

# 如何落地 1-5-10

Kindling-OriginX 全球首创的故障根因推理引擎

# 如何发现业务真实告警——传统告警的弊端

传统告警的目的是识别排障方向

不断积累踩坑告警越来越多

人员流动，告警缺少治理

故障来临的时候，告警太多，不知道该往哪里排查

真正有用的告警，淹没在误告警当中

# 如何发现业务真实告警——基于SLO的告警

什么指标影响真实业务？

延时

错误率

访问量

SLO目标反应真实业务要求

P95延时参考历史数据

成功率大于99%

访问量最少 最多

SRE的黄金指标

# 故障一旦发生如何做到最快识别可行动的根因

可行动的根因定义

依据根因，  
可以制定  
标准化自  
愈流程



自动化分析每个用户请求智能化给出  
根因结论：为什么慢和错



人肉分析每个用户请求不靠谱

## 海量Trace分析工作量大

不一致的性能问题表现：同一个性能问题可能在不同的span或服务调用中显现，这使得确定问题的确切原因变得更加困难

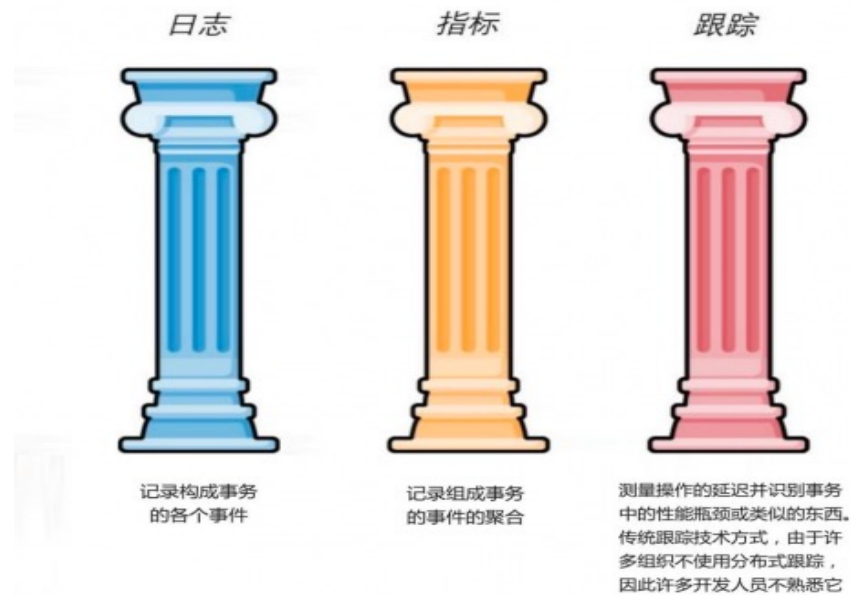
信号干扰与隐藏的性能瓶颈：性能问题的真正原因可能被各种干扰信号掩盖，如垃圾收集（GC）造成的延迟。这些干扰项可能使分析变得复杂，特别是在定位那些不常出现或隐蔽的性能问题时

动态和复杂的系统环境：云原生环境，可能在不同的实例或不同的情境下表现不同，增加了诊断的复杂性

# 机器分析的结论真的靠谱吗？

故障需要具备可解释性获得SRE认可

## 日志、指标与跟踪如何组合才具有可解释性？



可观测性三大支柱数据

# 可观测性数据有机组合在一个故障报告之中

## 标准化的排障方式简化了 用户体验设计

在排障过程中，什么时候该看什么数据，已经根据专家经验在界面流程当中设计好了，排障的特异化需求就能被固化并同步下来

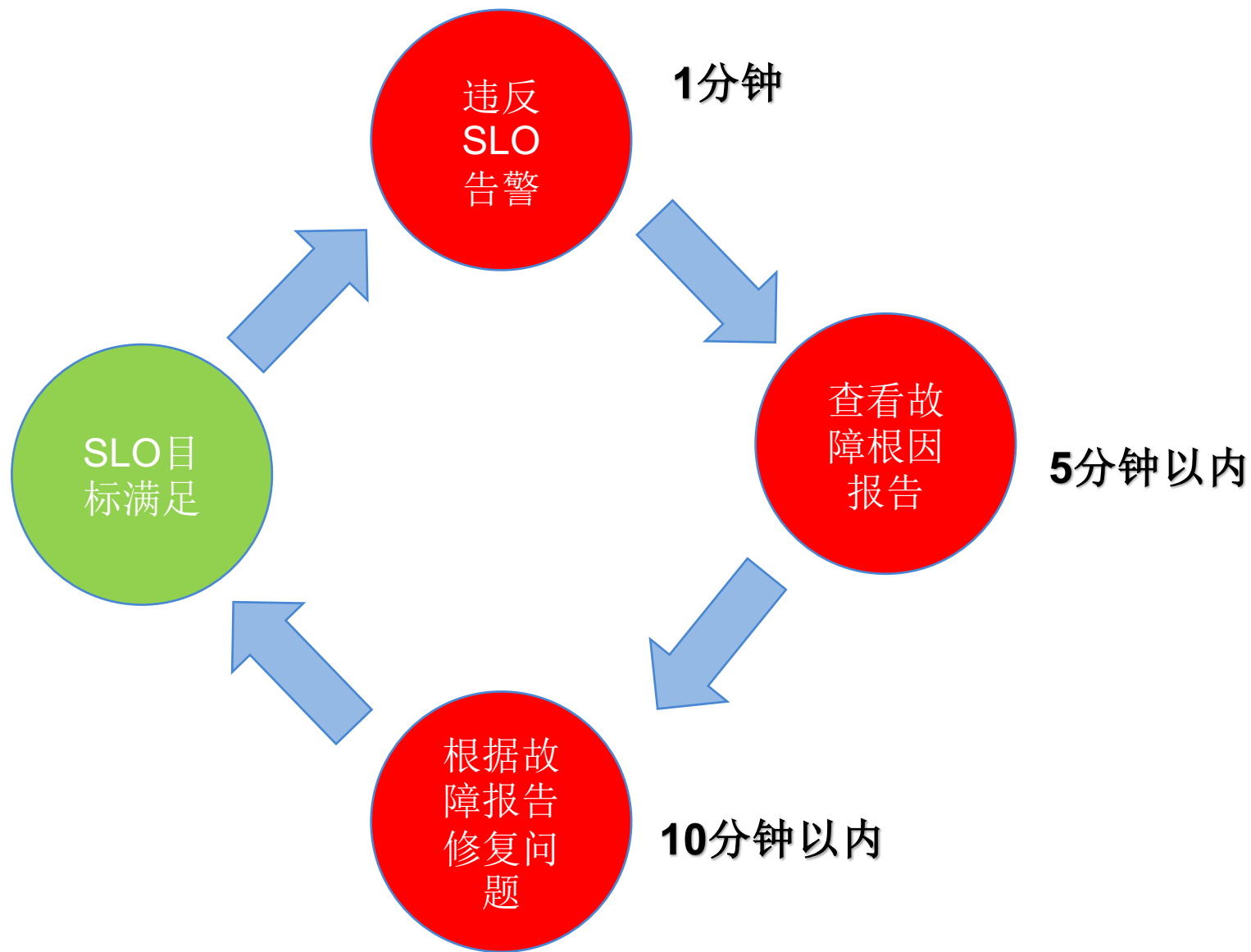
## 自动化智能化查询关联 数据

标准化排障过程当中所需的数据自动化关联查询关联出来

## 故障根因结论

补充排障人员知识短板，比如runq的场景绝大多数运维人员看不懂

# 违反SLO的告警闭环处理——落地1-5-10





# 第一步设置SLO目标

## 修改配置信息 ✕

URL  
POST:/gateway/order/list  
服务入口 \*

请输入您的服务别名便于您快速找到它

### ①设置SLO目标

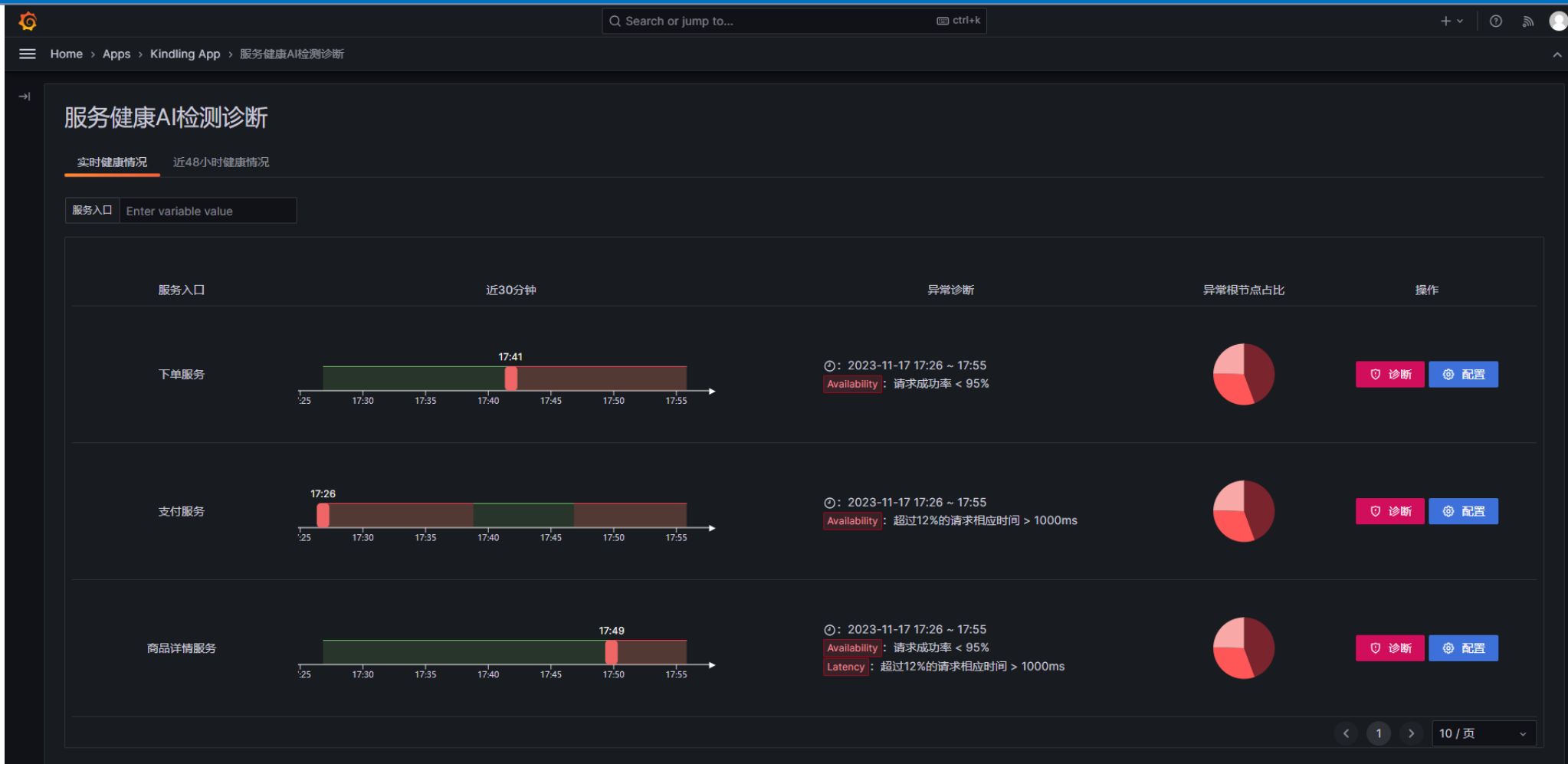
Availability: \* 请求成功率 >=  %  
历史参考值: last 1h=98% last 12h=95% last 24h=95% last48=95%

Latency \* 超过 >=  % 请求响应时间 <  ms  
历史参考值: P90=200ms P95=184ms P99=179ms

## 第二步，业务SLO执行情况

针对违反SLO的业务，进行理解

- 了解问题何时发生，
- 以及影响范围有多大



# 第三步，通过诊断发现有问题的请求

## 查看有问题的典型故障请求列表

### 故障列表

业务入口端点: POST:/api/v1/travelservice/trips/left  
服务入口别名: 余票 请求成功率 = 100% SLO可用性目标是95 % P99 = 757.14ms SLO延迟性目标是498.94 ms

故障服务名  水位线过滤

🕒 2023-12-21 20:03:45 to 2023-12-21 20:05:00 🔍 ↻

**Slow Trace List** 8 **Error Trace List** 0

故障服务名	请求时间	业务入口请求耗时	TraceID
> 192.168.1.5-ts-travel-service (7)	12/21/2023, 20:04:08	499.00 ms	07c4406b6c15437aa0d8277bd38698ed.16876.17031602483706973
192.168.1.29-ts-basic-service	12/21/2023, 20:04:07	824.00 ms	07c4406b6c15437aa0d8277bd38698ed.413.17031602478828611

# 第四步，查看故障报告，理解故障根因

## 查看故障报告，直接得到故障根因





Kindling-OriginX 全球首创的故障根因推理引擎

THANKS FOR YOUR WATCHING