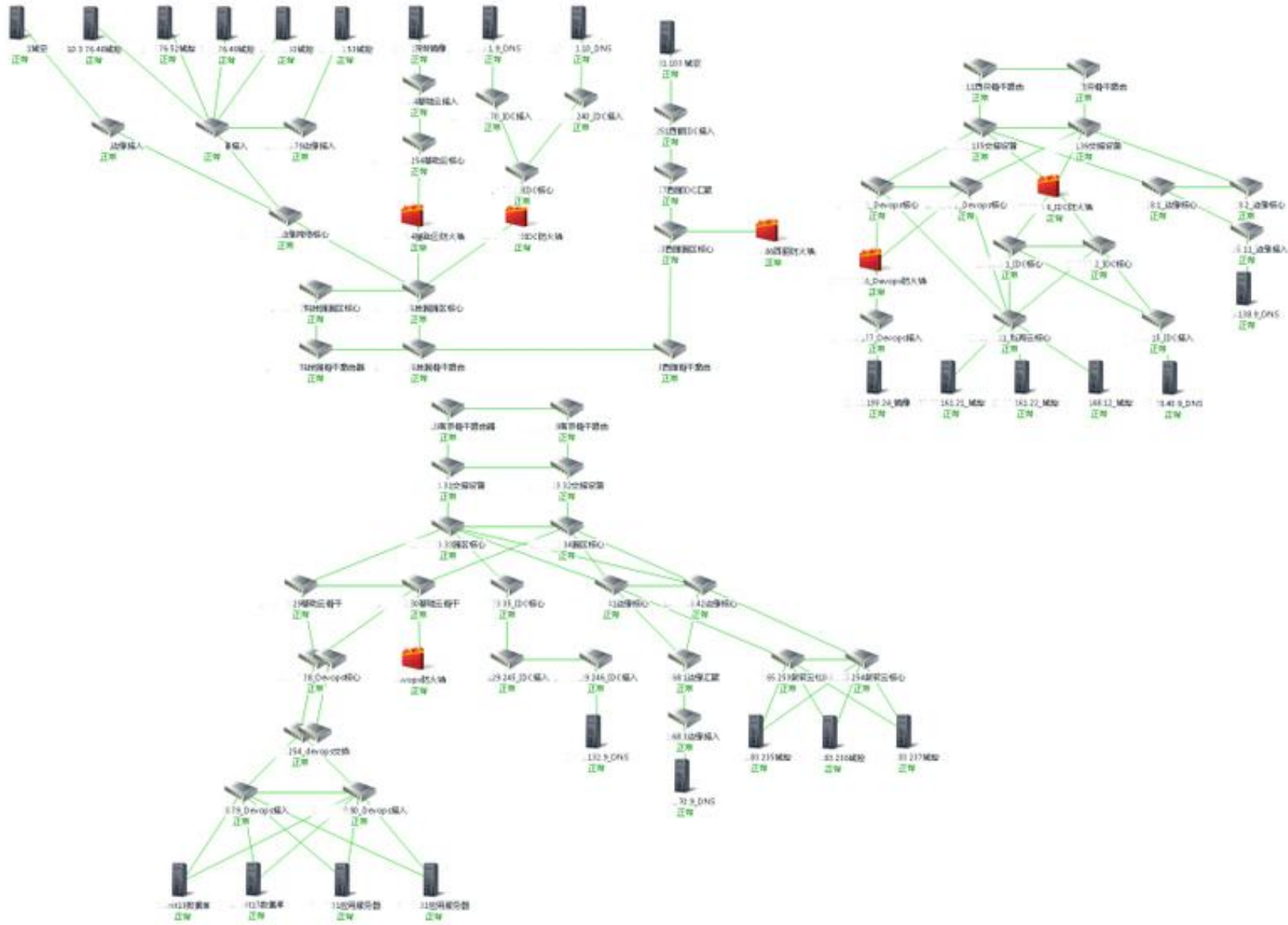


基于Zabbix的监控

基于指标项的告警



# 全局的总图



- 显示全局状态
- 服务器上排查问题



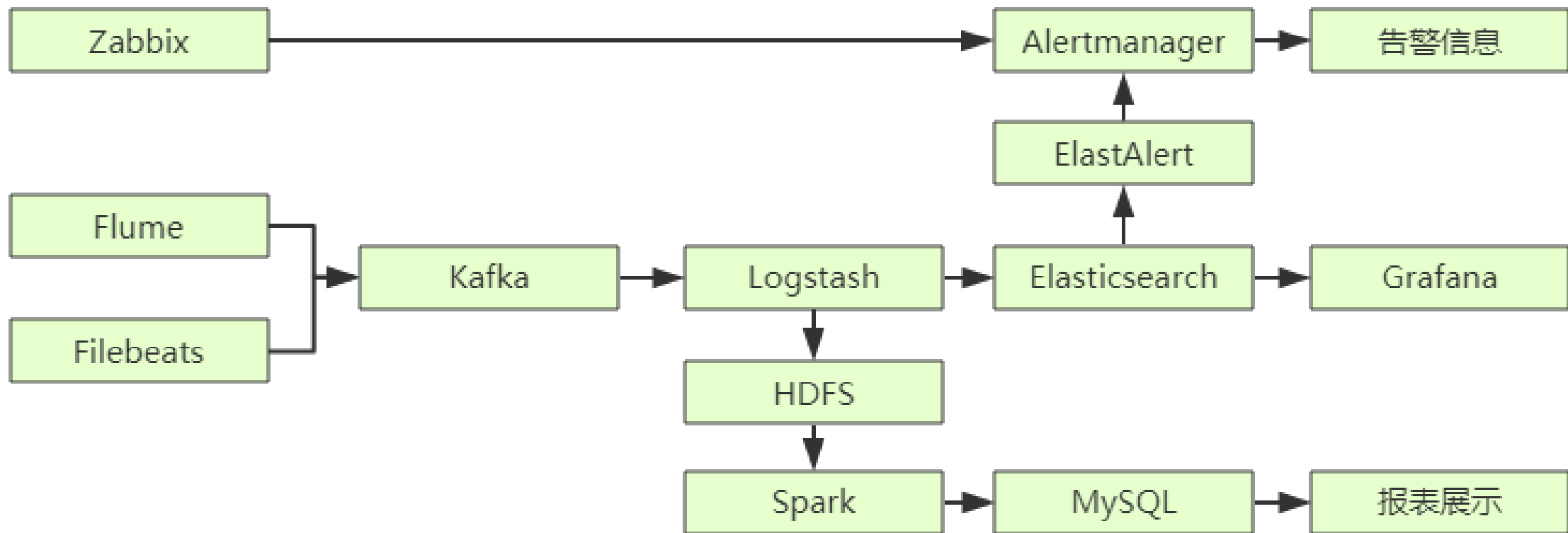
用户体验方面的监控

日志数据分散

告警繁多

# 03 架构设计

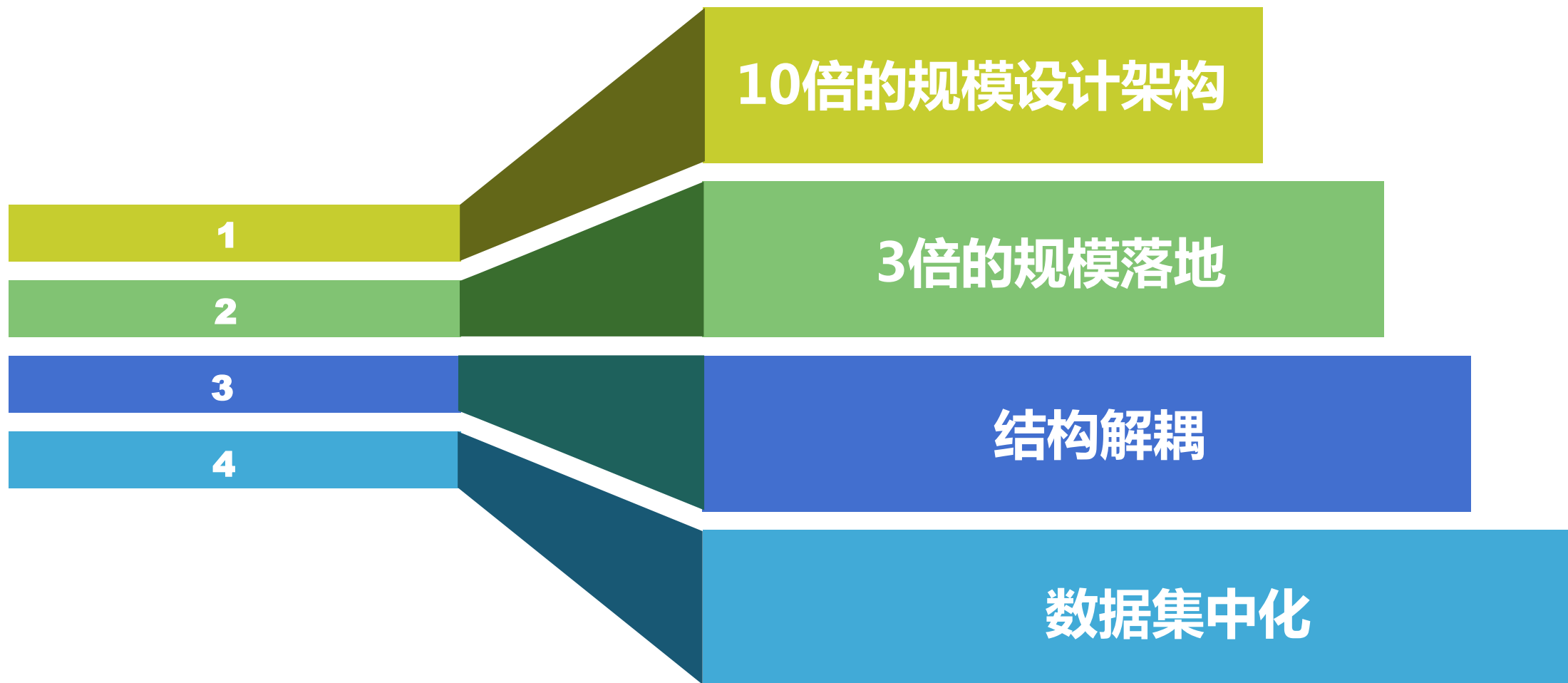






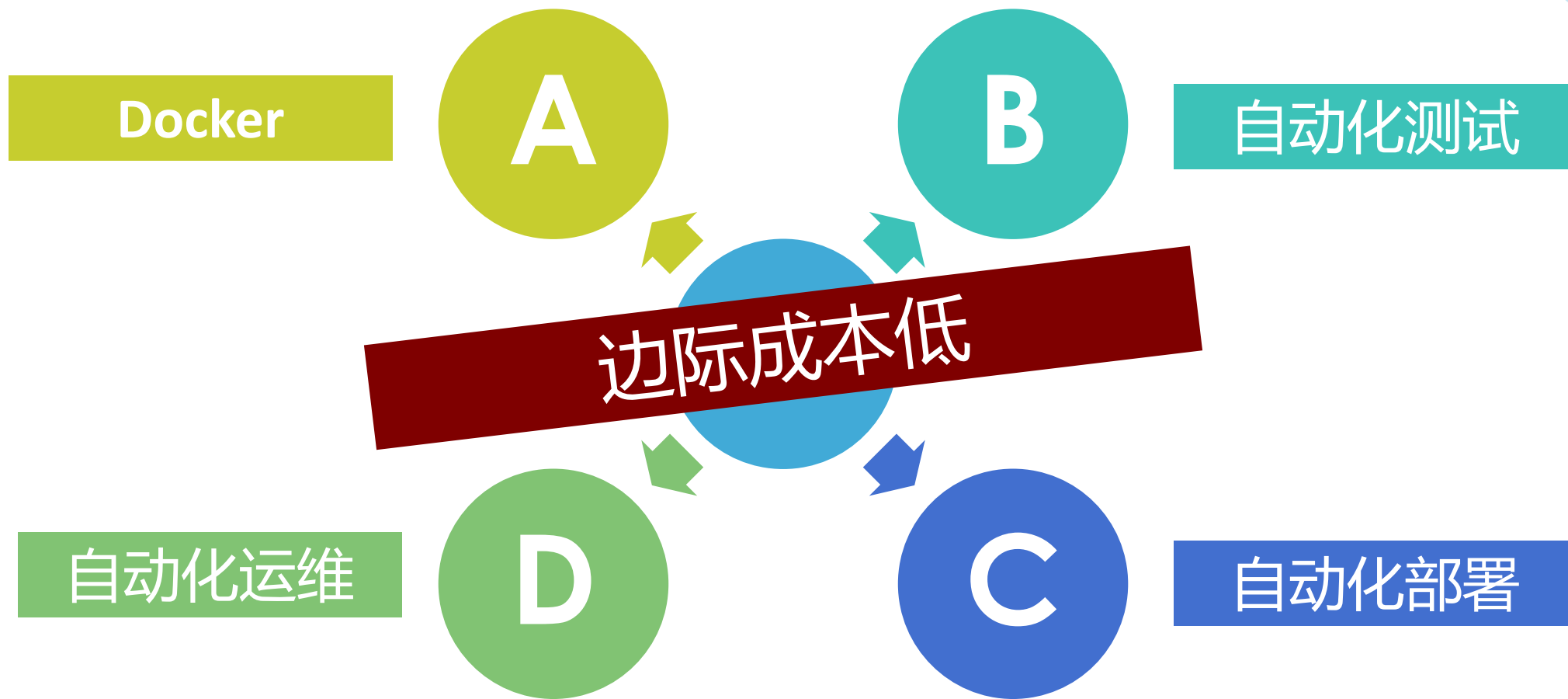
One more thing...

One more thing...





# 10倍规模的设计





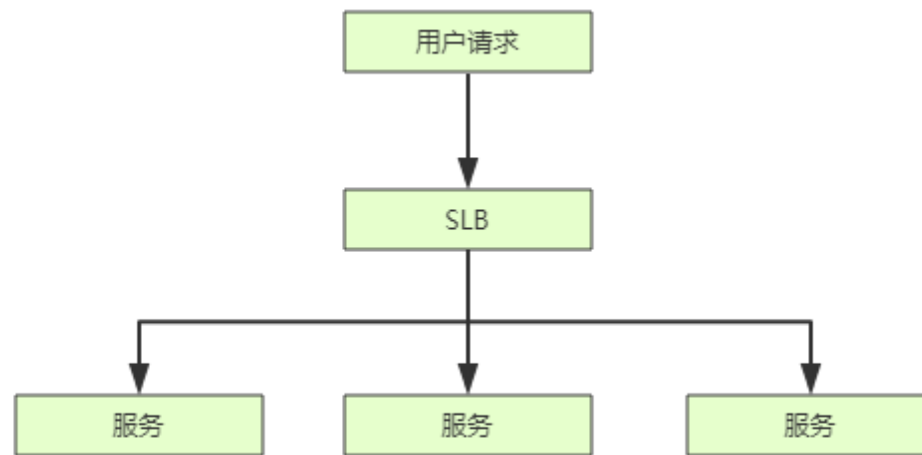
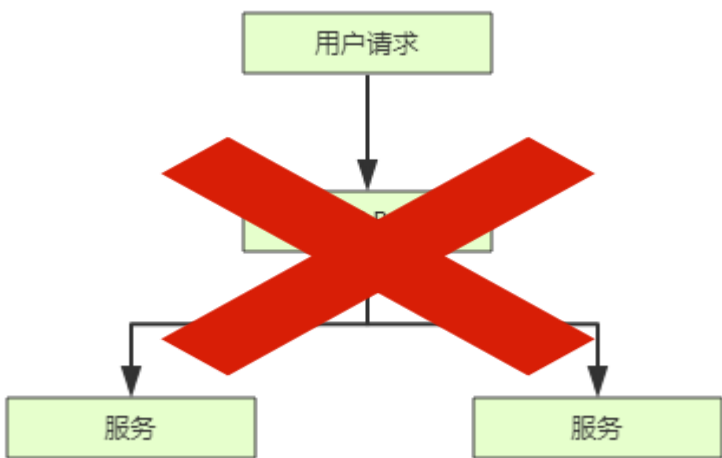
如果规模扩张10倍！

哪些工具的边际成本很高？配置？维护？

最理想：只增加服务器

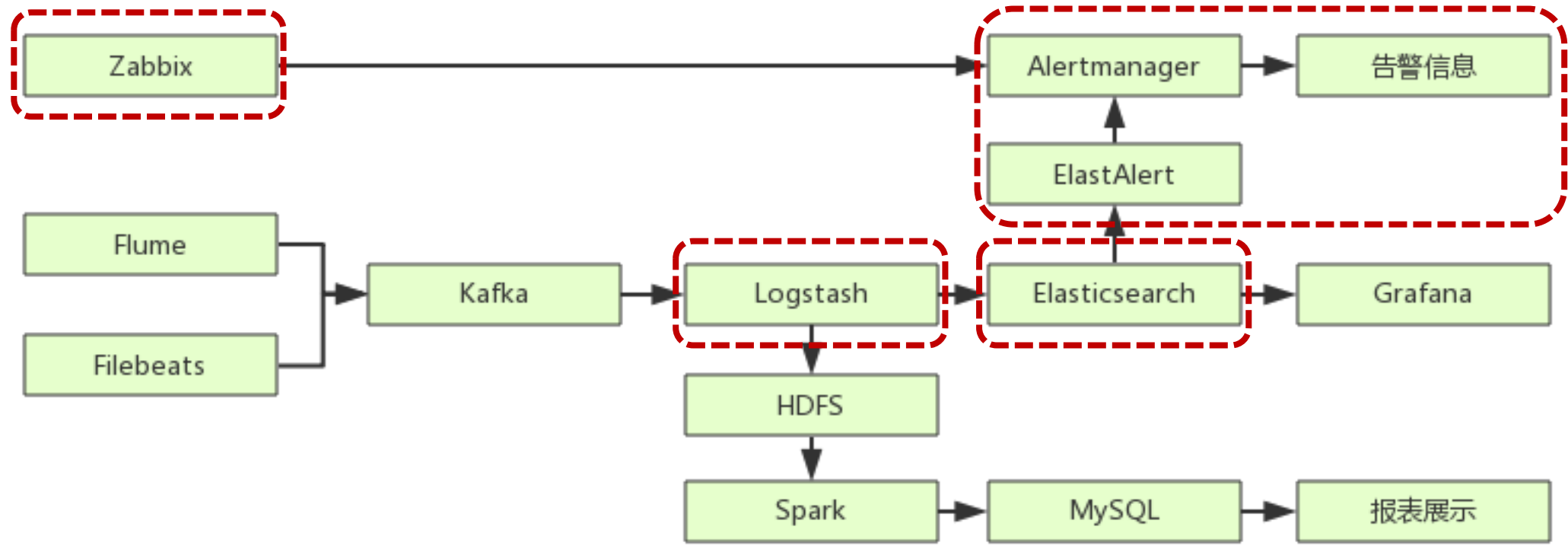


# 3倍规模的部署



雪崩效应









痛点1：

**监控告警混合**

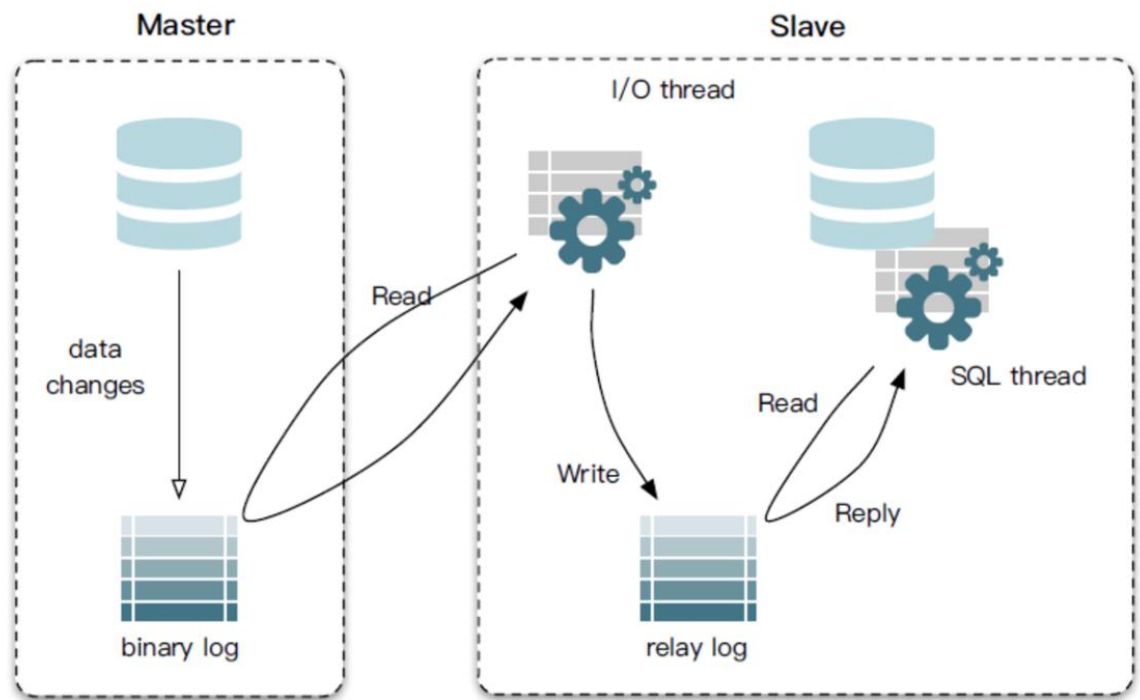
痛点2：

**监控体量上限**

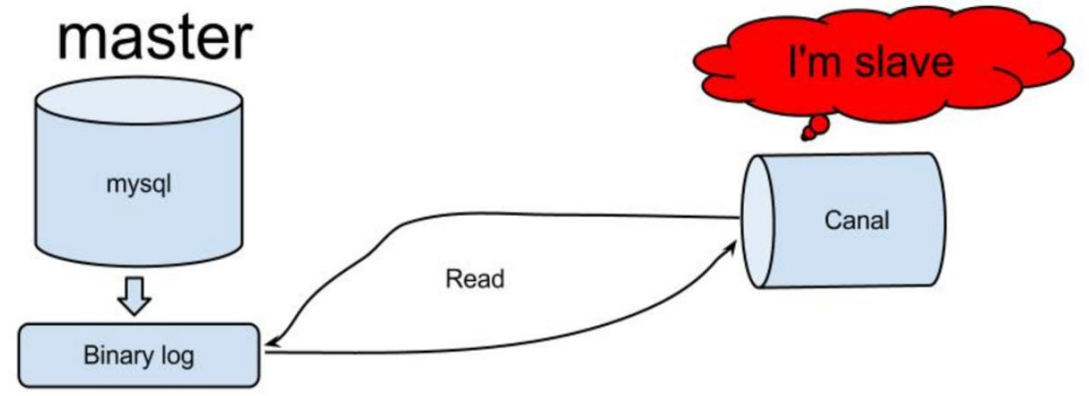
痛点3：

**数据无法汇聚**





canal的工作原理





## 痛点

- 数据清洗工具，无法提供更复杂的处理
- 批量处理效果差
- 处理规则散落
- 无分布式版本

## 解决方案

- Spark Streaming处理引擎
- 分布式计算
- 规则集中
- 能力随资源增加



## 痛点

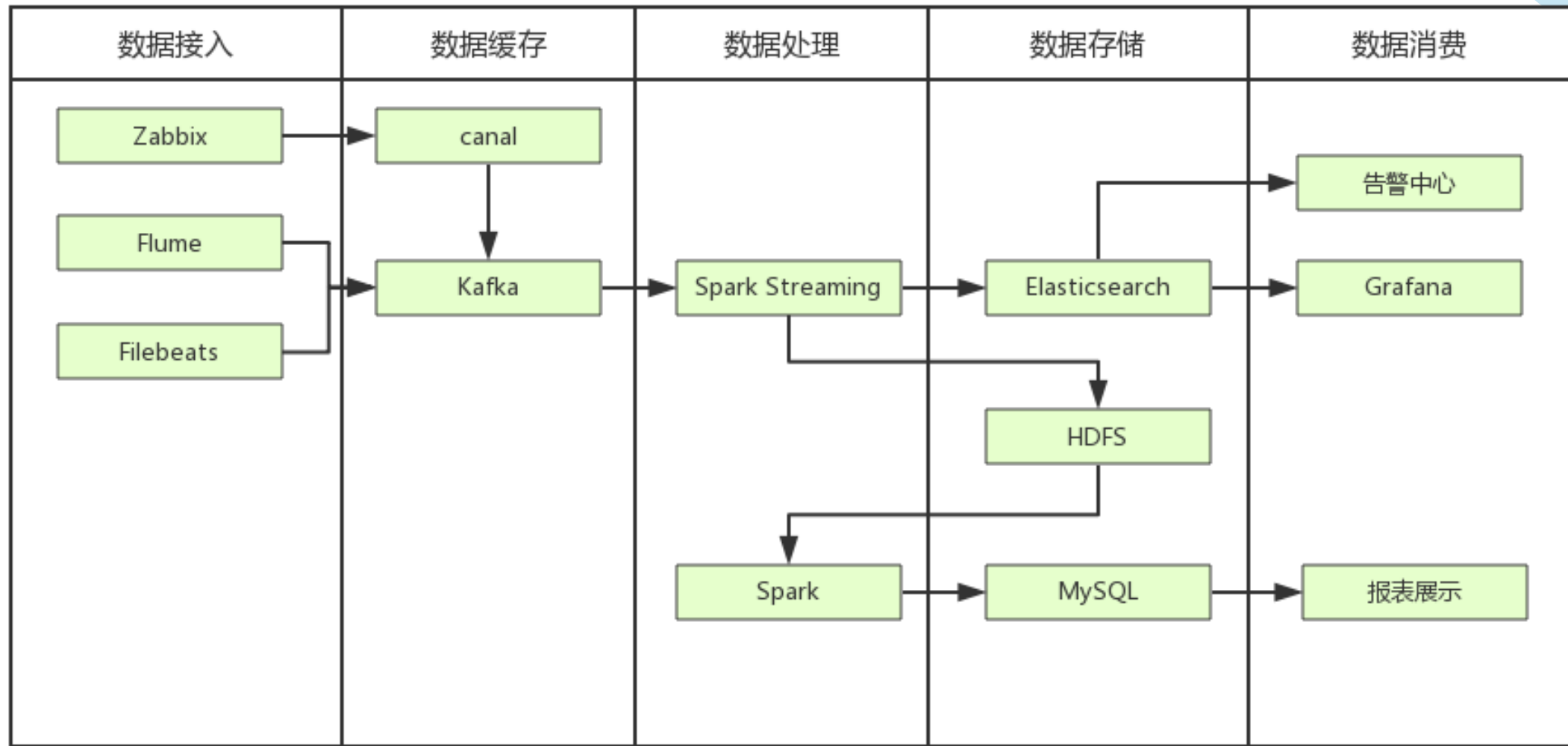
- 数据容量
- 查询效率

## 解决方案

- 数据的分类，部分数据直接不入库
- Spark Streaming提前处理计算逻辑



# 最终架构图





canal的方案依旧是过度，最终我们想改Zabbix的server端

告警数据的亲缘性

自动化运维

# 04 总结



开源工具满足了最初的需求

深度使用后最大的难点在于非分布式工具的管理问题

解耦开源工具，在外面封装是DevOps技术发展进阶之路



**谢谢聆听！ THANKS FOR LISTENING!**