



Flashcat 让监控分析变简单

Flashcat 产品技术交流

北京快猫星云科技有限公司




公司简介：快猫星云是一家云原生智能运维科技公司



由知名开源项目“夜莺”的核心开发团队组成：

夜莺是一款开源云原生监控工具，是中国计算机学会接受捐赠并托管的第一个开源项目，在GitHub上有超过8500颗星，上百位社区贡献者，上万家企业用户，是国内领先的开源可观测性解决方案。



创始团队均来自国内一线互联网公司：创始团队在  阿里巴巴  百度  DiDi 等互联网公司，长期担任基础设施、云计算、稳定性保障等方向的负责人。



由国内顶级投资机构连续投资。

快猫星云是国内开源监控领域最具专业性的团队之一

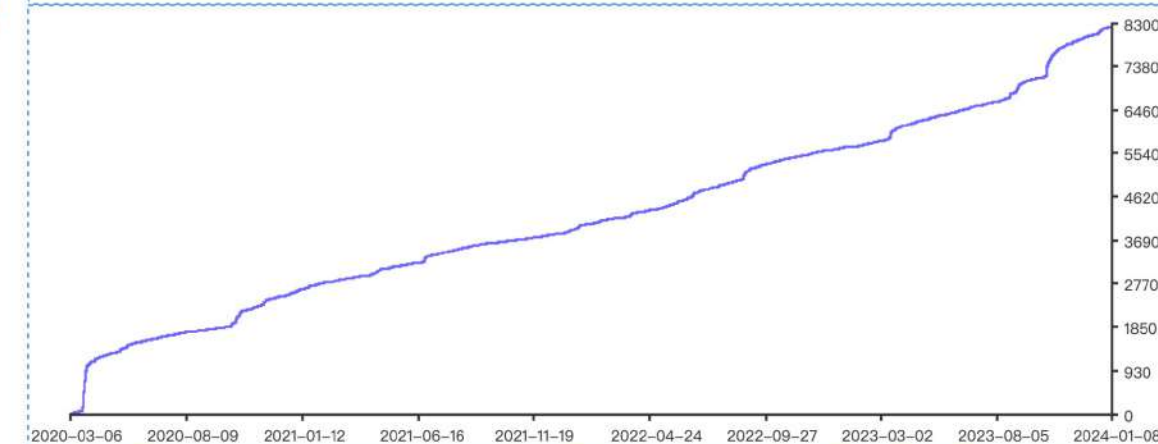


夜莺Nightingale

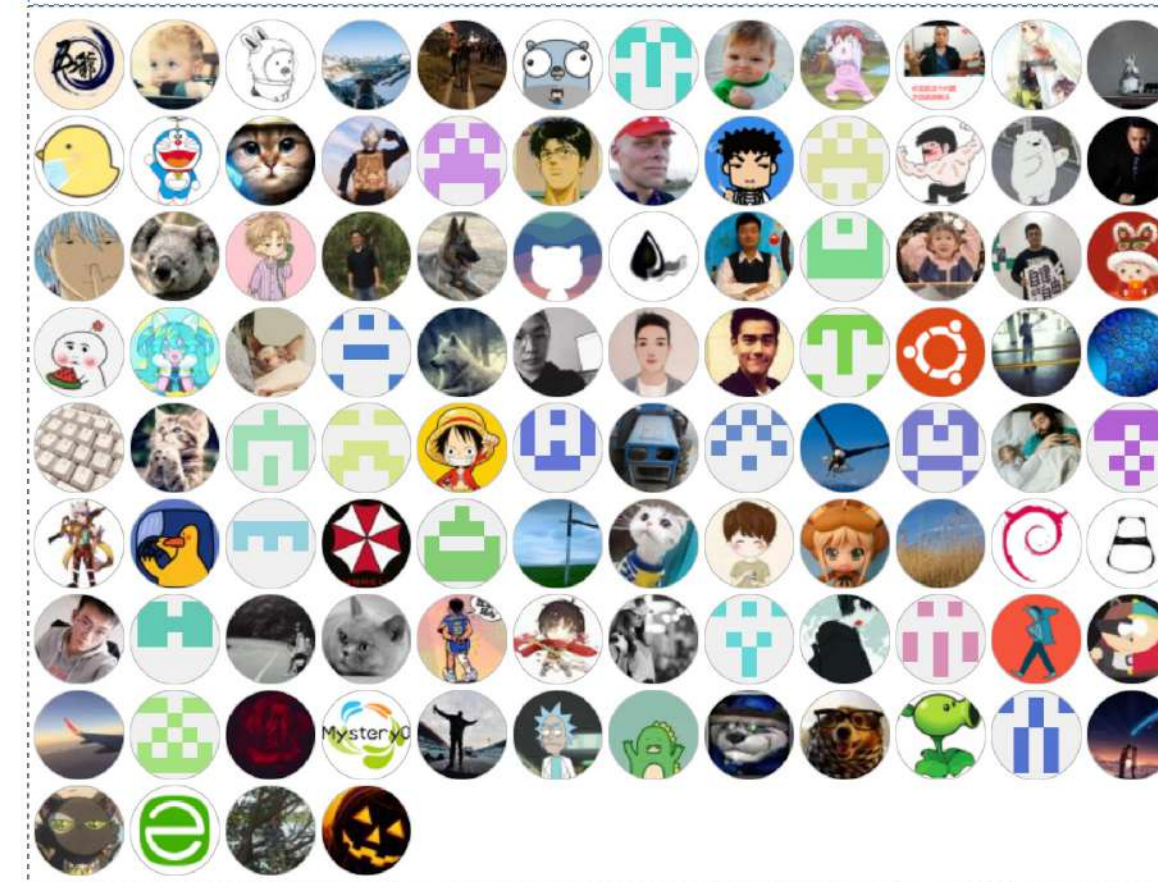
是中国计算机学会接受捐赠并托管的第一个开源项目
是国内使用最广泛的开源监控工具之一



8500+ Stars



100+ Contributors



137 次 Releases!

- Apache-2.0 license
- Activity
- Custom properties
- 8.5k stars
- 154 watching
- 1.3k forks

Report repository

Releases 137

v7.0.0-beta.1 Latest
last month

+ 136 releases



超过 35 万次下载，上万家企业信赖使用

Catgraf + Nightingale

快猫星云技术团队是夜莺Nightingale 和 Catgraf 的主要贡献者，是夜莺项目管理委员会的核心成员

Flashcat 是什么



81%的企业采用2个或多个公有云

Gartner survey 2019

61%的企业实践可观测性方法

study from ClearPath Technologies 2021

绝大多数企业使用了超过6个以上的监控工具



构建统一的
可观测体系



解决监控系统分散
维护成本高
数据无法串联打通的问题



- 物理机、微服务、云原生架构
- 公有云、多云、混合云
- Metrics、Logging、Tracing、Events
- RegionA、RegionB



构建智能的故障
发现定位体系

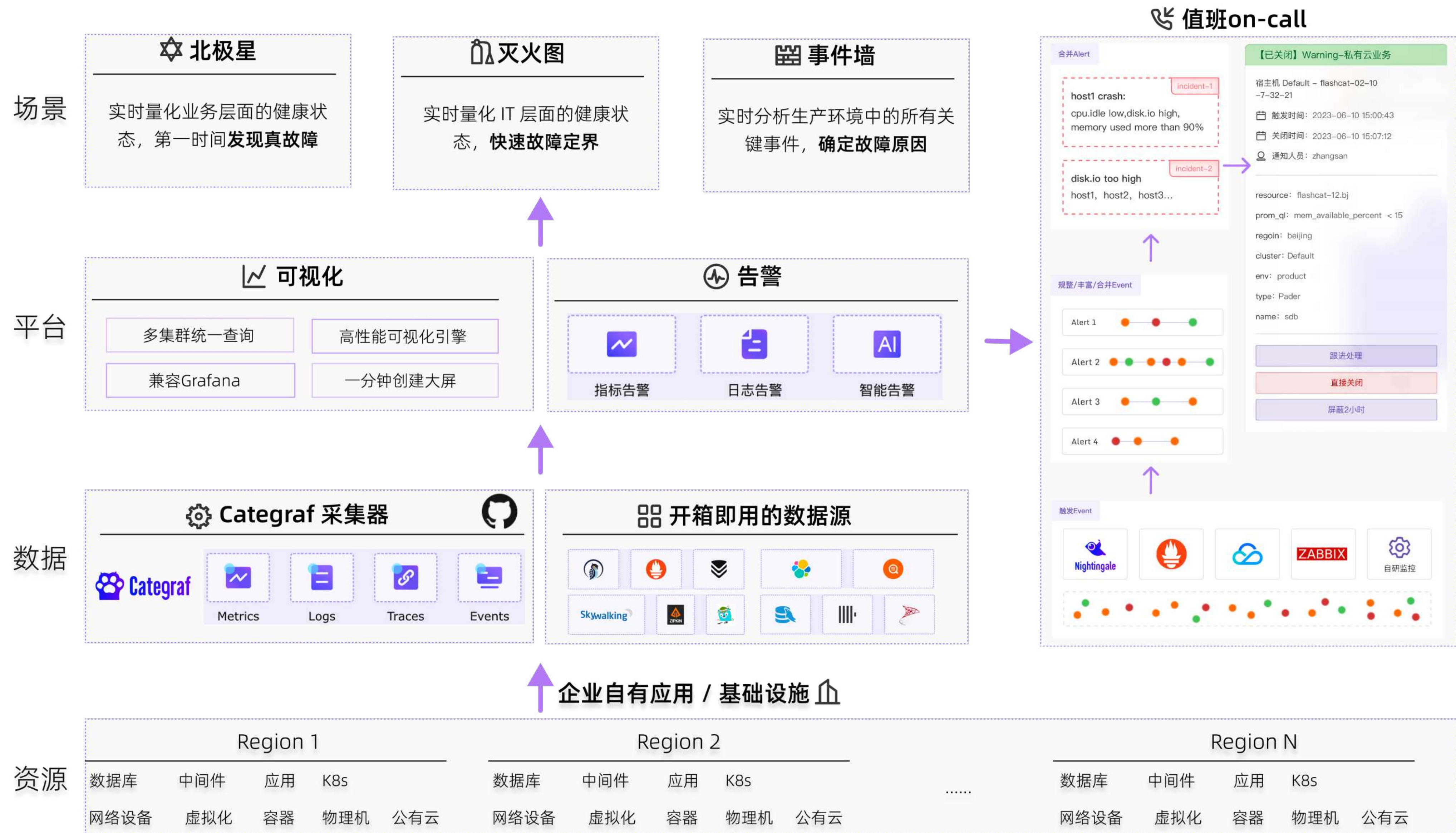


预置最佳实践
及时从业务侧发现异常
加速故障处理过程



- 总是后于用户反馈发现故障
- 难以快速确认故障影响面
- 难以快速找到故障的直接原因
- 故障处理进度不透明
- 理解数据的成本太高

Flashcat 构建了一个数据、平台、场景打通的一体化方案



Flashcat 的特点



统一采集

采用插件化思路，内置集成上百种采集插件，服务器、网络设备、中间件、数据库、应用、业务，云上云下，均可监控，开箱即用。



统一告警

支持指标告警、日志告警、智能告警，支持几十种数据源对接，收集各类监控系统的告警事件，进行统一的告警收敛、降噪、排班、认领、升级、协同，大幅提升告警处理效率。



统一观测

将 Metrics、Logs、Traces、Events、Profiling 等多种可观测性数据融会贯通，并预置行业最佳实践，既提供全局业务视角、技术视角的驾驶舱，也提供层层下钻的故障定位能力，有效缩短故障发现和定位时间。



统一采集

All-in-One 的数据采集器



release v0.3.57 docker pulls 313k Stars 654 Forks 210
contributors 61 license MIT Powered By Flashcat

All-in-One

一个agent同时采集
指标、日志、链路追踪数据

vs Telegraf

“Telegraf 数据格式
与Prometheus生态不兼容”

开源、开放

Categraf 源代码开放
遵循OpenTelemetry标准
社区驱动，近百项插件

vs Exporters

“需要安装维护的Exporter数量庞大，
Exporter质量良莠不齐”



Categraf

- 是一款 All-in-One 的开源的 telemetry 数据采集器，支持指标、日志采集；
- 支持 Tracing 数据的收集；
- 支持物理机、虚拟机、交换机、容器、K8s、多种中间件/数据库的数据采集，云上云下，均可监控；
- 汇聚领域最佳实践，开箱即用；



内置仪表盘模板和告警模板

仪表盘模板 开箱即用

常用告警策略 一键添加

内置仪表盘

EN 超管

分类

Q

AliYun

Kubernetes

Linux

Canal

Ceph

CloudWatch

ElasticSearch

Gitlab

HAProxy

HTTP HTTP_Response

IPMI

JMX

Kafka

Logstash

MinIO

MongoDB

MySQL

N9E

Net_Response

仪表盘列表

采集说明

Q

批量克隆

批量导出

<input type="checkbox"/> 仪表盘名称	标签	操作
<input type="checkbox"/> ARMS-API	ARMS JVM	查看 克隆 导出
<input type="checkbox"/> ARMS-Application	JVM ARMS	查看 克隆 导出
<input type="checkbox"/> ARMS-DB	ARMS	查看 克隆 导出
<input type="checkbox"/> ARMS-Machine	ARMS	查看 克隆 导出
<input type="checkbox"/> 阿里云 ARMS-JVM	JVM ARMS	查看 克隆 导出
<input type="checkbox"/> 阿里云CDN	CDN	查看 克隆 导出
<input type="checkbox"/> 阿里云ECS		查看 克隆 导出
<input type="checkbox"/> 阿里云MongoDB		查看 克隆 导出
<input type="checkbox"/> MSE监控大盘		查看 克隆 导出
<input type="checkbox"/> 阿里云-MSE Ingress监控中心		查看 克隆 导出
<input type="checkbox"/> 阿里云NAT	NAT	查看 克隆 导出
<input type="checkbox"/> 阿里云OSS		查看 克隆 导出
<input type="checkbox"/> 阿里云POLARDB-MySQL	polardb 阿里云	查看 克隆 导出
<input type="checkbox"/> 阿里云RDS		查看 克隆 导出
<input type="checkbox"/> 阿里云RDS_N		查看 克隆 导出
<input type="checkbox"/> 阿里云REDIS		查看 克隆 导出
<input type="checkbox"/> 阿里云REDIS 集群版		查看 克隆 导出
<input type="checkbox"/> 阿里云REDIS_N		查看 克隆 导出
<input type="checkbox"/> 阿里云REDIS 标准版		查看 克隆 导出

内置告警规则

EN 超管

分类

Q 搜索

Kubernetes

Linux

Ceph

ElasticSearch

Gitlab

HTTP HTTP_Response

IPMI

Kafka

MinIO

MongoDB

MySQL

Net_Response

Network

Ping

PostgreSQL

Process

Processes

Procstat

RabbitMQ

Radic

规则列表

采集说明

apiserver

Q 搜索

批量克隆

批量导出

<input type="checkbox"/> 规则名称	附加标签	操作
<input type="checkbox"/> KubeClientCertificateExpiration-S2		查看 克隆 导出
<input type="checkbox"/> KubeClientCertificateExpiration-S1		查看 克隆 导出
<input type="checkbox"/> AggregatedAPIErrors		查看 克隆 导出
<input type="checkbox"/> AggregatedAPIDown		查看 克隆 导出
<input type="checkbox"/> KubeAPIDown		查看 克隆 导出
<input type="checkbox"/> KubeAPIErrorBudgetBurn-S1-120秒	long=1h short=5m	查看 克隆 导出
<input type="checkbox"/> KubeAPIErrorBudgetBurn-S1-900秒	long=6h short=30m	查看 克隆 导出
<input type="checkbox"/> KubeAPIErrorBudgetBurn-S2-3600秒	long=1d short=2h	查看 克隆 导出
<input type="checkbox"/> KubeAPIErrorBudgetBurn-S2-10800秒	long=3d short=6h	查看 克隆 导出

共 9 条 < 1 > 10 条/页

Categraf 增强功能

采集配置集中下发

筛选条件  

全部机器 



机器预览

采集配置 使用须知 

* 插件类型

elasticsearch

* 插件配置 模板选择

```
1  ## collect interval
2  # interval = 15
3
4  #####
5  # !!! uncomment [[instances]] to enable this plugin
6  [[instances]]
7  # # interval = global.interval * interval_times
8  # interval_times = 1
9
10 # append some labels to metrics
11 # labels = { cluster="cloud-n9e-es" }
12
13 ## specify a list of one or more Elasticsearch servers
14 # servers = ["http://localhost:9200"]
15 servers = []
16
17 ## Timeout for HTTP requests to the elastic search server(s)
18 http_timeout = "10s"
19
20 # either /_nodes/stats or /_nodes/_local/stats depending on this setting
21 local = false
22
23 ## Set cluster_health to true when you want to obtain cluster health stats
24 cluster_health = true
```

网络设备SNMP采集模板

网络设备采集模板

* 模板名称

switch-MS210-24P

```
1  [[instances]]
2  ## Agent addresses to retrieve values from.
3  ##   format:  agents = ["<scheme://><hostname>:<port>"]
4  ##   scheme:  optional, either udp, udp4, udp6, tcp, tcp4, tcp6.
5  ##             default is udp
6  ##   port:    optional
7  ##   example: agents = ["udp://127.0.0.1:161"]
8  ##             agents = ["tcp://127.0.0.1:161"]
9  ##             agents = ["udp4://v4only-snmp-agent"]
10 #agents = ["udp://127.0.0.1:161"]
11 agents = [{"Schema"}://[{.IP}]:[{.Port}]}"]
12
13 ## Timeout for each request.
14 # timeout = "{.Timeout}}s"
15
16 ## SNMP version; can be 1, 2, or 3.
17 # version = {.Version}}
18
19 ## Unconnected UDP socket
20 ## When true, SNMP reponses are accepted from any address not just
21 ## the requested address. This can be useful when gathering from
22 ## redundant/failover systems.
23 # unconnected_udp_socket = false
24
```

保存

取消

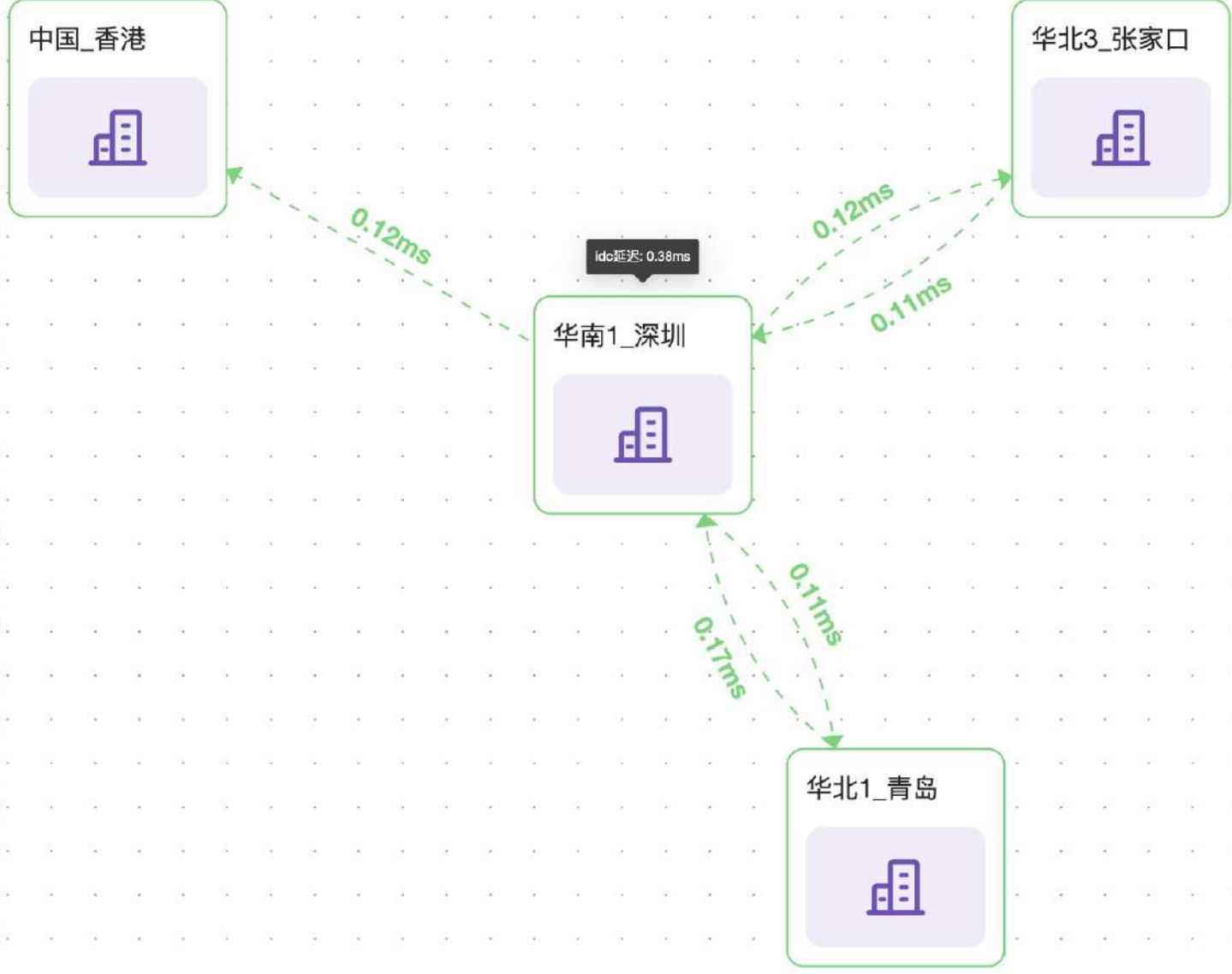
Pingmesh


中国_香港




华南1_深圳

华北3_张家口


华北1_青岛






ICMP  延迟 丢包

2024-03-26 13:22:09   off 

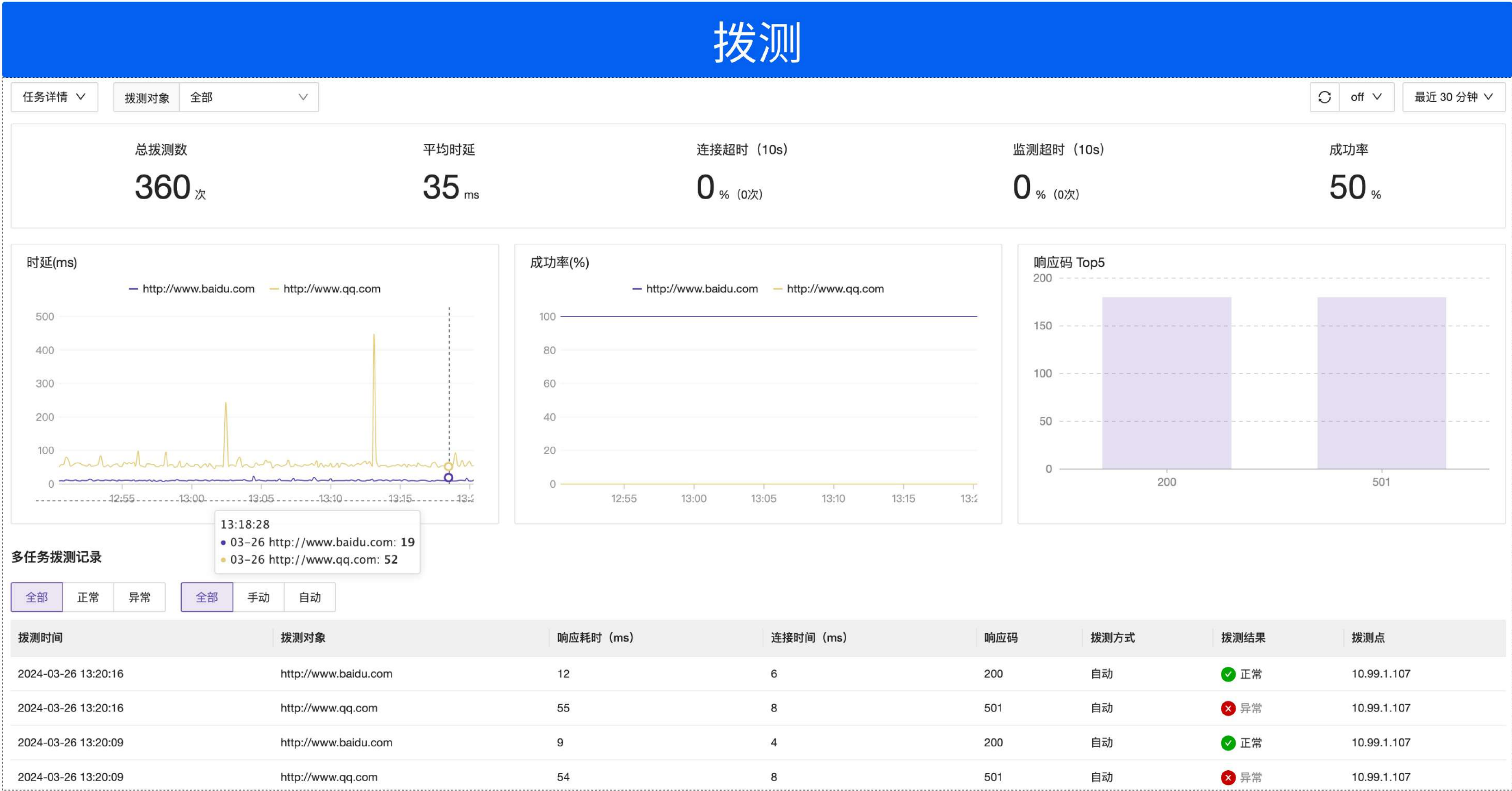
	目的	172.22.1.2/32	172.22.1.3/32	172.22.1.4/32
源				
172.22.1.2/32		0.2ms	0.13ms	0.11ms
172.22.1.3/32		0.08ms	0.07ms	0.24ms
172.22.1.4/32		0.13ms	0.13ms	0.21ms

ICMP  延迟 丢包

2024-03-26 13:29:57   off 

	目的	172.22.1.2/32	172.22.1.3/32	172.22.1.4/32
源				
172.22.1.2/32		0%	0%	0%
172.22.1.3/32		0%	0%	0%
172.22.1.4/32		0%	0%	0%

Categraf 增强功能



多协议：
HTTP、TCP、UDP、
ICMP、WSDL

多拨测点：
可选择安装了Categraf
的一个或多个设备

多数据源集成

时序数据源

时序数据源

事件源

日志源

Tracing源

可用于 北极星、灭火图、仪表盘、告警管理



InfluxDB

添加



Oracle

添加




Prometheus Like

添加



MySQL

添加




Zabbix

添加



PostgreSQL

添加




ClickHouse

添加



SQLServer

添加



JSON API

添加

事件数据源

时序数据源

事件源

日志源

Tracing源

可用于 事件墙

变更 ^



自定义事件

添加



Jira

添加



Kubernetes

添加

告警 ^



自定义事件

添加



Prometheus

添加



Zabbix

添加



Nightingale

添加



Open-Falcon

添加



腾讯云监控 CM

添加

Tracing 数据源


时序数据源

事件源

日志源


Tracing源

可用于 链路分析




Zipkin

添加




Jaeger

添加




Skywalking

添加




自定义跳转

添加




Elastic APM

添加




SLS Trace

添加




阿里云 OpenTelemetry

添加




腾讯云 APM

添加




Arms Trace

添加




Tempo

添加



OpenTelemetry

添加



PINPOINT

添加

日志数据源


时序数据源

事件源

日志源


Tracing源

可用于 日志分析、告警管理、仪表盘




kafka

添加




Elasticsearch

添加




阿里云SLS

添加




ClickHouse

添加




腾讯云CLS

添加




OpenSearch

添加



Loki

添加



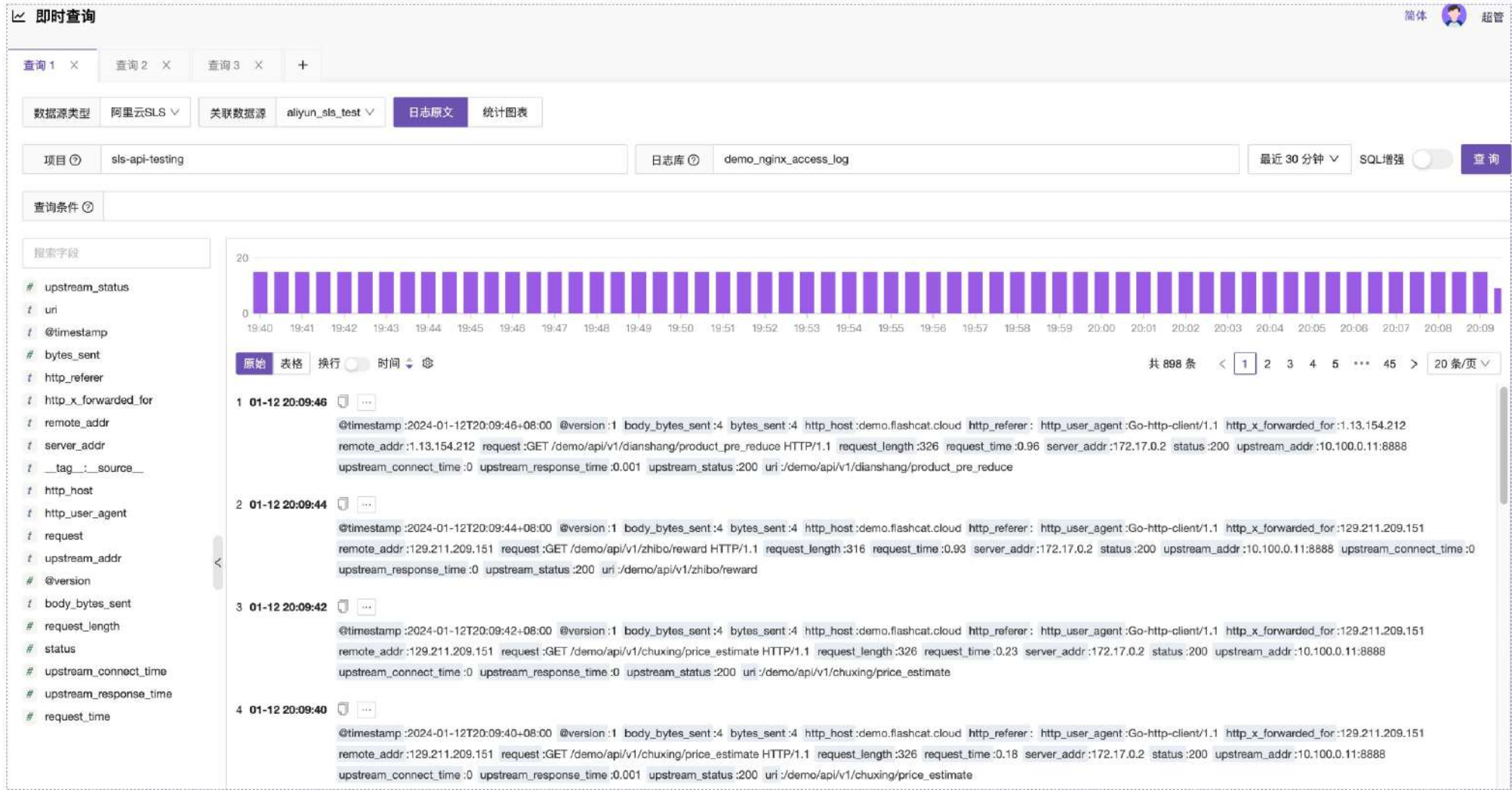
DORIS

添加

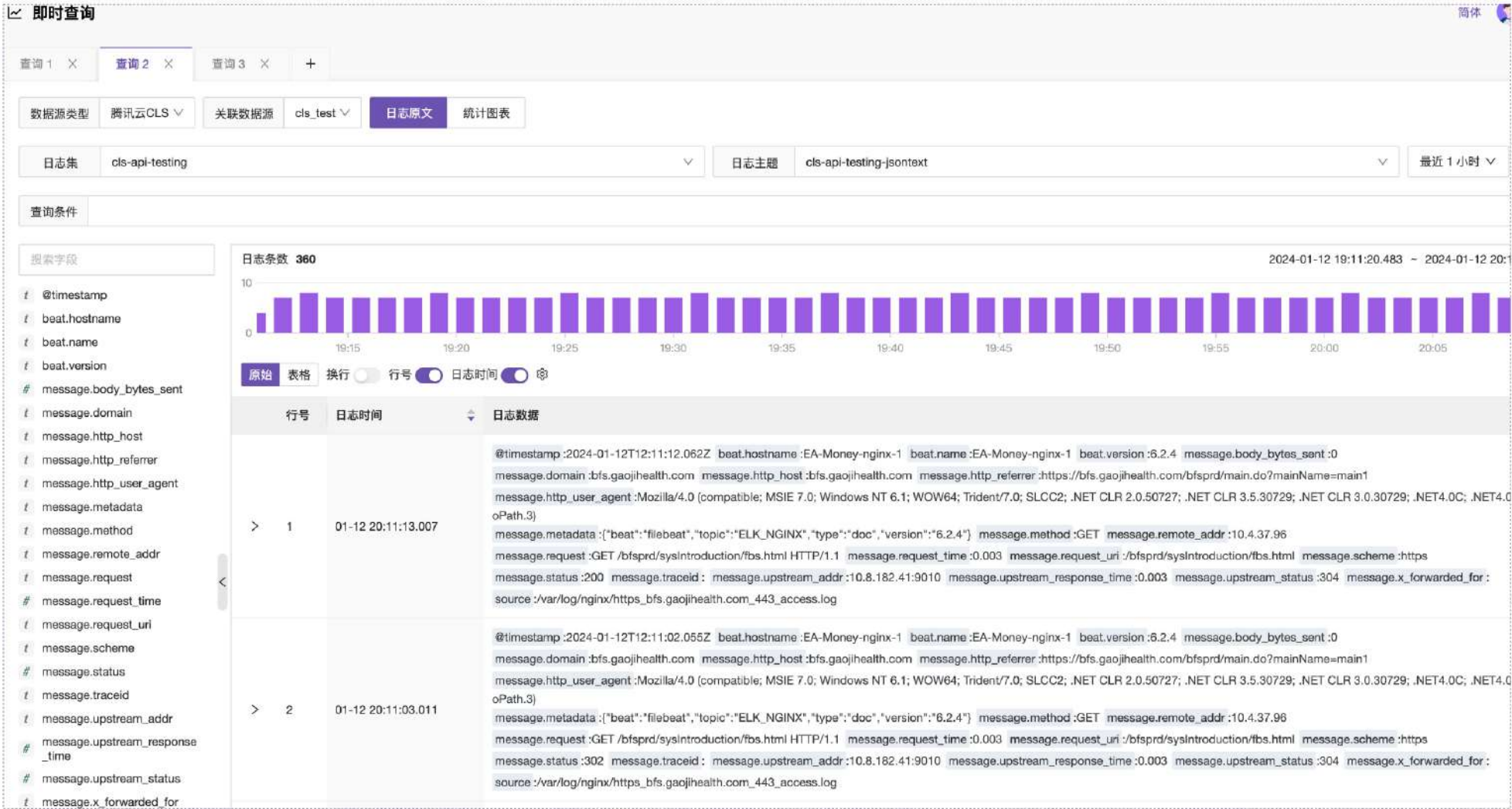


日志统一可视化

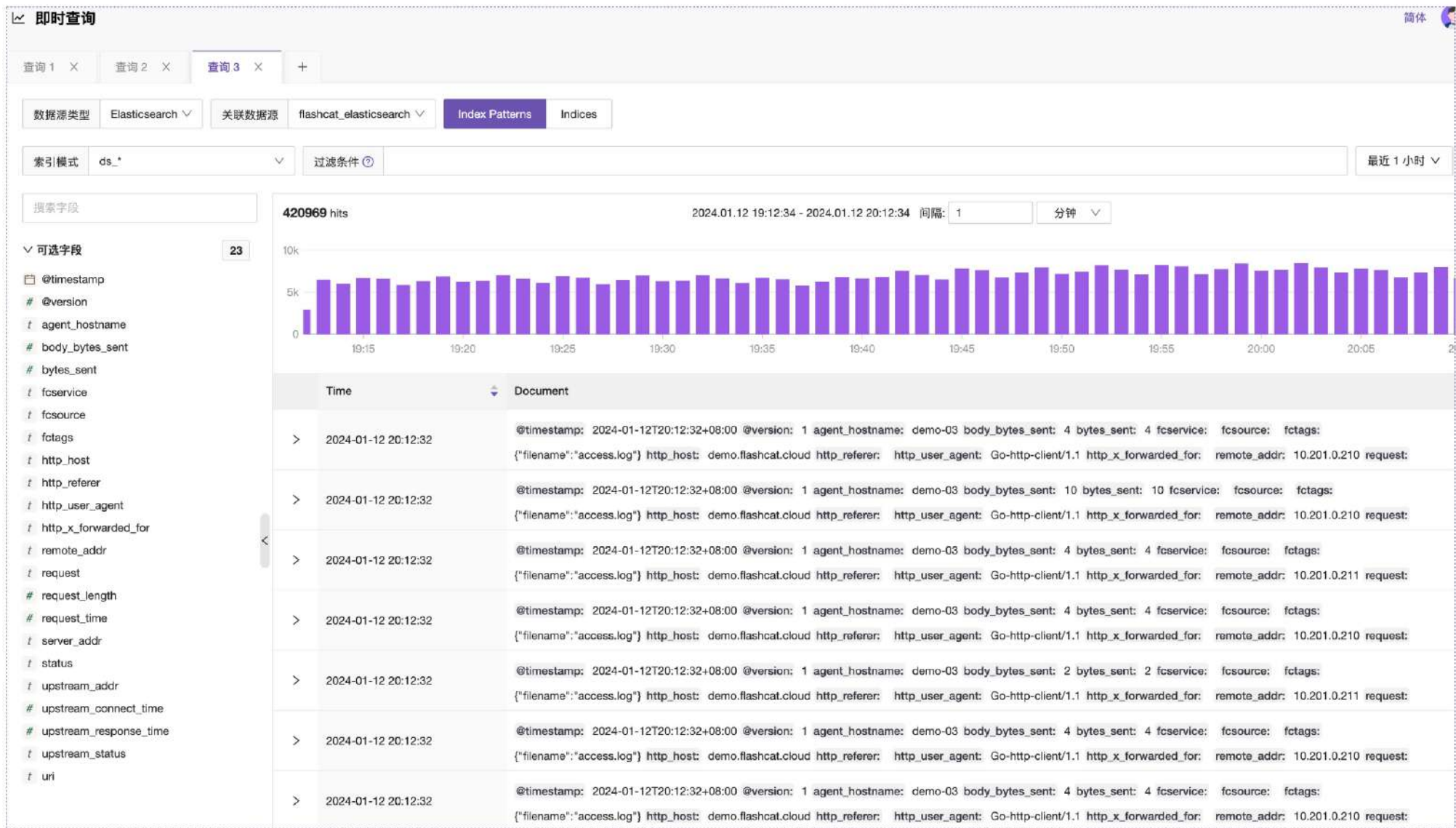
阿里云 SLS 日志可视化



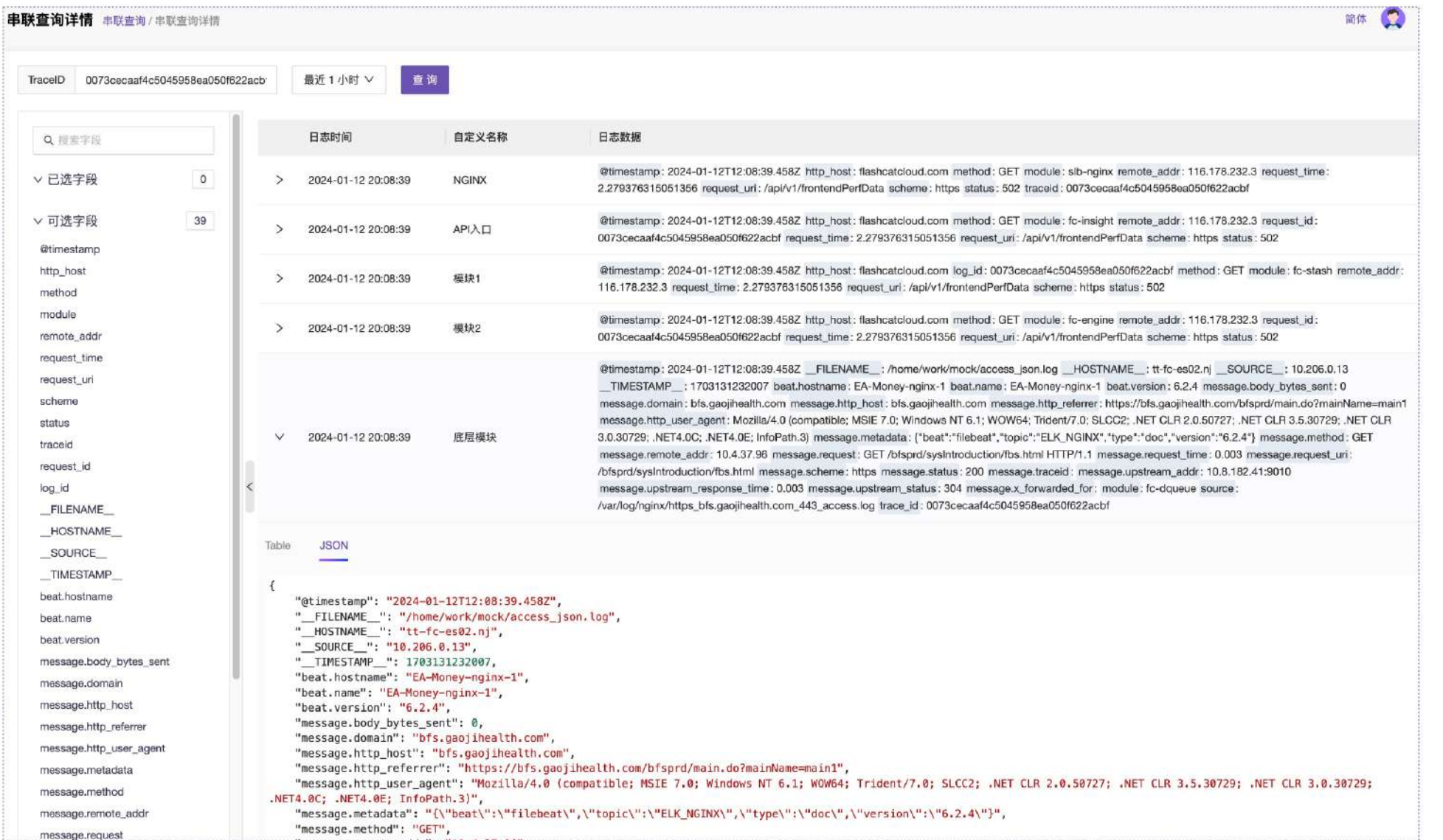
腾讯云 CLS 日志可视化



ElasticSearch 可视化



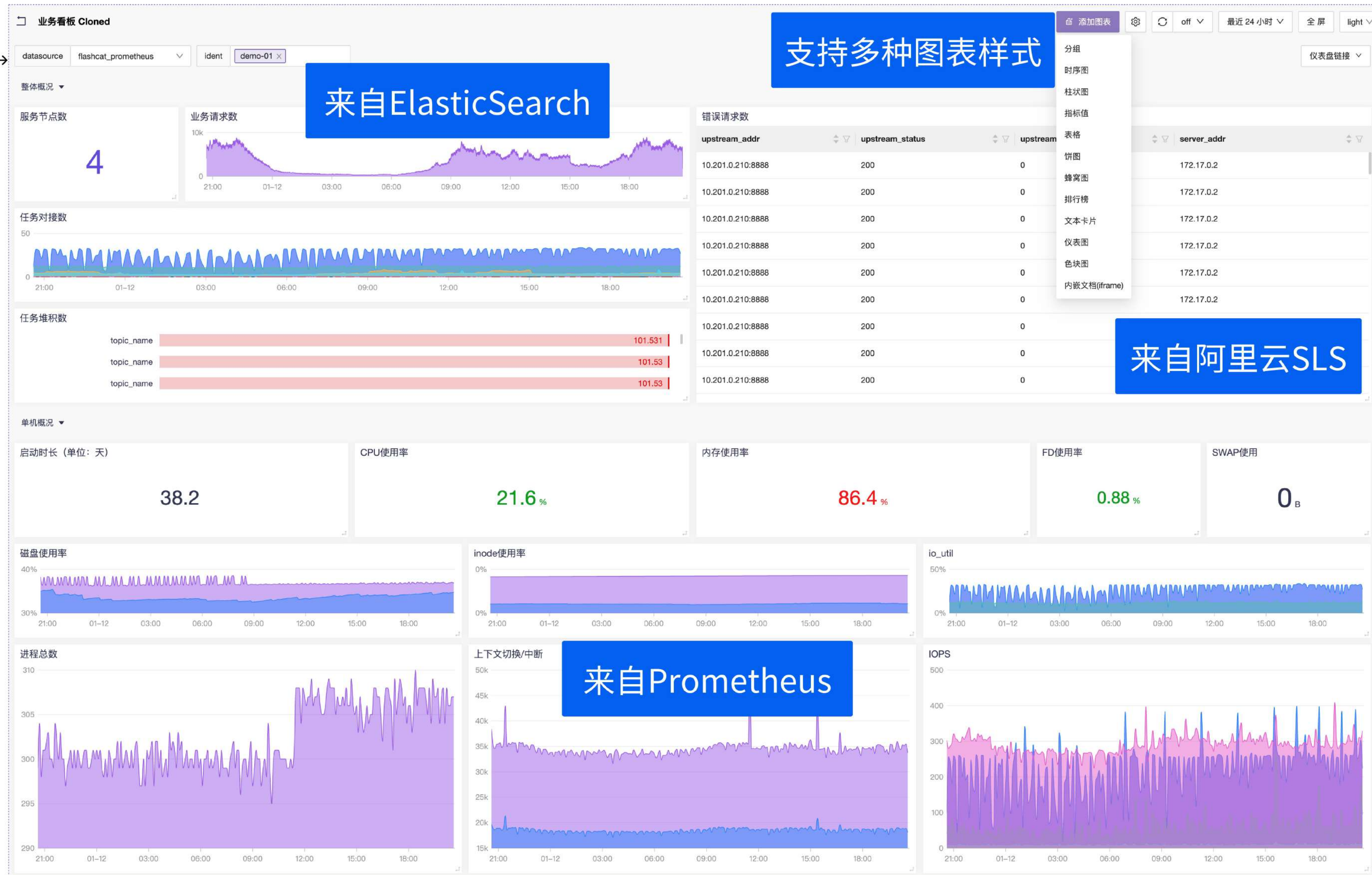
跨数据源联动查询



统一的仪表盘

在一个仪表盘面板中
展示不同来源的数据

支持导入Grafana仪表盘





统一告警



统一告警

阿里云SLS告警

MetricHostLogAnomalyPro

数据源类型* 关联数据源

阿里云SLSaliyun_sls_test

查询统计

A项目sls-api-testing日志库demonginx_access_lo

查询区间最近1小时SQL增强

查询条件*|SELECT count(1) as total_pv

辅助配置数据预览

告警条件

简单模式表达式模式

A>1000

触发告警：一级报警二级报警三级报警

执行频率(s)15持续时长(s)0

高级设置

生效配置

立即启用

生效时间开始时间结束时间

周一周二周三周四周五周六周日00:0023:59

ElasticSearch日志告警

MetricHostLogAnomalyPro

数据源类型* 关联数据源

Elasticsearchflashcat_elasticsearch

查询统计

A索引ds_*

过滤条件

日期字段@times...

数值提取count

Group By

数据预览

告警条件

简单模式表达式模式

A>1000

触发告警：一级报警二级报警三级报警

执行频率(s)15持续时长(s)0

高级设置

生效配置

立即启用

生效时间开始时间结束时间

周一周二周三周四周五周六周日00:0023:59

多集群Prometheus告警

MetricHostLogAnomalyPro

数据源类型* 关联数据源

Prometheusflashcat_prometheus北京机房(阿里云)

告警条件级别抑制

PromQLelasticsearch_cluster_health_status{color="yellow"} == 1

触发告警：一级报警二级报警三级报警

数据预览

PromQLrate(cpu_usage_idle{ident="demo-01"}[2m]) < 0.01

触发告警：一级报警二级报警三级报警

数据预览

执行频率(s)15持续时长(s)0

生效配置

立即启用

生效时间开始时间结束时间

周一周二周三周四周五周六周日00:0023:59

服务日历第一个节假日日历

异常检测

说明文档



off ▾

最近 12 小时 ▾

智能检测 ✓

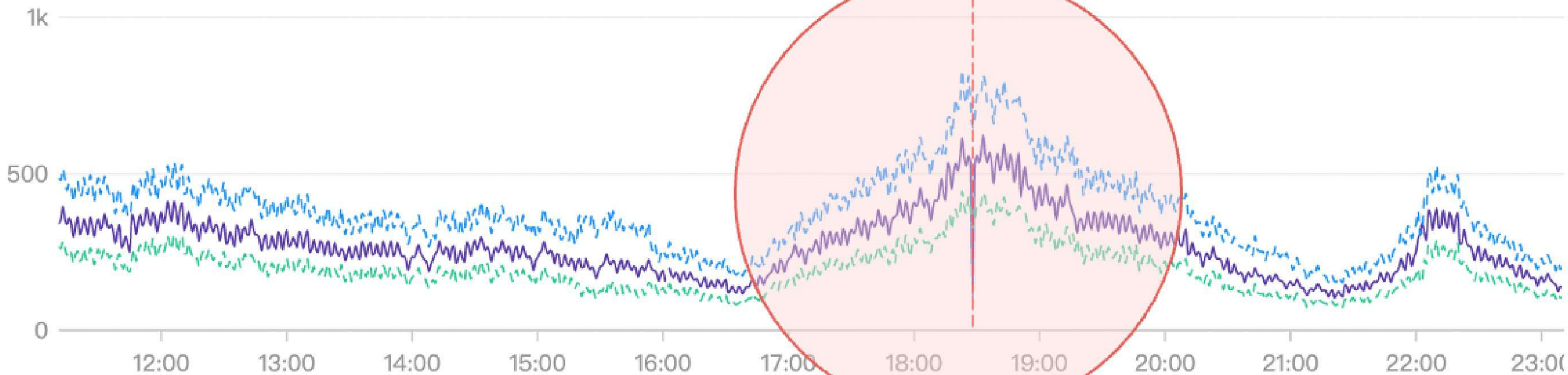
同环比检测 ✓

数据中断检测 ✓

阈值条件

区间预测曲线

☒ 显示越界异常点



该指标智能报警开启推荐:

推荐开启



告警太多，怎么办？

告警数量多

技术团队每天接收大量告警

告警响应慢

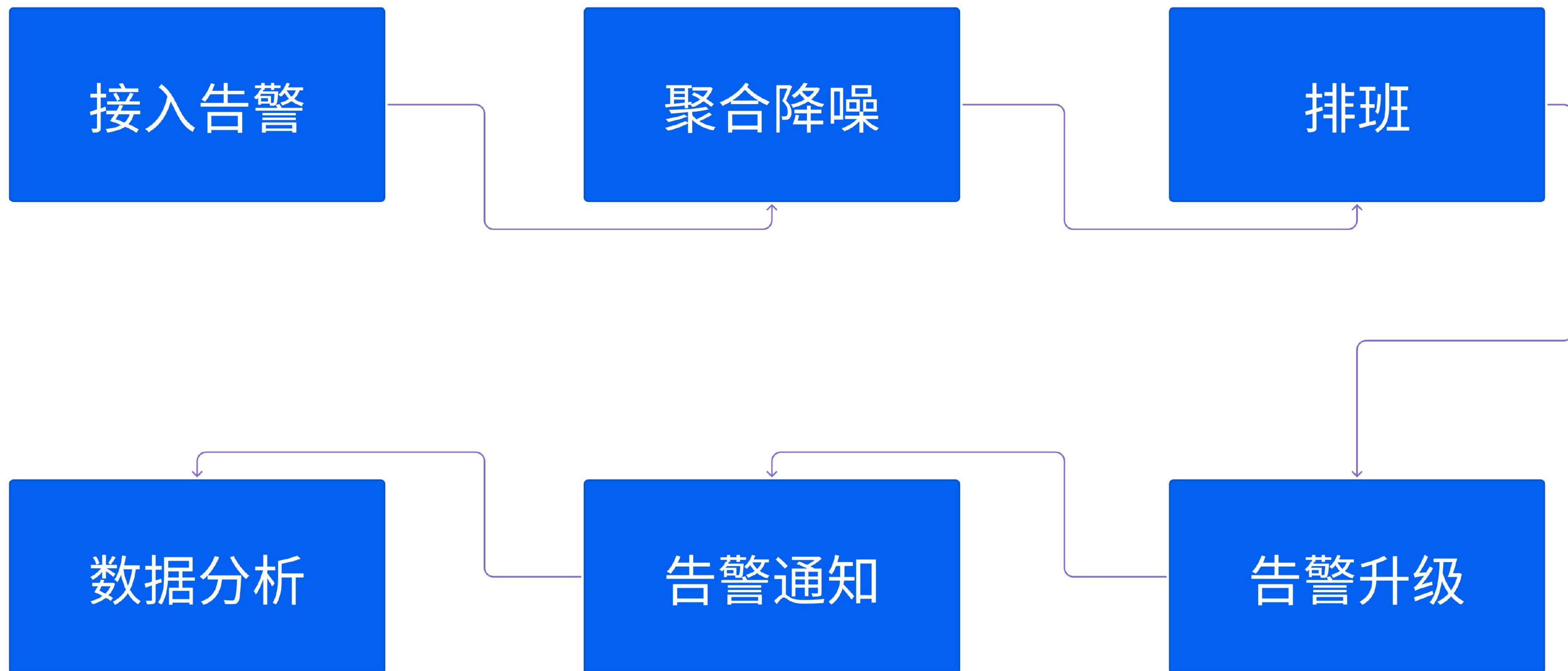
很多告警长时间无人响应
长期无人问津

处理协同差

告警处理缺乏协同，处理过程不透明
信息难以贡献，知识不易沉淀

IT 满意度低

客户往往先于技术团队发现故障
对 IT 满意度持续走低



快速接入各种告警事件

选择数据源

获取告警推送地址

修改 AlertManager Webhook 地址

集中查看和处理告警


告警事件

变更事件


即时消息

Webhook


搜索集成类型




自定义事件




Prometheus




Zabbix



Nightingale



Open-Falcon



腾讯云监控 CM



阿里云监控 CM



腾讯云 EventBridge



Grafana



PagerDuty



Influxdata



AWS CloudWatch



百度云监控 BCM



华为云



华为云监控 CES



阿里云 SLS



蓝鲸智云



阿里云 ARMS



Uptime Kuma

prom-demo 启用中

推送地址 未收到告警事件

https://api.flashcat.cloud/event/push/alert/prometheus?integration_key=de...c647f167973

编辑路由规则

如果 severity == "Critical"

将告警投递到协作空间 laiwei01，然后停止匹配下一路由

默认路由 如果未匹配到任何路由

将告警投递到协作空间 laiwei01 测试环境

编辑

停用

步骤 1: 配置 Alertmanager

1. 登录您的 Alertmanager 实例
2. 找到并打开配置文件，一般为 Alertmanager 部署根目录下的 alertmanager.yml
3. 在 receivers 配置部分，增加一个快猫星云 webhook 类型的 receiver，如下

```
receivers:
- name: 'flashcat'
  webhook_configs:
  - url: '<替换为prometheus集成推送地址>'
    send_resolved: true
    http_config:
      proxy_url: 'http://proxyserver:port'
```

您需要替换 url 对应的参数值为集成的推送地址，注意 query string 参数部分需要携带 integration_key。

如果您需要通过代理请求快猫星云，可以额外设置 http_config 的 proxy_url 参数为代理地址。

4. 在 route 配置部分，更改默认 route 指定 receiver 为刚才配置的 webhook，如下：

```
route:
...
receiver: 'flashcat'
```

您也可以把 receiver 添加到非默认 route，但这样您只会同步对应 route 的告警事件到快猫星云，而非全部告警事件。

5. 保存配置文件
6. 通过重新加载配置文件（向进程发送 SIGHUP 信号，或 POST 请求 /-/reload api），使更改生效
7. 回到集成列表，如果展示了最新事件时间，说明配置成功且收到事件
8. 完成

步骤 2: 配置 Timestamp

默认情况下，系统使用当前时间作为事件触发时间。如果您希望自定义时间，您可以额外设定一个 timestamp 字段来标识每一次告警发生的准确时间。

1. 登录您的 Prometheus Server 实例
2. 打开告警规则相关配置文件
3. 对于每一条告警规则，更改 annotations 部分，添加 timestamp 字段，如下：

```
annotations:
  timestamp: '{{ with query "time()" }}{{ . | first | value }}{{ end }}'
...
```

协作空间 > 空间详情

订单系统

负责团队 研发团队

过去一周故障 MTTA ③

0秒 ↓ 0%

故障列表

集成数据

分派策略

降噪配置

全部故障

最近 30 天

+ 筛选

全部故障

未关闭故障

待处理故障

处理中故障

已关闭故障

[Uptime Kuma Monitor Down] [flashcat] Flip UP to DOWN

待处理 0 / 1 11天23小时

[Uptime Kuma Monitor Down] [Flashduty] read ECONNRESET

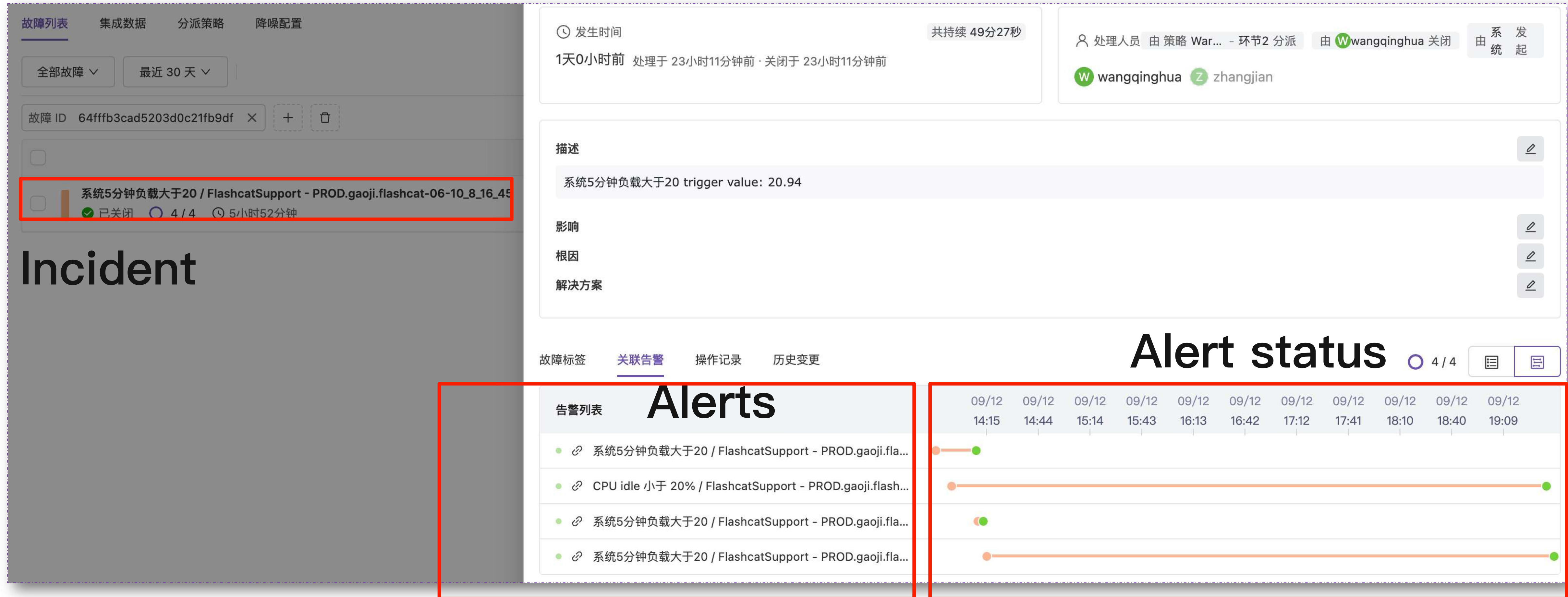
待处理 0 / 1 18天21小时

[Uptime Kuma Monitor Down] [flashcat] Flip UP to DOWN

待处理 0 / 1 22天16小时

测试FlashDuty发送 / victormetrics - flashcat-saas-03

处理中 0 / 8 25天22小时



- 告警的聚合分两层：1) 将Alert的状态变化进行聚合、 2) 多个相近的 Alerts 进行聚合。
- 告警的通知，以 incident 为最小单位进行发送，降低通知的数量（80%下降）。

告警排班



值班管理 > 值班详情

监控管理 故障管理 费用中心 访问控制 审计 支持 设置

演示 ☒ 启用中 ☐ 研发团队

禁用

设置

当前值班 现在 ~ 1月13日 00:00
值班人 laiwei

下一值班 1月13日 00:00 ~ 1月15日 00:00
值班人 ysyneu qinyening

值班提醒

< ● > 2024 年 1月15日 - 1月22日

时间线

最终值班



值班规则

+ 新增规则



临时调班

+ 新增规则




可创建多个规则



临时调班

1. 提前规划值班表，可以让 on-call 工作更有计划性，减少疏忽和失误。
2. 通过值班表，可以有效的减少告警对非值班 team 的打扰，提升工程师的工作体验。


告警日历

对于一些业务，会有交易日和非交易日的场景，在非交易日的时候，服务会关闭，期间不需要任何告警通知。此时可以使用服务日历的功能，配置好哪些是非交易日，在告警规则中关联了服务日历之后，只有在交易日告警规则才会生效，不再需要频繁地修改规则的生效时间。

<  > 2024 年

 工作日  休息日

日 一 二 三 四 五 六

关联日历 

一月 2024

日	一	二	三	四	五	六
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

二月 2024

日	一	二	三	四	五	六
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29		

三月 2024

日	一	二	三	四	五	六
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31						

四月 2024

日	一	二	三	四	五	六
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30				

五月 2024

日	一	二	三	四	五	六
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	

六月 2024

日	一	二	三	四	五	六
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30						

七月 2024

日	一	二	三	四	五	六
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

八月 2024

日	一	二	三	四	五	六
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

九月 2024

日	一	二	三	四	五	六
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30					

十月 2024

日	一	二	三	四	五	六
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

十一月 2024

日	一	二	三	四	五	六
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30

十二月 2024

日	一	二	三	四	五	六
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

中国节假日

中国香港节假日

日本节假日

新加坡节假日

英国节假日

美国节假日

告警升级

一线: Tier 1

升级

二线: Tier 2

环节 1

团队 > 一线团队-演示

个人

值班

以单聊渠道通知

遵循个人偏好

遵循统一设置

请确认已完成设置

每个人可以走不同的通知方式，去 [账户设置](#) 页面更新配置。

以群聊渠道通知

循环通知设置

超过 30 分钟后如果故障

未关闭

未关闭且未认领

，则升级到下一环节。

环节 2

团队 > 二线团队-演示

个人 > laiwei

值班

以单聊渠道通知

遵循个人偏好

遵循统一设置

请确认已完成设置

每个人可以走不同的通知方式，去 [账户设置](#) 页面更新配置。

以群聊渠道通知

循环通知设置

每隔 10 分钟通知 1 次，最多通知 1 次。

超过 30 分钟后如果故障

未关闭

未关闭且未认领

，则升级到下一环节。



通过告警升级的机制，有效的协调一线和二线的工作安排，避免告警漏处理。

告警通知



常见的告警接收方式

- > 飞书应用 (已设置)
- > 钉钉应用 (已设置)

取消

description : Send order message failed too many times. trigger value: 100
resource : flashcat-001.bj, flashcat-002.bj, flashcat-003.bj
business : FlashDuty
check : send order message failed
cluster : Default
collector : categraf
env : release
promql : increase(send_msg_error_count{channel="order"}[1m]) > 0
region : Beijing
runbook_url : https://flashcat.cloud/categories/flashduty/
service : event-engine
topic : fc-notify
trigger_value : 20

自定义消息字段和样式

- > 企业微信应用 (已设置)
- > Slack 应用 (已设置)
- > 飞书机器人 (已设置)
- > 钉钉机器人 (已设置)
- > 企业微信机器人 (已设置)
- > Telegram 机器人 (已设置)
- > Slack 机器人 (已设置)
- > Zoom 机器人 (已设置)
- > 邮件 (已设置)
- > 短信 (已设置)

通知模板 开箱即用

交互式的消息卡片

FlashDuty Online



FlashDuty

【处理中】Warning-订单系统

测试FlashDuty发送 / victormetrics - flashcat-saas-03

触发时间: 2023-06-10 15:00:43

认领时间: 2023-06-10 15:07:12

聚合告警: 8条 (风暴中)

分派人员: @快猫星云

description: 测试FlashDuty发送 trigger value: 5.99194

name: cpu_usage_active

check: 测试FlashDuty发送

metric: cpu_usage_active

promql: cpu_usage_active>0

跟进处理

直接关闭

当前故障已聚合8条告警, 触发告警风暴, 请加急处理!

9:41



FlashDuty Online



FlashDuty

【已关闭】Warning-私有云业务

主机 Default - flashcat-02-10-7-32-21

触发时间: 2023-06-10 15:00:43

关闭时间: 2023-06-10 15:07:12

分派人员: zhangsan

resource: flashcat-12.bj

promql: mem_available_percent < 15

region: beijing

cluster: Default

env: product

type: Pader

name: sdb

跟进处理

直接关闭

屏蔽2小时

数据统计

过去一周降幅比例 🕒

71.07 % ↑ 5.52%

过去一周故障 MTTA 🕒

6 小时 ↑ 1550.47%

过去一周故障 MTTR 🕒

2 小时 ↓ 31.8%

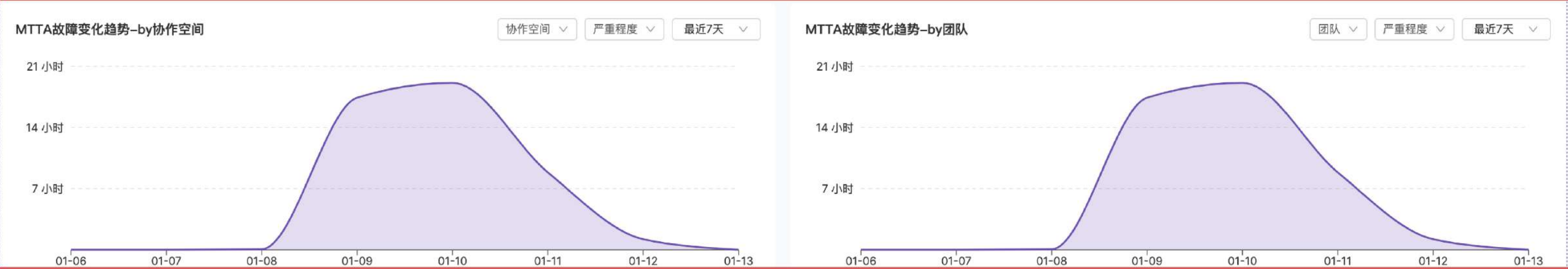
过去一周响应比例 🕒

20.91 % ↑ 278.89%

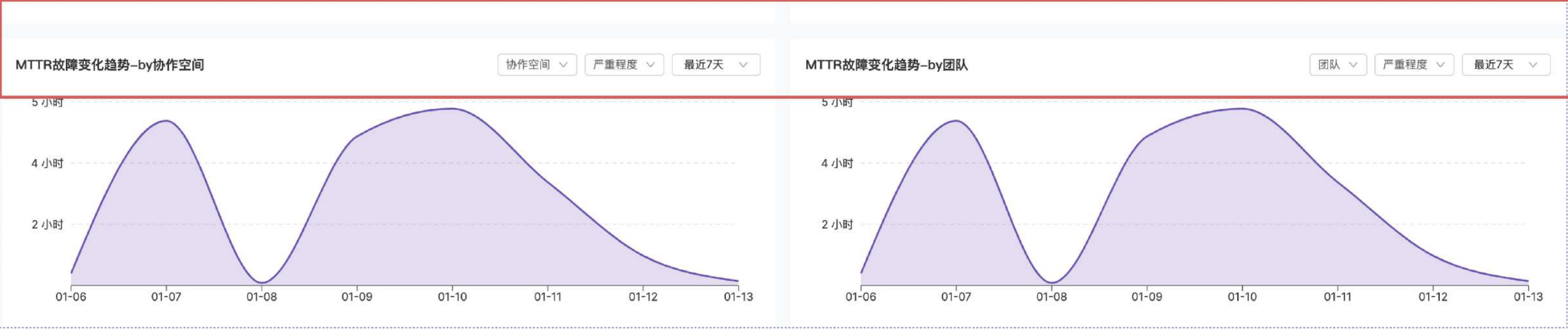
过去一周故障数量

263 条 ↓ 14.61%

MTTx 统计



MTTx 分类趋势



TopN 告警统计

Top20告警检查项

协作空间

最近7天

名称	数量
Binlog同步延迟	226
SaaS-HTTP请求出错	128
SystemDefault_acs_ecs_dashboard_CPUUtilization	68
连接mysql数据库超时	62
请求数突增	44
...	...

Top20告警对象

协作空间

最近7天

名称	数量
PRC shcat-06-10_8_16_45	178
customers-vpn-01	144
dev-flasheye-02/123.56.237.129	77
PRC shcat-04-10_8_16_43	48
api-...	44
...	...

工作量统计

个人指标

成员

团队

成员 +10

严重程度

最近7天

姓名	被分派故障	认领故障	关闭故障	MTTA	MTTR
guoyu	39	0	0	0秒	0秒
liming	48	44	43	7小时58分钟	8小时45分钟
yushu	79	4	2	14分16秒	26分42秒
wangq	23	0	0	0秒	0秒

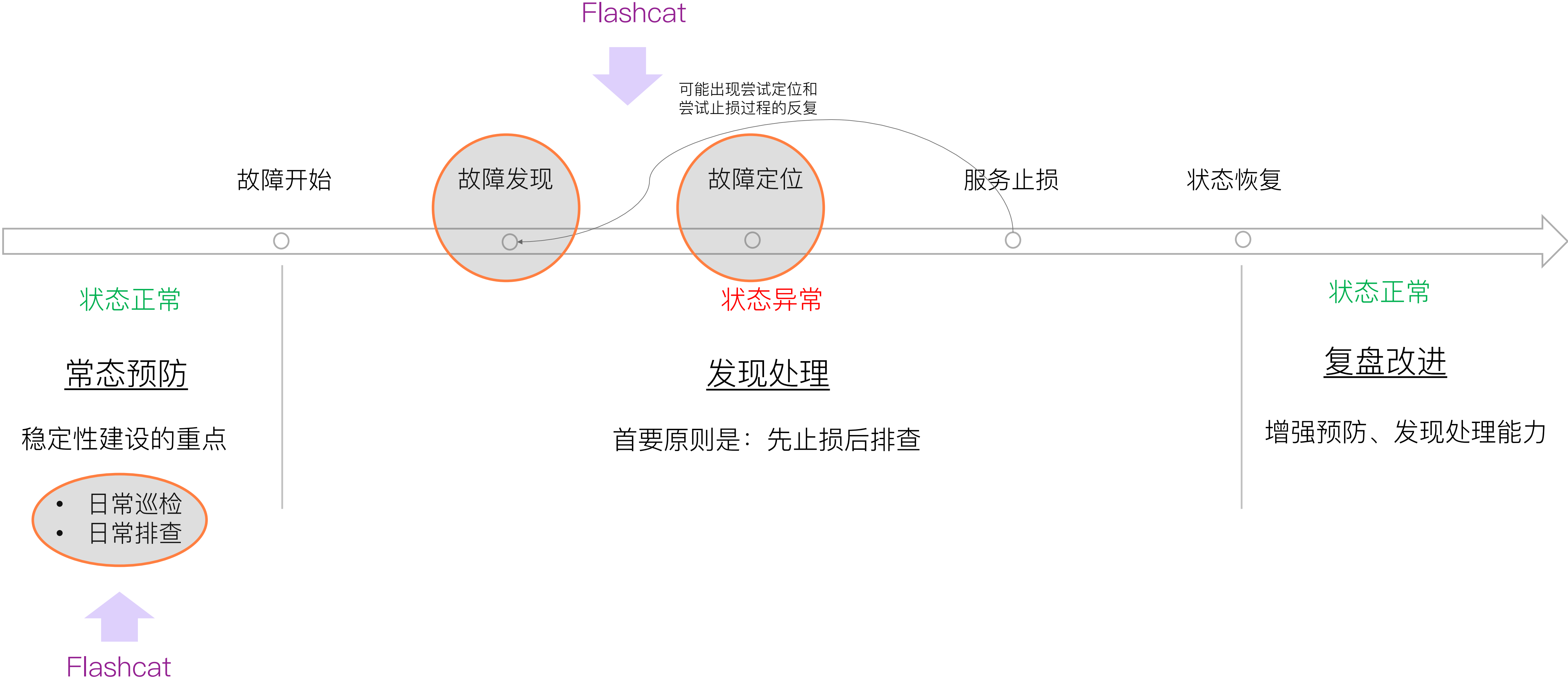


统一观测

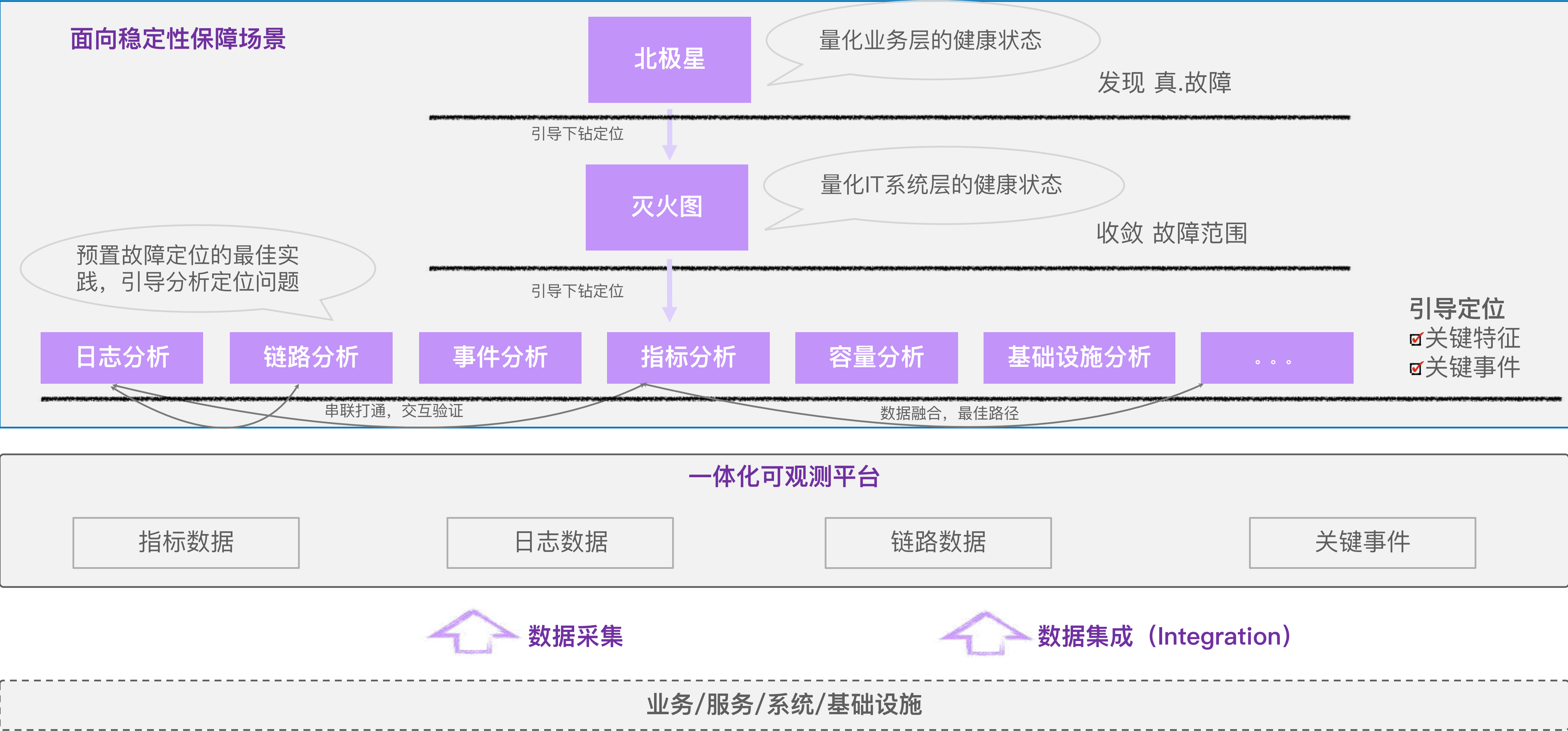
Flashcat 面向服务稳定性保障场景



以故障处理为中心的稳定性保障模型



面向业务视角的故障发现定位体系



北极星：第一时间发现真故障



北极星指标

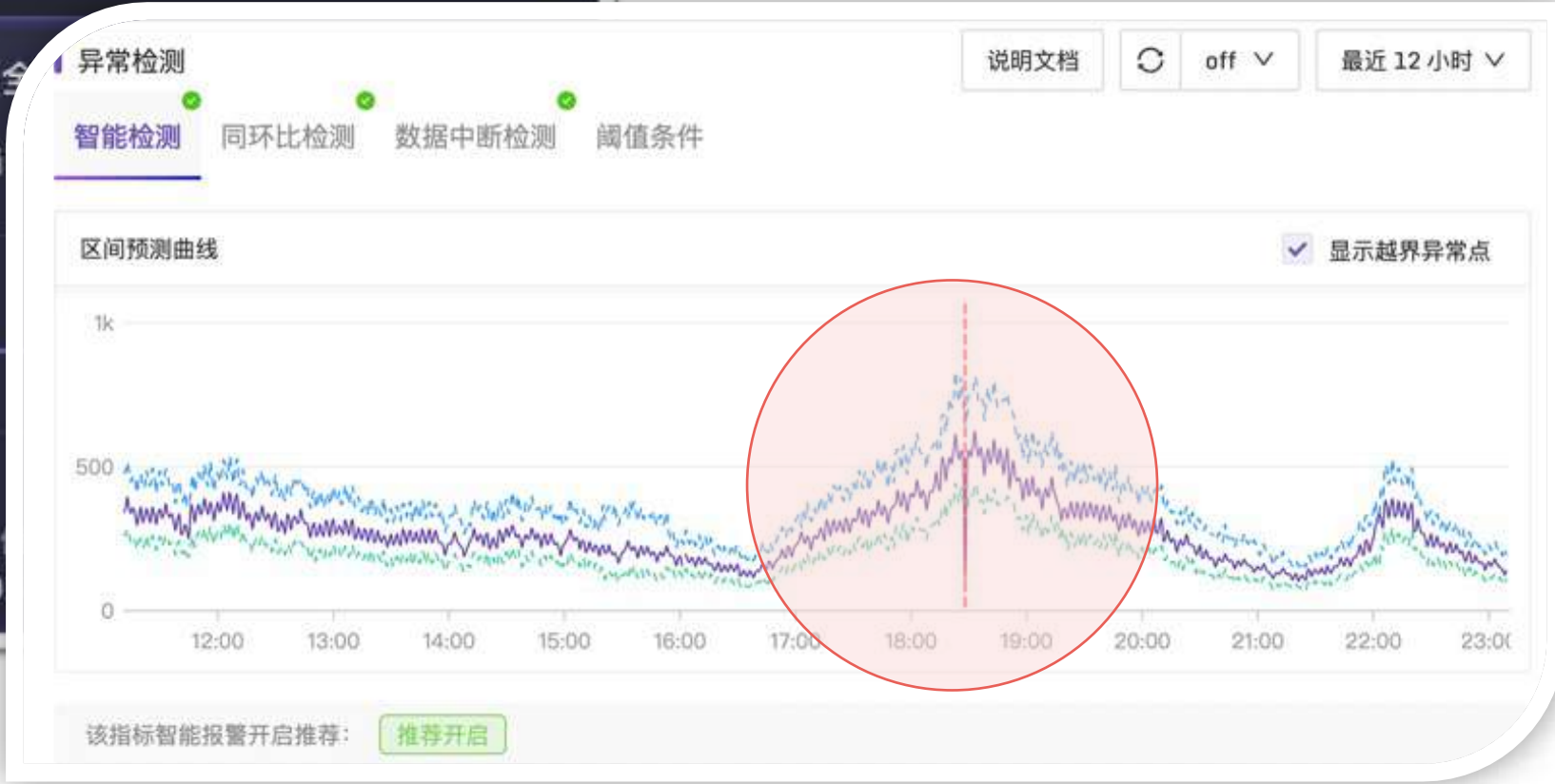
对业务或者用户体验的量化，才是衡量系统是否稳定的关键。

真故障

对业务或者用户产生影响的故障，才叫真故障。

智能检测

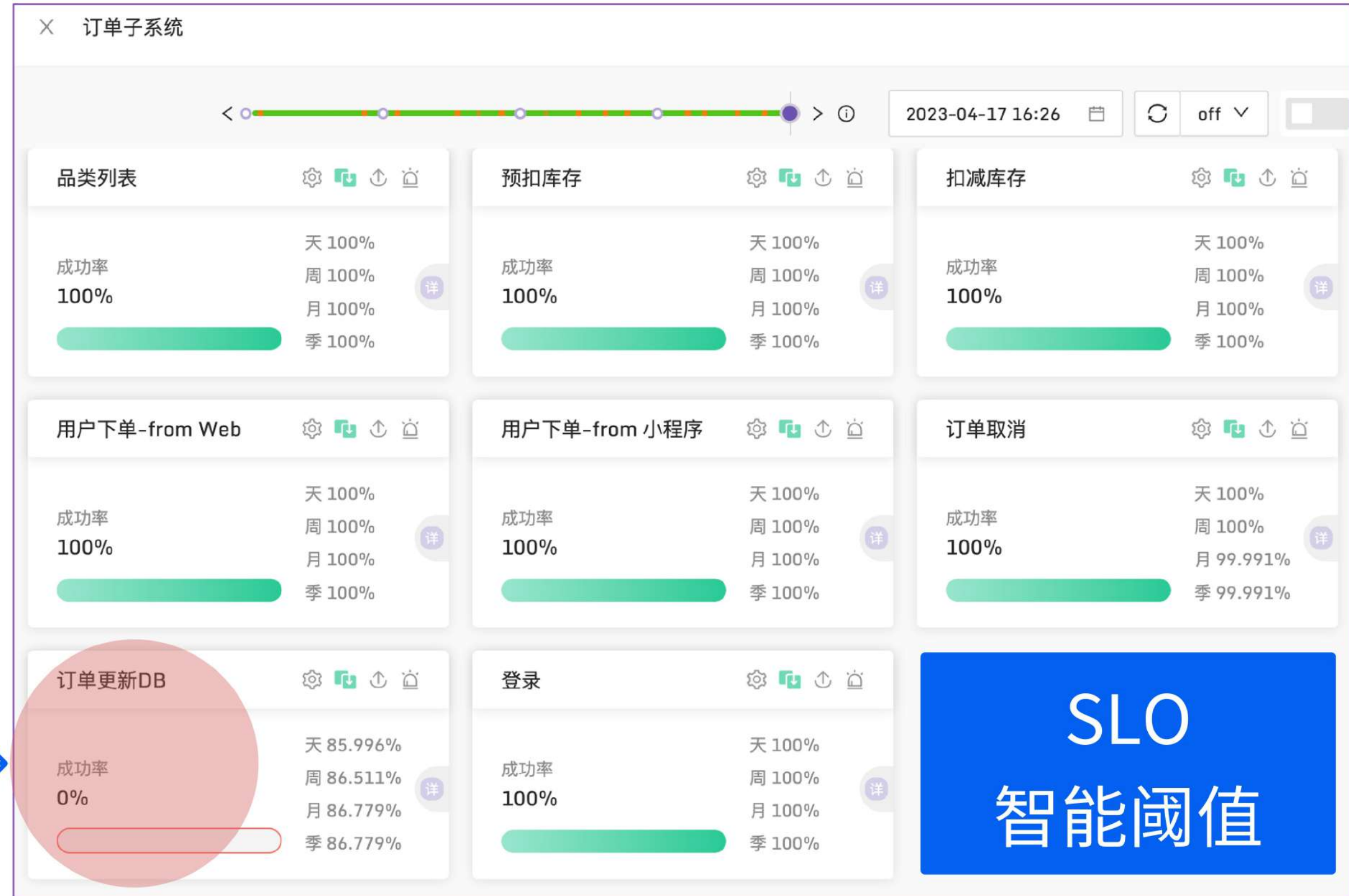
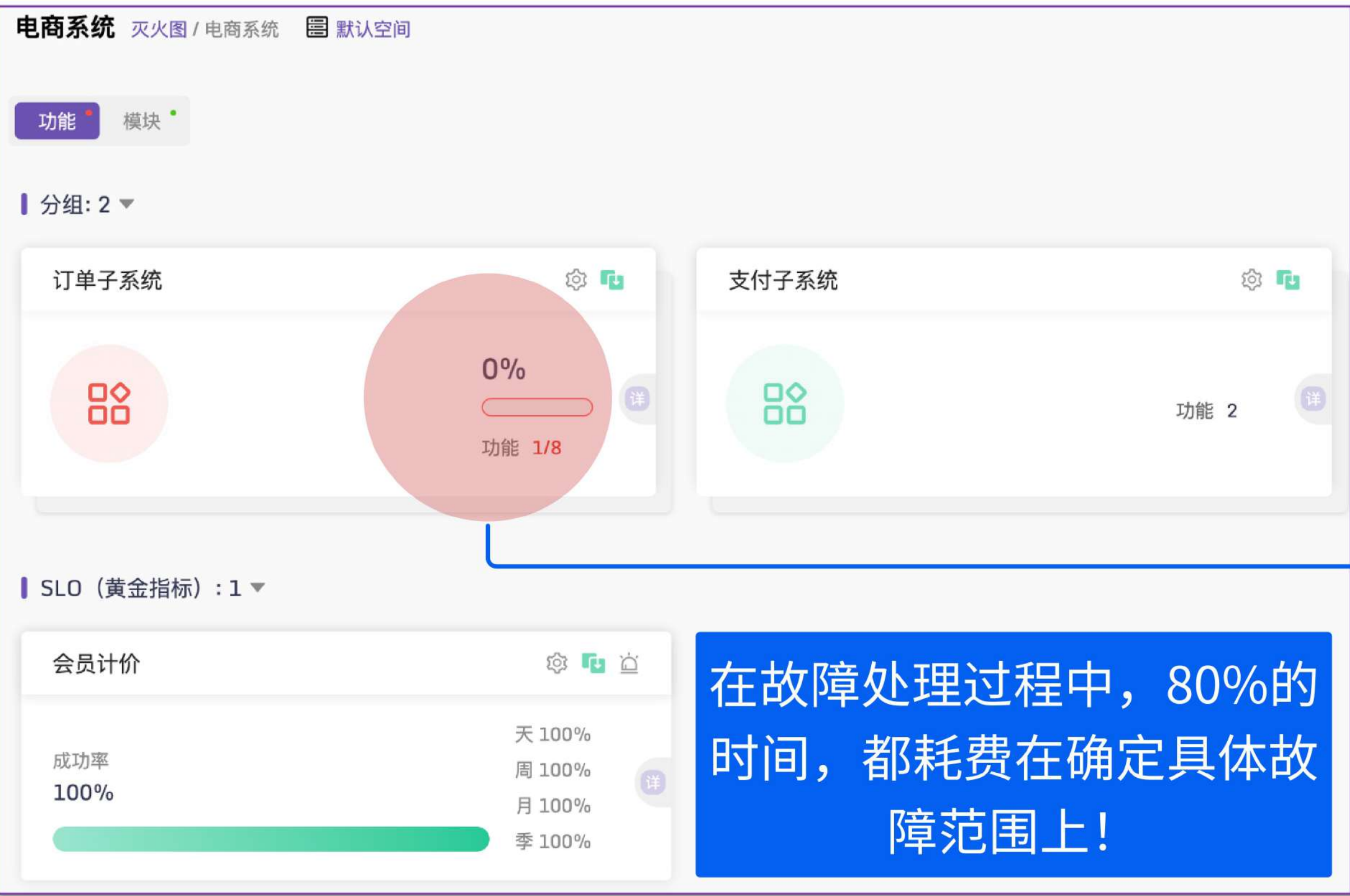
当北极星指标发生波动，会第一时间被检测到，并通知相关技术团队。



灭火图：故障快速定界



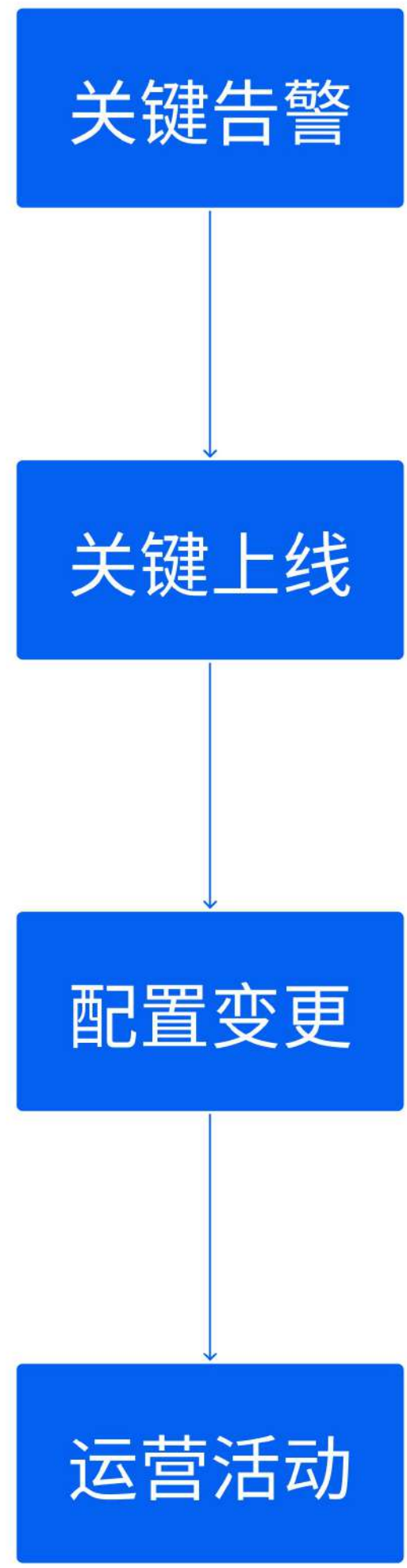
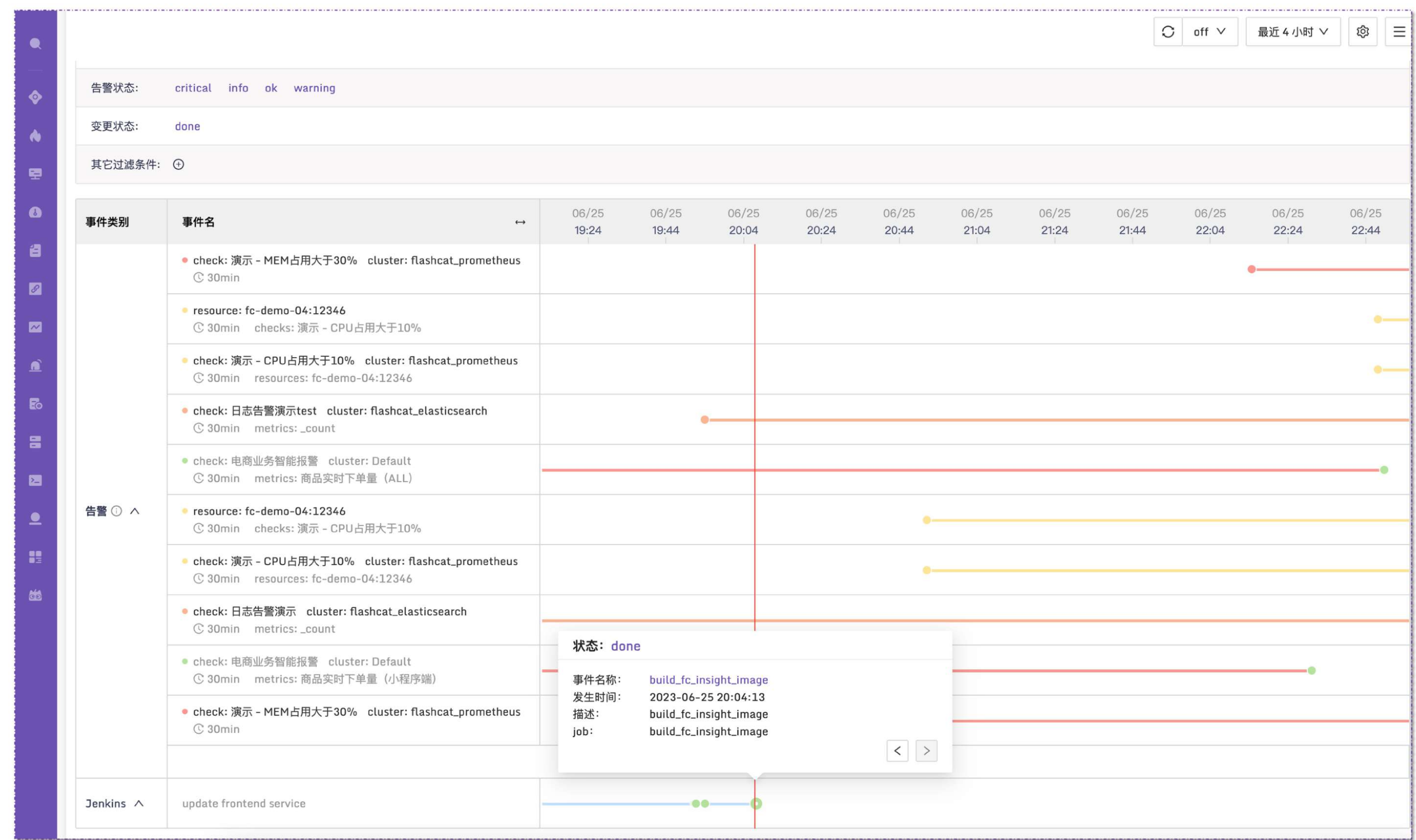
灭火图：实时度量 IT 服务的核心功能/核心模块的健康状况，快速收敛故障范围

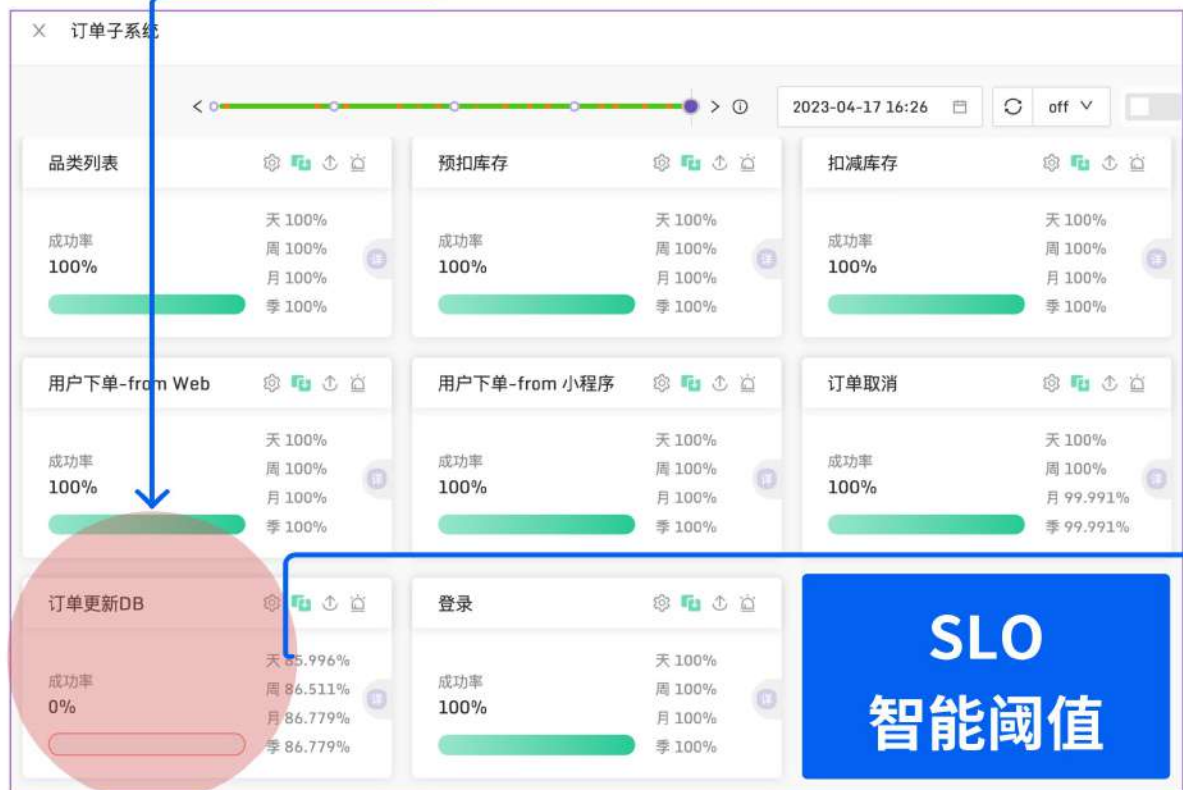
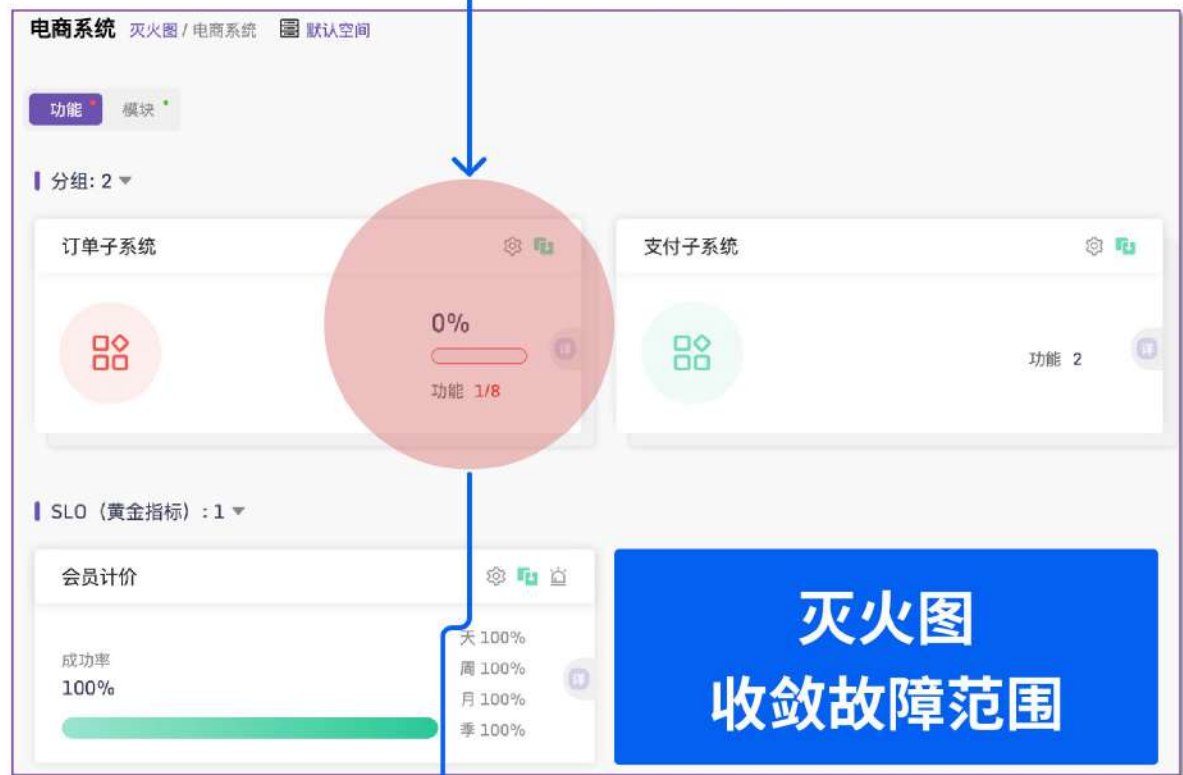
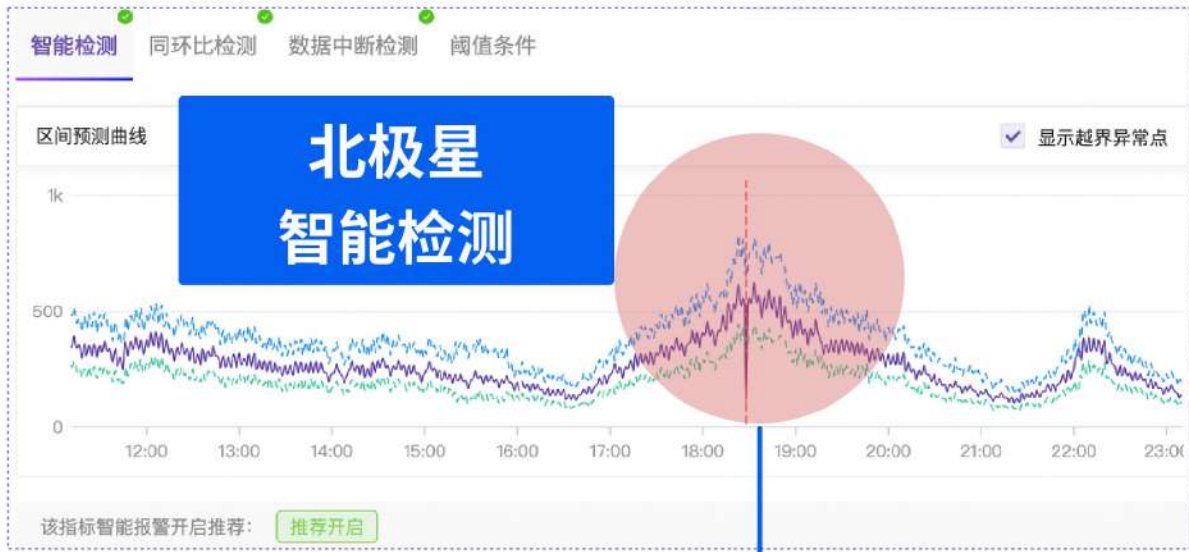


事件墙

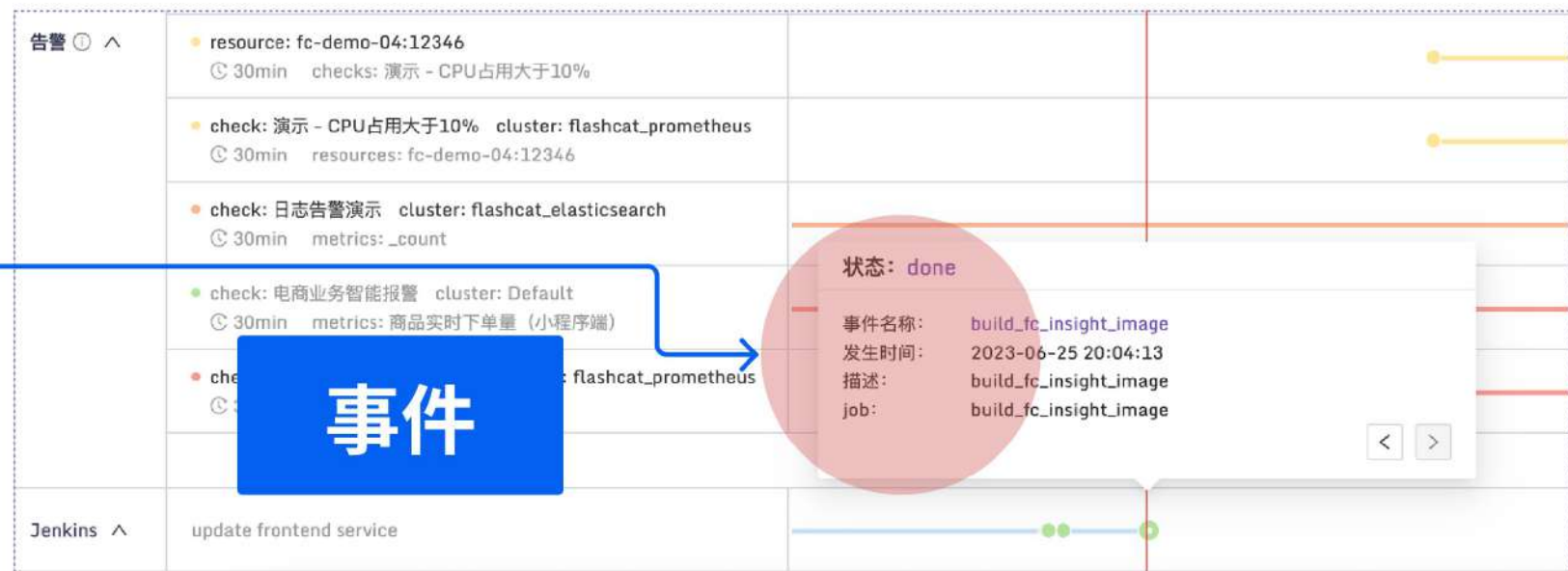
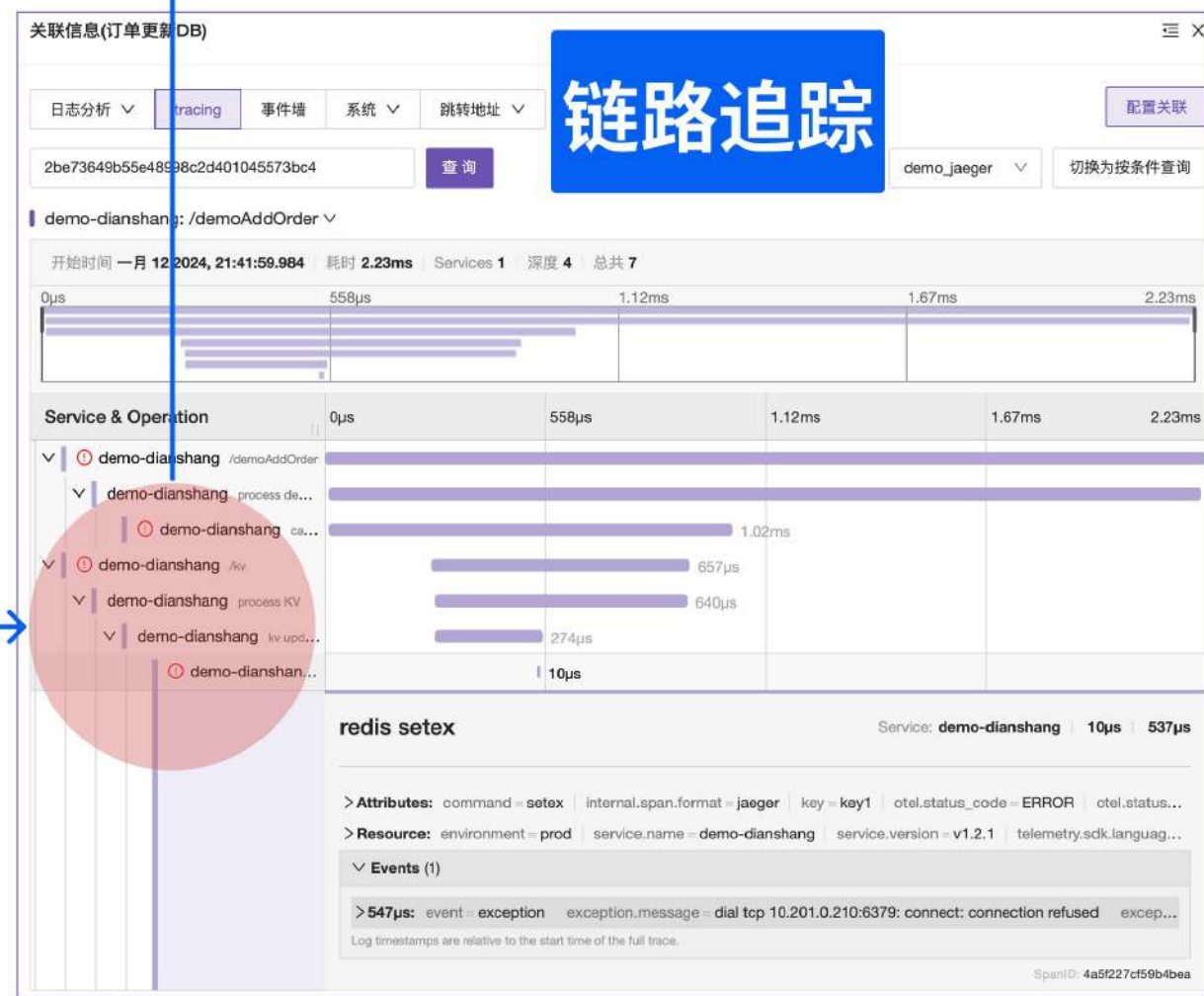
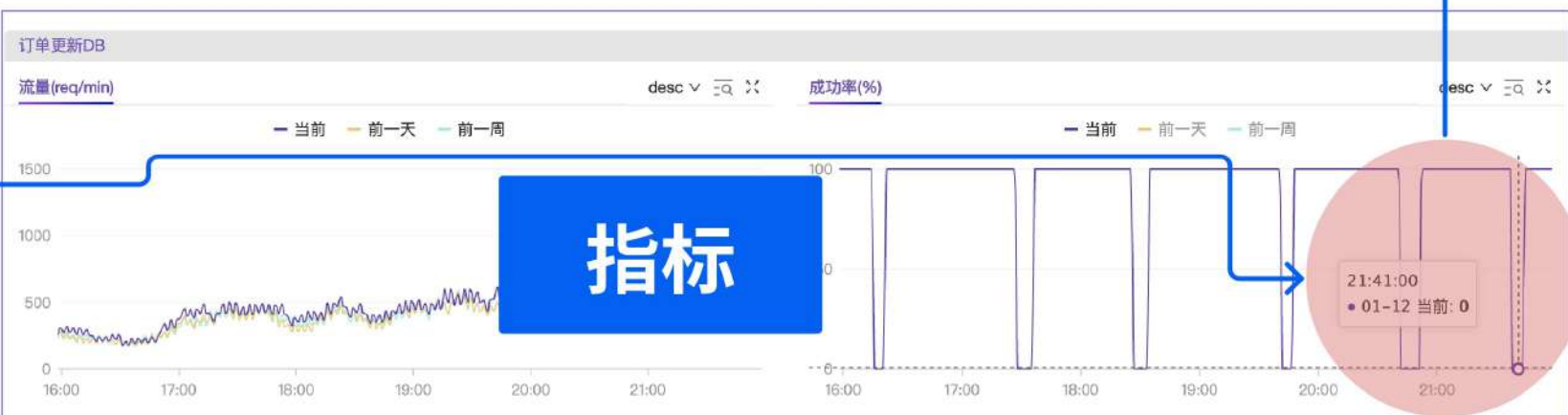
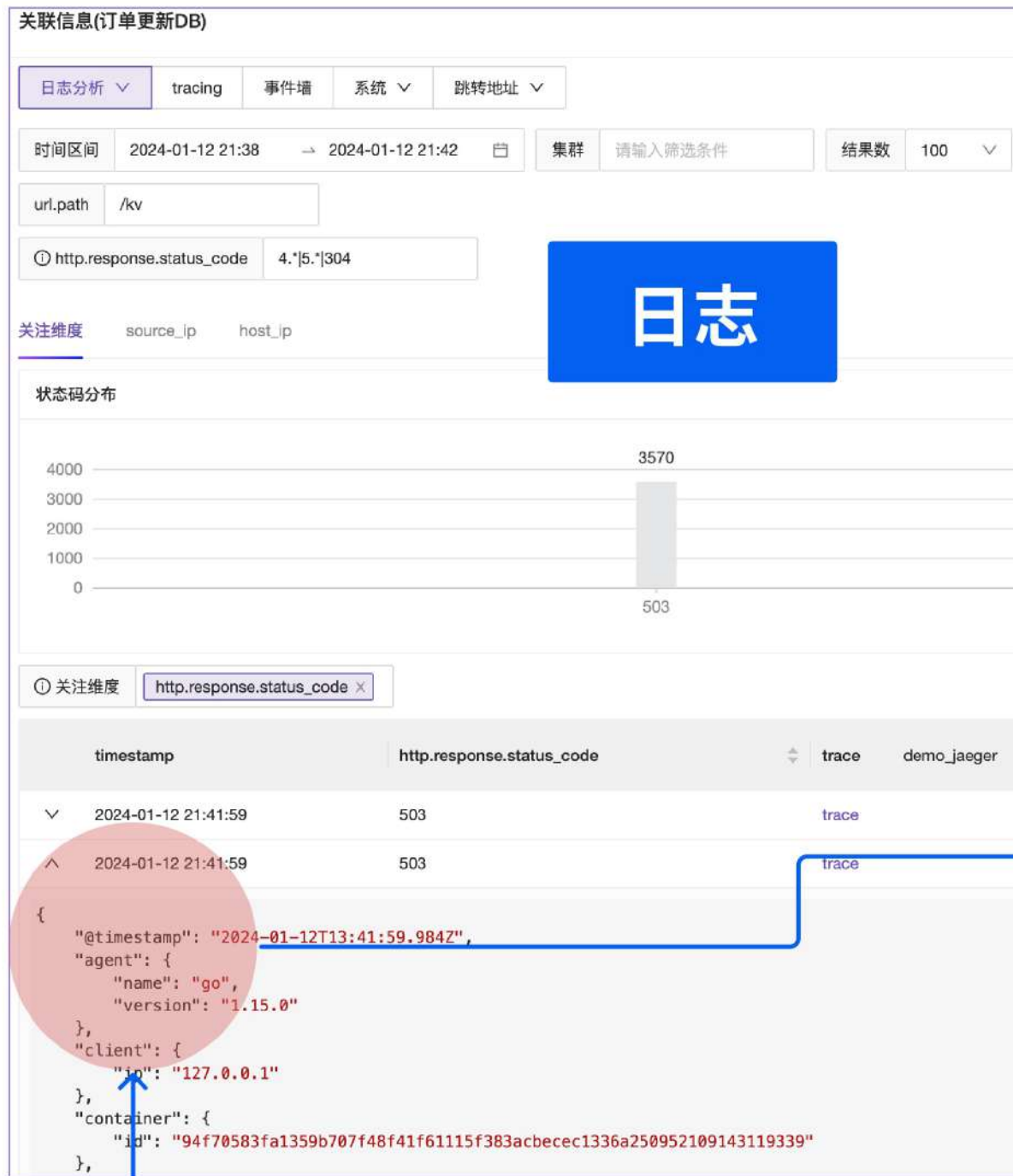
事件墙：快速确定或者排除「可疑事件」

70%的故障都是由变更引起的！





层层下钻



Flashcat 优势



屏蔽了多个分散的监控工具

轻松监控多云Region，从业务、到应用、基础设施，开箱即用。



内置了故障处理的最佳实践

当业务受损时，总能第一时间发现，并和 IT 系统深入联动，辅助技术团队快速展开调查。



IT 架构无关，只需一个平台

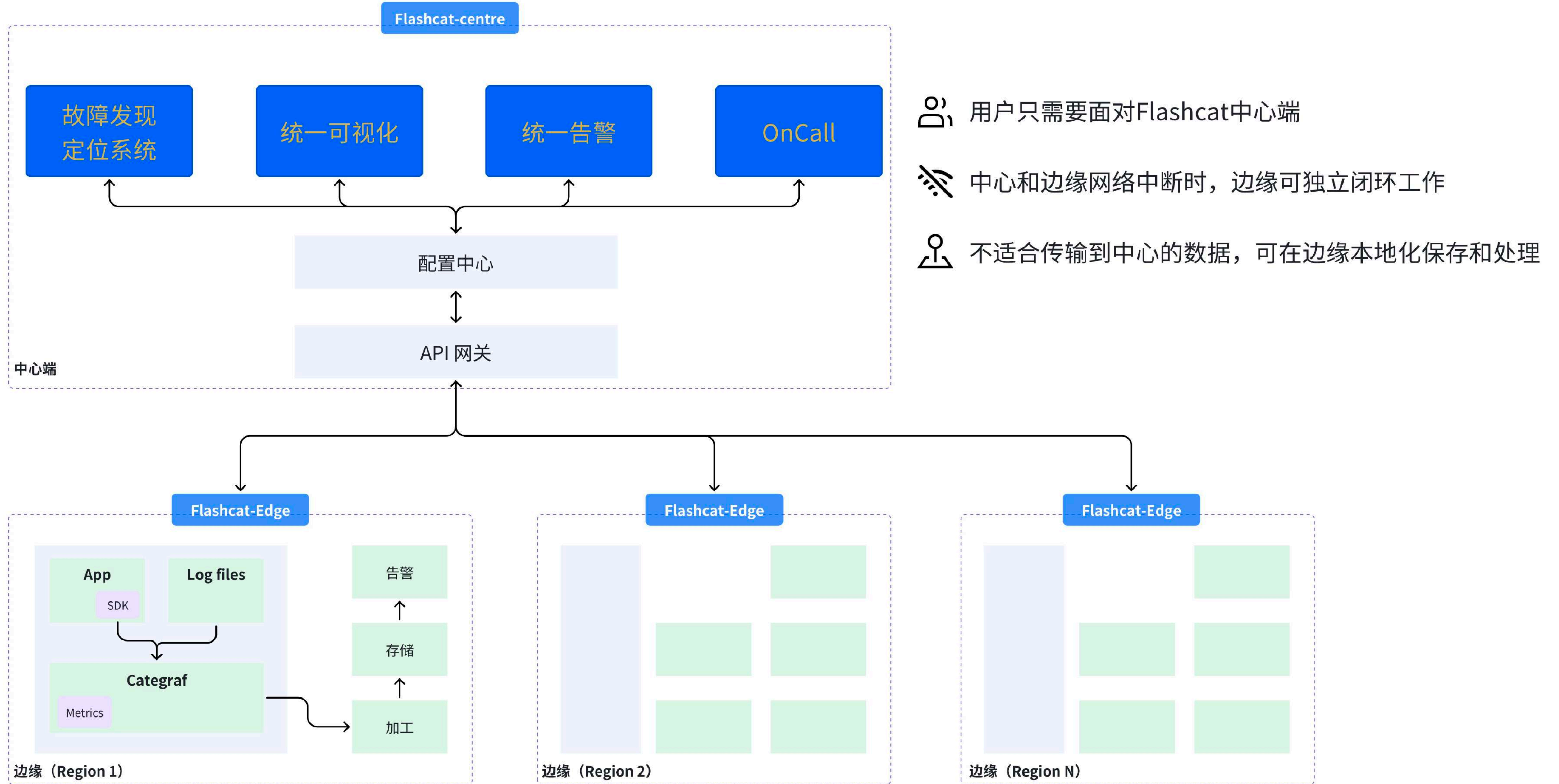
支持物理机、网络设备、容器、K8s，微服务、云上云下，无论采用什么样的 IT 架构，只需要一个平台。

我们的用户喜欢 Flashcat



					
海底捞	海大集团	益丰大药房	高济健康	莉莉丝游戏	悠星网络
					
哈啰	阳光出行	叮当快药	UU跑腿	香港医管局	国泰君安期货
					
小马智行	Zenlayer	吉野家	途游游戏	路特斯科技	地平线
					
六分科技	鹿客科技	畅捷通	八维通	海康威视	作业帮
					
中国电信	顺丰航空	当当	马泮齿科	方正证券	华东师范大学

创新的边缘部署模式



六分科技——国内领先的高精定位服务产品专业提供商



快猫助力六分科技，打造统一观测平台，构建全局稳定性视图

通过 Flashcat 平台，六分科技整合了 Prometheus、ClickHouse、日志、云监控等多个数据源，其中包括近 10 个 Prometheus 集群，十余个日志主题，实现统一的报警管理、数据可视化，降低了监控工具的维护成本，只有一位工程师负责监控产品的对接，就满足了内部对于监控、报警功能的使用，节省了人力，节省出的人力就可以投入到其他更有挑战的方向上。

了解更多

<http://flashcat.cloud/blog/liufen/>



六分科技：

基于虚拟参考站技术原理，依托在全国自建的约3000个CORS基站，自研终端RTK算法与组合导航算法，以“网-云-端”一体化解决方案为海量用户提供5系统16频点、全天候、实时厘米级和亚米级的高精度定位服务。公司高精度定位服务已覆盖智能驾驶、共享出行、精准农业、测量测绘、智慧城市、大众应用等多个领域。

痛点：

1. 监控工具太多，维护和使用都很麻烦
2. 缺少业务维度的监控
3. 缺乏统一的稳定性视图，缺乏故障定位的驾驶舱

效果：

1. 通过Flashcat平台，整合了Prometheus、ClickHouse、日志、云监控等多个数据源，其中包括近10个 Prometheus 集群，十余个日志主题，实现统一的报警管理、数据可视化，降低了监控工具的维护成本，**仅需投入一个人力**。
2. 建立了一整套稳定性的量化体系，依靠北极星第一时间发现故障，依靠灭火图定位故障，利用FlashDuty实现告警值班和故障协同处理，**缩短了整个故障处理的时间**。

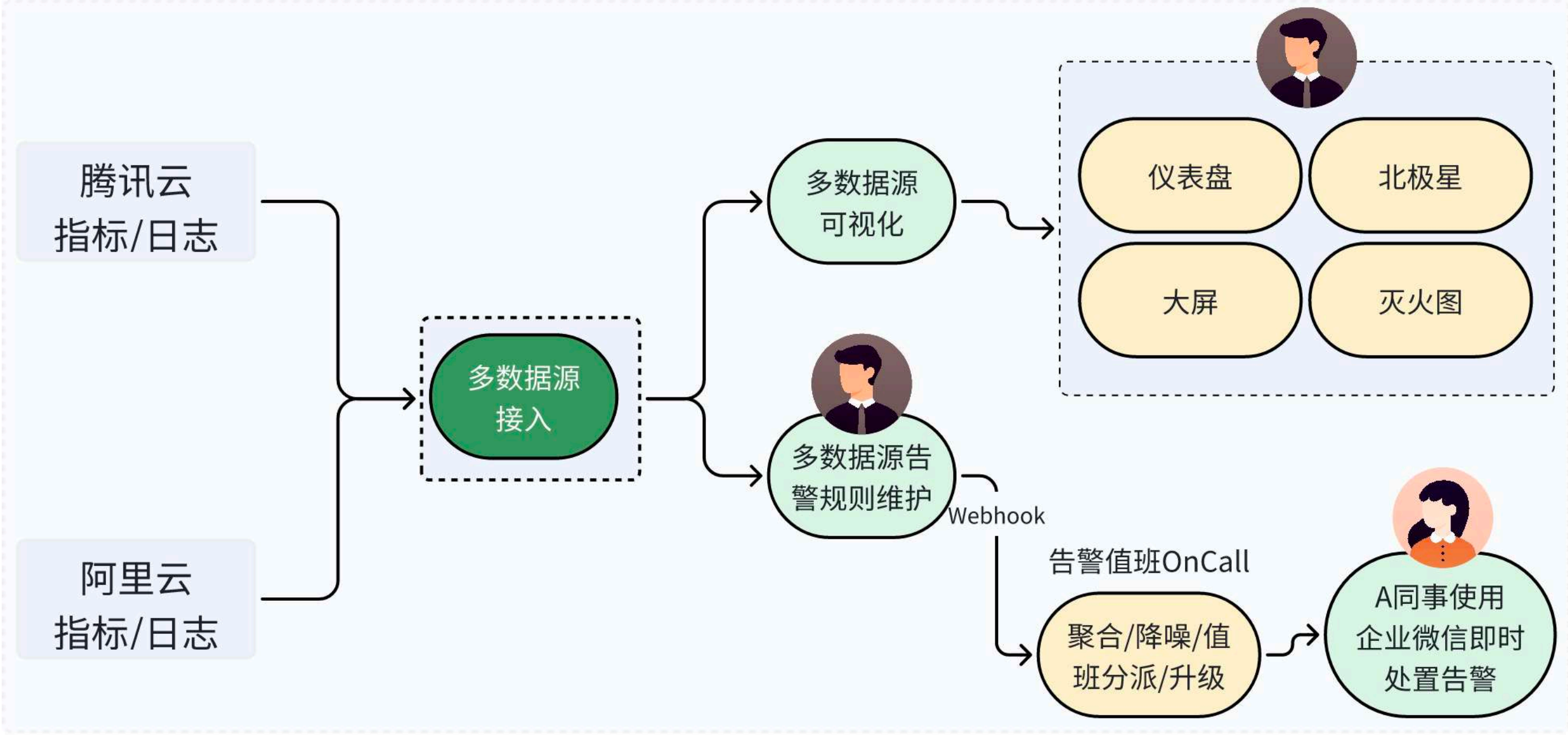
阳光出行——国内领先的出行科技公司



阳光出行是国内领先的出行科技公司，致力于为用户提供温暖的网约出行服务，阳光出行以司乘安全为前提，带给用户便捷、有性价比和人性化的服务。

阳光出行每天承载着数百万次出行需求，IT系统的可靠性至关重要。同时出行服务的场景具有非常明显的潮汐效应，因此弹性也是构建IT系统需要考虑的关键因素。阳光出行技术团队，依托国内领先的公有云提供商，采用多云架构，在可用性、弹性、成本、供应商依赖、最佳实践等方面，拥有领先的优势，积累了丰富的经验。

相应的，多云架构也给技术团队带来了一定的复杂度和技术挑战，最显著的就是如何高效的构建跨云的可观测性体系，提升故障发现、问题排查、性能分析等方面的能力。



挑战：

- ❑ 跨多云的监控数据权限管理难、安全隐患大
- ❑ 监控工具多且分散，维护和使用成本高
- ❑ 跨多云的故障发现和定位体系缺失，稳定性保障难度高

落地效果：

- ✓ 多云统一的可观测性数据权限管理
- ✓ 多云统一的仪表盘
- ✓ 多云统一的告警管理
- ✓ 高效的故障发现定位体系



医药健康企业基于 Flashcat 有效加强 IT 服务故障管理能力

通过和快猫团队合作，建设并落地 Flashcat 平台，目前公司 A 级产品线北极星指标监控实现了全覆盖，P3 级及以上故障北极星监控发现率为100%，MTTI 控制在 5 分钟以内。真正做到了先于用户发现问题，让故障处理变被动为主动。

[了解更多](#)

<http://flashcat.cloud/blog/case-flashcat-in-medicine-company/>

背景介绍：

一家专注大健康领域的医药健康产业集团。在国内有覆盖广泛的连锁药店（10000+连锁实体门店），同时拥有包括在线医疗、药品配送等医疗相关业务。在中心端我们的 IT 服务保障着全国药店和在线业务的高效运营。

痛点：

- 1. 故障发现慢，主要依赖用户保障
- 2. 缺乏基于业务视角的全链路监控，故障定位耗时较长
- 3. 缺乏对重要故障场景的应急预案的梳理和演练

效果：

- 1. 通过落地Flashcat平台，公司A级产品线北极星指标监控实现了全覆盖，**P3级及以上故障北极星监控发现率为100%**，MTTI控制在5分钟以内。真正做到了先于用户发现问题，让故障处理变被动为主动。
- 2. 故障定位能力建设也已取得重要进展，我们和业务一起梳理了公司A级产品线核心主流程依赖的接口和模块，并将梳理结果落地到Flashcat灭火图系统，并建立了北极星和灭火图的关联，**完成了服务全景图的建设，加速了故障处理和团队间的协同效率。**

4多

4问题

门店多	全国上万家实体药房，300个地级市。	故障发现慢，依赖用户端报障
业务多	智慧药房、药急送、慢病管理、创新支付、互联网医院。	故障影响面、影响程度难确认
系统多	会员、运营、问诊、处方、履约、支付、门店、中台、云平台。	故障处理进度不明确，靠少数技术人员对外反馈
链条多	线上线下融合，互联网和大数据技术驱动，涉及众多链条，业务流程复杂。	故障处理中团队协作难，信息不透明

某中国领先的火锅连锁企业



某知名火锅连锁企业是中国领先的餐饮企业，近2000家门店遍布全球，由于门店餐饮行业的特殊性，需要靠前部署服务，所以在每家餐厅中，会部署相应的服务器，及相应IT设备，本地会运行POS、会员、下单等业务。

公司有众多的餐厅门店，各个门店业务流量不同，门店的IT设备由于城市、开业时间等因素，其型号也不相同，服务器、应用程序分散式部署，给应用管理、IT运维、以及先于门店发现问题，带来了极大的挑战。

痛点和挑战

- ❑ 如何高效的集中监控所有的门店？
- ❑ 如何度量、发现、治理有 IT 隐患的门店？
- ❑ 如何让总部 IT 先于门店发现故障？

解决方案：

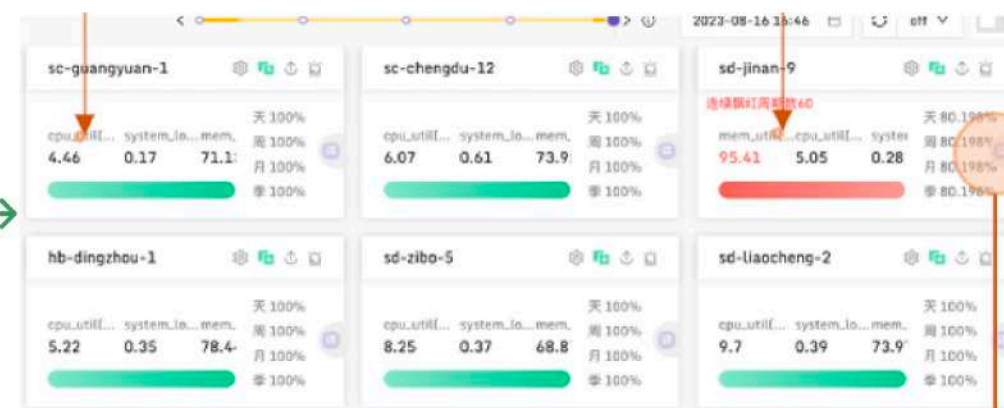
Flashcat 连锁门店集中监控方案，基于All-in-One的开源采集器Categraf，加上业界领先的开源监控夜莺（Nightingale），集中化的监控所有的门店，并采用数据驱动的理念，对所有的门店 IT 健康状态进行科学的量化，真正做到先于门店发现问题，及时高效治理有IT 隐患的门店。

- ✓ 层次化展示餐厅门店的IT质量，满足不同的角色对于门店IT质量的可视化需求。
- ✓ 下钻关联的能力，可以支持层层下钻，找到异常餐厅，以及具体的异常原因。
- ✓ 多样化的可视化效果，可以灵活的展示异常餐厅的地理分布、异常趋势变化等，便于问题的定位排查。

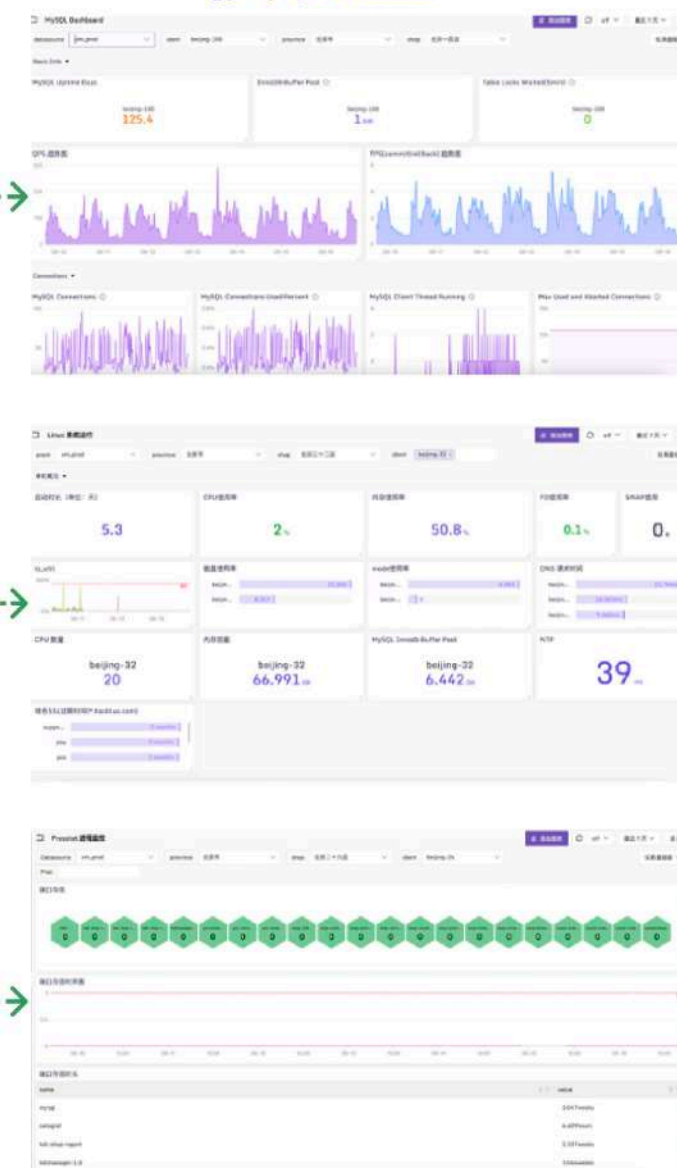
北极星



灭火图



仪表盘



企业版 vs. 开源版

更详细对比请访问 <https://flashcat.cloud/docs>

北极星 >

灭火图 >

事件墙 >

日志分析 >

On-Call 值班中心 >

数据源管理 >

数据采集器 >

仪表盘 >

告警管理 >

告警自愈 >

分布式链路追踪 >

基础设施 >

人员组织 >

系统配置 >

操作审计 >

技术支持 >

咨询实施 >

开源版



企业版





Flashcat 开源监控引领者 故障定位真帮手



访问 www.flashcat.cloud 了解更多