基于大模型的根因分析实战

文吉 畅捷通信息技术股份有限公司

演讲嘉宾

文吉

十年以上SRE实战经验，特别是对ToB场景有丰富实战经验

用友集团 P9高级专家

多次对外分享，融合大模型能力升级智能运维

荣获了信通院颁发的“稳定性优秀案例”

目录
CONTENTS

PART 01
背景

畅捷通是做什么的?

畅捷通信息技术股份有限公司是用友旗下成员企业，成立于2010年3月，于2014年在港交所上市，是中国领先的小微企业财税及业务云服务提供商。

业务架构复杂

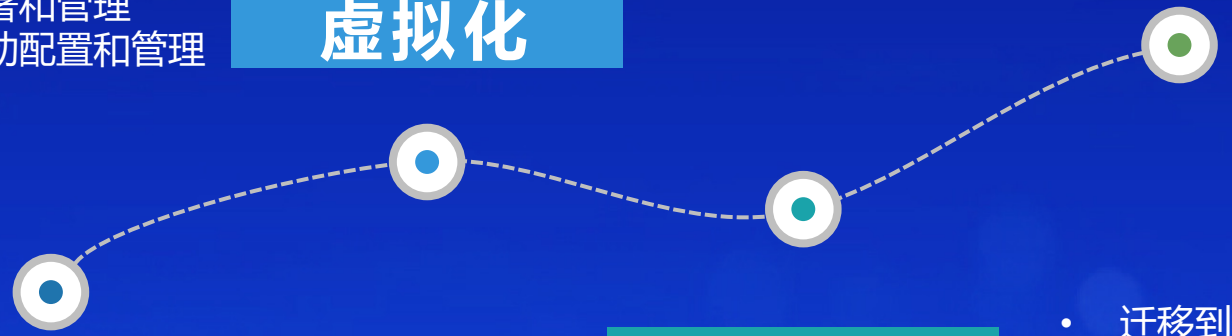
C端用户量 + B端客户体量

要保障每个用户的体验

业务迭代速度快

畅捷通运维转型之路——目标0-2-5-10

业务从自建机房逐步转向全面采用公有云容器化架构，为业务发展提供了更强大的基础，但同时也带来了运维复杂性的指数级增长。



自建机房

- 手工操作
- 资源利用率低
- 部署和扩展速度慢
- 不易实现弹性伸缩

虚拟化

- 提高资源利用率
- 简化部署和管理
- 仍需手动配置和管理

云计算

- 迁移到公有云或私有云平台
- 实现弹性、灵活性和成本效益
- 可根据需求快速调整资源

容器化

- 实现更快速的部署
- 弹性伸缩和持续交付
- 提高开发和部署效率

PART 02
问题/痛点

从一次飞机撞鸟说起

2023年11月1日，旭日8409飞机起飞离地时，发动机遭遇鸟击。

情况万分危急，关系到机上183人的生命安全。

客机起飞时突遭鸟击，机组人员28分钟安全返航获嘉奖

新京报
2023-11-03 20:46 发布于北京 新京报官方账号

+ 关注

新京报讯（记者 吴采倩）“稍等一下，旭日8409，刚才撞鸟了。”11月1日，“航班遭鸟击机组带173名旅客安全返航”一事登上微博热搜，网友表示，驾驶舱内的对话“满满安全感”。

这段对话发生在8月26日，长安航空9H8409机组人员在执飞广西梧州至海南三亚航班时，飞机起飞离地时遭遇鸟击。从鸟击发生到安全返航，机组人员用28分钟化险为夷，保障了机上173名旅客和10名机组人员的生命安全。

收集现象

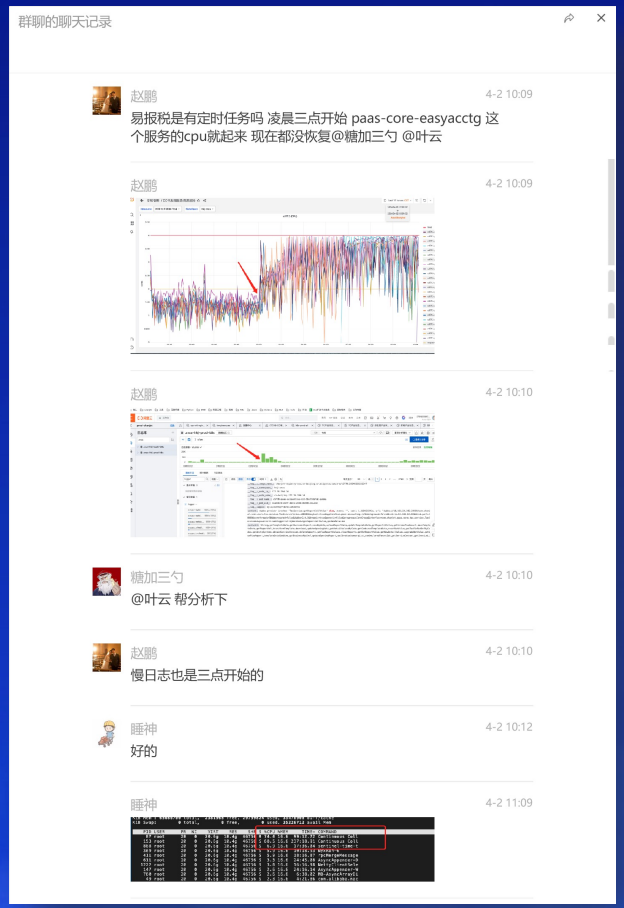
执行检查单

做出正确决策

总耗时 **7** 秒

畅捷通运维面临什么样的压力?

发生故障时难以定位



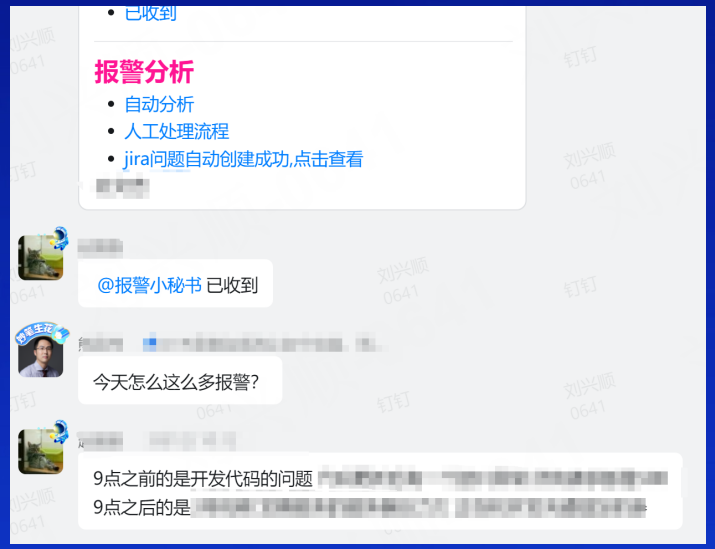
定位一个问题，需要：

- 打开3-5个看板
- 执行2-4次分析脚本

90%的问题此时就能找到原因，耗时10分钟。

但另10%的问题，才会产生大的故障，且往往难以定位原因

无法快速判断爆炸半径



- 怎么判断报警严重性?
- 报警爆炸半径多大?
- 是否正在处理?谁在处理?
- 恢复了吗?

畅捷通运维面临什么样的压力?

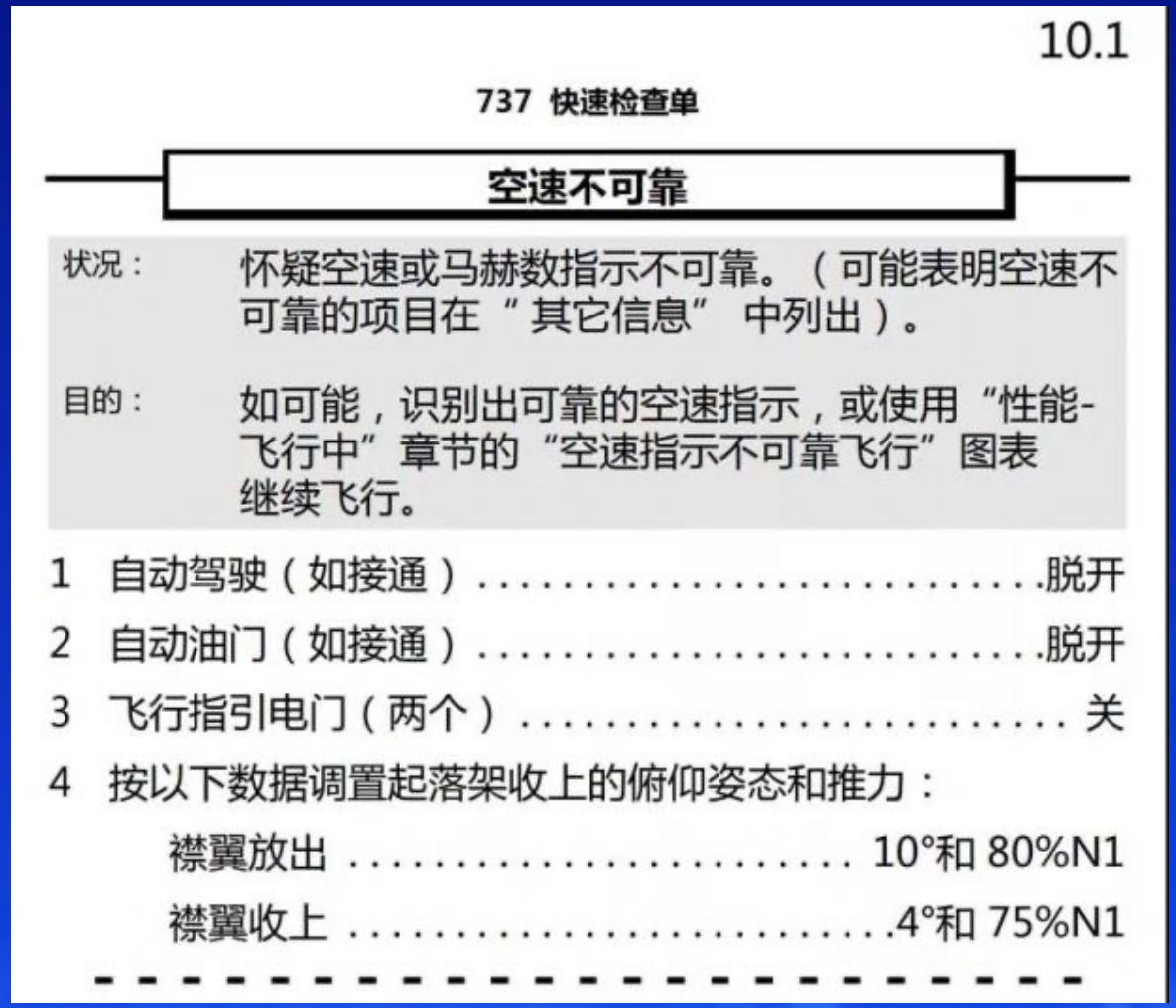
客户不能等

线上无法复现 → 压力山大

无迹可寻

PART 03
解决思路/整体方案

关键要素：检查单



10.1

737 快速检查单

空速不可靠

状况： 怀疑空速或马赫数指示不可靠。（可能表明空速不可靠的项目在“其它信息”中列出）。

目的： 如可能，识别出可靠的空速指示，或使用“性能-飞行中”章节的“空速指示不可靠飞行”图表继续飞行。

1 自动驾驶（如接通）.....脱开

2 自动油门（如接通）.....脱开

3 飞行指引电门（两个）..... 关

4 按以下数据调置起落架收上的俯仰姿态和推力：

襟翼放出 ..... 10°和 80%N1

襟翼收上 .....4°和 75%N1

1. 吸收了所有故障排查经验
2. 紧急时刻不需要思考
3. 谁都可以执行，无门槛
4. 资料集中，查阅方便

运维领域现状-传统AIOps的缺陷

- 运维团队积累的专家经验很难编码到算法模型中。通常，这些经验会被简化为阈值或复杂的规则，不仅难以维护，也难以传承。

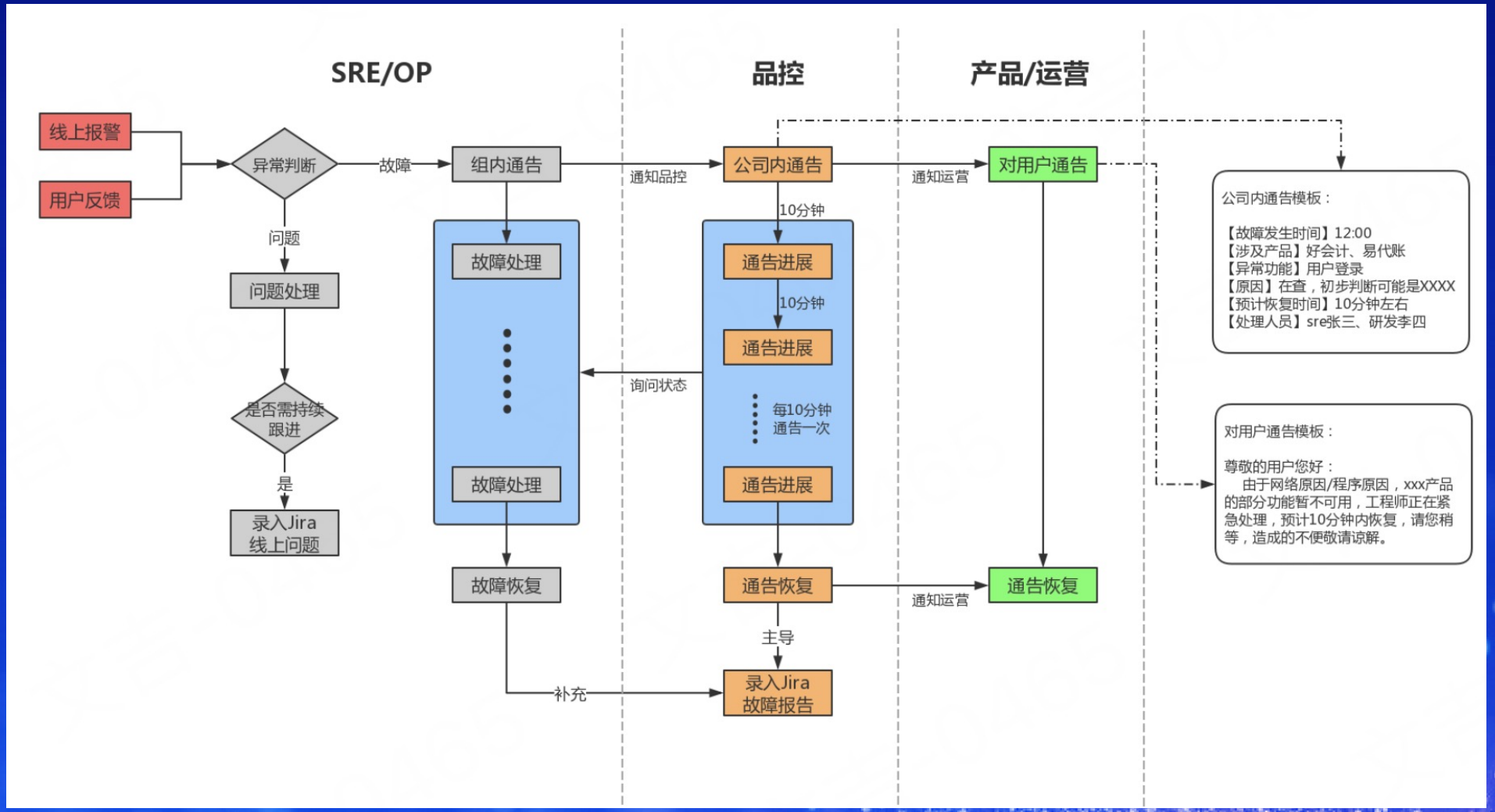
如何打造运维检查单?

- 接入和维护成本高，需要业务和算法团队深入理解业务逻辑和算法模型。
- 未遇到过的故障很难被解决，因为它们超出了模型的训练范围。
- 方案需要用户理解模型并精确地传递参数

可落地的协同处理流程

建立故障处理流程；

高效协同多个组织；



可落地的协同处理流程

建立业务高峰期预防应急机制。

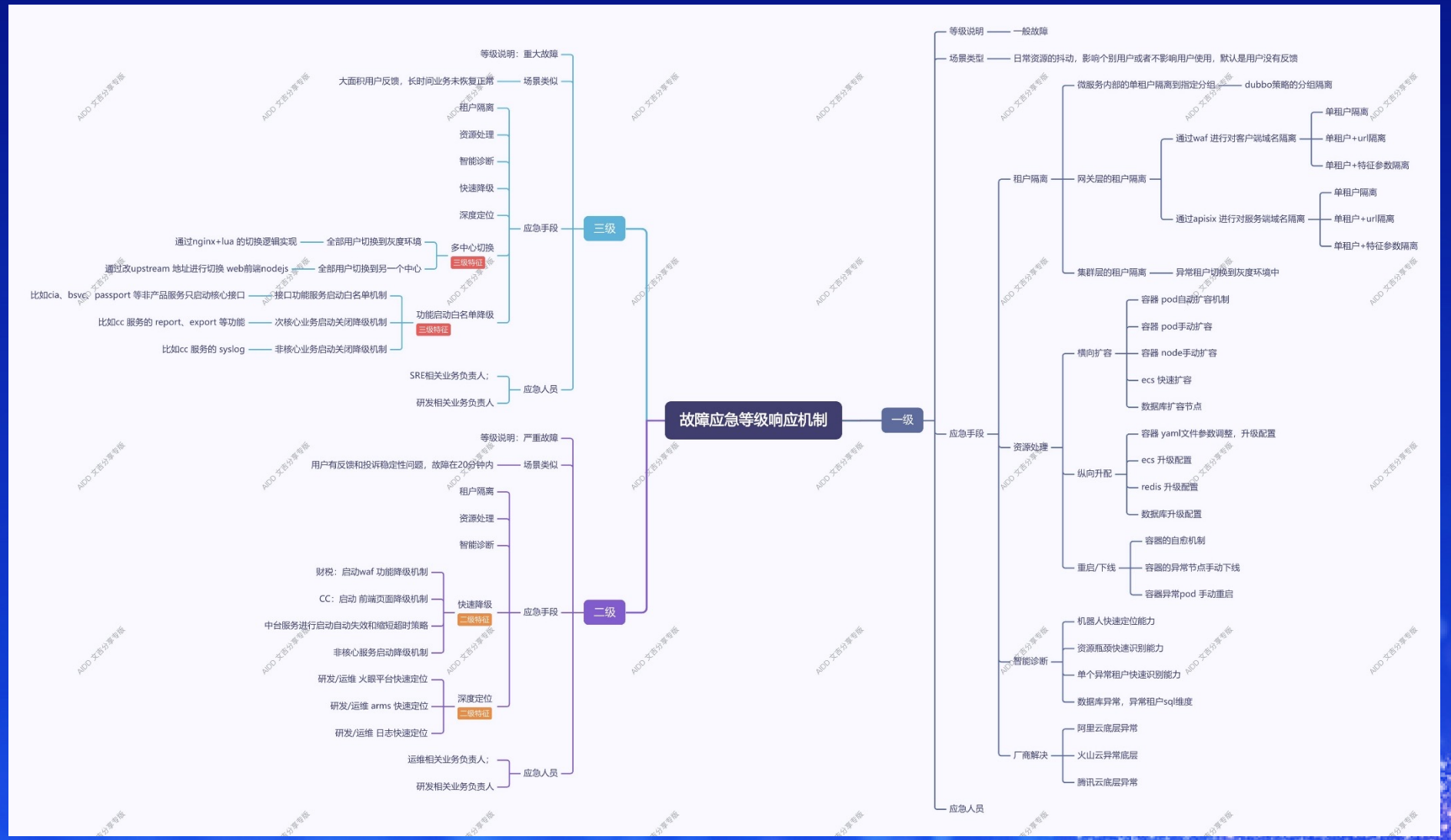
4.6 财税业务--大税期预防和应急机制

由 文吉创建, 最后修改于六月 30, 2024

- 预防篇：
  - 一级响应机制-处置方案：（高等风险处理机制）
    - 运维团队：
      - 1、遵循2-5-10原则，告警2分钟内须响应，5分钟内须定位到问题，10分钟内须对问题进行处理
      - 2、线上报警异常后，及时定位到问题，如问题不能立刻修复，须打开紧急预案，保障用户核心功能可用
      - 3、线上发生问题，及时打开快速定位面板，最短时间内定位问题
      - 4、如果是单个节点异常，应及时从集群摘除节点，防止影响扩大
      - 5、需要根据经验对问题严重性进行评估，如果达到一级响应级别，要及时通知研发，拉好问题处理专项群进行问题跟进与处理
    - 研发测试团队：
      - 1、ui自动化报警须及时跟进问题，进行解决，保证报警的准确性
      - 2、ui自动化大规模报警，须及时发出预警，拉好问题处理相关负责人的群进行专项问题跟进与处理
      - 3、研发同学在进入专项问题跟进群后，须集中精力处理线上问题，直到问题解决
    - 服务团队：
      - 1、共性问题及时反馈，及时发现和处理共性问题
      - 2、大规模共性异常问题走紧急流程，要求开发运维及时跟进处理
  - 二级响应机制-处置方案：（中等风险处理机制）
    - 运维团队：
      - 1、值班人员负责对告警面板进行巡检，不可存留告警信息，及时反馈和解决。前期是grafana面板，后期为监控中心的驾驶舱
      - 2、智能巡检提供的异常巡检及时验证。（24小时内的全部异常都要解决掉）
      - 3、容量评估预测风险：
      - 4、历史问题进行分析和提前预警：
      - 5、线上须研发跟进问题要及时创建jira给对应研发负责人进行跟进
    - 研发团队：
      - 1、ui自动化要求必须全部有效和正常运行。（24小时内的全部异常都要解决掉）
      - 2、研发团队及时分析异常日志信息。
    - 服务团队：

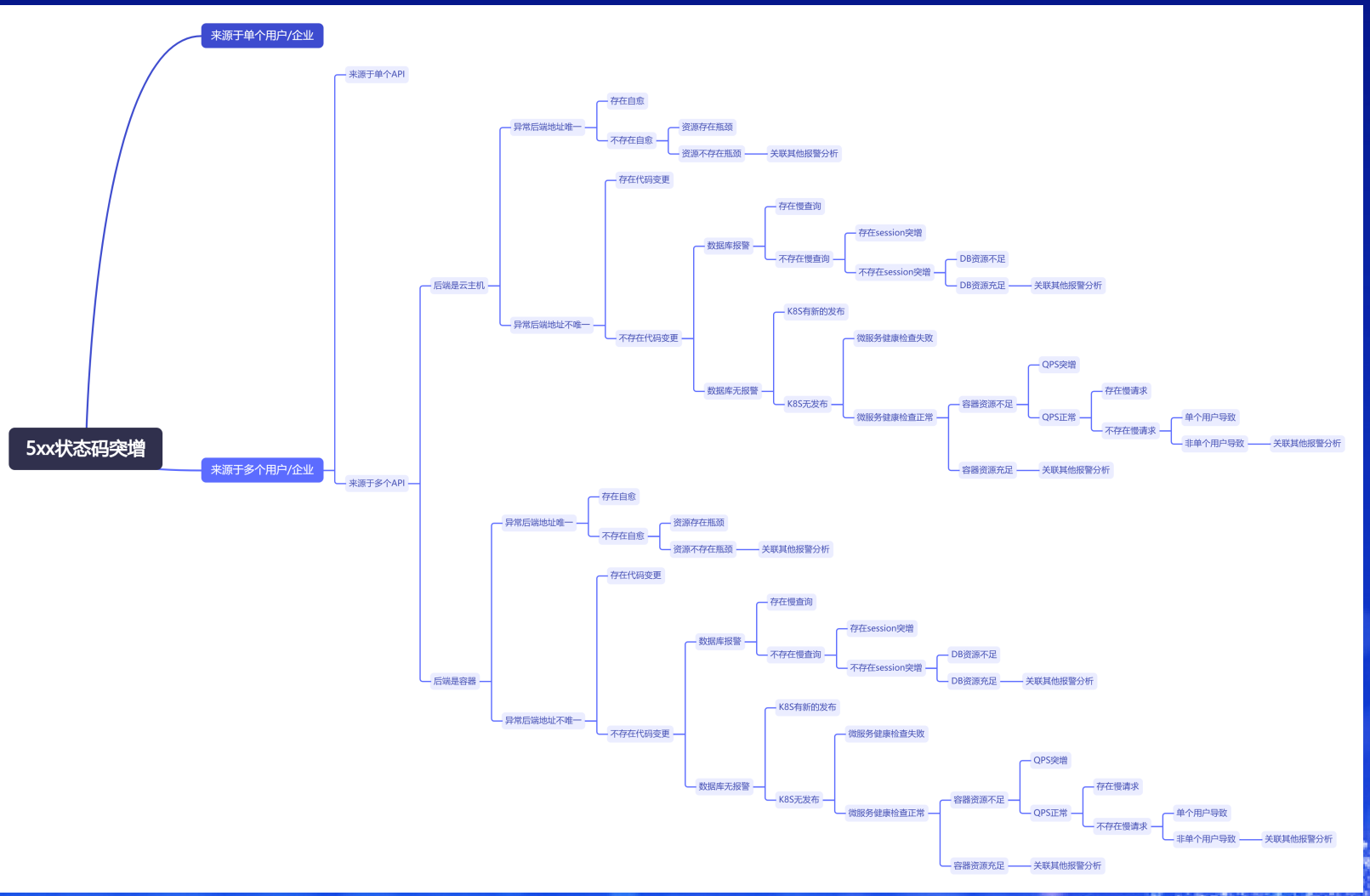
应急止损方法论——应急止损

建立应急止损操作流程和工具。



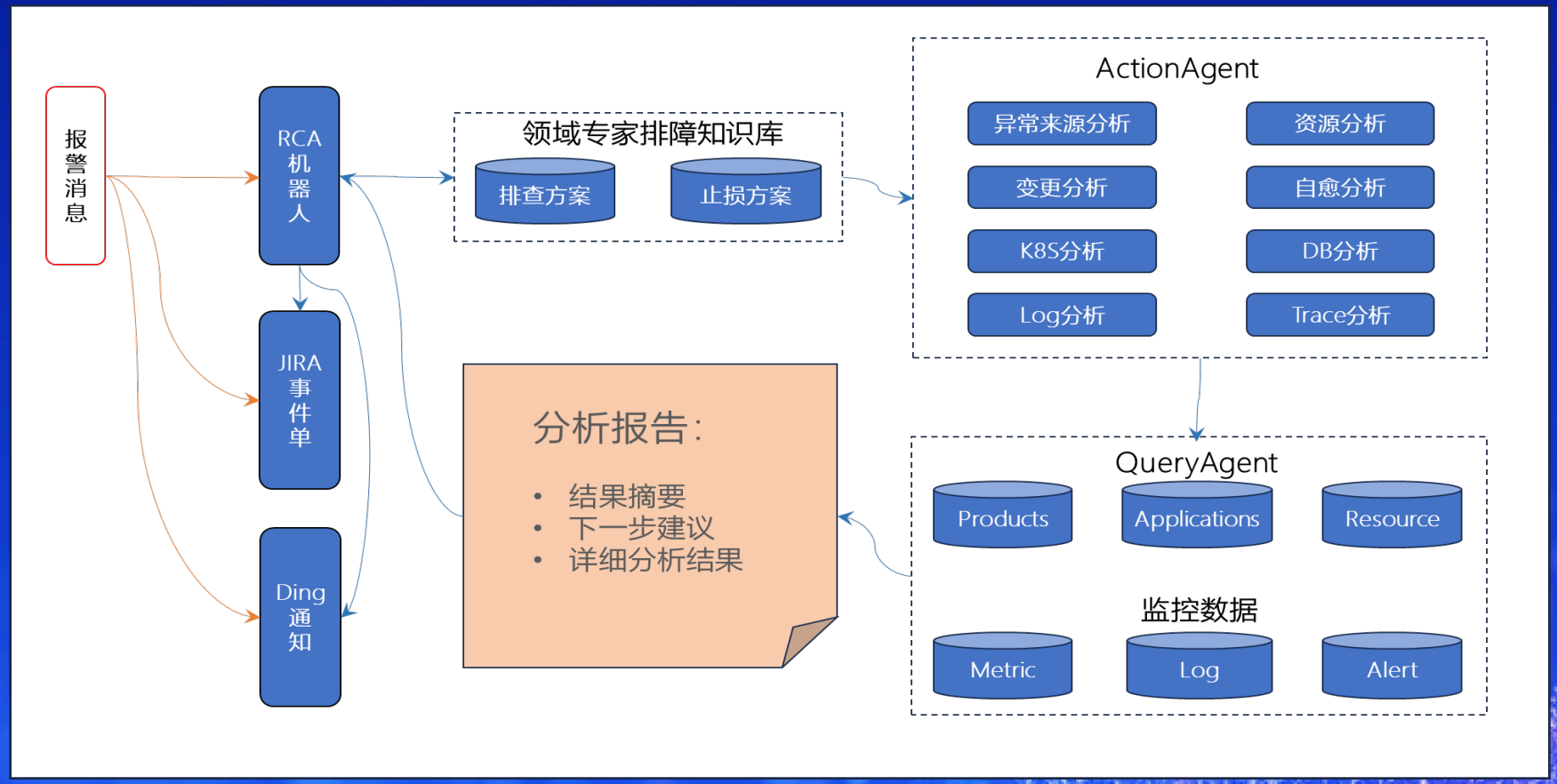
应急止损方法论——排障树

建立故障排查的专家经验排障树。



基于大语言模型的根因诊断（RCA）Agent 框架

我们定义了一些工具和插件，这些工具和插件是用于出现故障时进行检测。除了工具和插件，我们还设计了工作流编排，可以自动化的故障处理流程。此外我们构建了一个知识库，它包含了历史故障数据、专家经验和故障处理策略，这些都是进行有效根因分析的关键资源。



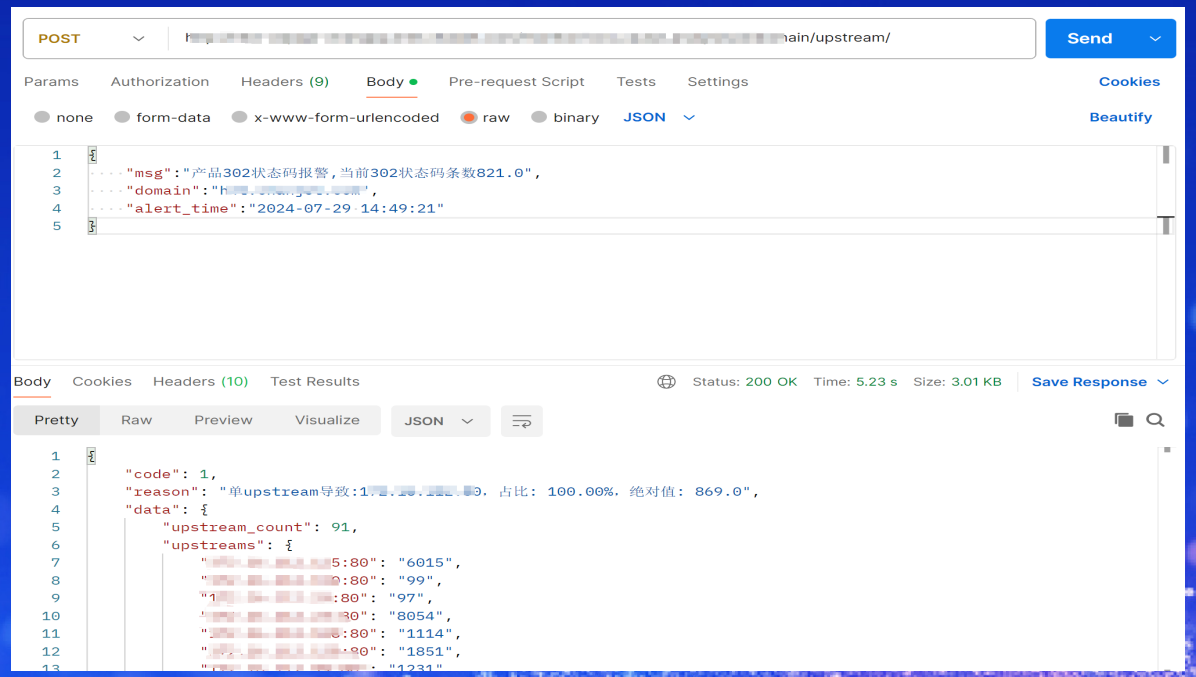
基础工具的构建

将传统的针对多模态运维数据的异常检测方法变成工具（Agent），用户仅需维护指标项即可。

比如我们定义变更查询工具，该工具可以用于确定问题是否由线上变更导致。这样的工具有很多，一般都是基于运维专家日常的排障经验，可以是一个简单的脚本，也可以是一个API，或者是一个命令，这些工具可以完成故障排查过程中某一个环节的任务。

服务器资源瓶颈分析

域名错误量upstream分布分析



工作流的构建

构建工作流，我们在prompt和文档中预先设置了不同报警的分析流程，即应该先后检查哪些数据，从而得出结论。

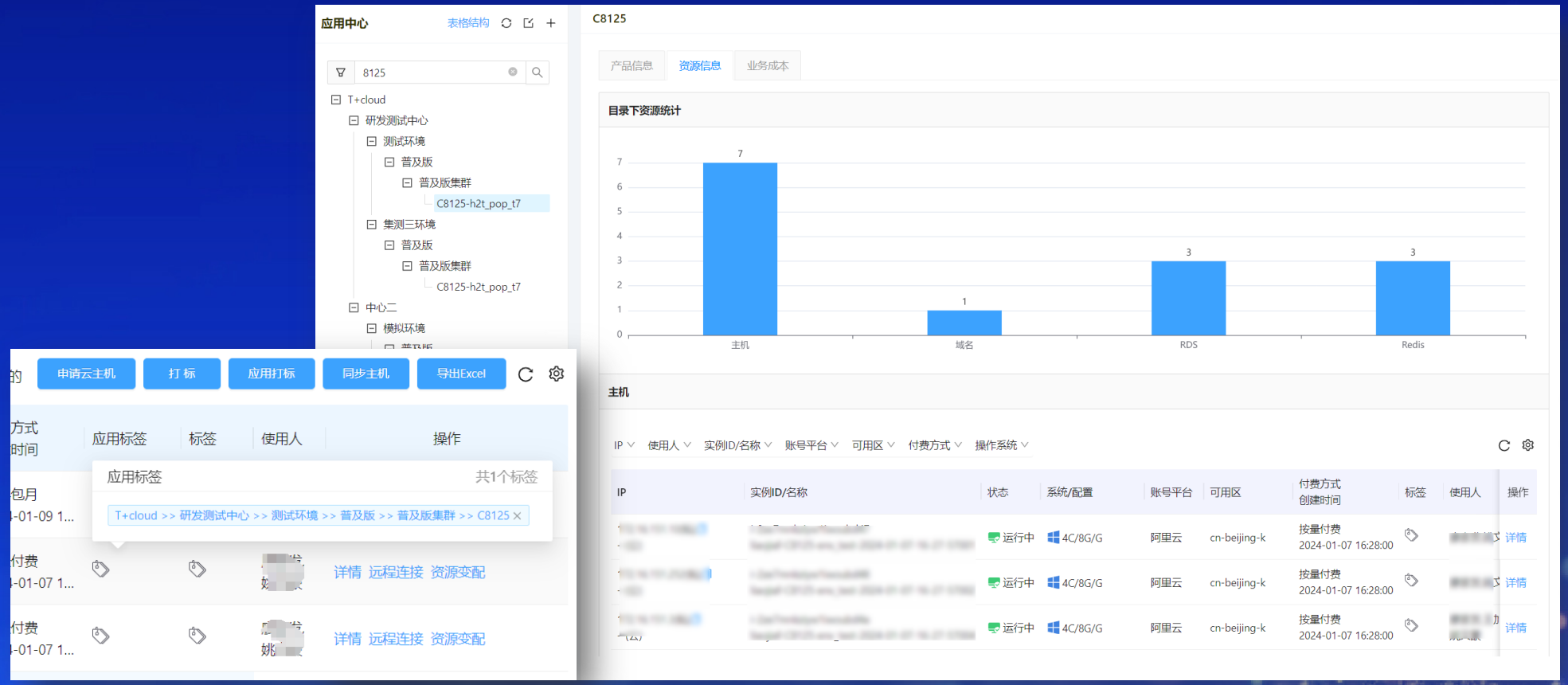
这个工作流类似飞机检查单（SOP），不同的现象对应不同的检查项，类似一个树状结构，最终一定会递归找到一个叶子节点然后返回。比如当某个域名出现5xx状态码报警，我们需要先判断这些状态码是否来源于同一个用户的请求，再判断这些请求是否都打到了同一个upstream节点，后端承载流量的微服务、容器和node是否存在问题，最后再检查是否是第三方依赖存在问题等。

这是一种妥协方案，我们可以选择对通用大语言模型进行训练，十七能够根据用户的SOP文档直接生成工作流，但是大模型训练的成本是非常高的，一方面是资源成本，另一方面是对大模型人才需求的成本。

PART 04
具体实现/技术实践

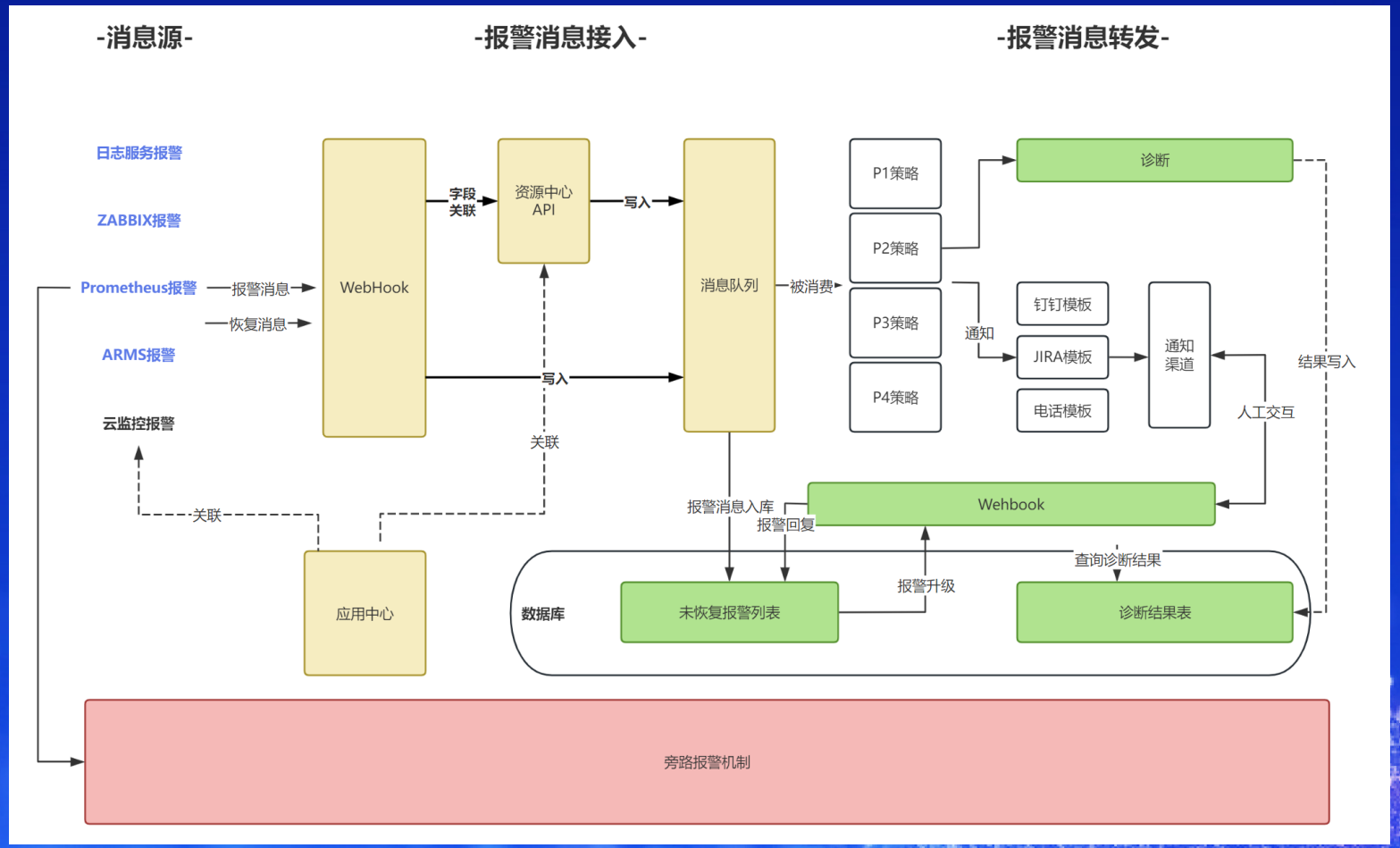
数据治理——CMDB建设

将资产标签化，将标签目录化，得到完整的产品六级目录，既有业务信息，又有资产实例的关联关系，每种资源都拥有自己的身份证号：六级目录。这是AIOps落地的基石。



数据治理——监控统一

来源于不同监控工具的报警必须满足最小字段集合，这样以来所有的报警都能标准的关联到具体的业务、产品，从而关联出所有的资源、中间件等信息。同时我们也完成了CMDB的自动化维护，形成了包含业务、基础资源、人员、代码仓库、配置等关联关系的大型数据字典，本身也为webUI提供了许多API，这些API都将作为Agent被注册。



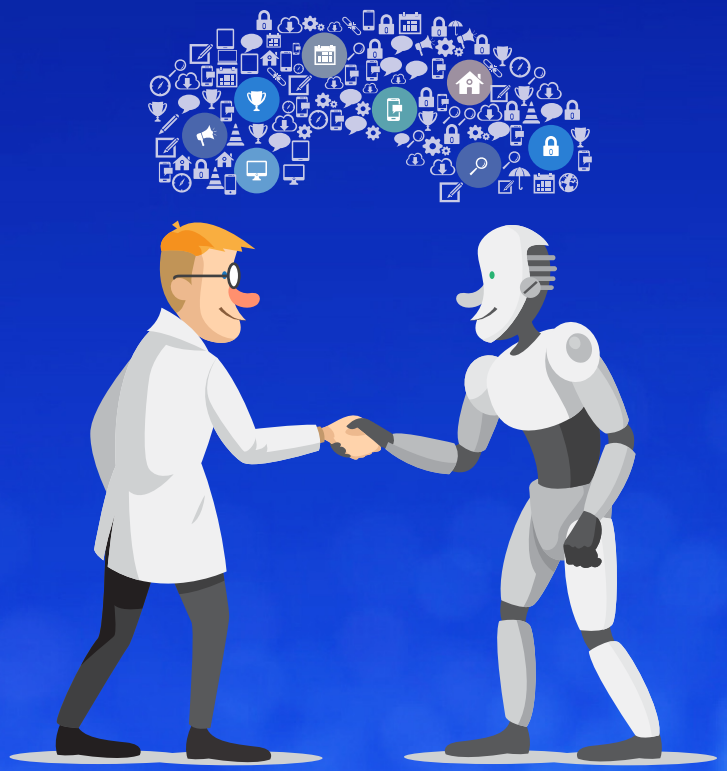
数据治理——监控统一

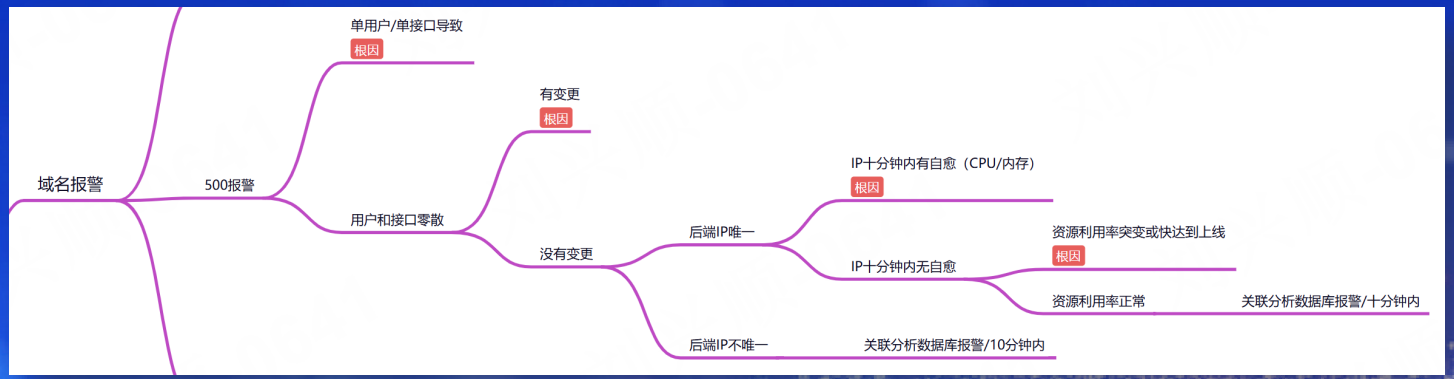

云擎平台
全局搜索菜单/资源
值班表
监控中心
驾驶舱
监控项看板
监控项配置
默认监控项
业务监控项
资源监控项
用户监控项
监控类型配置
域名接口管理
域名监控项管理
ARMS-K8S集...
驾驶舱
P1-Disaster
0个
P2-High
0个
P3-Info
171个
P4-Notclassified
218个
全量未恢复报警
P1
P2
报警来源
请输入报警内容
报警来源 报警时间 持续时间 报警级别 报警内容 资源类型 报警实例 产品线 中心 环境 cid 操作 ACK
暂无数据
历史报警流水
近8小时
报警级别
报警来源
请输入报警内容
时间 报警级别 资源类型 报警实例 报警内容 产品线 中心 环境 CID 报警状态 持续时间 报警来源 手动关闭 / ACK
2024-08-16 16:46:24 P4 ecs 172.16.43.1... win内存使用... T+cloud 中心二 线上环境 C2053-devh... firing 27 ZABBIX No / No
2024-08-16 16:43:05 P4 pod c0850-hsy-... 容器 CPU 资... 好生意-好业财 中心一 线上环境 C0850-publi... firing 226 Prometheus No / No

SOP定义——专家经验的沉淀

针对每种现象，我们都梳理了运维专家的排障脑图，将故障排查过程固化下来。

根节点表示现象（报警），分支节点代表一个分析操作，每个分支节点会再分化出是和否两个分支，直到最外围的叶子节点，无法再进行下一步分析为止。外围节点会有两种状态：根因和非根因





工具构建之查询类Agent

查询类Agent融合了CMDB（产品、应用、资源的关联关系）、IT资产清单、CICD配置、config数据的查询。
查询类Agent的还包含了历史故障单的查询，让AI具备寻找历史相似事件的能力。

Schema 查看 OpenAPI-Swagger 规范 ＋ 从 URL 中导入 例子

```
"servers": [
  {
    "url": "https://[redacted].chaniet.com"
  }
],
"paths": {
  "/[redacted]_host_info/": {
    "get": {
      "summary": "Get machine host info",
      "parameters": [
        {
          "name": "pageSize",
          "in": "query",
          "required": true,
          "schema": {
```

可用工具

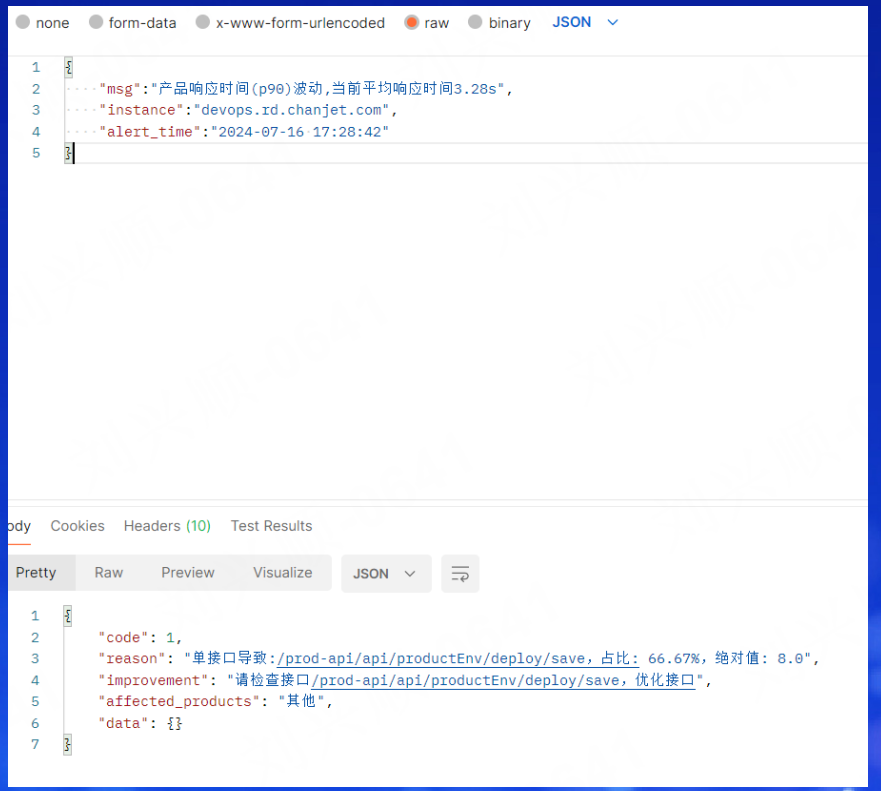
名称	描述	方法	路径	操作
assetecsmachine_host_info_get	Get machine host info	get	[redacted]host_info/	测试
tagV2show_ctags_get	获取服务器的应用标签	get	[redacted]tags/	测试

工具构建之动作类Agent

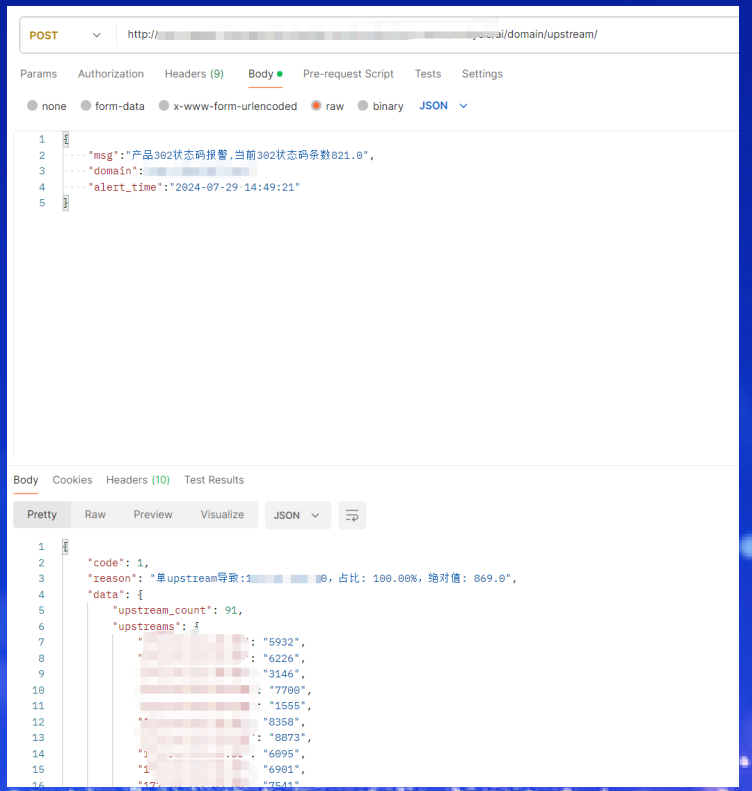
动作类的Agent就是前文提到的，对于排障脑图中某个具体节点的对象的分析过程，我们可以非常原子化的进行这些Agent的定义，比如下面是我们定义的一些Agent

服务器资源瓶颈检查

异常访问来源检查



异常访问upstream检查



流程编排

GitLab Projects Groups More Search or jump to...

A airca-prompt

- Project overview
- Repository
  - Files
  - Commits
  - Branches
  - Tags
  - Contributors
  - Graph
  - Compare
- Issues 0
- Merge Requests 0
- CI / CD
- Operations
- Analytics
- Wiki
- Collapse sidebar

[redacted] committed 1 day ago

prompt 4.14 KB Edit Web IDE

```
你是一个根因分析专家，专注于解析报警信息以确定问题根本原因。报警信息将以JSON格式提供，需仔细审查以下核
 - `instance_type`：指示报警涉及的实例类型，涵盖domain、ecs、log、数据库、K8S组件等。
 - `instance`：具体的报警实例详情。
 - `product`、`center`、`env`、`cid`：这些字段共同标识特定的业务环境。
 - `pd`：表示特殊产品线中的子产品名称；若pd为空，则无需将其纳入API查询参数

### 针对提供的instance_type进行系统化分析流程，遵循以下步骤以精准定位问题根因，并按规范格式返回结果
1. **代码变更优先检查**：  - 首先检查过去30分钟内是否有代码变更。若有变更，收集变更内容及变更人作为根
2. **instance_type特定分析**：
- **domain实例**：
 - 查看异常来源分布，异常来源分布结果会有以下四类：
    - 如果是单IP导致，直接返回作为根因。
    - 或者是单IP+URL导致，直接返回作为根因。
    - 如果是单接口导致，请勿将此作为根因返回，而是继续下面的步骤。
    - 或者其他情况，请勿将此作为根因返回，而是继续下面的步骤。
 - 若异常来源分布未明确根因，分析后端异常分布；若非单一upstream，则总结当前分析并通知未找到根因。
 - 继续对单一upstream执行服务器资源分析，发现自愈行为或资源瓶颈即为根因；否则，告知未找到根因。
- **ecs实例**：
 - 直接进行服务器资源分析，发现自愈或资源瓶颈作为根因；否则，告知未找到根因。
- **其他instance_type**：
 - 根据实际情况推理最合适的分析路径。
3. **终极分析步骤**：若上述所有步骤未能确定根因，无论instance_type是什么，都进行关联报警分析。分析
- 关联报警为空，此时告知用户该业务线10分钟内无其他报警。
- 关联报警不为空,此时告知用户10分钟内有多少条报警，并列举每条报警的关键信息（什么资源，发生了什么事）。

### 分析任务要求 你的响应必须结构严谨，包含以下组成部分：
1. **result**（布尔值）：表明是否成功识别出根因。`True` 表示已发现，`False` 表示未发现。
2. **root_cause**：明确指出根因（如找到）。
3. **analysis_abstract**：中文总结分析过程，包括检查的要素及得出的结论，不要使用换行符等特殊字符。
4. **improvement**：提出针对发现的问题的具体改进建议（中文表述）。
5. **markdown_result**：整合上述所有信息为Markdown格式的字符串，遵循特定风格指南，确保颜色编码正
#### Markdown格式示例```## <font color=#0096fe >AI大模型智能诊断结果 </font> \n ### <font
```

编辑动作

名称

智能诊断action

Schema 查看 OpenAPI-Swagger 规范 + 从 URL 中导入 例子

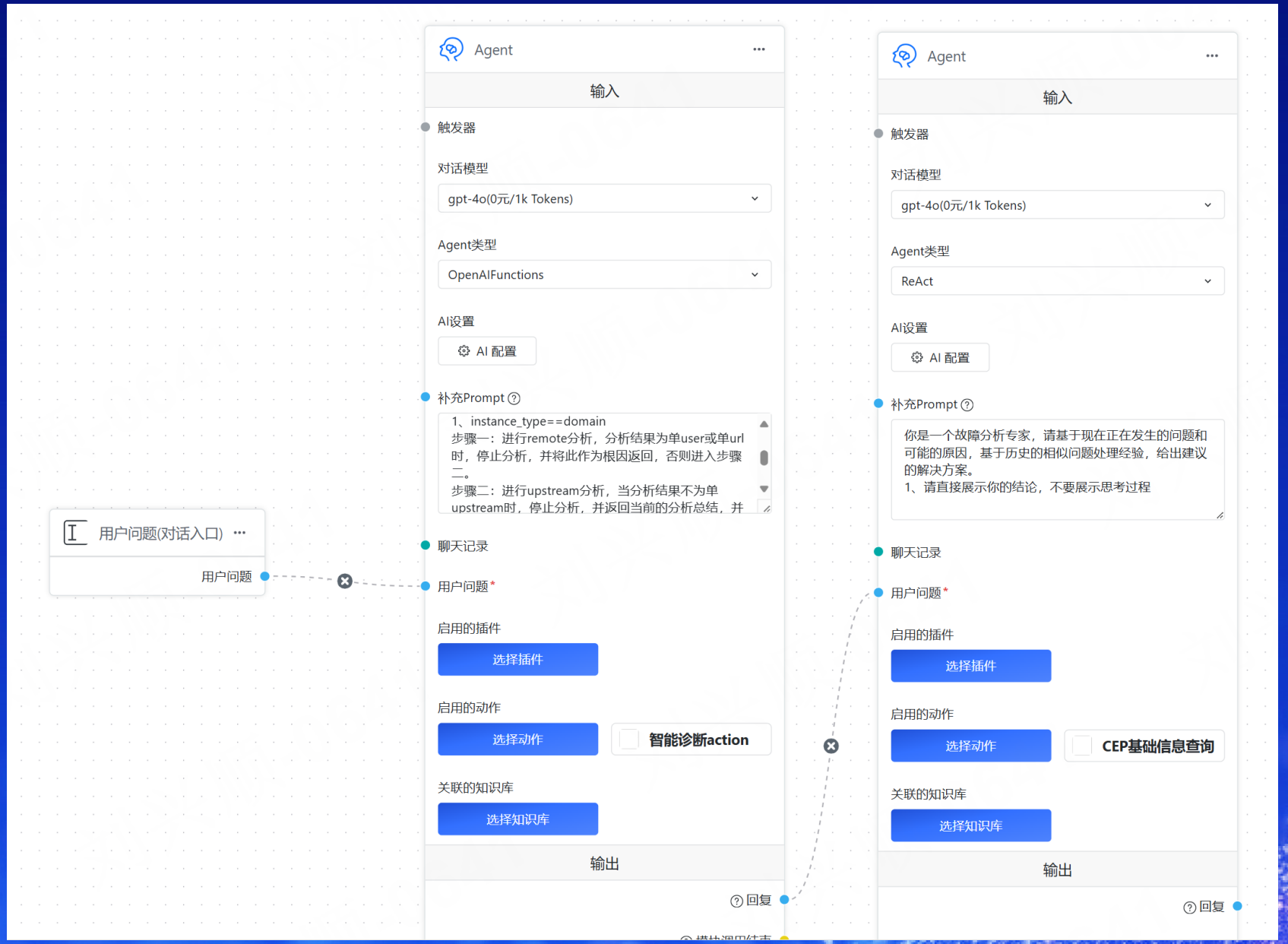
```
openapi: 3.1.0
info:
  title: Root_Cause_Analysis_API
  description: API for monitoring and analyzing root causes of alerts
  version: 1.0.0
servers:
  - url: https://[redacted]jet.com
    description: Production server
paths:
  /monitor/root_ca[redacted]:
    post:
      operationId: remote_root_cause_analysis
      summary: Analyze the root cause of alerts pertaining to either a single user or domain,
exclusively when the instance_type is designated as domain
```

可用工具

名称	描述	方法	路径
remote_root_cause_analysis	Analyze the root cause of alerts pertaining to either a single user or domain,	post	/monitor/root_cause_analysis/ai/domain

移除 取消 保存

流程编排


用户问题(对话入口)
用户问题
Agent
输入
触发器
对话模型
gpt-4o(0元/1k Tokens)
Agent类型
OpenAIFunctions
AI设置
AI 配置
补充Prompt
1、instance_type==domain
步骤一：进行remote分析，分析结果为单user或单url时，停止分析，并将此作为根因返回，否则进入步骤二。
步骤二：进行upstream分析，当分析结果不为单upstream时，停止分析，并返回当前的分析总结，并
聊天记录
用户问题*
启用的插件
选择插件
启用的动作
选择动作
智能诊断action
关联的知识库
选择知识库
输出
回复
Agent
输入
触发器
对话模型
gpt-4o(0元/1k Tokens)
Agent类型
ReAct
AI设置
AI 配置
补充Prompt
你是一个故障分析专家，请基于现在正在发生的问题和可能的原因，基于历史的相似问题处理经验，给出建议的解决方案。
1、请直接展示你的结论，不要展示思考过程
聊天记录
用户问题*
启用的插件
选择插件
启用的动作
选择动作
CEP基础信息查询
关联的知识库
选择知识库
输出
回复

效果升级

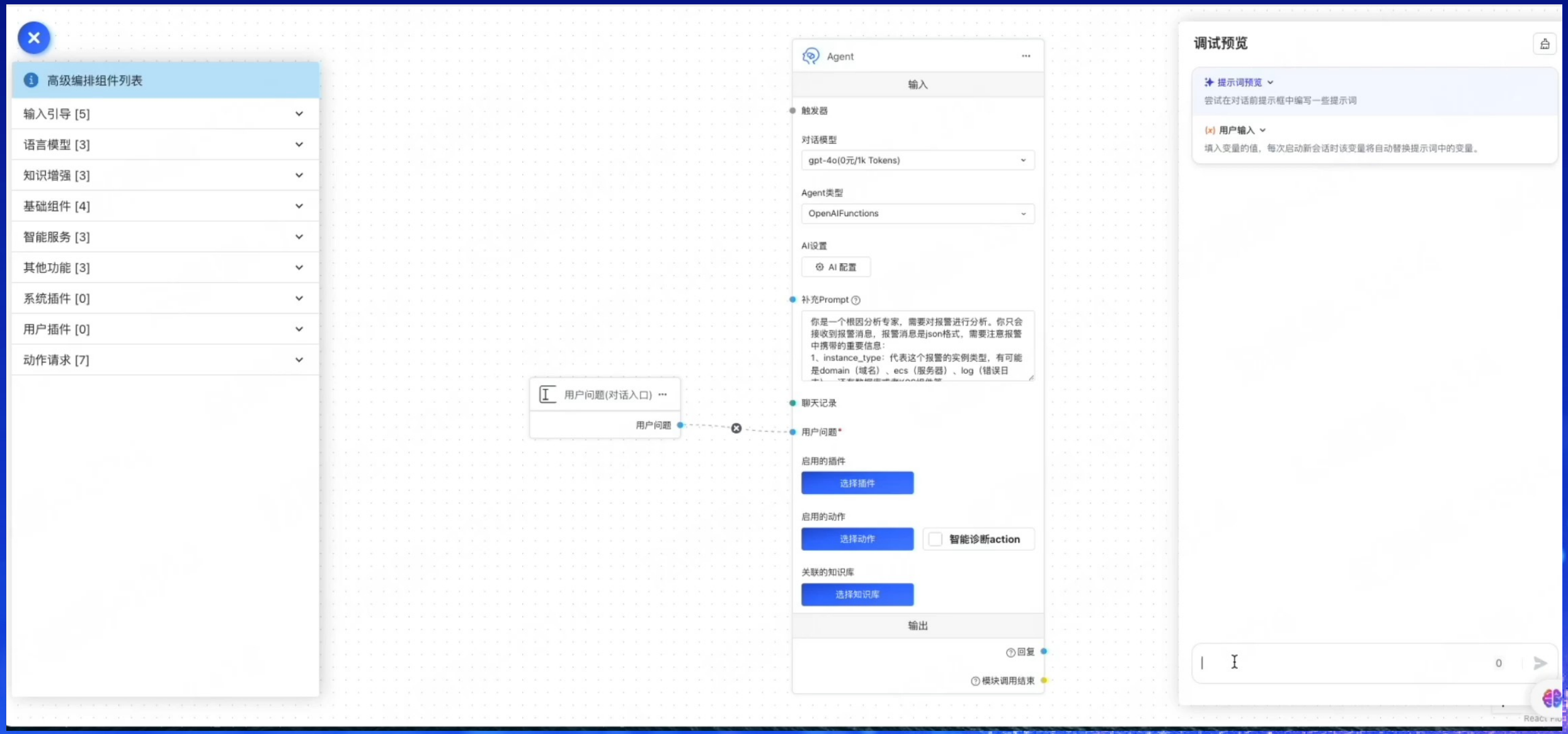
降低编码的复杂性和成本输出

实际运用

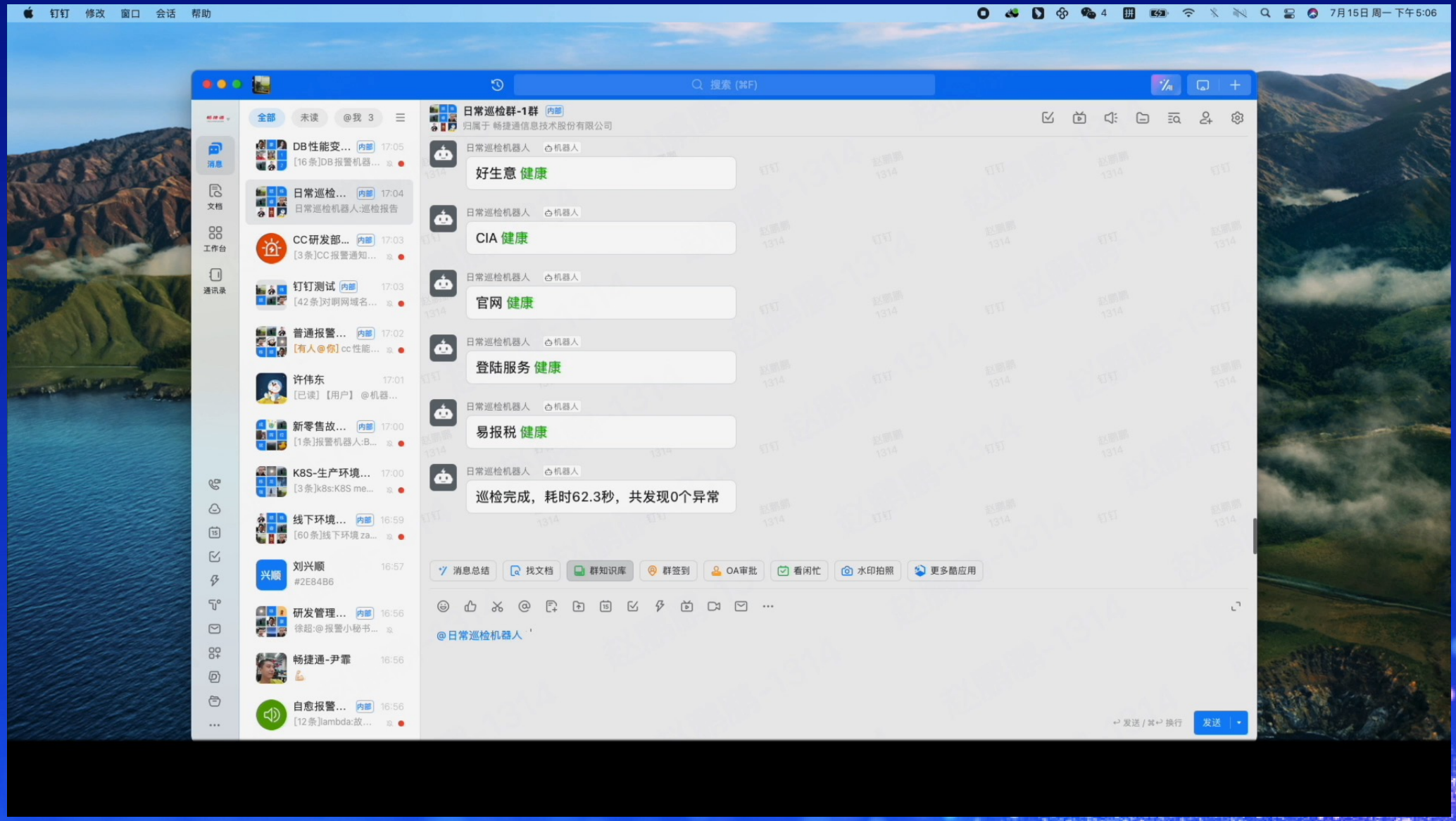
我们目前已经实现了所有线上报警的自动分析，目前根因的召回率已经超过了50%，随着Agent和流程编排的完善，召回率还会逐渐提升。

对于成功召回根因的报警，机器人会自动关闭报警工单，同时支持钉群交互，形成闭环。

RCA效果展示


高级编排组件列表
输入引导 [5]
语言模型 [3]
知识增强 [3]
基础组件 [4]
智能服务 [3]
其他功能 [3]
系统插件 [0]
用户插件 [0]
动作请求 [7]
Agent
输入
触发器
对话模型
gpt-4o(0元/1k Tokens)
Agent类型
OpenAIFunctions
AI设置
AI 配置
补充Prompt
你是一个根因分析专家，需要对报警进行分析。你只会接收到报警消息，报警消息是json格式，需要注意报警中携带的重要信息：
1、instance_type：代表这个报警的实例类型，有可能是domain（域名）、ecs（服务器）、log（错误日
用户问题(对话入口)
用户问题
聊天记录
用户问题*
启用的插件
选择插件
启用的动作
选择动作
智能诊断action
关联的知识库
选择知识库
输出
回复
模块调用结束
调试预览
提示词预览
尝试在对话前提示框中编写一些提示词
用户输入
填入变量的值，每次启动新会话时该变量将自动替换提示词中的变量。
0

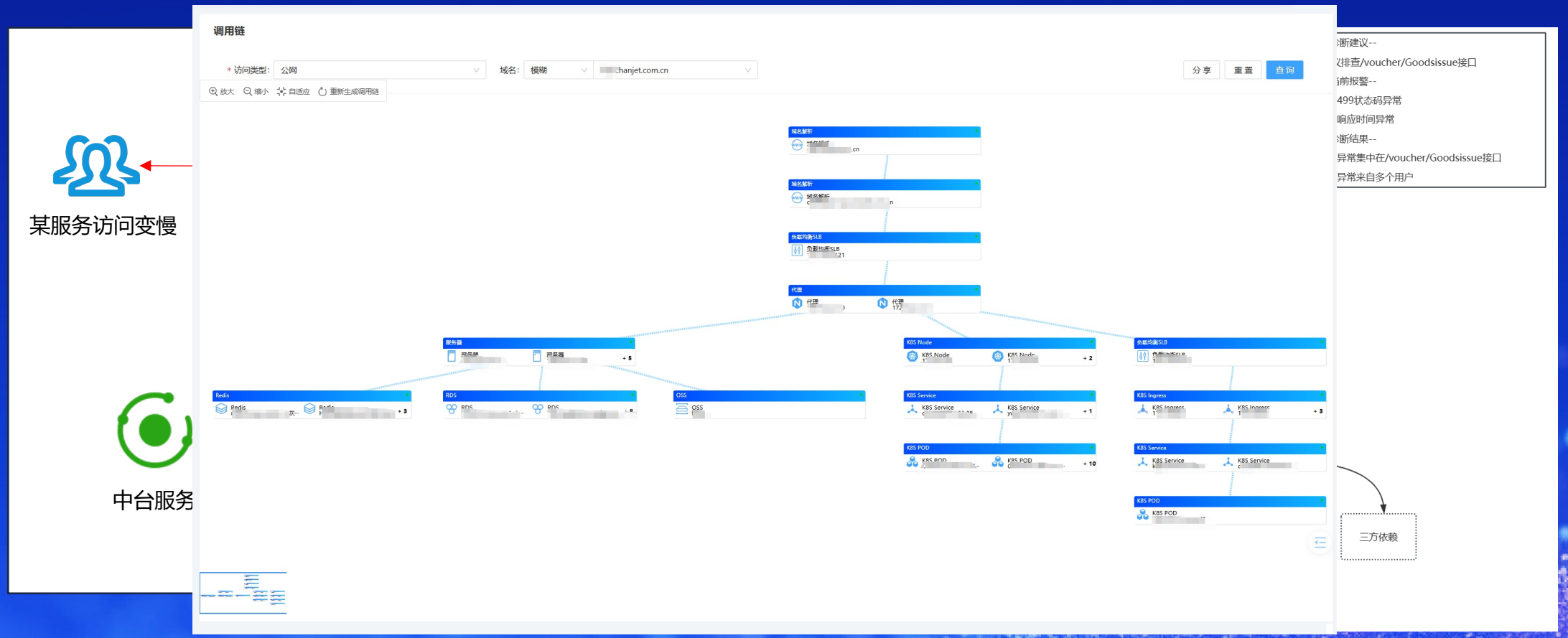
我们更进一步的尝试


钉钉 修改 窗口 会话 帮助
7月15日 周一 下午5:06
搜索 (⌘F)
全部 未读 @我 3
日常巡检群-1群 内部
归属于 畅捷通信息技术股份有限公司
日常巡检机器人 机器人
好生意 健康
CIA 健康
官网 健康
登陆服务 健康
易报税 健康
巡检完成，耗时62.3秒，共发现0个异常
消息总结 找文档 群知识库 群签到 OA审批 看闲忙 水印拍照 更多酷应用
@日常巡检机器人
发送

PART 05
总结与展望

方案总结——望、闻、问、切

本方案通过构建根因排查逻辑树、建立统一的报警字段集规范，建立多模态Agent集合，充分调度AI大模型文本推理的能力，对报警通知、报警事件单和根因分析过程进行了整合，实现了报警的自动化分析，整体耗时在1分钟以内，对于90%常见的报警都能分析出根因所在，即便是10%的不常见报警，也能完成分析过程，运维人员无需重复分析，为应急止损和故障定位争取了更多时间，保证了业务稳定性。



大模型时代，做AI的主人

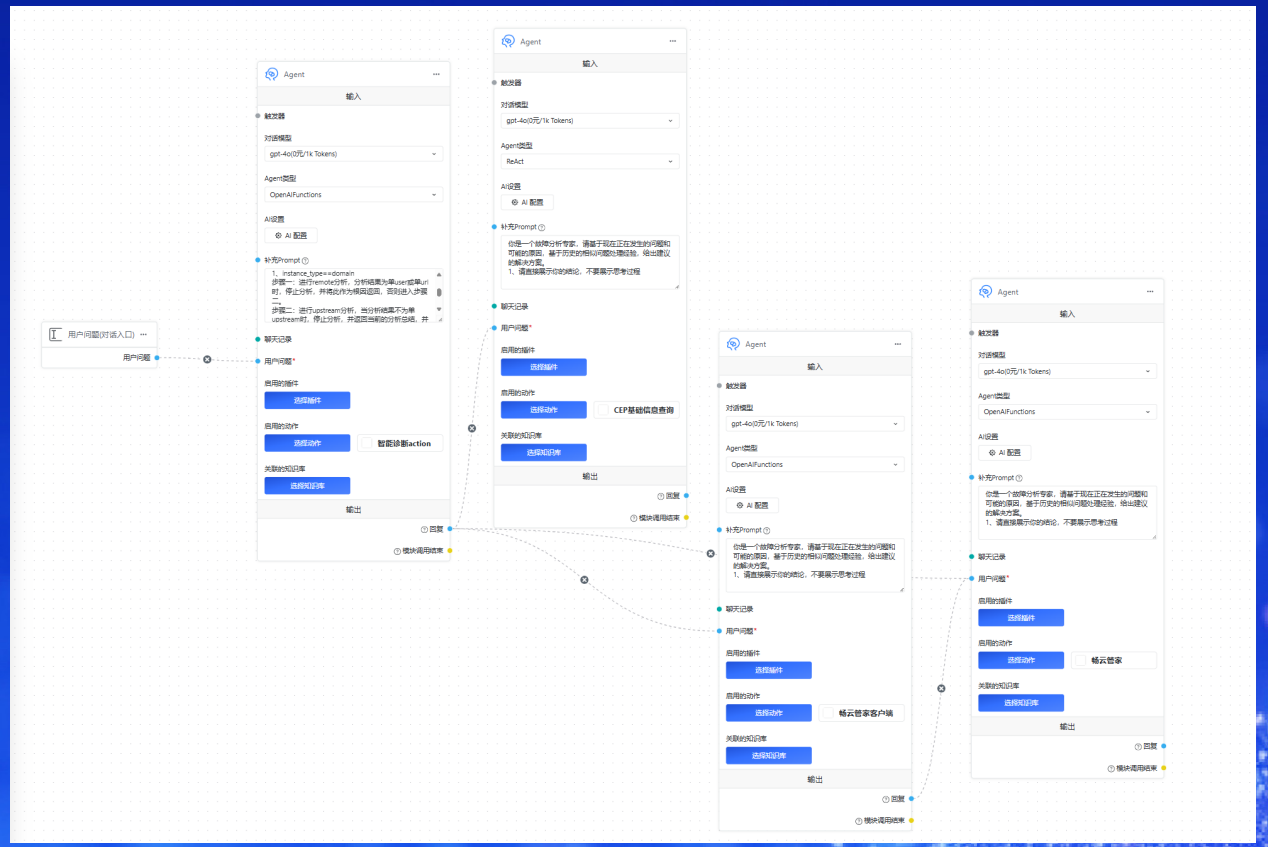
大模型技术诞生之后，已经颠覆了IT从业者的工作和思维习惯，大家的技术水平差距已经被大模型抹平了，而善于思考，能把问题想明白变这个事情，变得更加重要了。

其实用大模型技术完成推理+检索实现RCA应用的过程，其实就是在prompt或者知识库中定义了各种if else的逻辑，理论上只要能说得清的逻辑，就可以通过传统编码的方式实现，我们为什么要承担AI大模型偶尔“一本正经胡说八道”的风险?

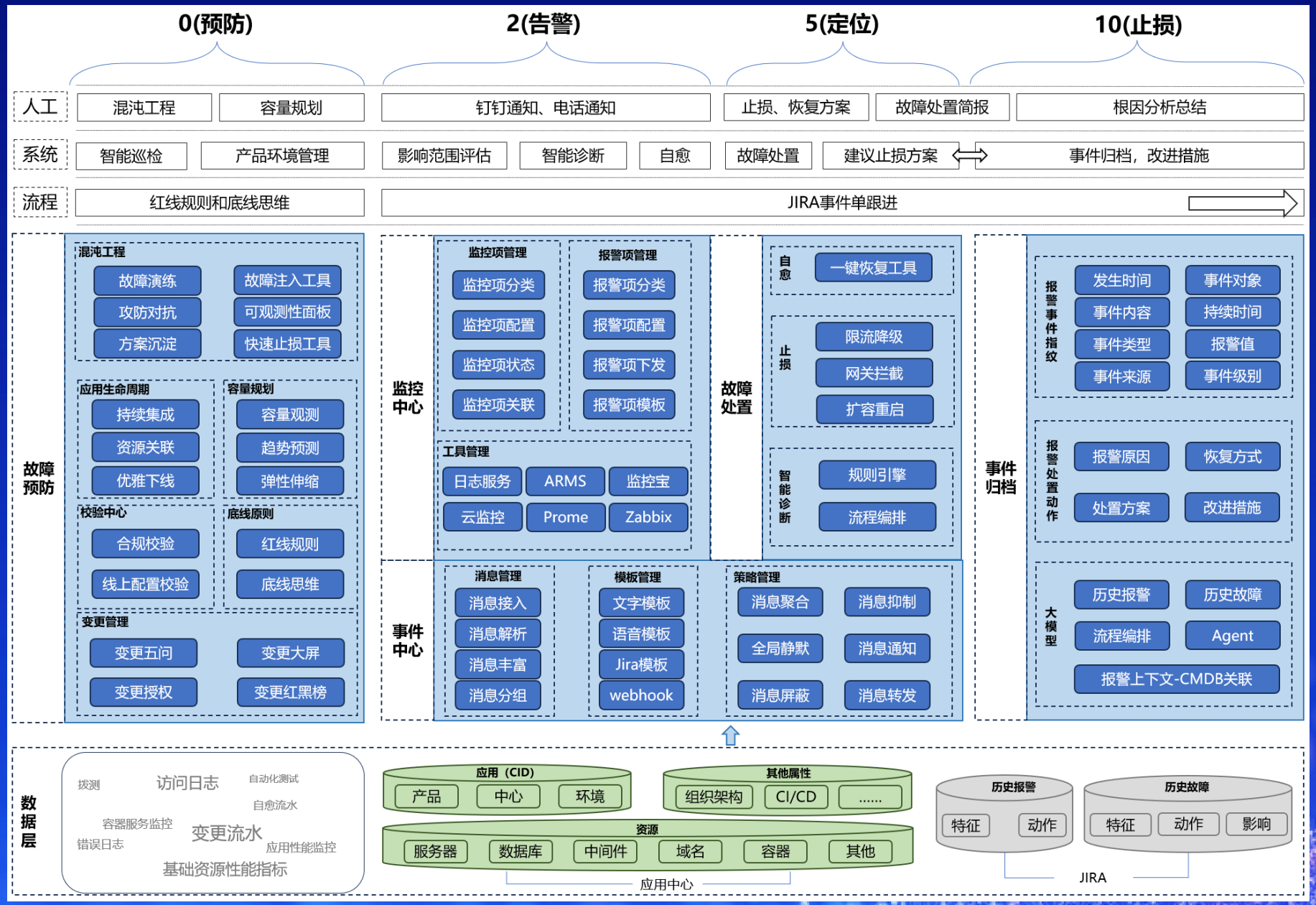
把问题想明白说清楚 > 专业技术强大

我们接下来会做的事情

- 更多工作流：让AI串联更多工作流程，比如监控、巡检、故障止损、智能容量预防、智能风险识别等
- 工作流插件化：让这些工作流变成插件，从而可以在大模型应用中进行调用
- 大总管的模式：面向对话框工作，所有的交互不再需要设计webUI，也不再需要设计问题，简化开发的过程，充分释放AI的能力



我们接下来会做的事情


0(预防)
2(告警)
5(定位)
10(止损)
人工
混沌工程
容量规划
钉钉通知、电话通知
止损、恢复方案
故障处置简报
根因分析总结
系统
智能巡检
产品环境管理
影响范围评估
智能诊断
自愈
故障处置
建议止损方案
事件归档，改进措施
流程
红线规则和底线思维
JIRA事件单跟进
故障预防
混沌工程
故障演练
故障注入工具
攻防对抗
可观测性面板
方案沉淀
快速止损工具
应用生命周期
持续集成
资源关联
优雅下线
容量规划
容量观测
趋势预测
弹性伸缩
校验中心
合规校验
线上配置校验
底线原则
红线规则
底线思维
变更管理
变更五问
变更大屏
变更授权
变更红黑榜
监控中心
监控项管理
监控项分类
监控项配置
监控项状态
监控项关联
报警项管理
报警项分类
报警项配置
报警项下发
报警项模板
工具管理
日志服务
ARMS
监控宝
云监控
Prome
Zabbix
事件中心
消息管理
消息接入
消息解析
消息丰富
消息分组
模板管理
文字模板
语音模板
Jira模板
webhook
策略管理
消息聚合
消息抑制
全局静默
消息通知
消息屏蔽
消息转发
故障处置
自愈
一键恢复工具
止损
限流降级
网关拦截
扩容重启
智能诊断
规则引擎
流程编排
事件归档
报警事件指纹
发生时间
事件对象
事件内容
持续时间
事件类型
报警值
事件来源
事件级别
报警处置动作
报警原因
恢复方式
处置方案
改进措施
大模型
历史报警
历史故障
流程编排
Agent
报警上下文-CMDB关联
数据层
拨测
访问日志
自动化测试
自愈流水
容器服务监控
变更流水
错误日志
应用性能监控
基础资源性能指标
应用（CID）
产品
中心
环境
其他属性
组织架构
CI/CD
.....
资源
服务器
数据库
中间件
域名
容器
其他
应用中心
历史报警
特征
动作
历史故障
特征
动作
影响
JIRA

未来的发展趋势

关键词：全文检索、逻辑推理、低代码

目标：

1、基于大模型的，减少知识获取难度

2、利用大模型擅于汇总、总结的能力

query

XXX当前有少量500状态码，都来自同一个用户的请求，结合历史判断目前正常……

THANKS