MTSC2019
中国移动互联网测试开发大会

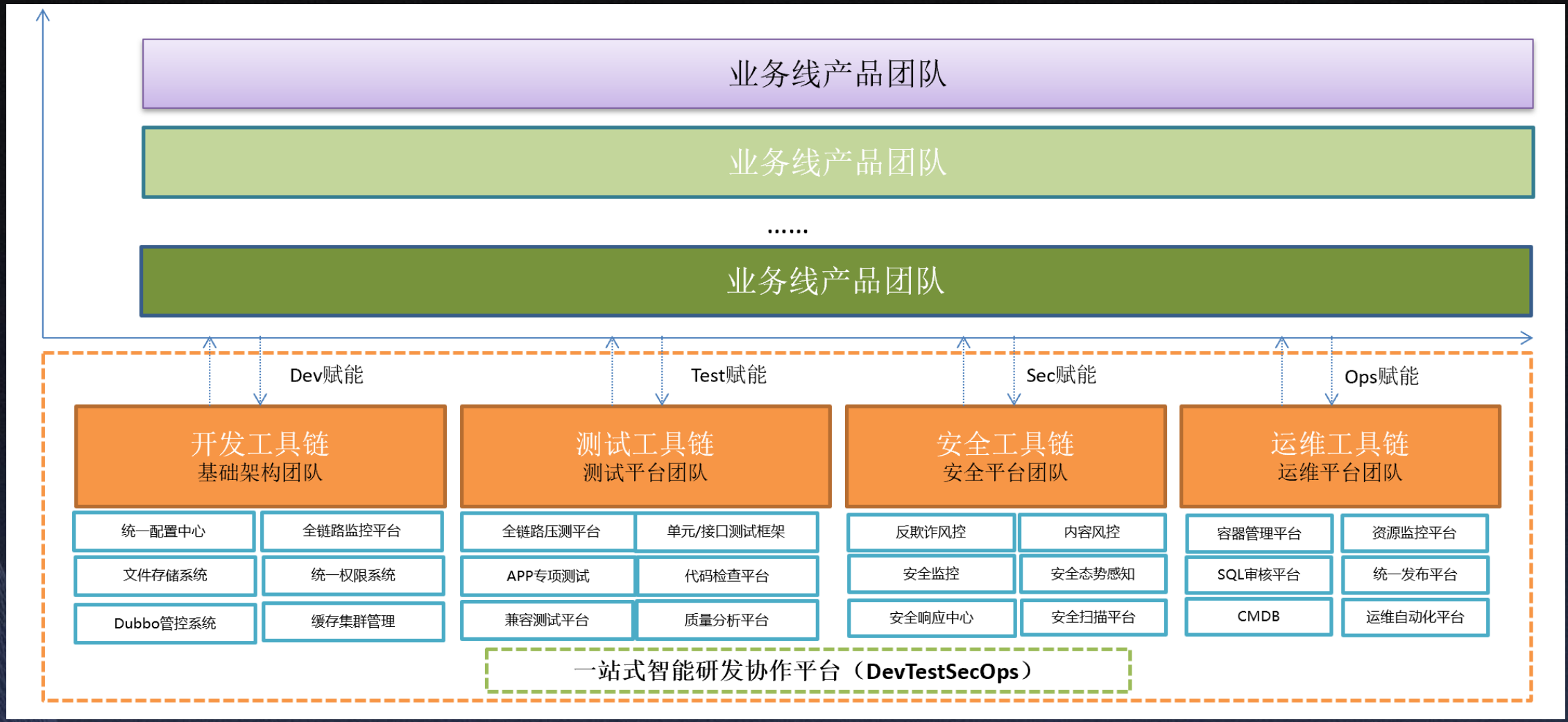
微医多维一体化监控平台实践

蒋刚毅（Cay）@微医

主办方：TesterHome 腾讯课堂

目录

写在前面：中台化技术平台建设



写在前面：一站式研发协作平台

应用管理

- 应用元信息
- 应用大盘
- 代码库模板
- 技术栈模板
- 公共应用库

交付流水线

- 静态代码扫描
- 单元测试
- 接口测试
- UI层测试
- 安全测试

部署作业

- 发布卡点
- 分批策略
- 过程监控
- 快速回滚

度量管理

- 代码指标
- 流水线指标
- 发布指标
- 运营指标
- 质量评分
- 多维看板

资源管理

- 标准stage
- 统一调度
- 标准运维模板
- 自愈与自助
- 一站申请

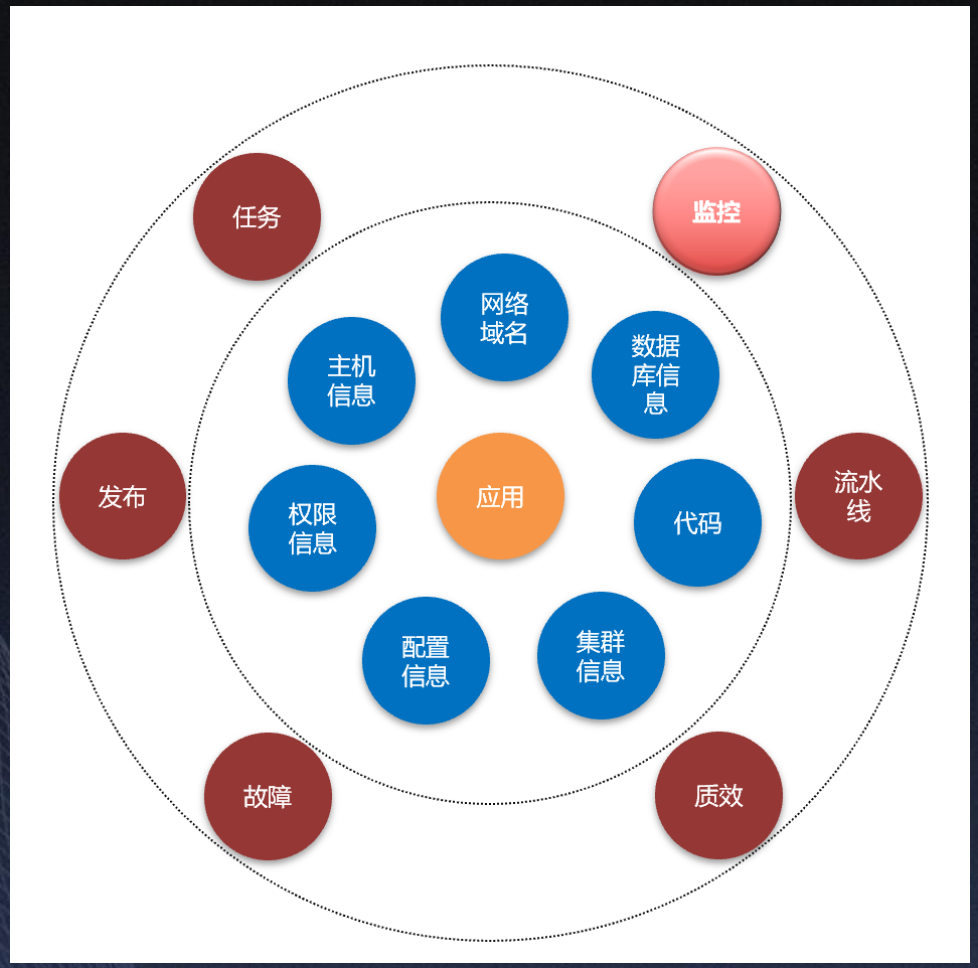
发布管理

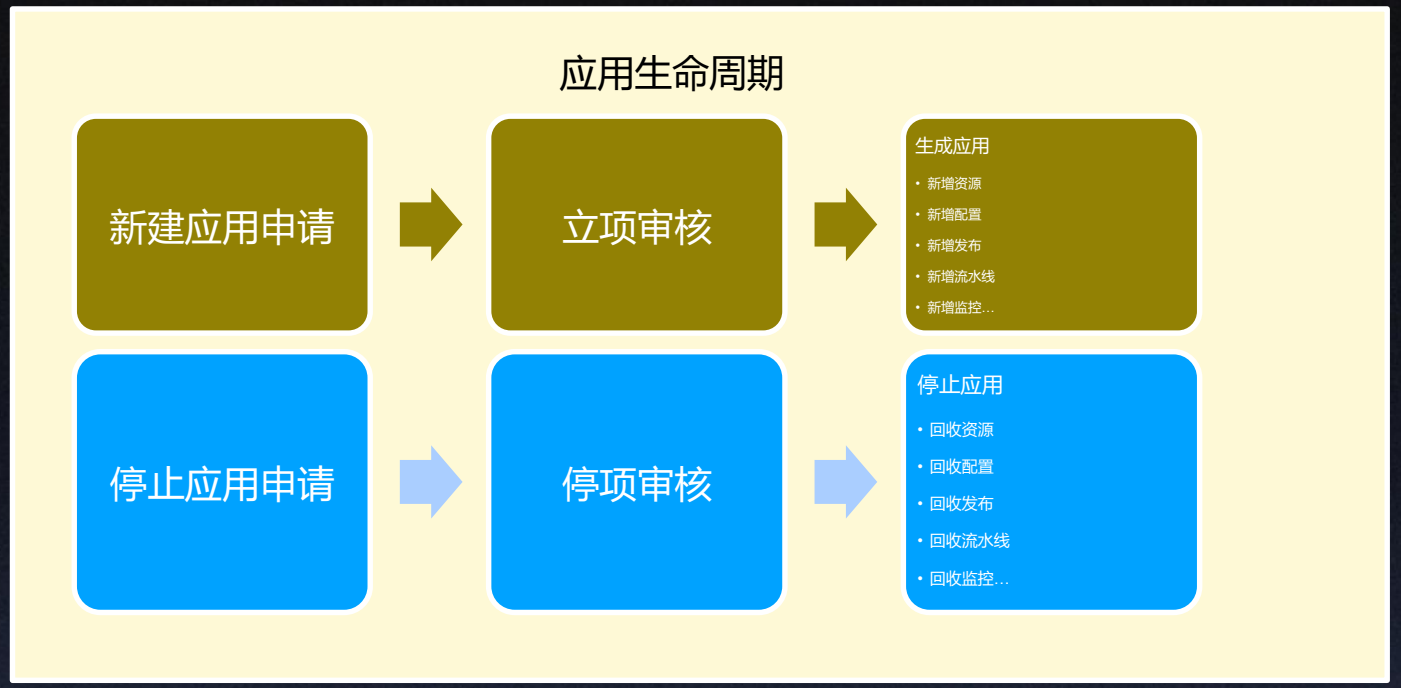
- 发布需求
- 发布步骤
- 发布进度
- 发布审核
- 回滚计划

监控管理

- 监控大盘
- 拨测监控
- 链路监控
- 运维监控
- 业务监控
- 安全监控

写在前面：应用为中心的研发协作





- 公司级别的标准应用库。
- 多维信息展示的应用大盘
- 以应用为中心，串联整个研发协作过程。
- 开发为应用全周期负责。

01

监控的概念和分类

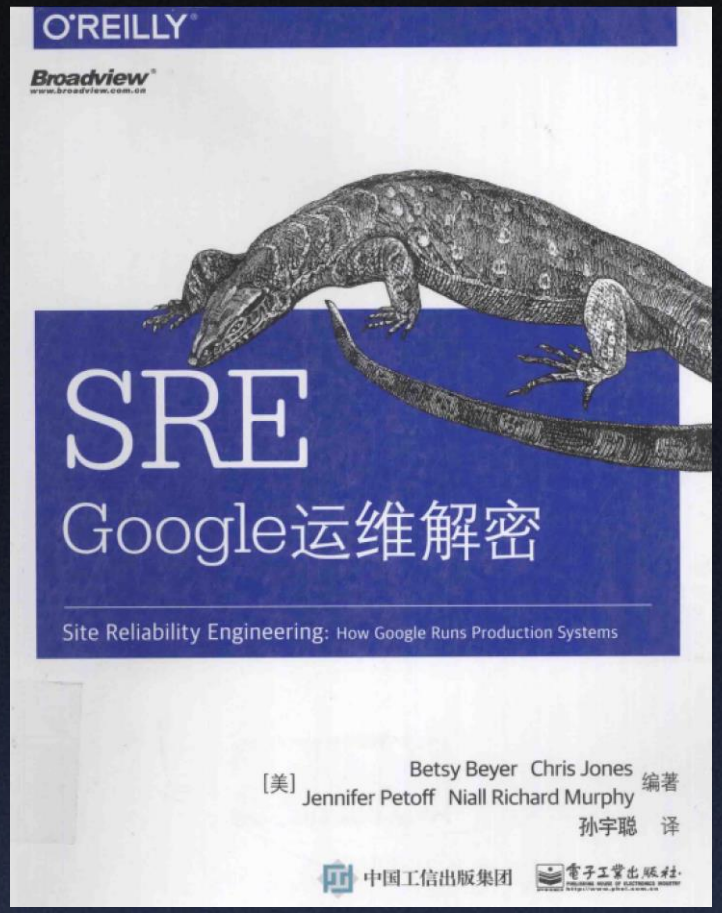
监控的常用概念

- **监控要解决的问题**
  a) 现象（什么东西出故障了）
  b) 原因（为什么故障）

- **黑盒监控**
  a) 通过测试某种外部用户可见的系统行为进行监控。
  b) 面向现象，代表了目前正在发生的问题，即“什么东西出故障了”。
  c) 黑盒监控可保证系统在某个问题正在发生，并造成某个现象的时候就会发出报警。

- **白盒监控**
  a) 依靠系统内部暴露的一些性能指标进行监控，如日志分析、链路分析等。
  b) 面向现象和原因，依赖对系统内部信息的检测，感知“为什么故障”。
  c) 白盒监控可以检测到系统即将发生的问题以及分析系统出现问题的原因。

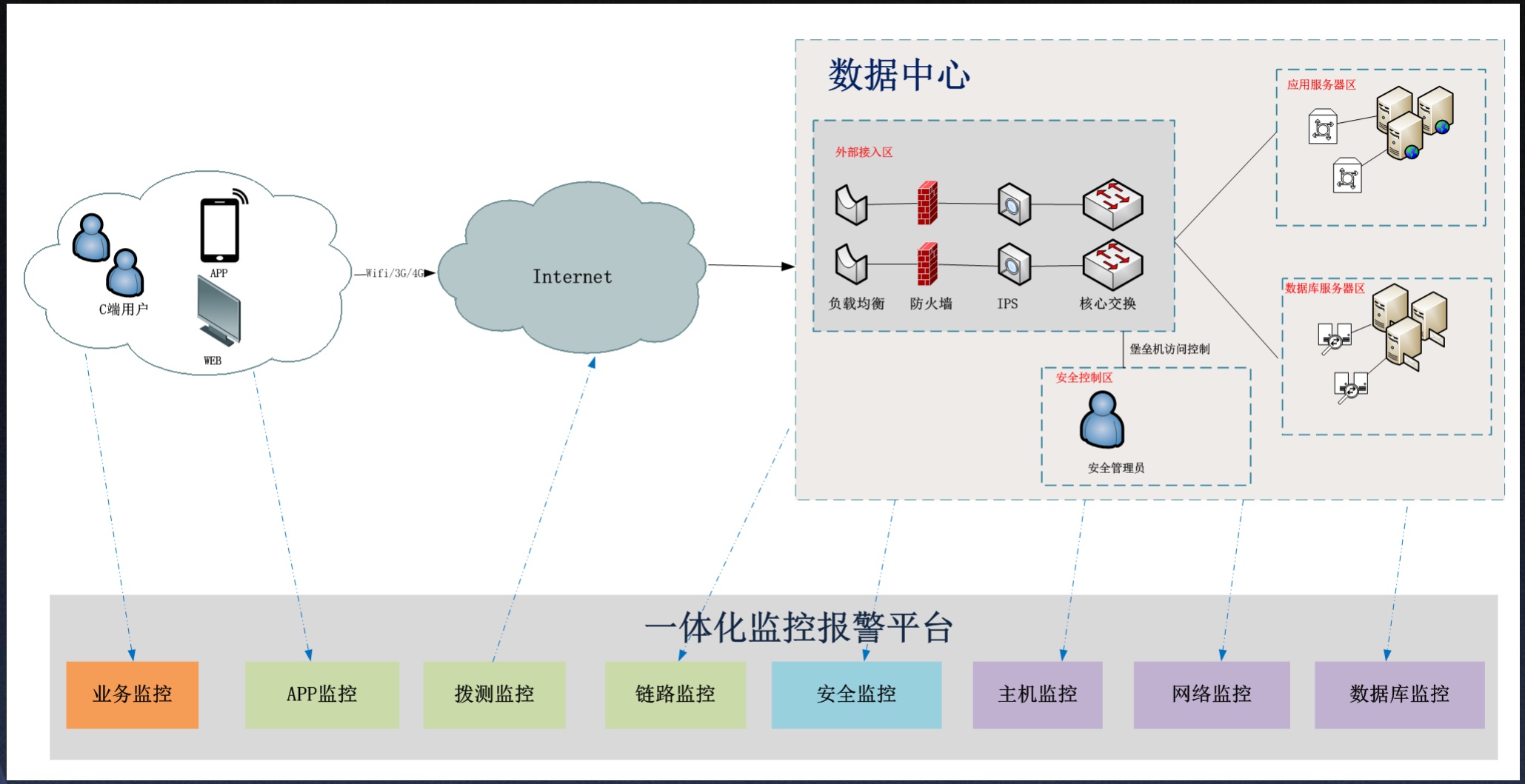
- **4个黄金指标**
  延迟、流量、错误、饱和度



监控的常见分类

- **应用层面的监控**
  包括服务的可用性、请求量、链路状态等，以及APP端的Crash率，卡顿等
- **运维层面的监控**
  包括主机资源、网络吞吐的情况，以及服务器上的各类中间件运行状况等
- **业务层面的监控：**
  包括核心的业务指标，用户行为，以及用户舆情等
- **安全层面的监控：**
  如安全态势感知，用户行为风控等

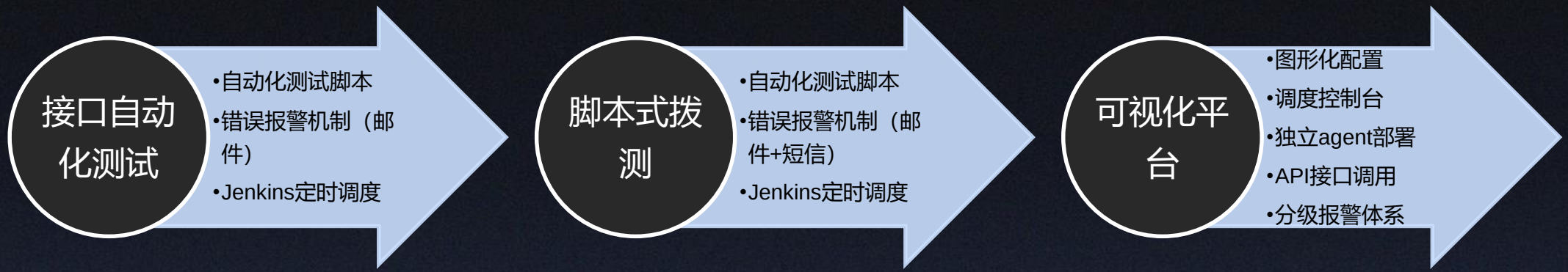
监控的常见分类


数据中心
外部接入区
负载均衡
防火墙
IPS
核心交换
应用服务器区
数据库服务器区
堡垒机访问控制
安全控制区
安全管理员
C端用户
APP
WEB
Wifi/3G/4G
Internet
一体化监控报警平台
业务监控
APP监控
拨测监控
链路监控
安全监控
主机监控
网络监控
数据库监控

02

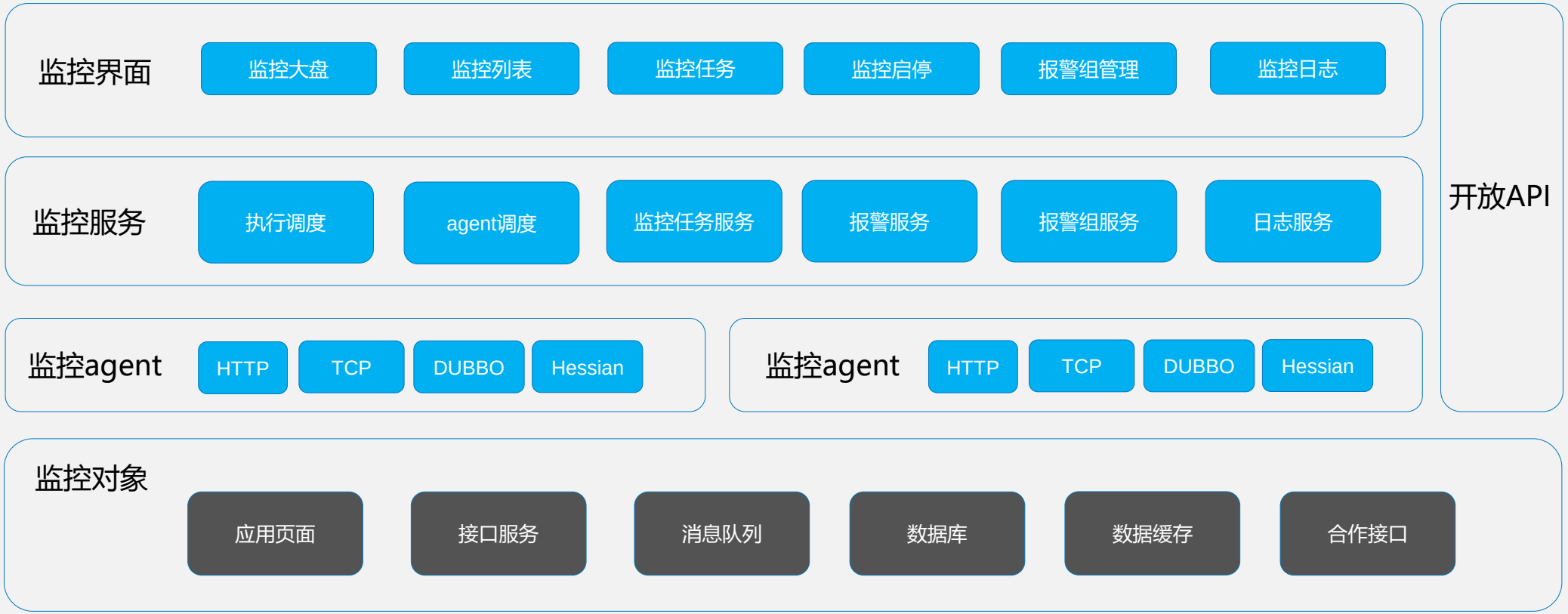
应用层面的监控

应用层面的监控 拨测监控



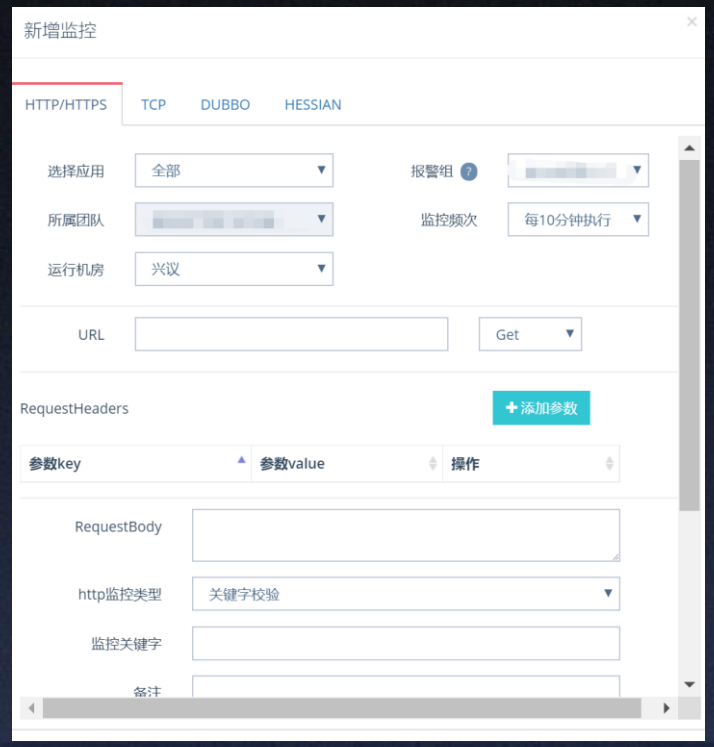
应用层面的监控 拨测监控

研发协作平台：拨测监控



应用层面的监控 拨测监控

新增监控 → 调试监控 → 定时执行 → 异常报警 → 错误日志 → 监控度量



应用层面的监控 链路监控

- **为什么需要链路监控**

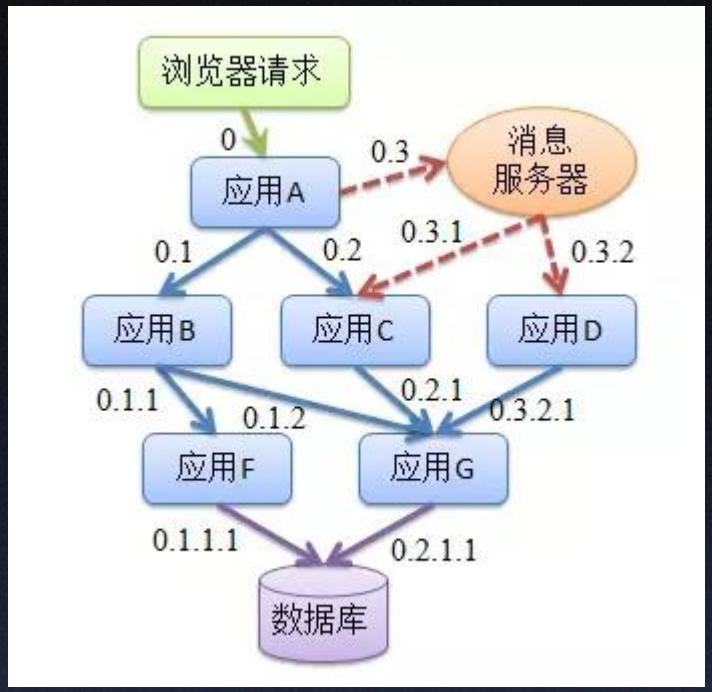
  随着微服务架构的实施，各大型系统解耦成多个微服务，一个完整的调用过程可能横跨多个服务，各服务调用情况会变得很复杂，复杂的调用导致很难对故障进行定位。

- **复杂调用链路带来的问题**

  如何快速发现问题?
  如何判断故障影响范围?
  如何梳理服务依赖以及依赖的合理性?
  如何分析链路性能问题以及实时容量规划?

- **什么是链路监控**

  链路监控的原理最早是由Google Dapper提出的，是一个分布式的监控系统。链路监控（TracingAnalysis）为分布式应用的开发者提供了完整的调用链路还原、调用请求量统计、链路拓扑、应用依赖分析等工具，能够帮助开发者快速分析和诊断分布式应用架构下的性能瓶颈，提高微服务时代下的开发诊断效率。



应用层面的监控 链路监控

- **gtrace客户端(SDK插件)**

  采集链路数据

- **gtrace服务端**

  为Web UI提供查询接口

  跑批量定时任务

- **Web UI**

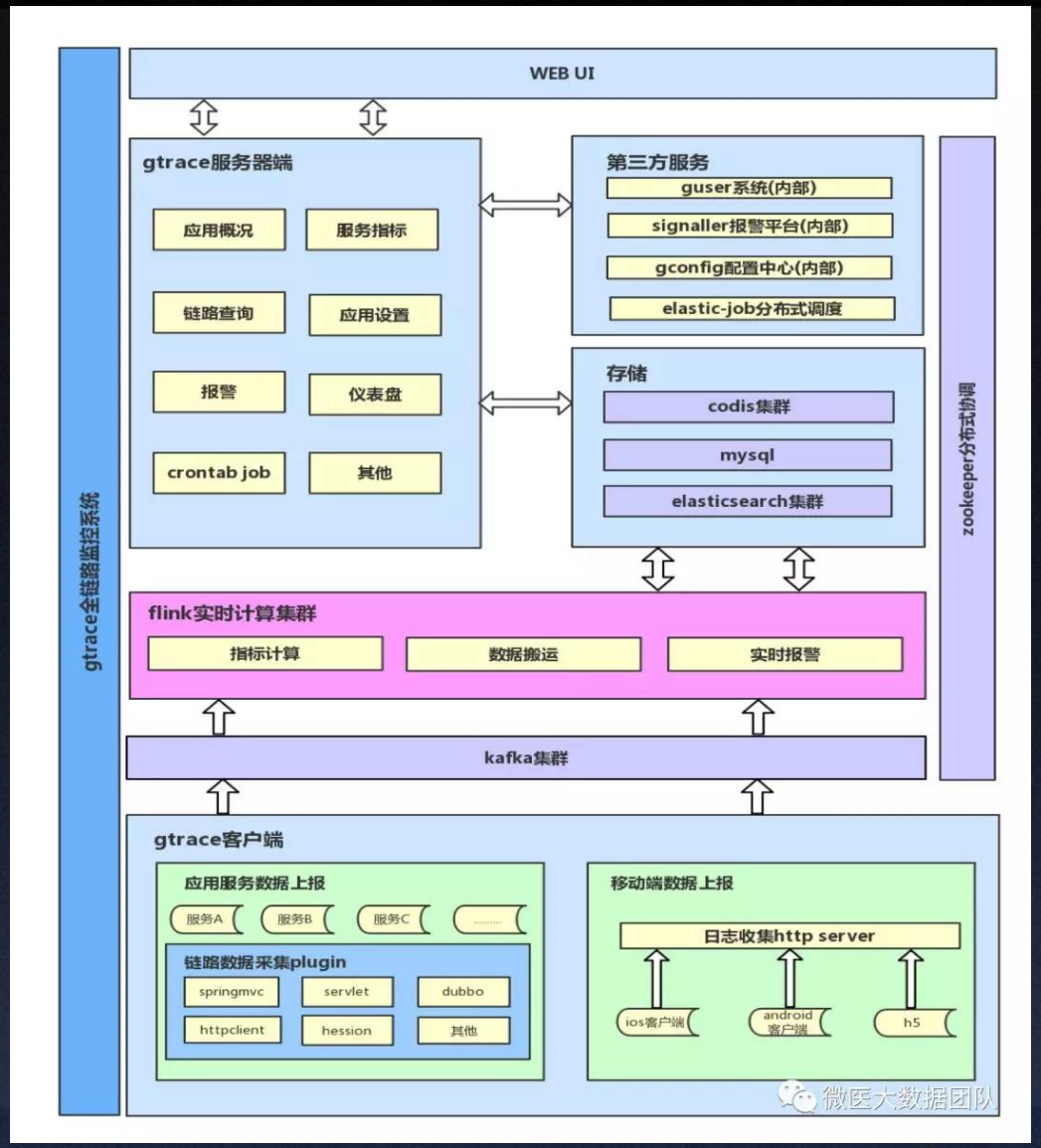
  展示链路数据、指标统计数据以及收集配置数据

- **flink实时计算集群**

  指标计算、数据搬运分发

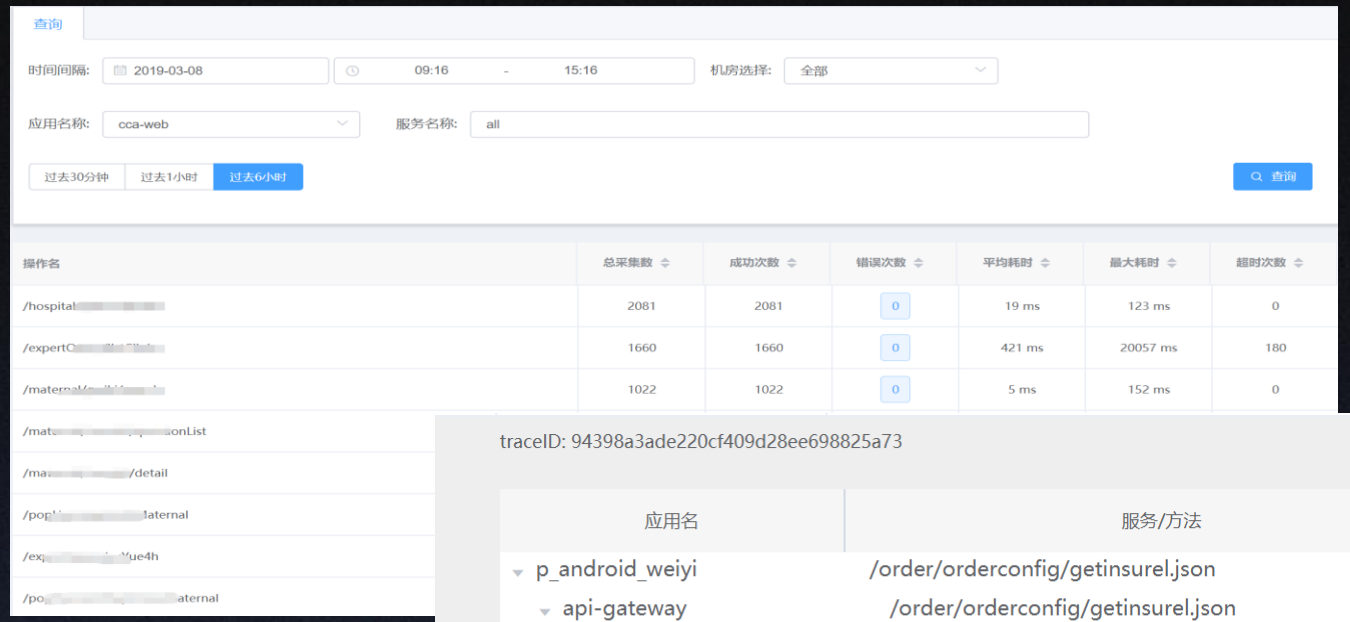
- **存储**

  es、mysql、redis等存储介质



应用层面的监控 链路监控

- 调用耗时
- 出入参数
- 附加业务id
- 服务异常日志
- 调用结果
- 返回码
- 强弱依赖分析



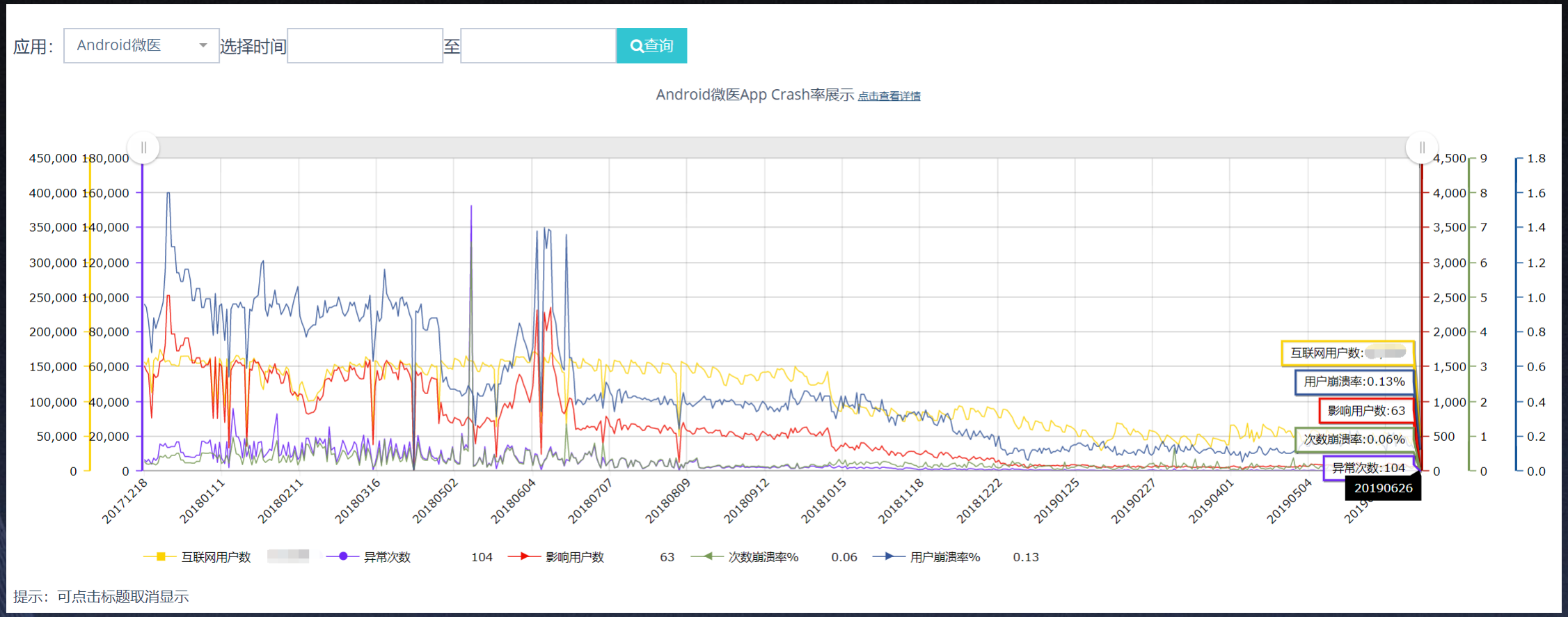
应用层面的监控

拨测监控 VS 链路监控

	拨测监控	链路监控
优点	1.资源需求低，一两台拨测服务器即可。 2.使用门槛低，简单图形化配置即可，与监控应用没有耦合。 3.扩展成本低，web/service/db/MQ都可以支持	1.可提供全局视野，快速分析问题，判断影响范围。 2.可提供链路性能分析能力，满足实时容量规划的需求。 3.可有效梳理服务依赖以及依赖的合理性。
缺点	1.黑盒监控，只能感知“什么东西出故障”，不能感知“为什么故障”。 2.监控粒度和覆盖率强依赖于人工设计。 3.单系统监控，报警原因定位比较困难，缺乏全局视野。	1.资源需求高，需要大量计算和存储等资源。 2.接入成本高，一般都需要各个应用在代码层做接入改造。 3.研发门槛高，应用接入少的话分析效果较差。

应用层面的监控 APP端监控

- 核心指标：Crash率、ANR、卡顿等



03 运维层面的监控

运维层面的监控

- **基础设施层**

  监控各个主机服务器资源，如CPU,内存,网络吞吐和带宽占用,磁盘I/O和磁盘使用等指标。

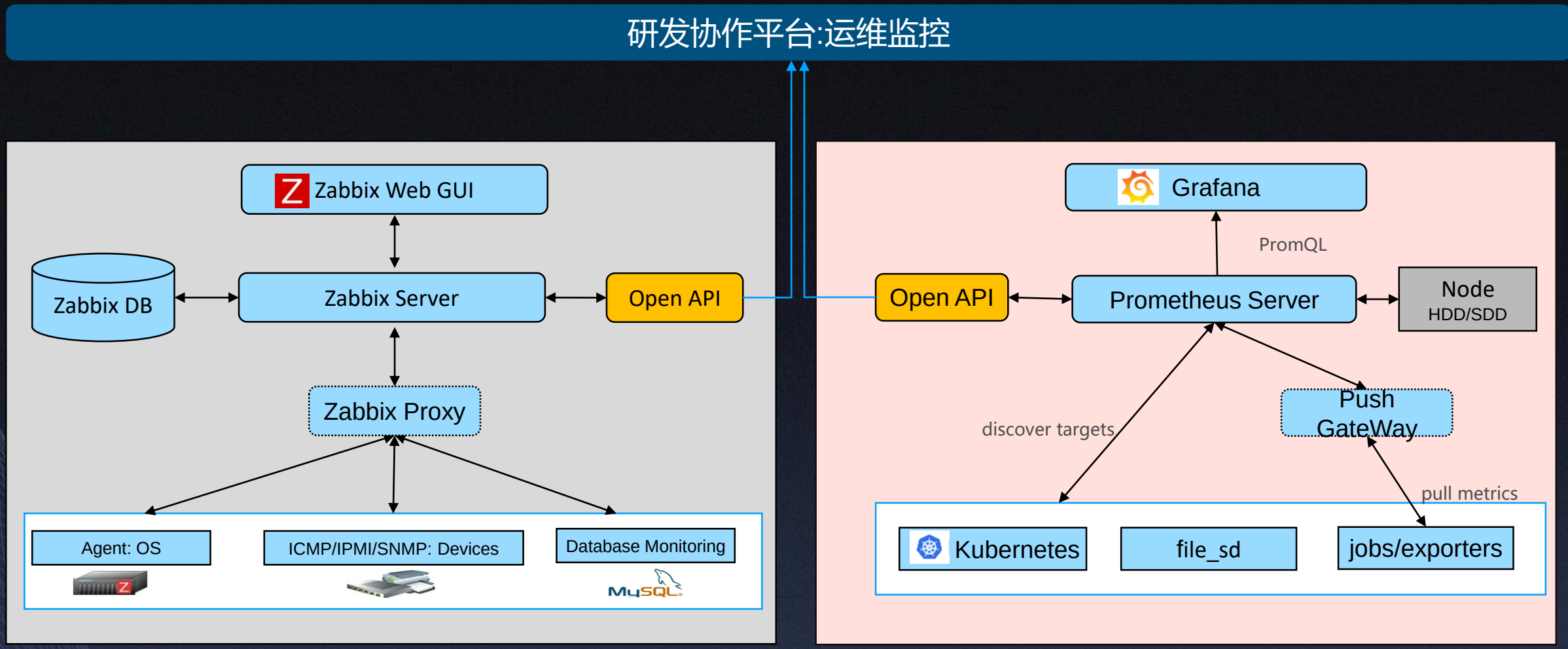
- **中间件层**

  监控独立部署于服务器上的各类中间件，例如：MySQL、Redis、RabbitMQ、ElasticSearch、Nginx等。

- **Kubernetes集群**

  监控Kubernetes集群本身的关键指标监控，以及Pod, DaemonSet, Deployment, Job, CronJob等各种资源对象的状态

运维层面的监控


研发协作平台:运维监控
Zabbix Web GUI
Zabbix DB
Zabbix Server
Open API
Zabbix Proxy
Agent: OS
ICMP/IPMI/SNMP: Devices
Database Monitoring
MySQL
Grafana
PromQL
Open API
Prometheus Server
Node
HDD/SDD
discover targets
Push
GateWay
pull metrics
Kubernetes
file_sd
jobs/exporters

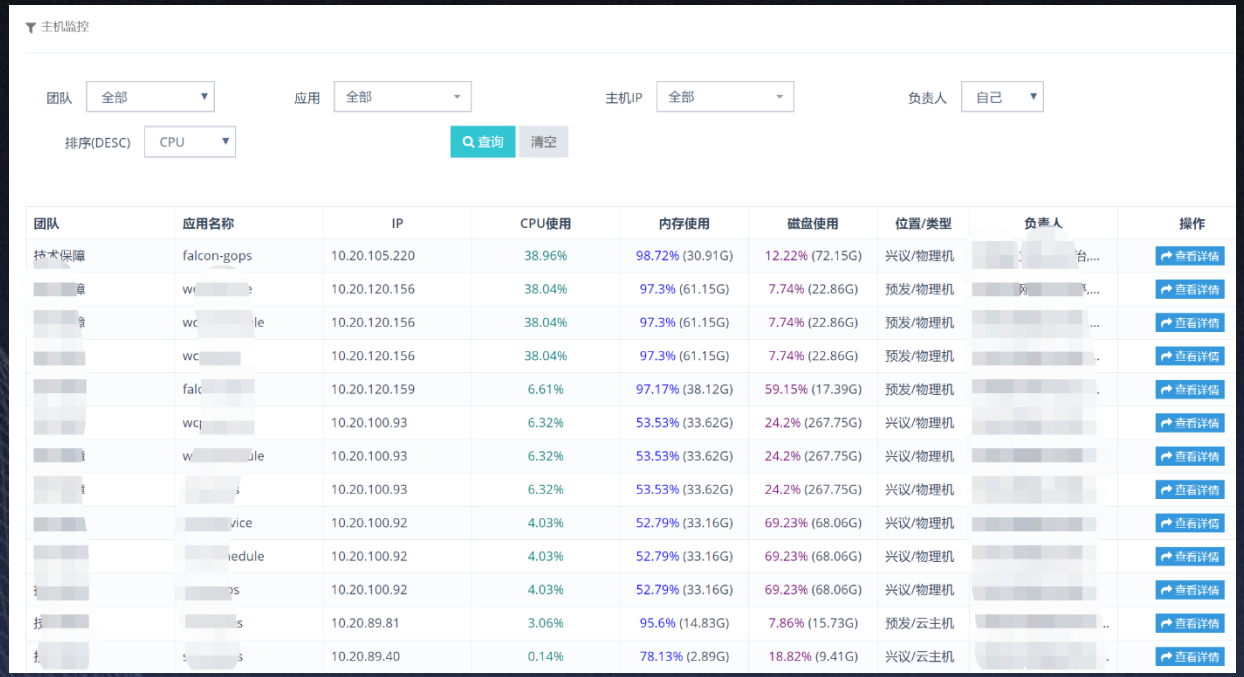
运维层面的监控

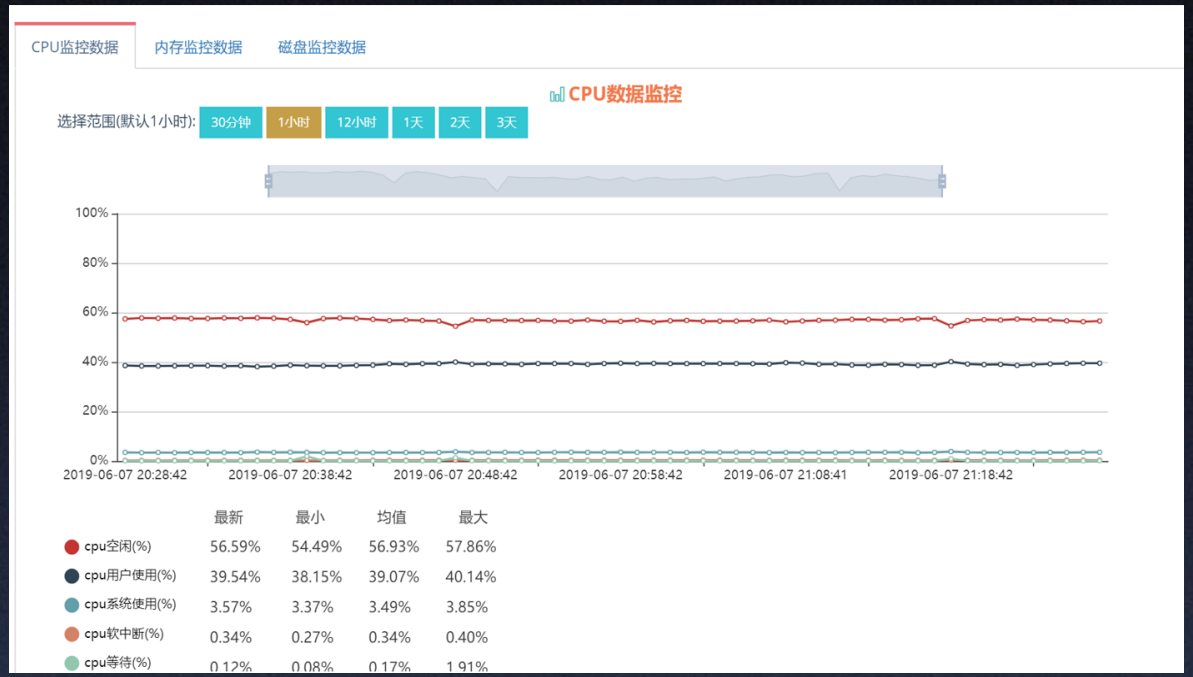
Zabbix VS Prometheus

Zabbix	Prometheus
后端用 C 开发，界面用 PHP 开发，定制化难度很高。	后端用 golang 开发，前端是 Grafana，JSON 编辑即可解决。定制化难度较低。
集群规模上限为 10000 个节点。	支持更大的集群规模，速度也更快。
更适合监控物理机环境。	更适合云环境的监控，对 OpenStack，Kubernetes 有更好的集成。
监控数据存储在关系型数据库内，如 MySQL，很难从现有数据中扩展维度。	监控数据存储在基于时间序列的数据库内，便于对已有数据进行新的聚合。
安装简单，zabbix-server 一个软件包中包括了所有的服务端功能。	安装相对复杂，监控、告警和界面都分属于不同的组件。
图形化界面比较成熟，界面上基本上能完成全部的配置操作。	界面相对较弱，很多配置需要修改配置文件。
发展时间更长，对于很多监控场景，都有现成的解决方案。	2015 年后开始快速发展，但发展时间较短，成熟度不及 Zabbix。

运维层面的监控 主机监控

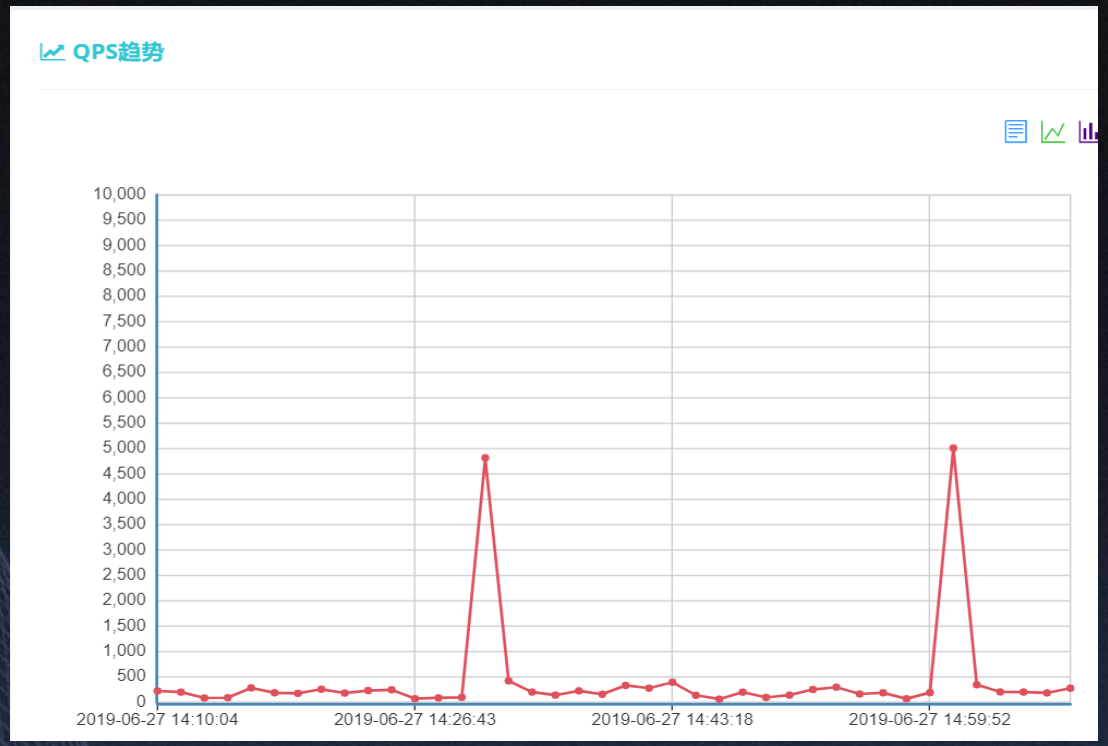
- 核心指标：CPU、内存、磁盘、网络吞吐等

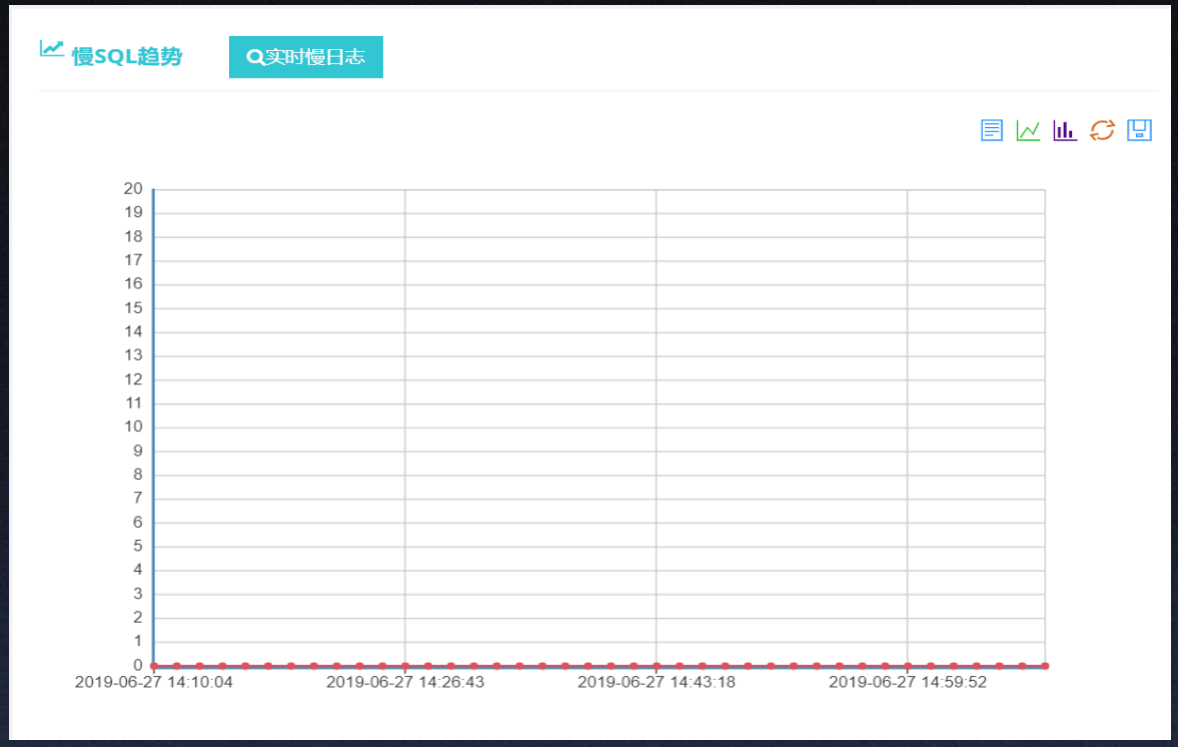




运维层面的监控 数据库监控

- 核心指标：QPS、连接数、慢SQL等

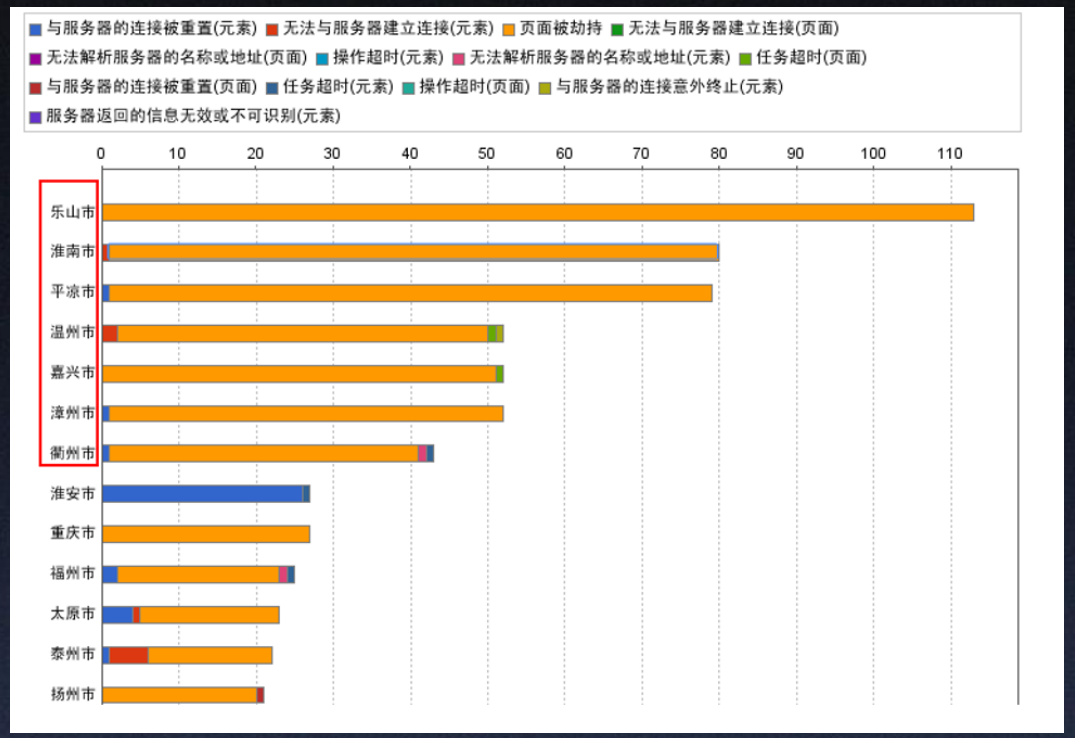




操作	任务ID	数据库名	用户名	慢SQL类型	百分之95慢SQL执行记录时间(秒)	总计花费时间(秒)	总计执行次数	首次出现慢SQL时间	最后一次出现慢SQL时间	最小慢SQL执行记录时间(秒)	最大慢SQL执行记录时间(秒)	ip	端口
详情	1722	anhao_four	shucang_anhao	commit	1.674	3.04	2	2019-06-21 06:01:02	2019-06-26 04:44:55	0.888	1.693	[redacted]	3315

运维层面的监控 网络监控

- 核心指标：多地监测、IDC流量监测、CDN监测

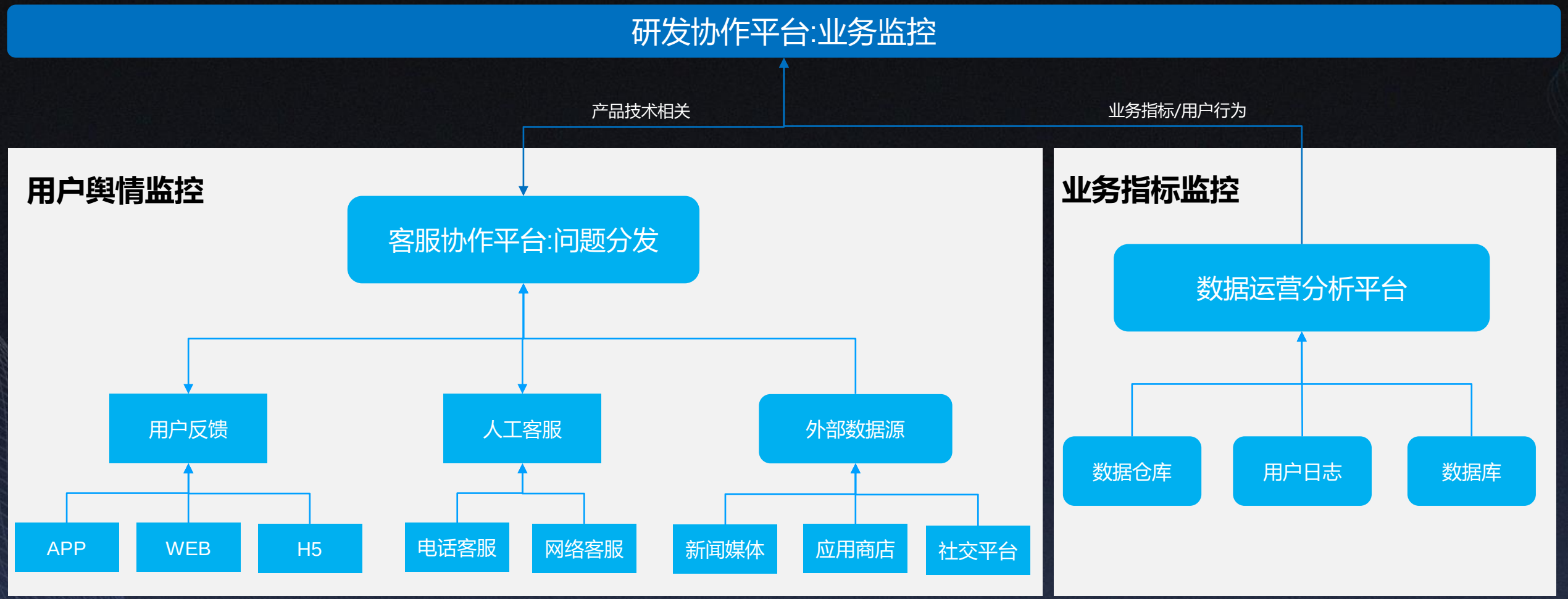


04 业务层面的监控

业务层面的监控

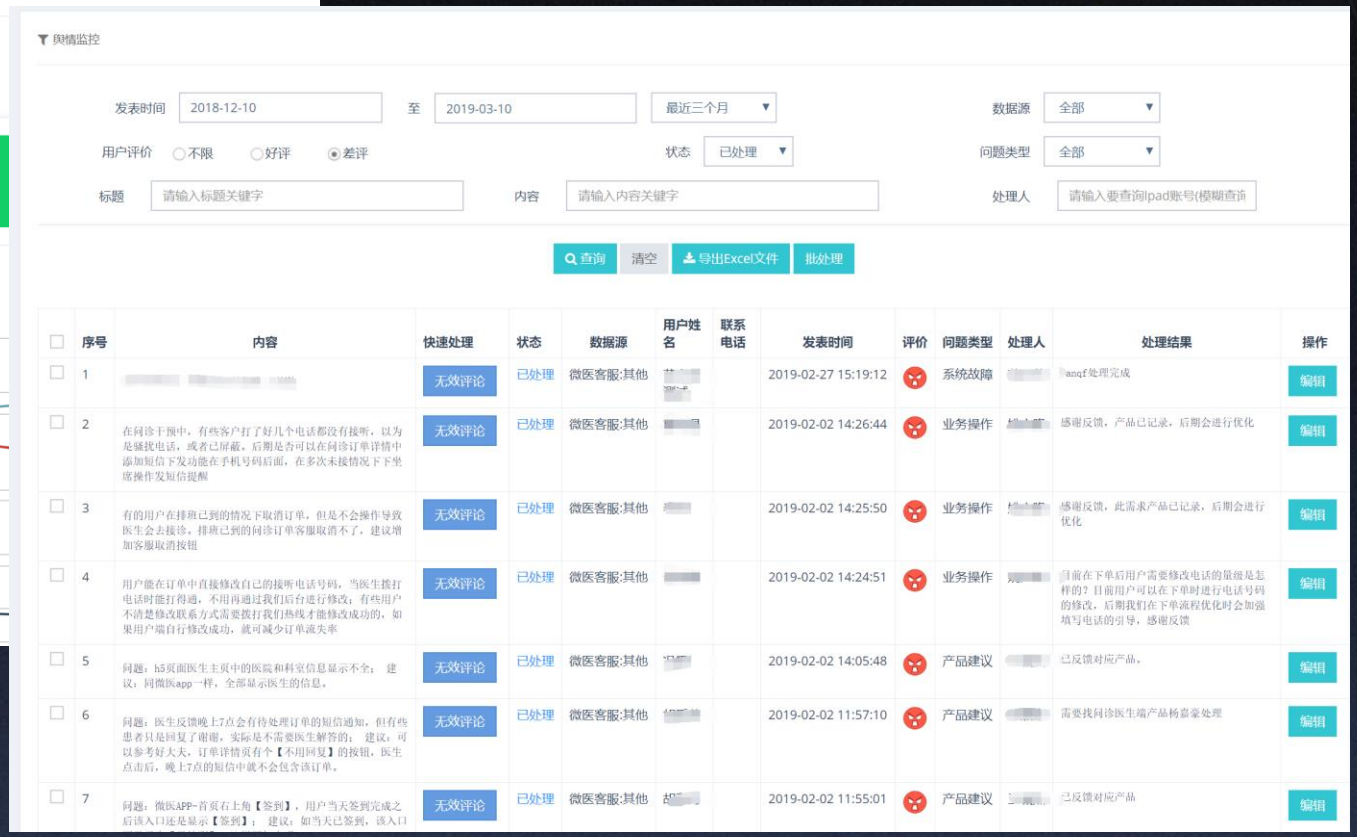
- **业务指标**
  - 用户量和趋势
  - 业务订单量和趋势
  - 资金流水量和趋势
  - 短信使用量和趋势
- **用户特征**
  - 页面访问统计
  - 终端比例(网络/终端/版本)
  - 用户行为习惯
- **用户舆情**
  - 客服反馈
  - 应用商店评论
  - 社区评论

业务层面的监控


研发协作平台:业务监控
产品技术相关
业务指标/用户行为
用户舆情监控
客服协作平台:问题分发
用户反馈
人工客服
外部数据源
APP
WEB
H5
电话客服
网络客服
新闻媒体
应用商店
社交平台
业务指标监控
数据运营分析平台
数据仓库
用户日志
数据库

业务层面的监控


WeData
运营分析
指标平台
流量分析
取数平台
资源分析
日期
2019-03-03
2019-03-09
昨日
最近7天
最近30天
客户端
用户端
产品线
页面
查询
终端
IOS
版本
全部
PV
1804950
UV（日累计）
671268
PV
UV
跳出率
350,000
300,000
250,000
200,000
150,000
100,000


舆情监控
发表时间
2018-12-10
至
2019-03-10
最近三个月
数据源
全部
用户评价
不限
好评
差评
状态
已处理
问题类型
全部
标题
请输入标题关键字
内容
请输入内容关键字
处理人
请输入要查询ipad账号(模糊查询
查询
清空
导出Excel文件
批处理
序号
内容
快速处理
状态
数据源
用户姓名
联系电话
发表时间
评价
问题类型
处理人
处理结果
操作
1
无效评论
已处理
微医客服:其他
2019-02-27 15:19:12
系统故障
aaqf处理完成
编辑
2
在问诊干预中，有些客户打了好几个电话都没有接听，以为是骚扰电话，或者已屏蔽。后期是否可以在问诊订单详情中添加短信下发功能在手机号码后面，在多次未接情况下下坐席操作发短信提醒
无效评论
已处理
微医客服:其他
2019-02-02 14:26:44
业务操作
感谢反馈，产品已记录，后期会进行优化
编辑
3
有的用户在排班已到的情况下取消订单，但是不会操作导致医生会去接诊，排班已到的问诊订单客服取消不了，建议增加客服取消按钮
无效评论
已处理
微医客服:其他
2019-02-02 14:25:50
业务操作
感谢反馈，此需求产品已记录，后期会进行优化
编辑
4
用户能在订单中直接修改自己的接听电话号码，当医生拨打电话时能打得通，不用再通过我们后台进行修改；有些用户不清楚修改联系方式需要拨打我们热线才能修改成功的，如果用户端自行修改成功，就可减少订单流失率
无效评论
已处理
微医客服:其他
2019-02-02 14:24:51
业务操作
目前在下单后用户需要修改电话的量级是怎样的？目前用户可以在下单时进行电话号码的修改，后期我们在下单流程优化时会加强填写电话的引导，感谢反馈
编辑
5
问题：h5页面医生主页中的医院和科室信息显示不全； 建议：同微医app一样，全部显示医生的信息。
无效评论
已处理
微医客服:其他
2019-02-02 14:05:48
产品建议
已反馈对应产品。
编辑
6
问题：医生反馈晚上7点会有待处理订单的短信通知，但有些患者只是回复了谢谢，实际是不需要医生解答的； 建议：可以参考好大夫，订单详情页有个【不用回复】的按钮，医生点击后，晚上7点的短信中就不会包含该订单。
无效评论
已处理
微医客服:其他
2019-02-02 11:57:10
产品建议
需要找问诊医生端产品杨嘉豪处理
编辑
7
问题：微医APP-首页右上角【签到】，用户当天签到完成之后该入口还是显示【签到】； 建议：如当天已签到，该入口
无效评论
已处理
微医客服:其他
2019-02-02 11:55:01
产品建议
已反馈对应产品
编辑

05

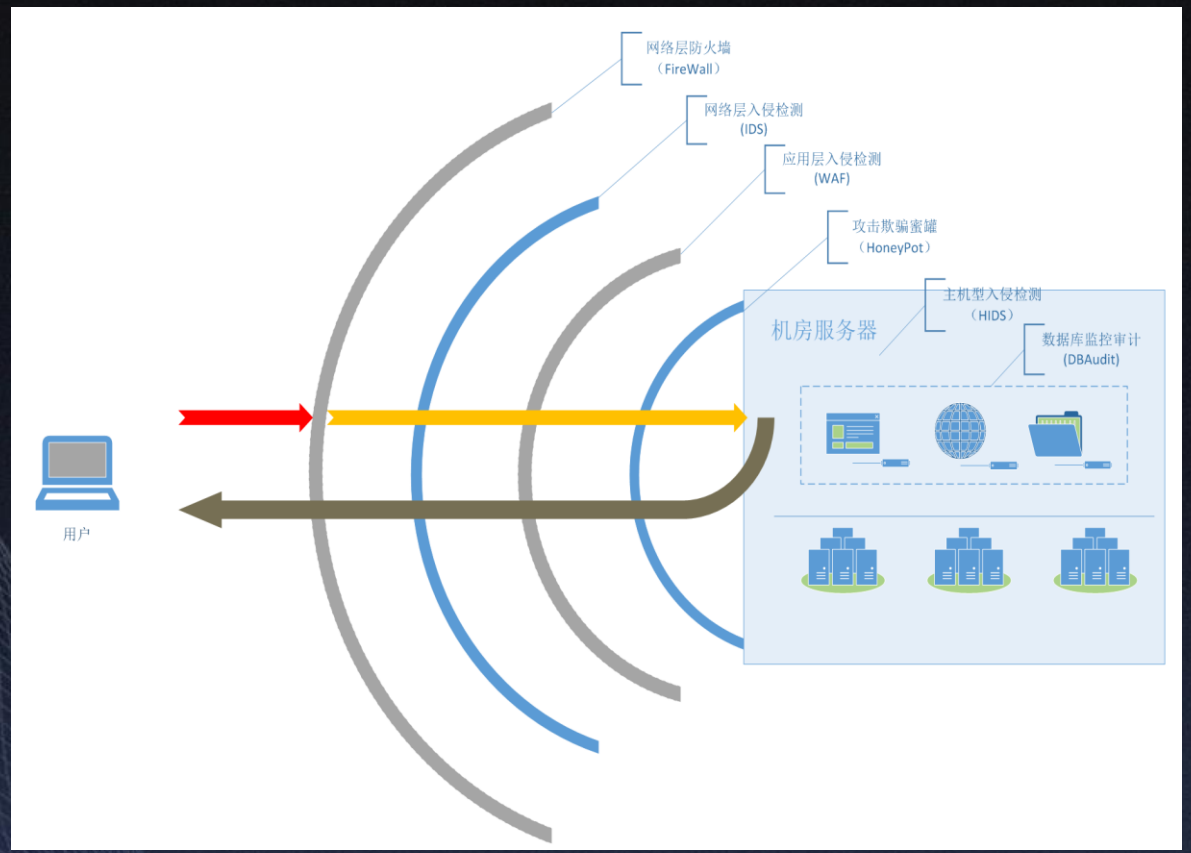
安全层面的监控

安全层面的监控

- **系统安全监控**
  - 主机安全
    - 主机入侵检测（HIDS）
    - 攻击欺骗蜜罐
  - 网络安全
    - 网络入侵监测（IDS）
    - 应用防火墙（WAF）
  - 数据库安全
    - SQL脱裤
    - 操作审计
  - 威胁情报
    - 开放漏洞库
    - 代码平台
    - 云盘
    - 安全社区、暗网等
- **业务安全风控**
  - 行为风控
    - 登录/注册/找密
    - 交易订单
    - 支付行为
    - 优惠券和活动
    - 医疗行为等
  - 内容风控
    - 文字
    - 图片
    - 语音
    - 视频等

安全层面的监控

系统安全



业务安全

安全层面的监控

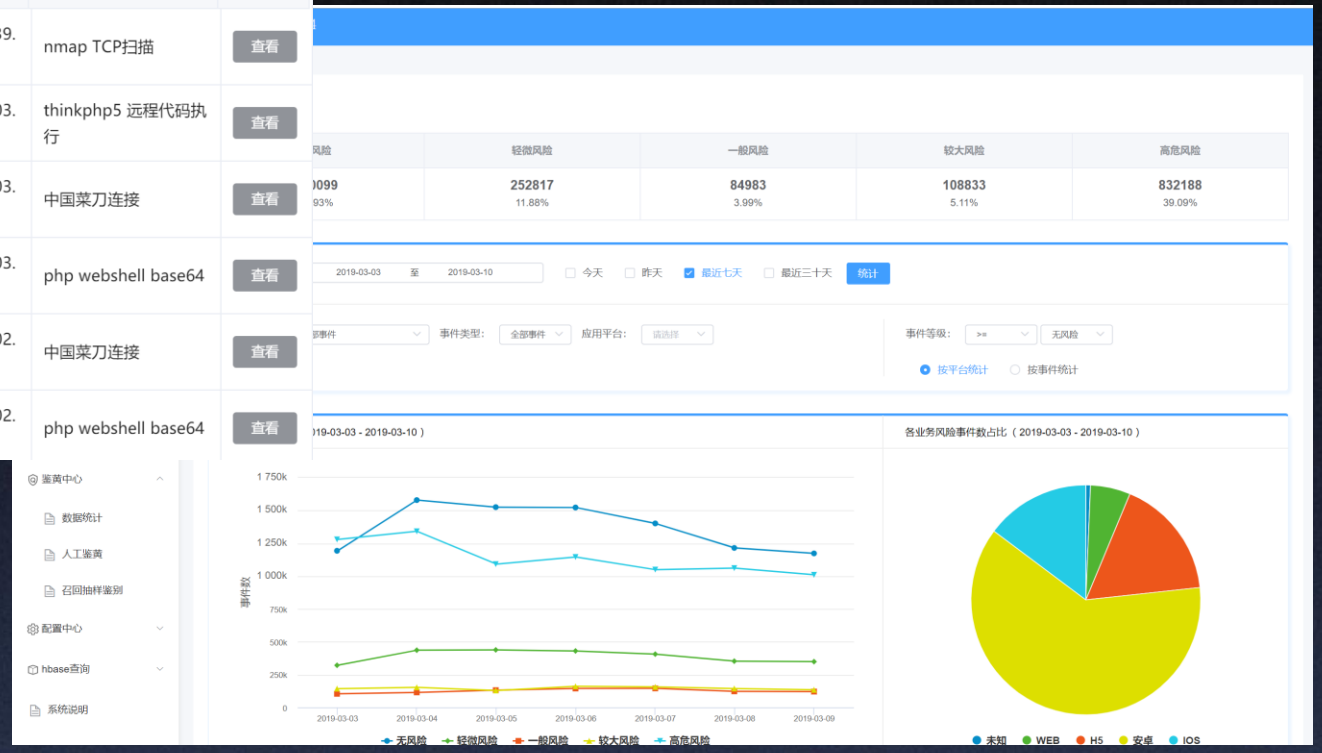
首页 / 网络入侵检测 / **监控日志**

攻击源IP: 请输入攻击源IP　攻击目的IP: 攻击目的IP　攻击源端口: 攻击源端口　攻击目的端口: 攻击目的端口

机房: 全部　触发规则: 全部　日志关键词: 日志关键词　日志时间: 开始时间 ~ 结束时间

查询

攻击源IP	攻击目的IP	攻击源端口	攻击目的端口	机房	攻击时间	触发规则	操作
140.207.35.5	[redacted]	80	7001	[redacted]	2019-06-13 09:00:39.0	nmap TCP扫描	查看
223.166.150.13	[redacted]	10206	80	[redacted]	2019-06-13 09:00:03.0	thinkphp5 远程代码执行	查看
223.166.150.12	[redacted]	61054	80	[redacted]	2019-06-13 09:00:03.0	中国菜刀连接	查看
223.166.150.12	[redacted]	61054	80	[redacted]	2019-06-13 09:00:03.0	php webshell base64	查看
223.166.150.125	[redacted]	21750	80	[redacted]	2019-06-13 09:00:02.0	中国菜刀连接	查看
223.166.150.125	[redacted]	21750	80	[redacted]	2019-06-13 09:00:02.0	php webshell base64	查看



06 多维一体化监控平台

多维一体化监控平台

- **单维度监控平台的不足**
  - **缺乏全局视角**
    主要反映的是单个业务或应用的运行状态，缺少全局的业务视角能反应整个“业务域”的上下游整体的运行情况。
  - **问题排查成本高**
    线上出现异常时，过于依赖开发人员个人的问题定位能力；很多时候都需要开发/测试/运维/DBA/安全等多个团队参与排查，各职能团队掌握的信息片面，问题原因定位难。
  - **监控标准不统一**
    各系统监控方式不统一，监控能力参差不齐，业务之间不能横向比较。
    各监控平台报警标准不统一，报警讯息泛滥、缺失或者混乱。

单维度监控效果有限，多维度立体化监控才是监控平台的根本之道。

多维一体化监控平台

研发协作平台

应用管理 → 代码管理 → 资源管理 → 交付流水线 → 发布管理 → 监控管理 → 度量管理

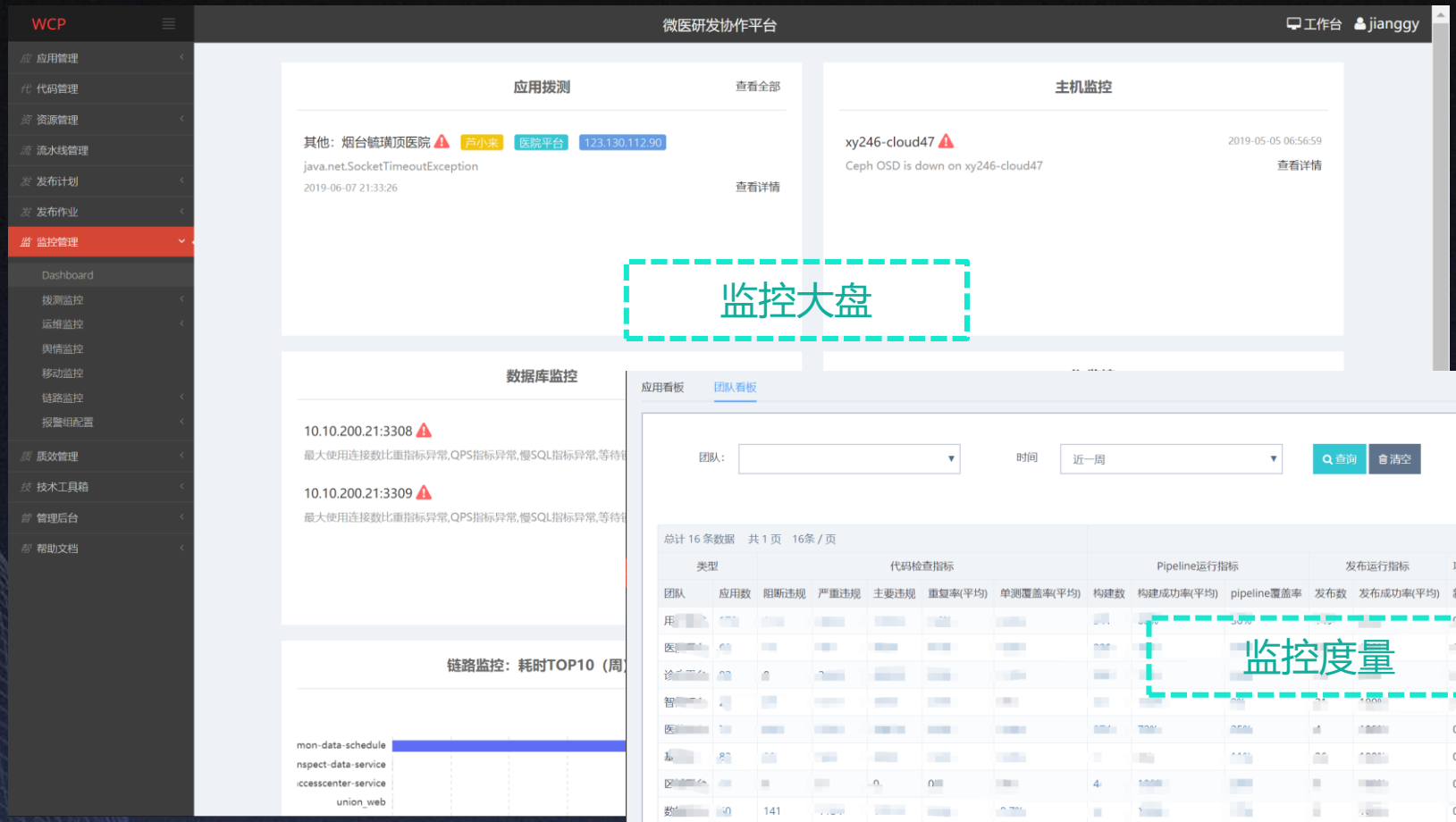
监控能力

- 应用监控
- 业务监控
- 运维监控
- 安全监控

监控组件

- 监控大盘
- 移动端监控
- 分级告警
- 监控度量

多维一体化监控平台


WCP
微医研发协作平台
工作台
jianggy
应用管理
代码管理
资源管理
流水线管理
发布计划
发布作业
监控管理
Dashboard
拔测监控
运维监控
舆情监控
移动监控
链路监控
报警组配置
质效管理
技术工具箱
管理后台
帮助文档
应用拔测
查看全部
其他：烟台毓璜顶医院
芦小来
医院平台
123.130.112.90
java.net.SocketTimeoutException
2019-06-07 21:33:26
查看详情
主机监控
xy246-cloud47
2019-05-05 06:56:59
Ceph OSD is down on xy246-cloud47
查看详情
监控大盘
数据库监控
10.10.200.21:3308
10.10.200.21:3309
链路监控：耗时TOP10（周）
mon-data-schedule
nspect-data-service
iccesscenter-service
union_web

应用看板
团队看板
团队：
时间
近一周
查询
清空
总计 16 条数据 共 1 页 16条 / 页
类型
代码检查指标
Pipeline运行指标
发布运行指标
项目指标
运营指标
资源指标
团队
应用数
阻断违规
严重违规
主要违规
重复率(平均)
单测覆盖率(平均)
构建数
构建成功率(平均)
pipeline覆盖率
发布数
发布成功率(平均)
新增漏洞
监控数
监控覆盖率
监控报警
外部缺陷
节点数
监控度量


23:00
微医集团(浙江)有限公司
数字化管理师
持"高薪"证书，提升职场竞争力
查看如何拿证
人事统计
7
0
考勤统计
未读日志
移动端监控
常用应用
签到
日志
考勤打卡
其他应用
监控大盘
微医CRM
OA办公系统
添加
消息
DING
微医
通讯录
我的

多维一体化监控平台 分级告警机制

- **常见的告警方法**

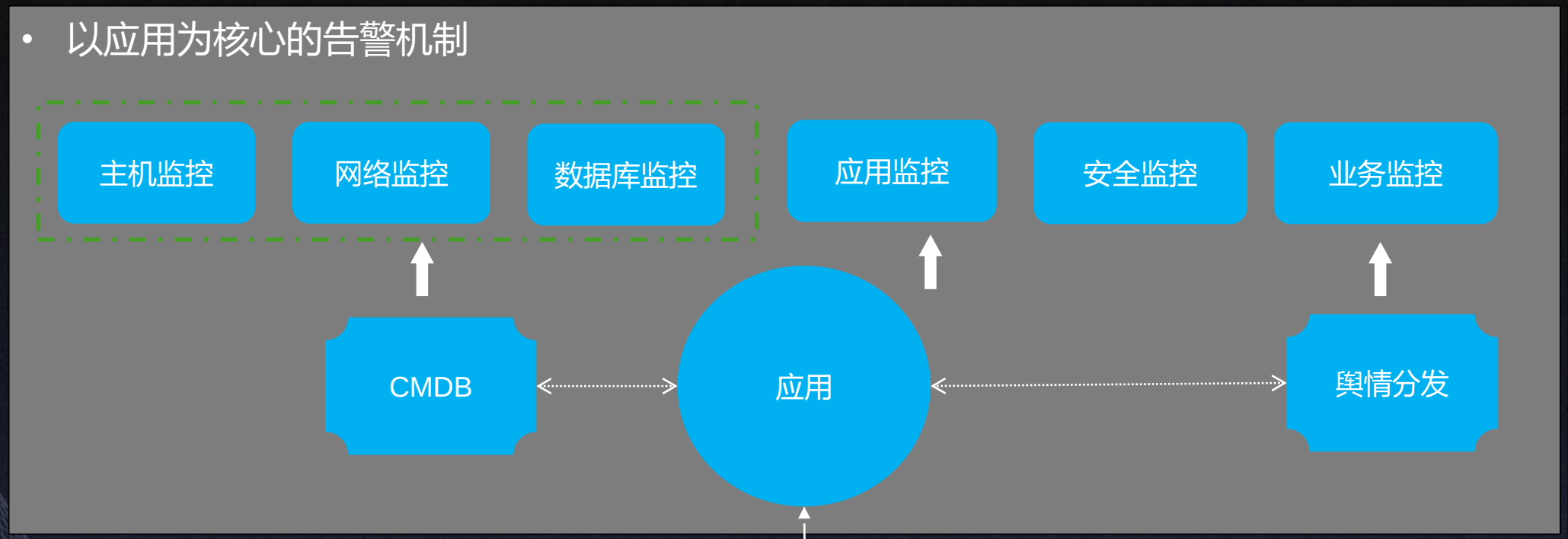
  短信：成本高，实时性好，到达率高
  邮件：成本低，实时性差，到达率高
  钉钉/微信：成本低，实时性中，到达率中

- **错误的告警方式**

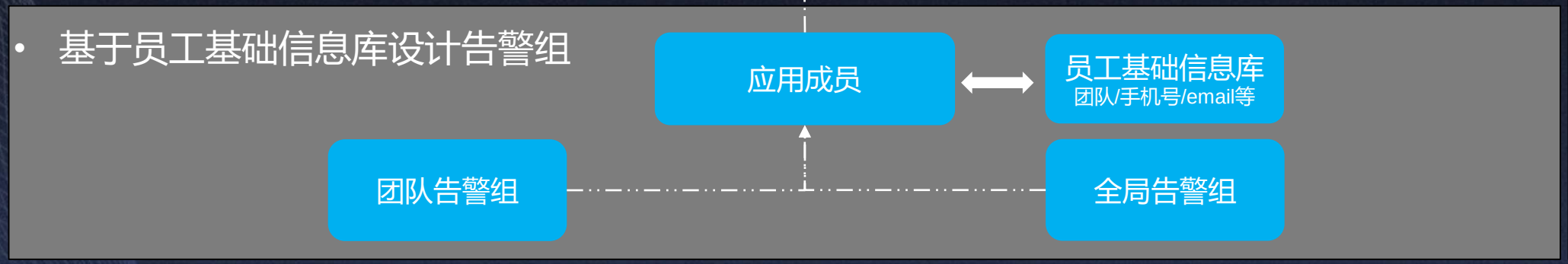
  系统负责人长期被告警短信刷屏，产生麻木感
  员工不重视告警，无法判断告警的优先级，leader又不知情，导致事故影响扩大
  告警信息来源不统一，格式混乱，让人无所适从
  系统负责人短时间内手机，邮箱，钉钉，微信同时对一个故障告警，导致产生巨大压力
  系统的告警人依赖手工配置，告警接收人信息维护困难，变更不及时

多维一体化监控平台 分级告警机制

- 以应用为核心的告警机制



- 基于员工基础信息库设计告警组



- 分级告警

问题分级

统一模板

频率收敛

时段区分

逐层上报

MTSC2019

中国移动互联网测试开发大会

Mobile Testing Summit China 2019

2019年6月28–29日 / 北京 国际会议中心

主办方：TesterHome 腾讯课堂