

GOPS

全球运维大会

2019 - AIOps 风向标

GOPS

深圳站

指导单位：



主办单位：



大会时间：2019年4月12日-13日

大会地址：深圳市南山区圣淘沙大酒店（翡翠店）

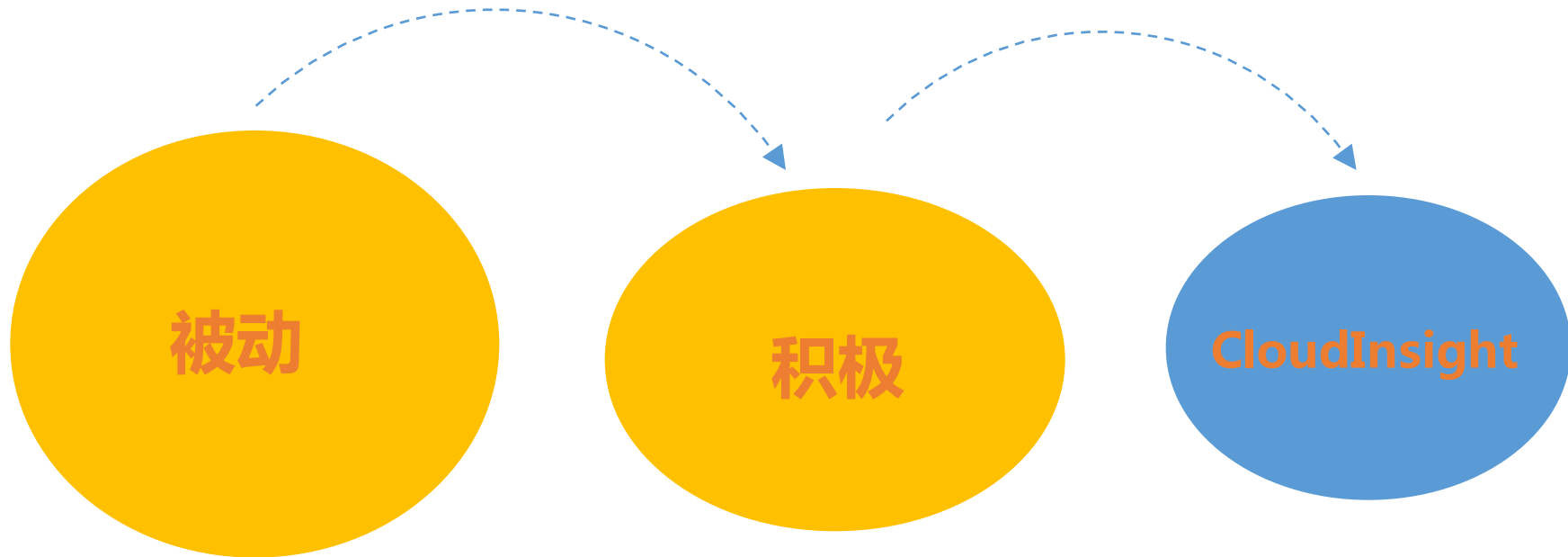
洞察 *CloudInsight*

百万级交易系统 *AI Ops* 架构实战

张俊卿 运行中心

引言

不放过任何一个故障！



目录



1

航信业务和运维情况概述

2

CloudInsight架构设计

3

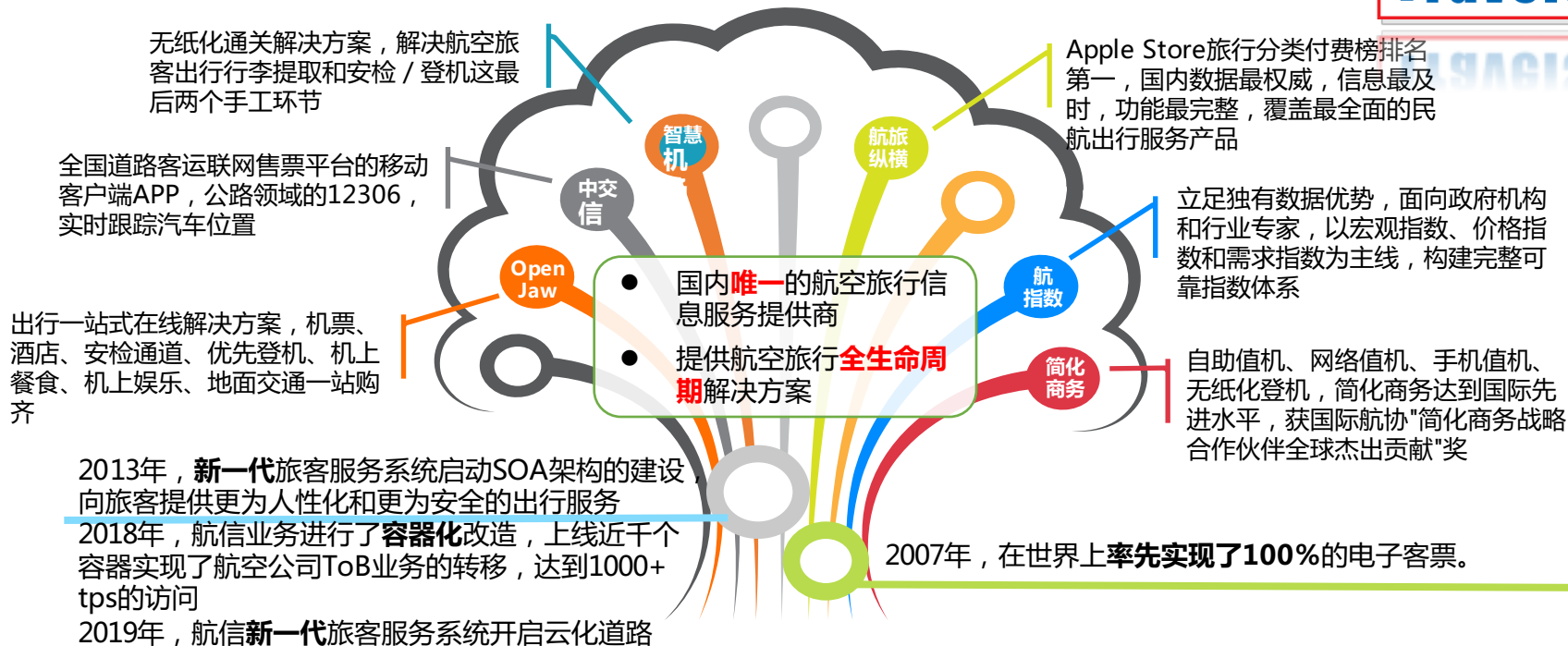
实际数据验证

4

未来努力的方向

中航信业务概述

中国航信
TravelSky



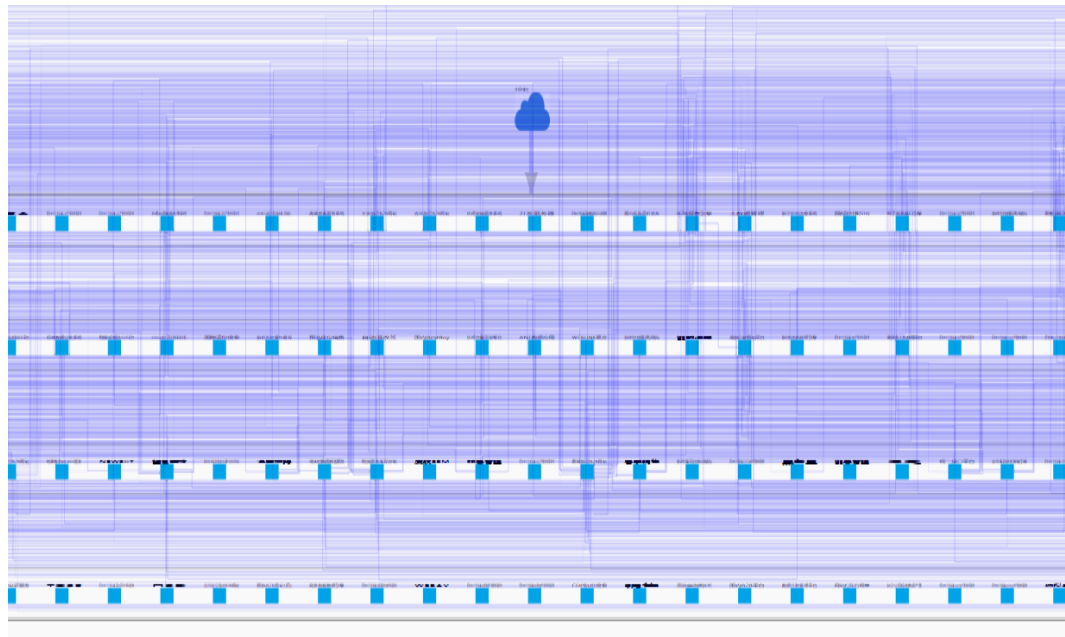
航信现有开放业务的特点

1. 业务种类架构繁多

- 根据功能区分为近400个业务
- 每个业务由多个产品集群组成
- 涉及4000+台服务器

2. 业务复杂度高

- 业务依赖关系复杂
- 核心业务基于SOA模式设计，但松耦合不彻底



航信开放运维的挑战

1. 开放业务日趋重要---》核心功能大多实现了外移
2. 业务量、服务器数量快速增加
 - 近三年以来，每年服务器数量以1000+以上的速度增长
 - 近三年总业务访问量以每年**30%**左右的速度增长
2. 变更频率及变更服务器台数快速增加
 - 变更频率从每周两次变更-->每天变更
 - 变更工单的服务器台数一年内有200+天可以达到100+台/次
 - 最大变更服务器数目达到**1500+**台/次

航信运维的目标

安全一直是我们追求的第一目标

- **把安全放在首位、让信息创造价值**

运维数据中存在着大量的宝藏：业务的特征与行为，正常与异常....

航信运维的组织分工



1. 产品运维能力极强

- 拥有近20年的运维经验，以主机运维的标准进行开放运维，拥有主机、系统、存储、数据库、网络、中间件等多种专业运维团队
- 运维安全性要求极高，业务分级维护，故障分级定性，四级业务系统要求零停机

2. 运维体系健全

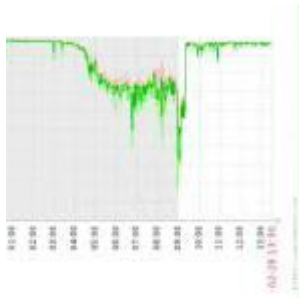
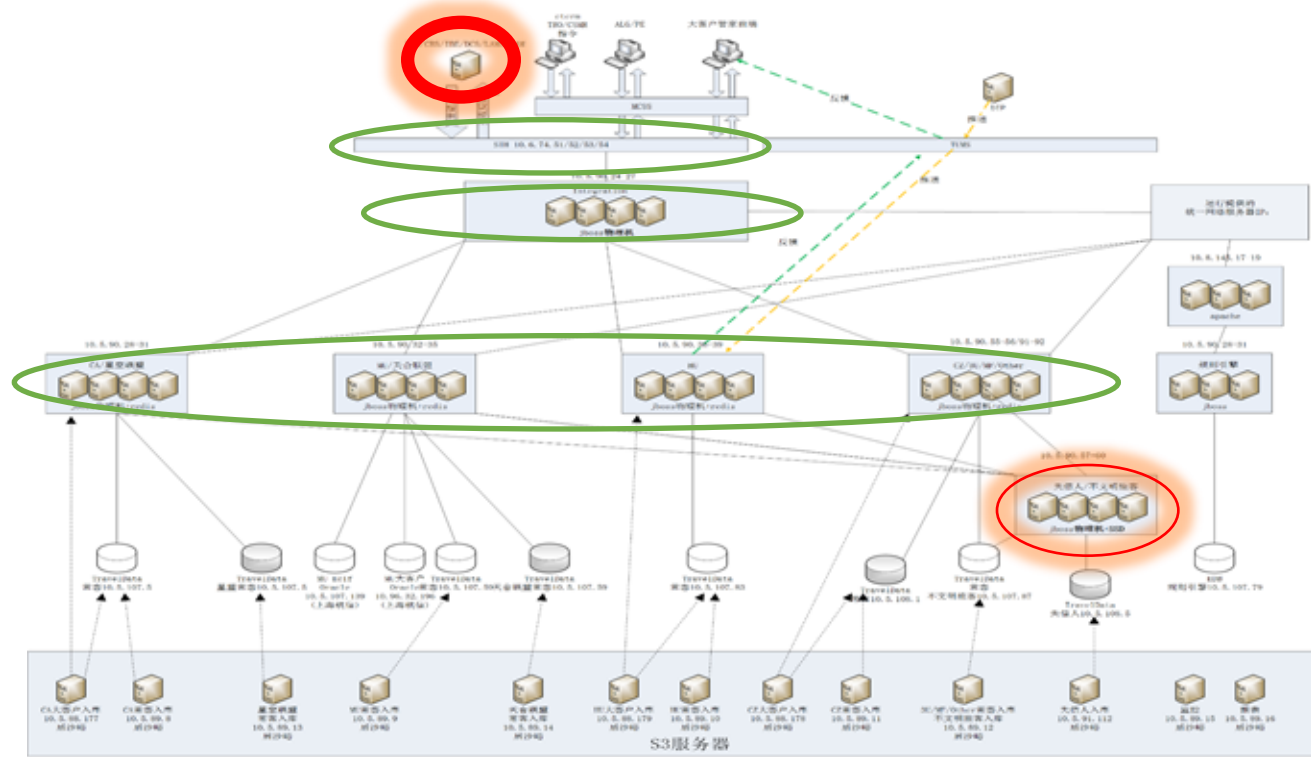
- 一线，二线，专家支持运维体系健全，各类传统运维工具完善

曾经发生的故障（开放某服务系统故障）- ？

业务监控

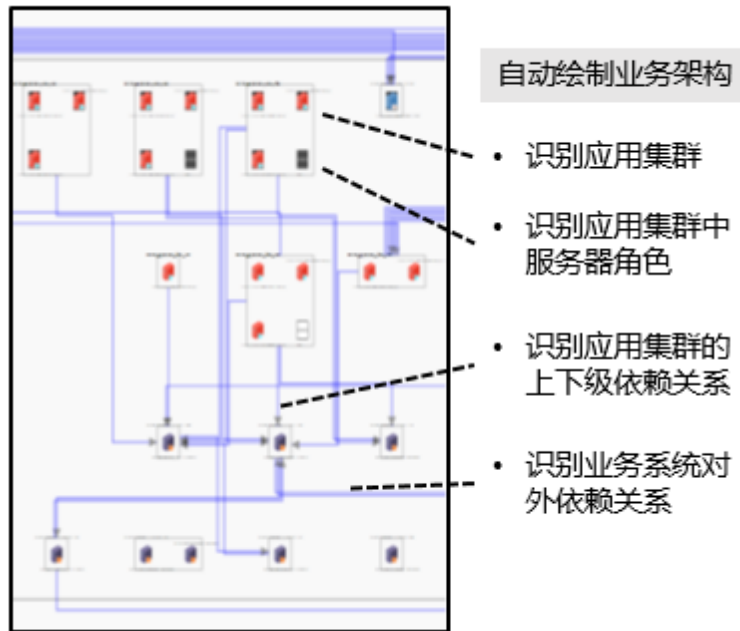
OOM监控

cpu监控



现有的运维工作成果

- 1 具有多种信息收集脚本和工具
- 2 完成自动识别服务器类别
- 3 完成了自动绘制业务架构
- 4 完成了业务之间和业务内部关系
数据的采集和存储



目录


1 航信业务和运维情况概述

➔ **2** CloudInsight架构设计

3 实际数据验证

4 未来努力的方向

CloudInsight的建设目标

- 整合现有数据，主动发现异常，定位问题 
- 根据异常综合判断，定位产品，快速启动产品的运维
- 建设运维知识库，传承运维经验

CloudInsight架构设计的关键要求

- 充分利用已有的成果和产品
- 架构具备**可扩展性**
 - 可扩展数据采集
 - 可扩展分析算法和规则
 - 可扩展存储架构和计算架构
 - 可扩展的可视化展示
- 架构具备**秒级处理性能**

CloudInsight架构图

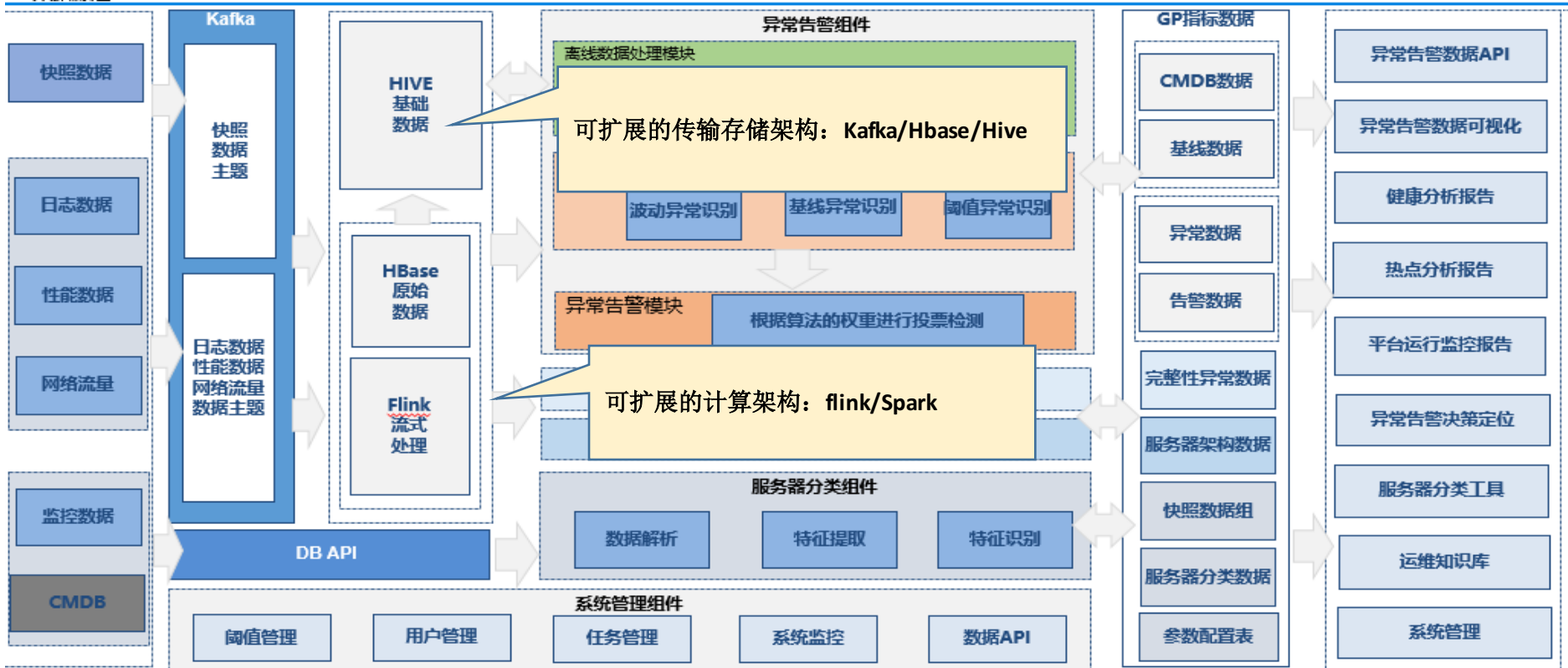
数据源层

数据采集预处理层

计算分析层

指标层

应用展示层



CloudInsight的数据输入

- 数据输入来自于快照数据、日志数据、指标数据、网络流量数据和监控数据，CMDB数据
- 按照数据特征分类为指标数据和信息数据
 - 指标数据：类似cpu idle信息、访问量信息等
 - 信息数据：应用输出日志、应用错误日志等
- 预处理后统一存储为如下格式：date|hostname|key|value

CloudInsight的数据示例

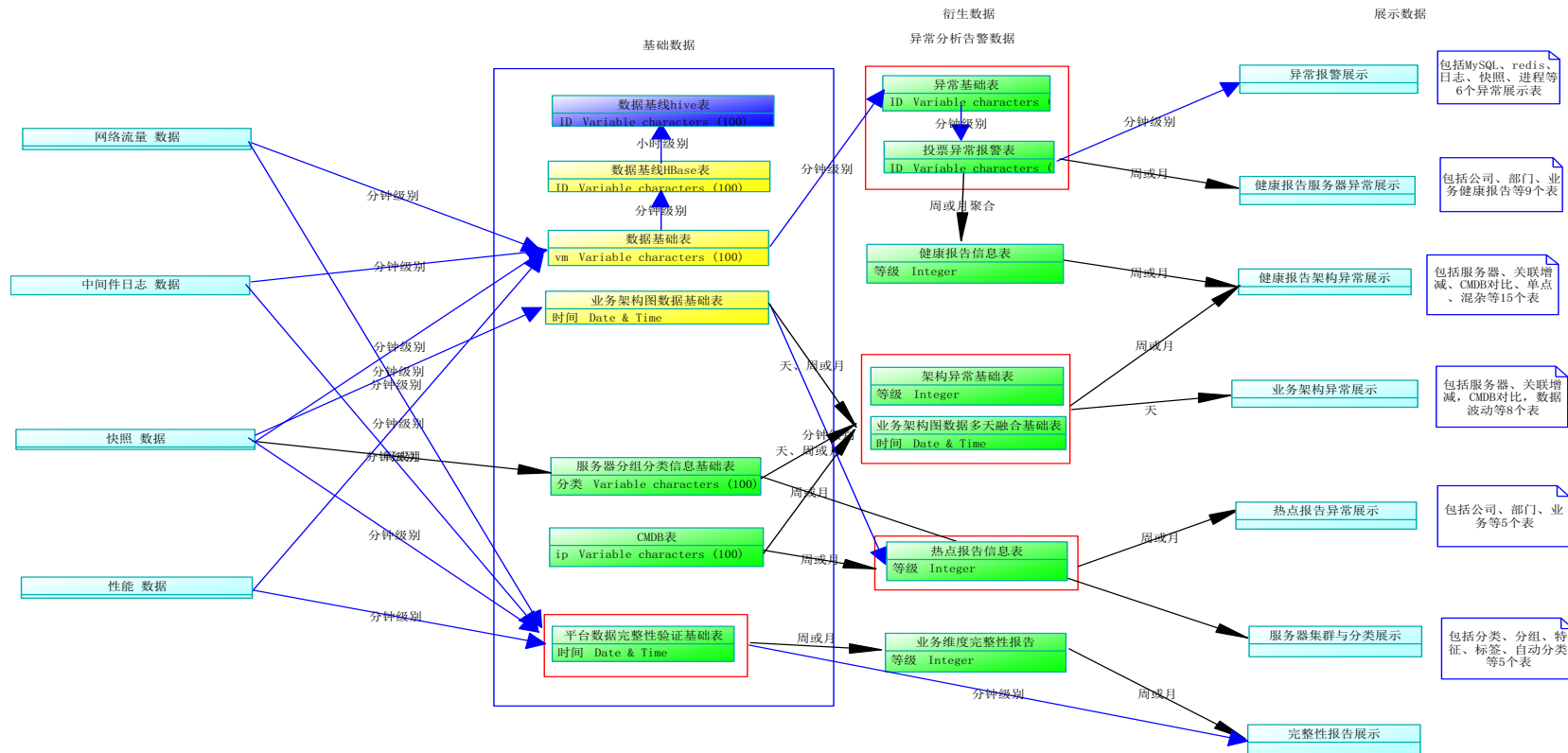
指标类型	指标	是否需要预处理
时序指标	源-目的-网络连接状态-个数	Y
	源-目的-接收包大小	Y
	源-目的-接收包数量	Y
	源-目的-发送包大小	
	源-目的-发送包数量	
	log- Apache-count	
	log- Apache-URL-Return_code	
	log- Apache-URL- receive_bytes	
	...	
	java-jdbcpool-剩余率	
	pid-vsز	
	pid-cpu	
	cpuinfo-user	
	cpuinfo-sys	
	cpuinfo-iowait	N
	cpuinfo-idle	N
信息指标	Jboss-server.log-error	Y
	Jboss-server.log-warning	Y
	Tode-TLOG.log	Y
	AMQ-activemq.log	Y
	lpcs	Y
	...	

Apache的URL作为访问的重要特征，一台服务器上会有多个，指标数据达到20+

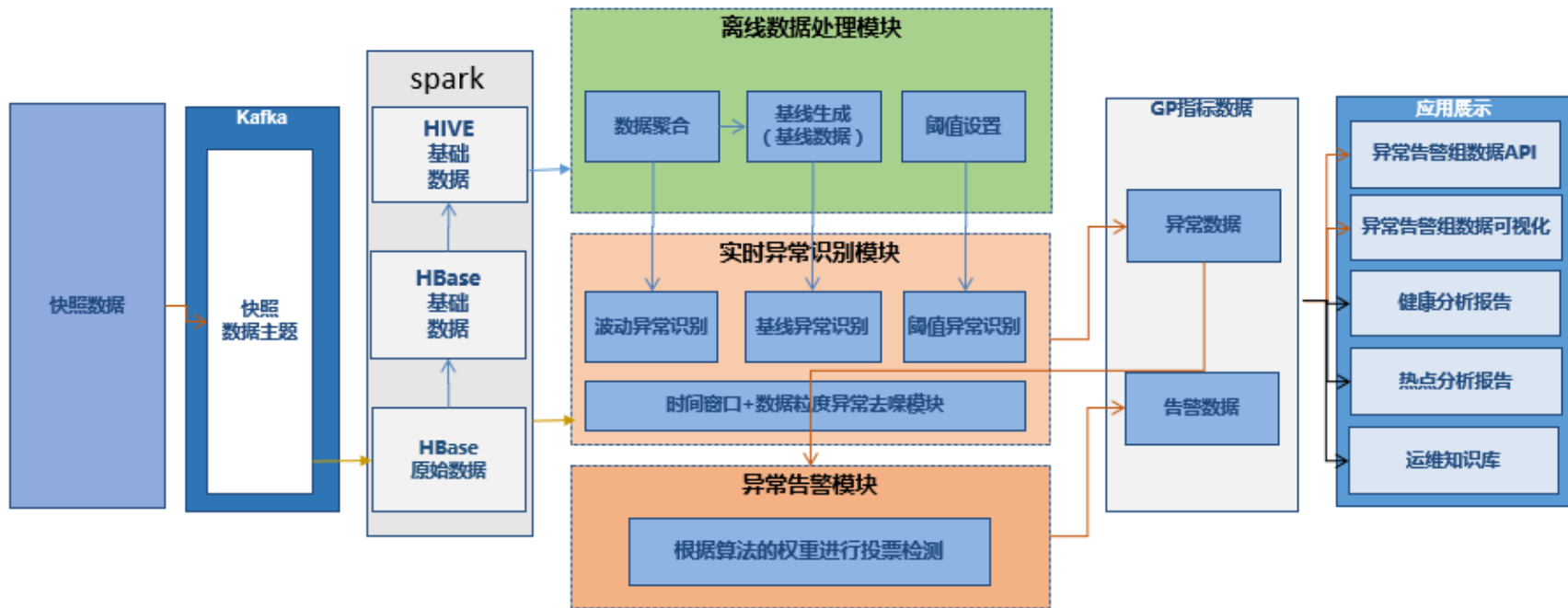
进程的vsz和cpu会根据服务器上的进程进行记录，一台服务器至少100以上

一台服务器根据类别计算之后至少要有200+的指标数据进行实时分析，按每一到两分钟进行采集计算，每秒钟需要计算1.5w个指标

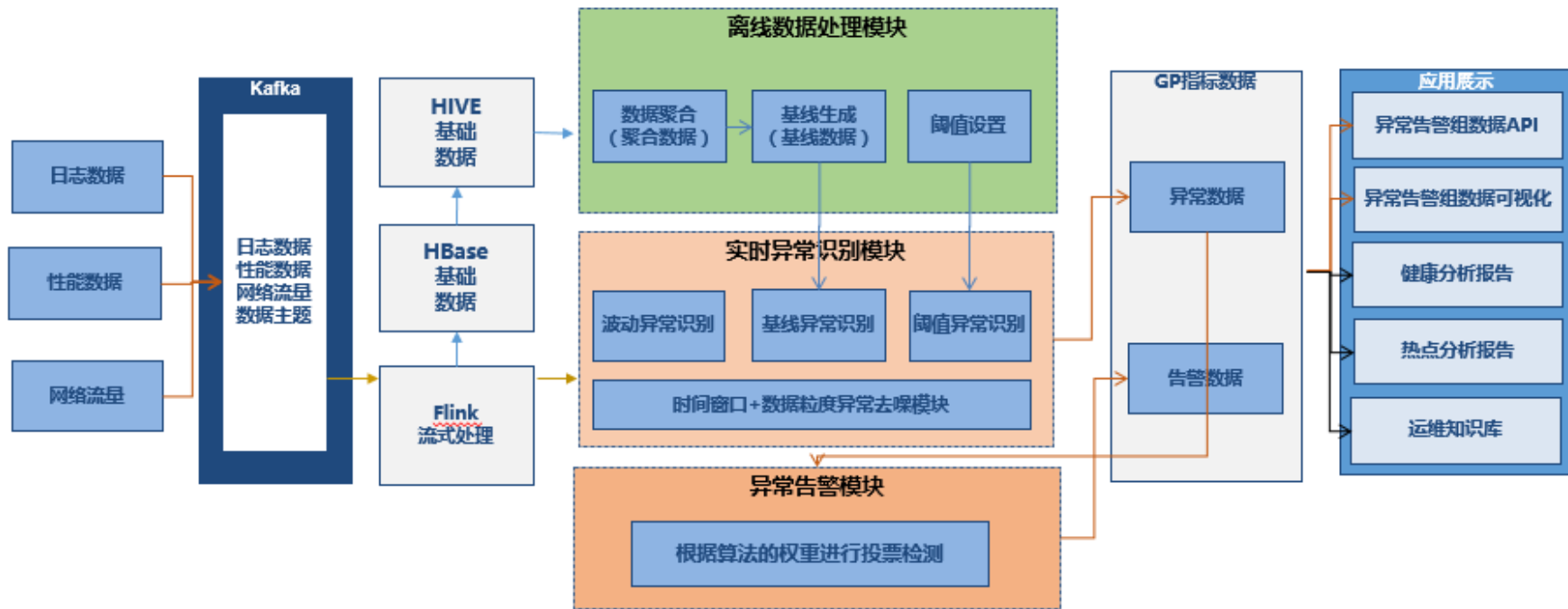
数据输入与展示的可扩展



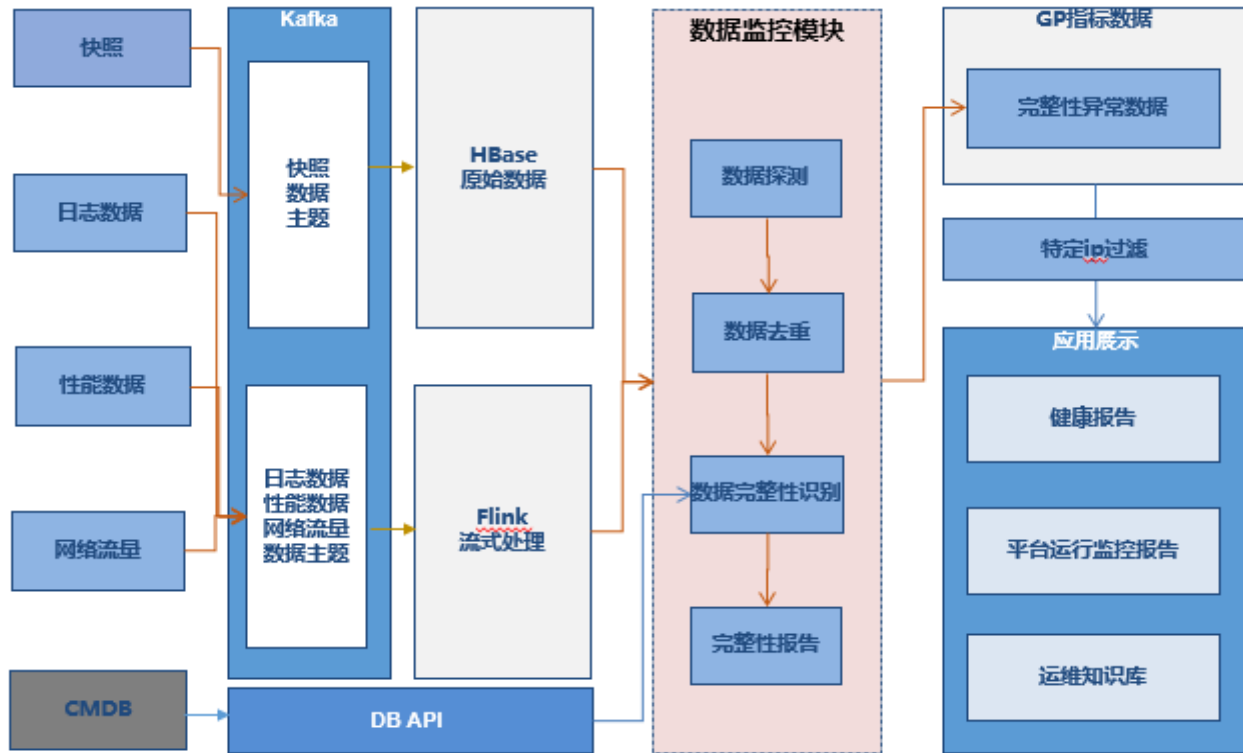
关键场景1:快照数据异常分析



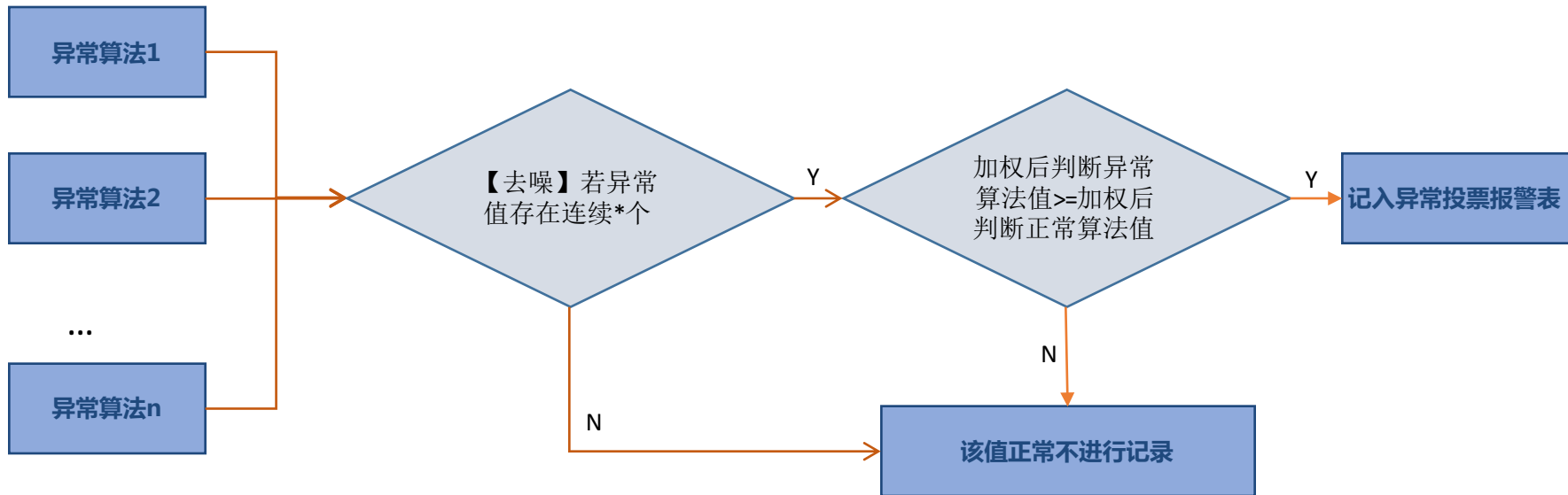
关键场景2：流数据异常分析



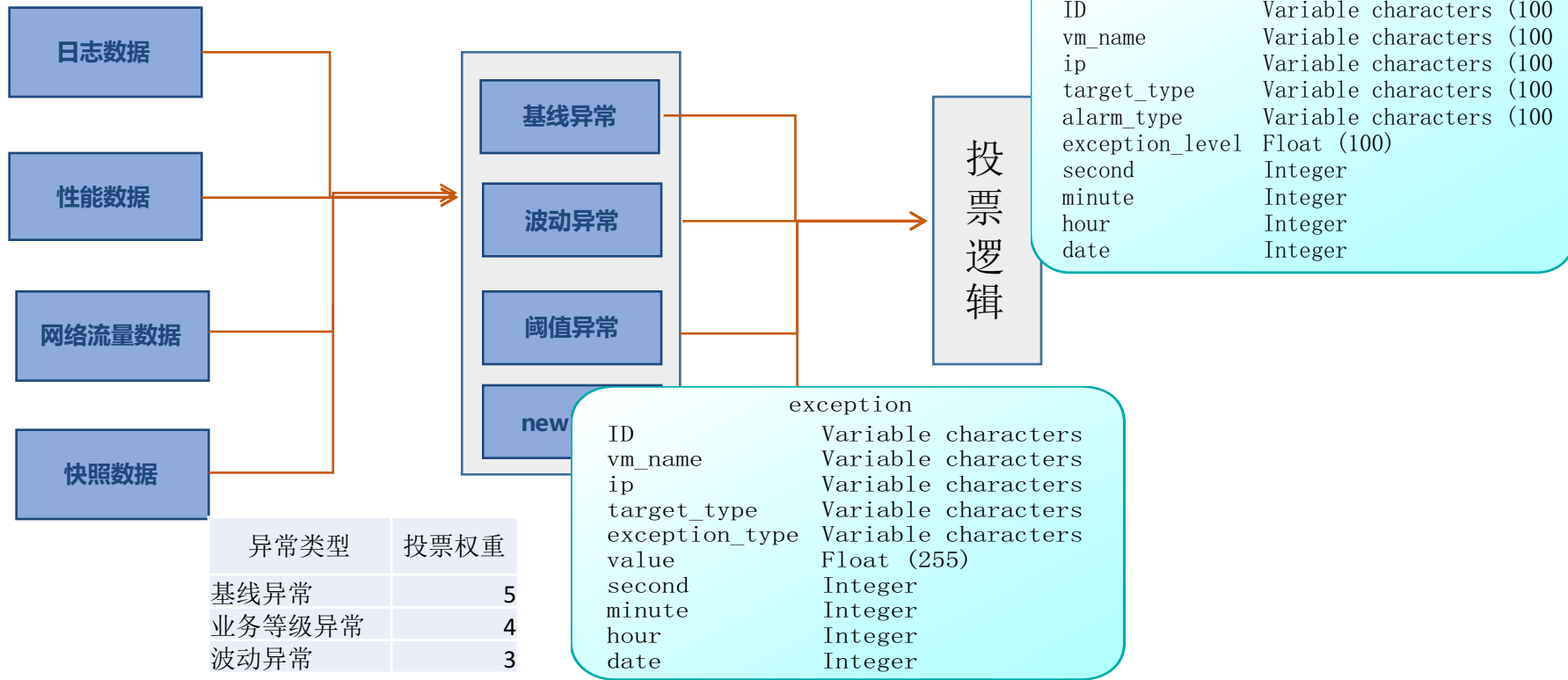
关键场景3：数据完整性验证逻辑



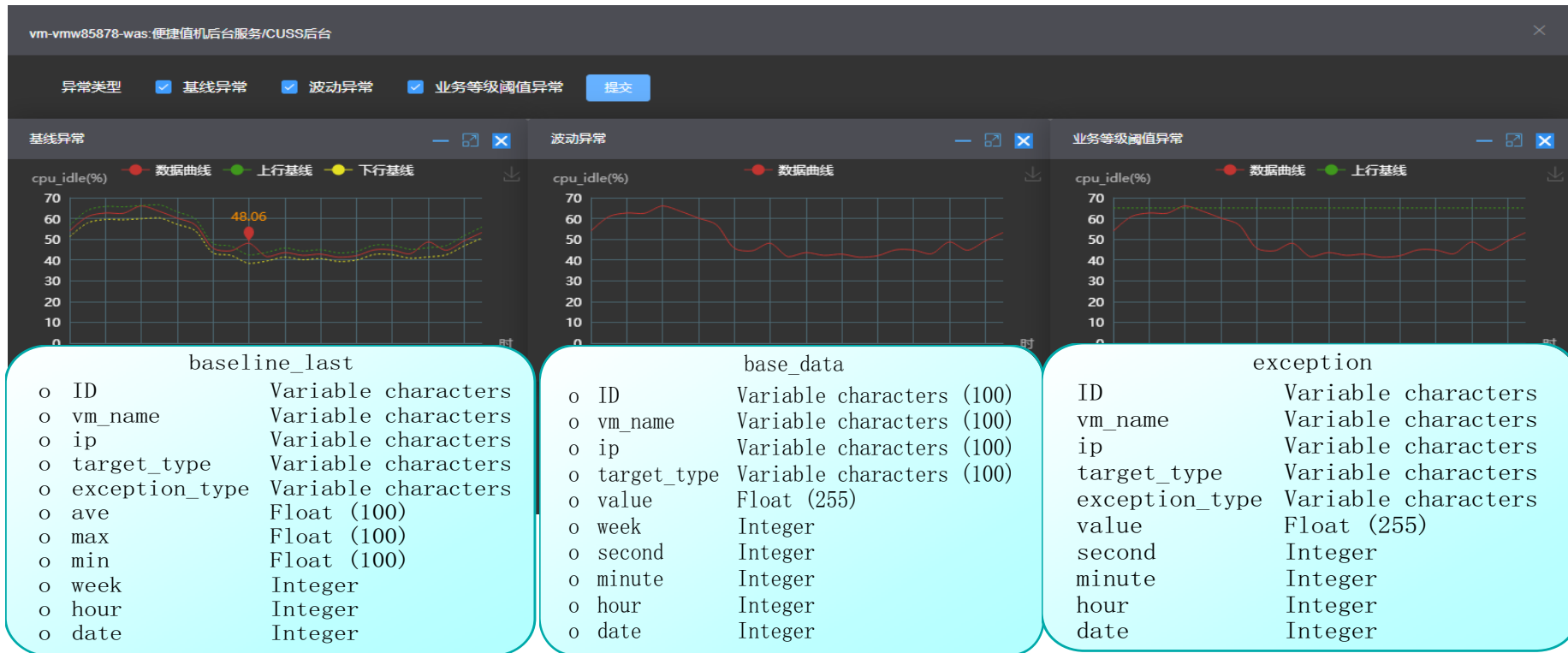
关键场景4：异常判定投票逻辑



关键场景5：可扩展的算法逻辑



关键场景6：可扩展的展示逻辑



目录

1 航信业务和运维情况概述

2 CloudInsight架构设计

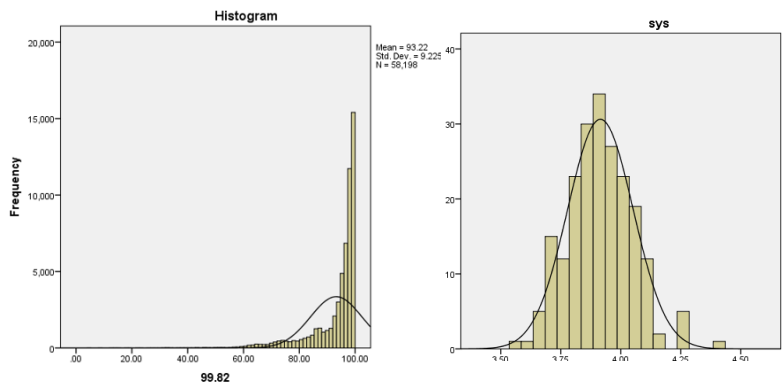
 **3** 实践展示

4 未来努力的方向

异常算法介绍-基线算法

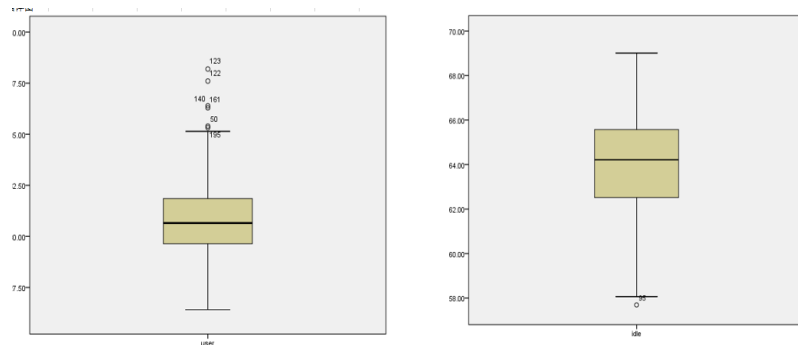
数据探索（各类数据分布确认）

Cpu idle/IO util /访问量在多周数据探测中某个固定小时数据分布并不完全符合正态分布



采用离群点的方式进行异常判定

以多天数据作为输入，根据week, hour进行基于箱体图上下限的时间序列的基线模型。



扩展异常算法-LSTM算法

基于LSTM的异常检测流程主要分为两部分

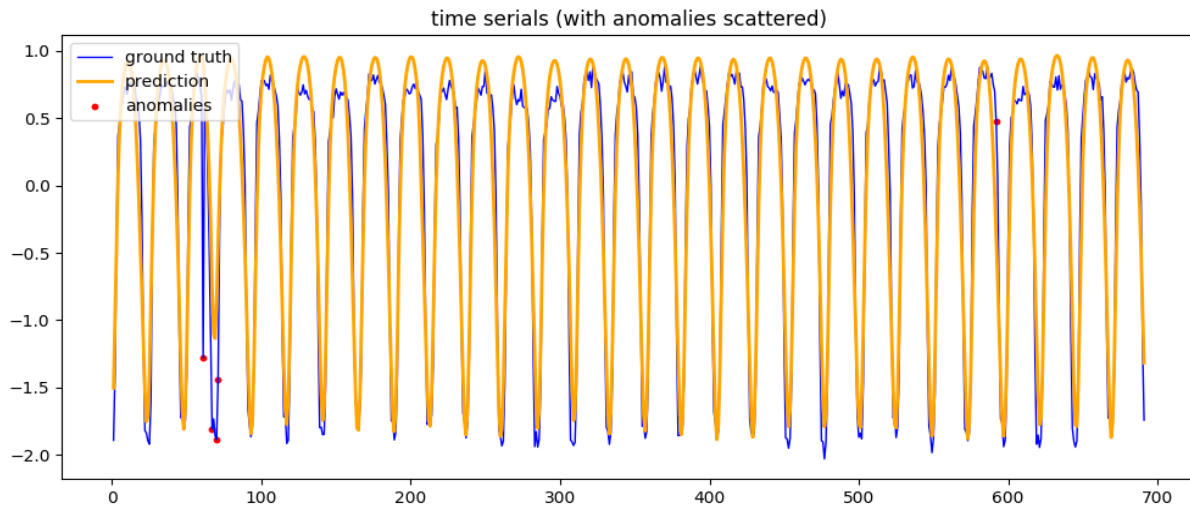
- 离线计算：使用了1个月的历史数据作为训练数据，并将训练模型导入HDFS中存储以供使用，学习出基于时序的趋势线；
- 实时计算：使用spark将实时数据导入模型，进行实时数据与趋势线的比对进行异常的判断

baseline_last

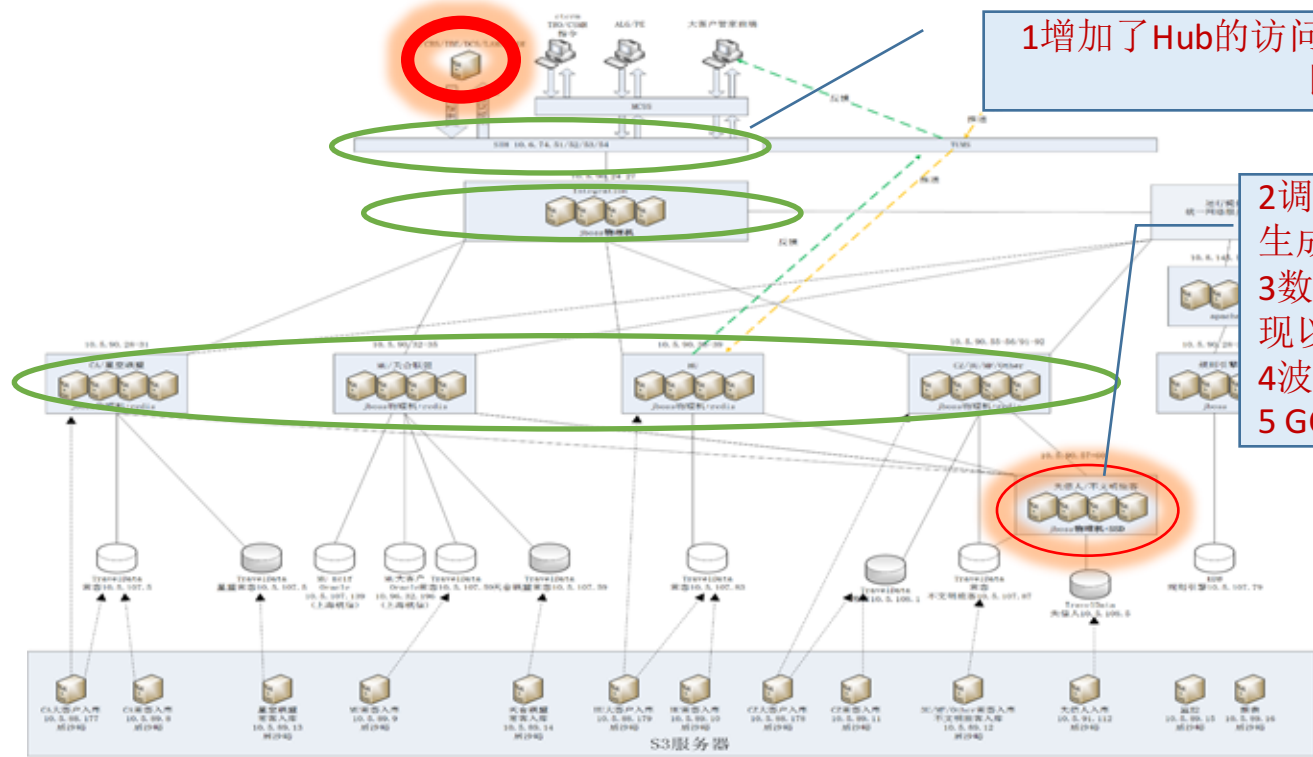
o ID	Variable characters
o vm_name	Variable characters
o ip	Variable characters
o target_type	Variable characters
o exception_type	Variable characters
o ave	Float (100)
o max	Float (100)
o min	Float (100)
o week	Integer
o hour	Integer
o date	Integer

exception

ID	Variable characters
vm_name	Variable characters
ip	Variable characters
target_type	Variable characters
exception_type	Variable characters
value	Float (255)
second	Integer
minute	Integer
hour	Integer
date	Integer



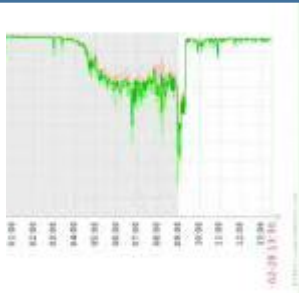
复盘曾经发生的故障（开放某服务系统故障） -



旅客管理搬迁后

1增加了Hub的访问链路调用tps/ART基线
比对

- 2调整异常策略，在基线/趋势线未生成前不参与投票
- 3数据完整性检查策略可保证快速发现以及部署新服务器
- 4波动异常可以发现异常
- 5 GC日志的频繁程度可以发现异常



目录

1 航信业务和运维情况概述

2 CloudInsight架构设计

3 实际数据验证

 **4** 未来努力的方向

未来努力的方向-1

1. 加入业务追踪、服务器追踪的能力

- 根据业务的调用关系以及服务器的调用关系，将异常影响进行深入的推演，去找到更广泛更准确的影响范围

2. 添加知识库的处理逻辑

- 将日常的处理文档和异常处理的工作经验与异常进行整合，真正做到可以在进行故障处理时快速获取帮助的能力；

未来努力的方向-2

1. 整合数据采集agent

- 性能数据/日志数据/快照数据加入统一的agent处理框架，进行统一管理和维护；

2. 实现从“算”到“想”

- 逐步实现从穷举的算推进到可以启发式的想，找到更多适用的算法加入进来获得更准确的异常分析！



生存：就是找到适合你的“缝隙”

AI：就是要找到适合的“架构”



Thanks

高效运维社区
开放运维联盟

荣誉出品

想第一时间看到高效运维社区
的新动态吗？

