

借助大模型实现工程师自驱式质量运营管理 助力业务线实现质效全面提升

董俊秋 众安 质量管理团队负责人

讲师简介



董俊秋

众安 质量管理团队负责人

CMMI 软件成熟度评估师、Scrum Master、EPG 过程改进专家、云从业者
曾任腾讯高级质量管理工程师、中兴智慧城市质量经理

涉及行业：ToB、ToC、云产品、金融、医疗、教育、通讯、人工智能

擅长领域：度量分析、质量建模、过程改进、研发效能度量






亮点介绍

【内容亮点】

- 全面解读0-1的质效搭建历程，为企业提供**质效度量探索**及北极星指标的**建设思路**
- 通过落地案例的实操呈现为企业提供**可借鉴和可复用的质效提升相关经验**
- 首次揭秘智能质效管理平台的**搭建全过程**，包含亮点功能：智能分诊、智能诊断等

【业务价值】

-  **合作方满意度**：树立QA团队口碑、打造影响力
-  **成本控制**：为企业大大节省了质量运营成本
-  **产品品牌形象**：通过质效全过程的心跳监控，帮助提前识别质效风险，确保产品交付质量，提升客户整体满意度

目录

- ➔ **问题：** 常见质量运营问题及质量管理痛点
- ➔ **解法：** 建体系 → 做度量 → 数据洞察 → 质效诊断
- ➔ **实践：** 质效度量体系建设、质量落地运营、智能质效管理平台
- ➔ **成果：** 短期收益、长期收益
- ➔ **未来：** 自驱式质量运营管理

问题



软件开发流程的数字化程度越来越高

面临devops、提效、数字化转型、变革时期

如何判断转型的好与坏？

主观判断？缺少抓手？缺少数据支撑？

指标太多了，搭建成本高且没有发挥价值

过度关注数据本身导致适得其反

指标并没有带来改进的牵引作用

数据的可信度不高

解法

调研现状

【业务线A】

- 指标**太多太泛**，大几十个，**看不到重点**，不聚焦
- 指标**不是我们想要的**，所以我们自己搭建了数据采集平台

【业务线B】

- 数据**没有分析结论**，看完不知道要做什么
- 很多结果指标里**未完结的阶段性数据也统计进去**

【QA组】

- **数据遭到挑战**，业务线多次在质量会议上质疑数据的准确性

根因分析

【指标设计】

- 指标没有指向性

【数据准确性】

- 采集路径口径不一致
- 业务线不认可采集方式/路径

【指标运用、数据分析】

- 不知道指标怎么运用
- 纯数据晾晒，没有质量建议、下钻分析等

实施改进

【质效指标建设】

- 优先梳理并对齐**北极星指标**

【采集路径对齐】

- 梳理度量采集表，所有指标的**详细采集路径一定要充分对齐**
- 选择**试点业务线**，试运行成功后再大范围推广

【质效度量诊断】

- 启动下钻分析及质效诊断(共性问题下钻分析及结论)
- 质量课程赋能：指标的价值意义解读、指标使用，如何通过指标改进提升

建体系 做度量
数据洞察 质效诊断



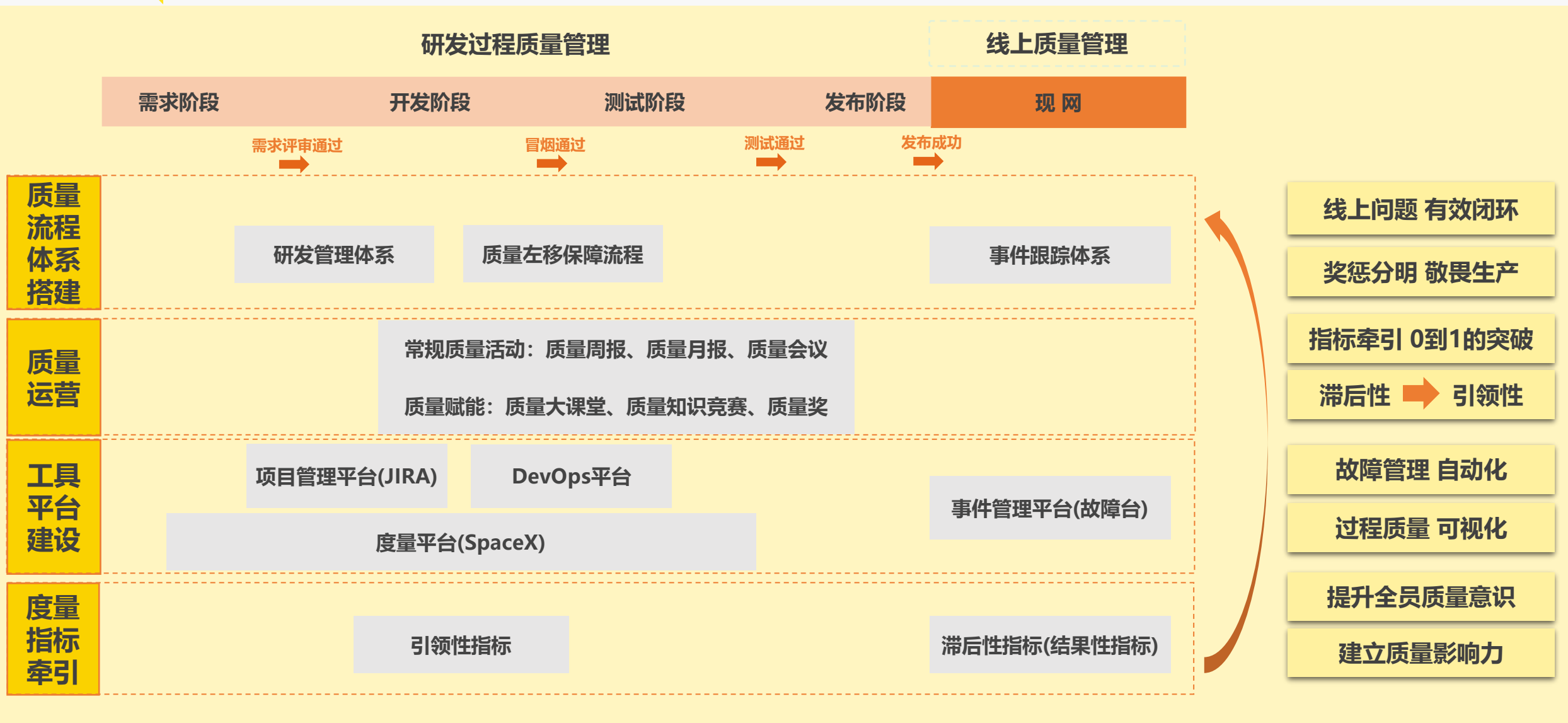
解法



实践 01-质效度量体系建设

实践

01-质效度量体系建设



实践

01-质效度量体系建设

线上管理

研发过程管理

工具自动化

度量可视化

快速分诊

智能质效诊断

✓ 搭建线上管理流程体系



2020

- ✓ 引入线上管理工具
 - ✓ 启动线上度量分析(质效)
 - ✓ 搭建研发过程管理体系
- 2021

- ✓ 引入研发过程管理工具
 - ✓ 启动研发过程度量分析(质效)
 - ✓ 建立度量可视化平台
- 2022

- ✓ 实现自助化质效诊断
 - ✓ 打造自驱式质量管理
- 2023

 ZA Bank 正式上线

实践 01-质效度量体系建设

确定度量目标

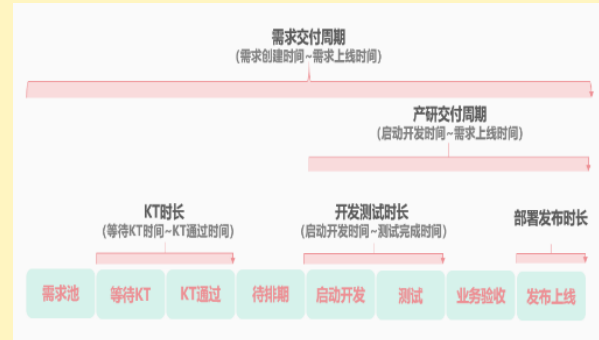
- 软件**成熟度**，所处**阶段**
- 度量**成本**和**预期**
- 度量效果**评估者的视角**
- 度量**实施**的**复杂度**

北极星指标建设

北极星指标

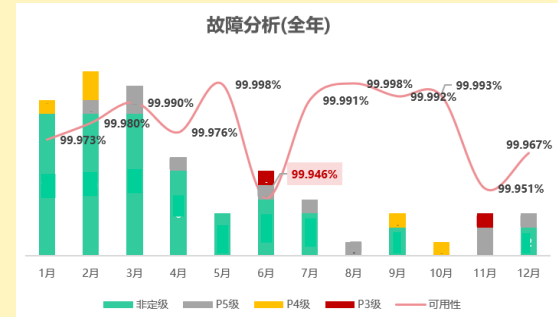
交付价值	产品价值达成率
	客户满意度
交付速率	需求吞吐量(流速率)
	产研交付周期
	缺陷修复周期 (致命严重、一般、轻微)
	故障数
交付质量	系统可用性
	平均故障恢复时长
	逃逸率

统计口径评审



指标项	指标作用	统计口径
需求交付周期	反映团队(包含业务、开发、运维等)对客户问题或业务机会的整体响应速度	从需求进入需求池至需求发布上线的时间周期 JIRA路径:【需求 workflow】A.01.新建需求-Closed
产研交付周期	反映研发的响应能力、交付速度及交付质量	从需求被研发团队确认并启动开发至需求发布上线的时间周期 JIRA路径:【需求 workflow】C.01.处理中-Closed
缺陷修复周期	1)反映研发的缺陷修复能力、处理问题速度等 2)修复成本的高低可间接反映代码质量情况	从缺陷录入至缺陷验证关闭的时间周期 JIRA路径:【迭代缺陷 workflow】A.01.新建bug-Closed

业务线试点



实践 02-质量落地运营

实践 02-质量落地运营

数据处理

大屏					大屏	
工厂内所有设备运行状态: 【数据大屏】查看:已部署、待部署、已完工、无数据、数据异常、数据异常、(不是问题)					APP: 数据下钻分析器	
序号	阶段	指标名称	指标定义及计算公式	统计口径说明	数据来源	
1	运维	★故障告警数	统计周期内的故障总数, 包括【故障告警】、【事件类型】为故障告警	故障告警/【事件类型】为故障告警	Space	
2	运维	★故障告警数	统计周期内已处理的故障总数 (P5及以上)	故障告警/【事件类型】为故障告警 【事件类型】为P5及以上	Space	
3	运维	★系统可用性	1. 单个系统可用性 = 1 - 系统不可用时长 / 系统总运行时长 2. 月度系统可用性 = 1 - 前月系统不可用时长 / 前月系统总运行时长 收集占比数据(CIS: 20%, MIS: 10%, DCI: 8%, PVS: 7%, RLN: 5%, RAS: 5%, KNC: 5%, MAS: 5%, BAS: 5%, PTC: 5%, UPS: 5%, LNS: 5%, 其余: 0.5%)	1. 系统不可用时长: 故障告警/故障告警 输出: 【与时间相关】 2. 系统总运行时长: 系统总运行时长/系统总运行时长 输出: 【与时间相关】 例如: 2022 年10月系统运行时长(min)=31*24*60	Space	
4	运维	★故障告警率	故障告警率 = 已提前已告警故障数 / 故障总数 故障告警率 = 故障告警数 / 故障总数	1. 已提前已告警故障数: 故障告警/故障告警 输出: 【与时间相关】 2. 故障告警率: 故障告警/故障告警 输出: 【与时间相关】 例如: 2022 年10月系统运行时长(min)=31*24*60	Space	

数据采集 数据聚合 数据清洗 数据复核

下钻分析

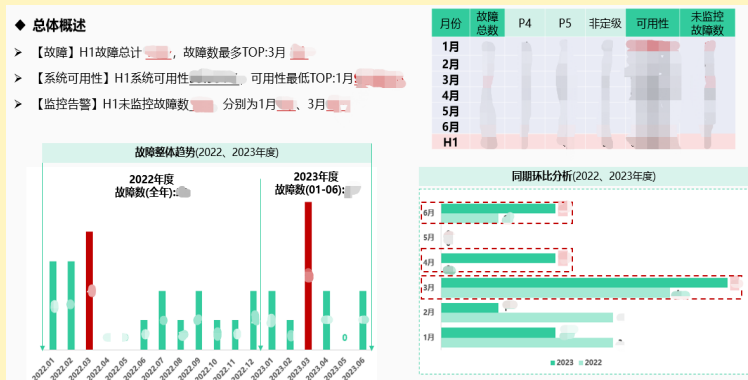
APP】数据下钻...	【Bank-...】数据...	【Bank-迭代质量】数据...
APP】数据下钻... 已关闭数是分子, 我当时统计是有13个未关闭 (38个关闭)	【Bank-...】数据... 提示成功率 BANK-... 前端没有开发完, 后端开关关闭技术上线, 配合前端进度分2个迭代提测 严重bug数占比: credit是一款新产品 7月有1/3是安全扫描的问题, 还有一些账务和出账的问题, 除去安全问题比率是正常的	【Bank-迭代质量】数据... Q3数据来看的话, 共有6个需求驳回, 其中BANK-... BANK-... BANK-... 三个与以上后提是一样的, 另外三个BANK-... BANK-... BANK-... 应该是开发操作问题, 最后只能就后提驳回的, 这几个可以看下

前端APP团队

后端某一级域

迭代质量下钻分析

质效分析报告



趋势分析、环比分析、异常预警、诊断建议、改进效果

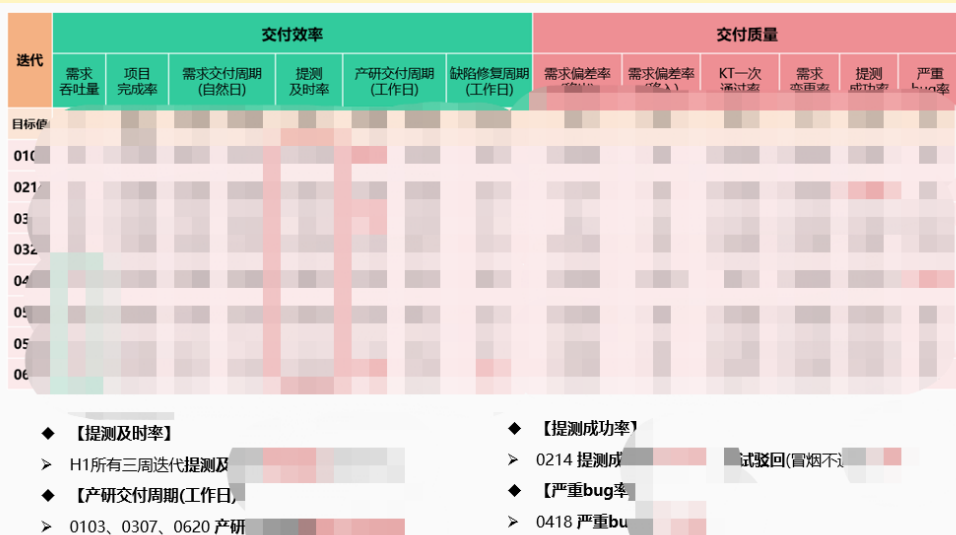
定位诊断

线上质效

迭代质效

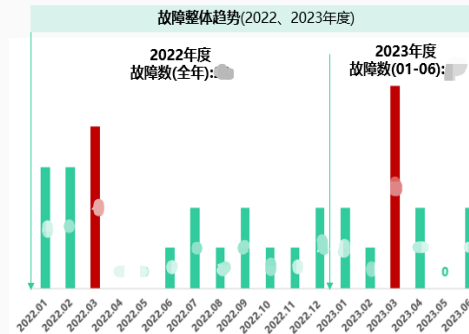
团队质效

质效数据看板
趋势环比
同期比
低健康度预警
诊断建议
改进效果呈现

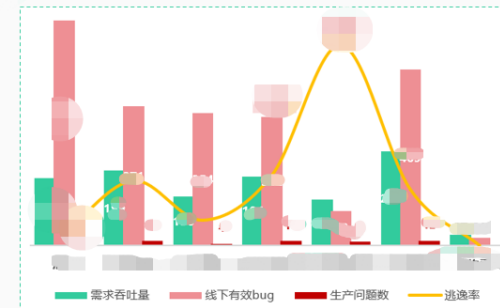
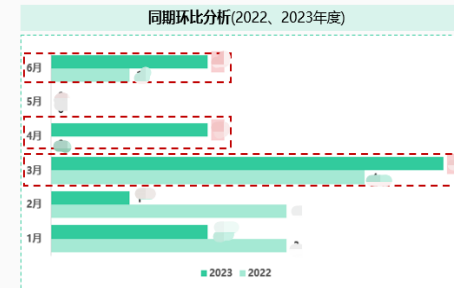


◆ 总体概述

- 【故障】H1故障总计 12, 故障数最多TOP:3月
- 【系统可用性】H1系统可用性 99.99%, 可用性最低TOP:1月
- 【监控告警】H1未监控故障数 1, 分别为1月、3月



月份	故障总数	P4	P5	非定级	可用性	未监控故障数
1月	1	0	0	0	99.99%	0
2月	1	0	0	0	99.99%	0
3月	1	0	0	0	99.99%	0
4月	1	0	0	0	99.99%	0
5月	1	0	0	0	99.99%	0
6月	1	0	0	0	99.99%	0
H1	12	0	0	0	99.99%	1



◆ 线上缺陷密度

【需求吞吐量】(上线需求数)
 【生产问题及故障数】
 【线上缺陷密度】

【需求吞吐量】(上线需求数)
 【生产问题及故障数】
 【线上缺陷密度】

【需求吞吐量】(上线需求数)
 【生产问题及故障数】
 【线上缺陷密度】

◆ 线上逃逸率

【线下bug数】(迭代内)
 【生产问题及故障数】
 【逃逸率】

【线下bug数】(迭代内)
 【生产问题及故障数】
 【逃逸率】

实践

02-质量落地运营

问题来了???

样例：业务线(某中大型规模业务线，150+研发人力)、**质量运营人员**(中级或高级)

➤ **人力投入：10-12人日/月**(工作日)

• 数据采集处理：4-5 人日/月

• 下钻分析：4-5 人日/月

• 编写诊断报告：1.5 人日/月

• 召开质量会议：0.5 人日/月

➤ **数据准确性：95%+-**(高级工程师)、**90%+-** (中级工程师)

➤ **报告时效性：每月15日**输出上个月质效报告

投入成本高

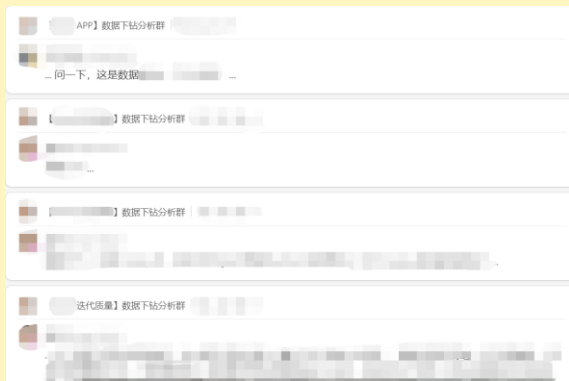
10-12人日/月

数据准确性难以保证

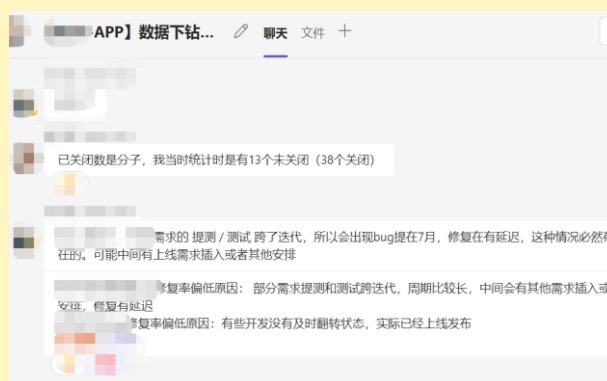
90%+-

报告存在滞后性

每月15日



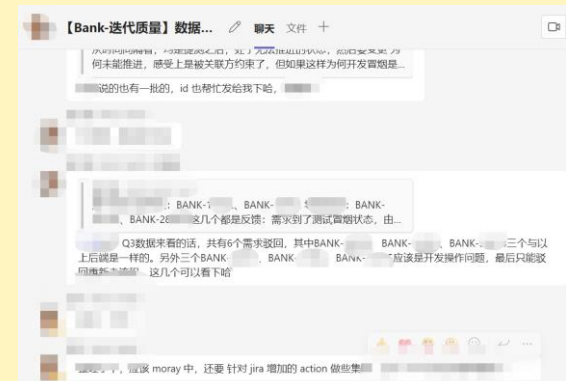
数据复核群



各团队下钻分析群(APP)



各团队下钻分析群(后端某一级域)



迭代质量下钻分析群

实践 03-智能质效管理平台

质效管理平台

【趋势观测】

支持数据**环比**、数据**同期比**，可实时观察研发全过程**走向及趋势**

【追根溯源】

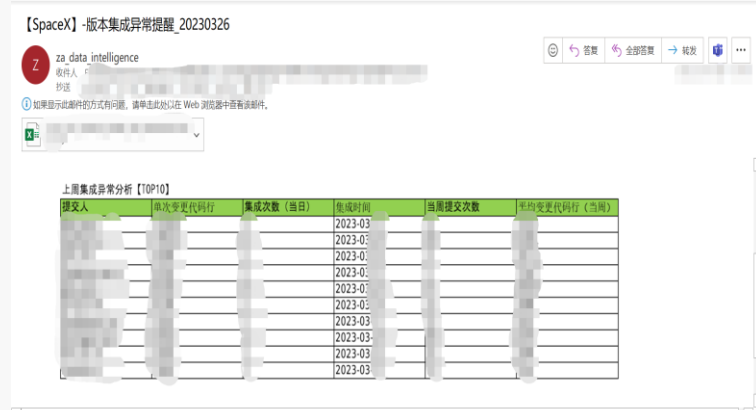
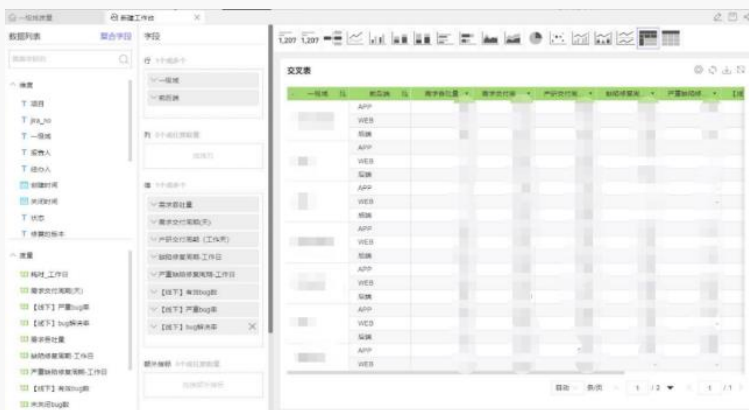
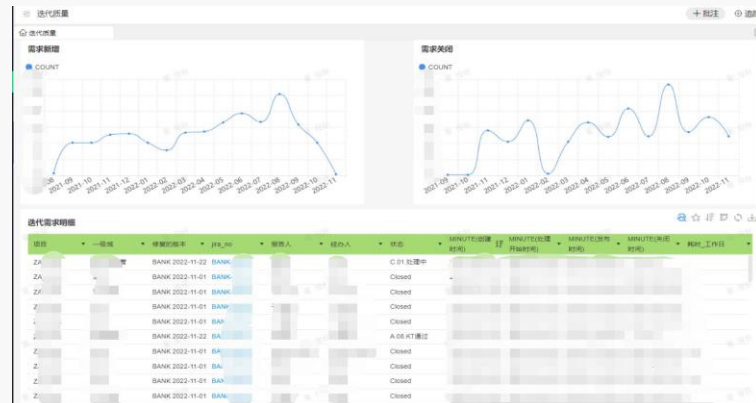
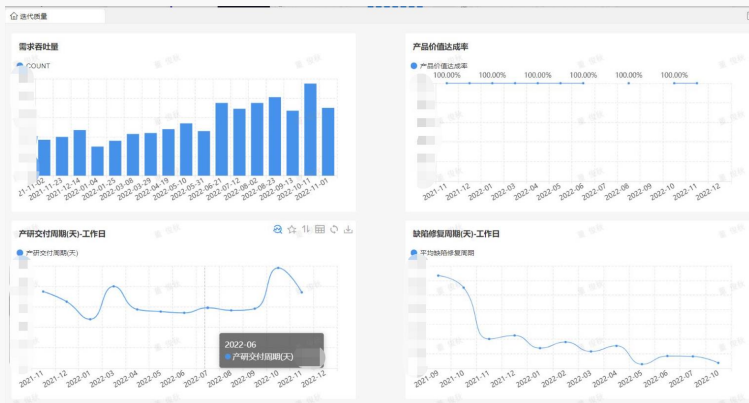
支持数据**下钻**，实现**一键追溯**原始数据，精准定位**异常数据**

【指标自助】

针对不同的关注重点，提供**指标自配**功能，所有复合指标均可自由定制

【风险预测】

实时监测质效心跳，实现质量**风险预警**，质量异常推送



质效管理平台

【趋势观测】

支持数据**环比**、数据**同期比**，可实时观察研发全过程走向及趋势

【追根溯源】

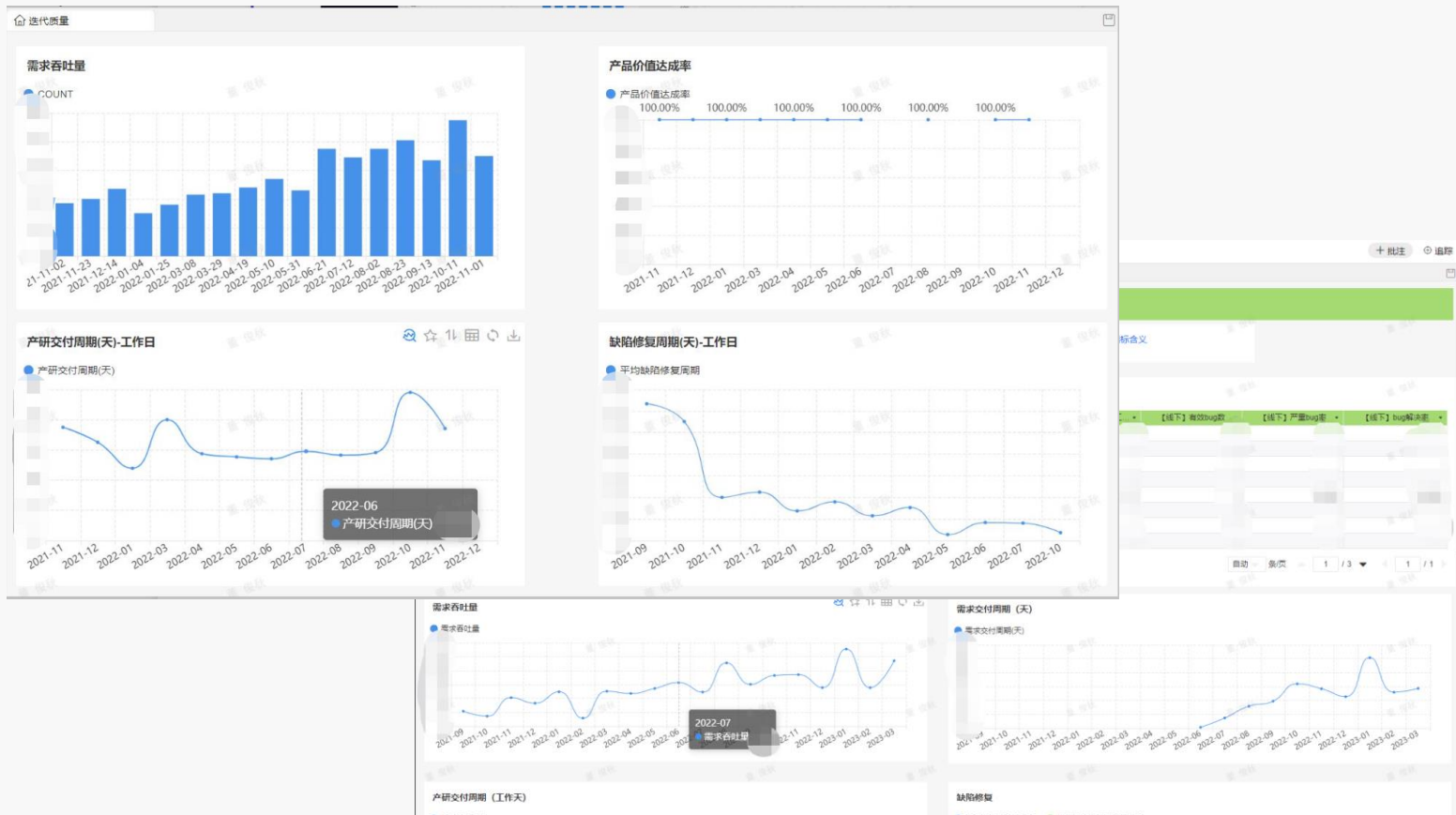
支持数据下钻，实现**一键追溯**原始数据，精准定位**异常数据**

【指标自助】

针对不同的关注重点，提供**指标自配**功能，所有复合指标均可自由定制

【风险预测】

实时监测质效心跳，实现质量**风险预警**，质量异常推送



- 分析维度：迭代、业务线、一级域
- 常用角色：PM、一级域leader、一线工程师

质效管理平台

【趋势观测】

支持数据环比、数据同期比，可实时观察研发全过程走向及趋势

【追根溯源】

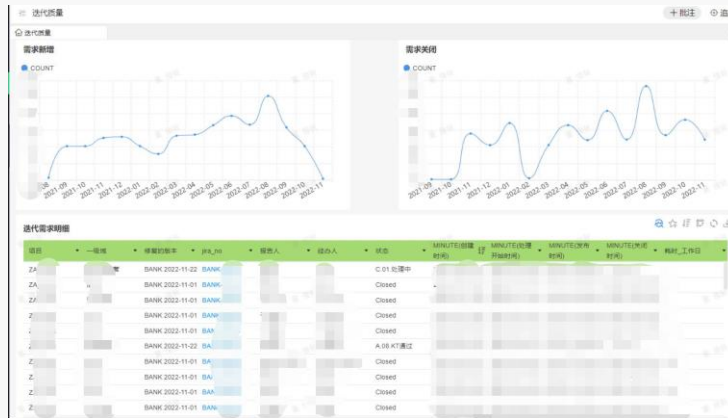
支持数据下钻，实现一键追溯原始数据，精准定位异常数据

【指标自助】

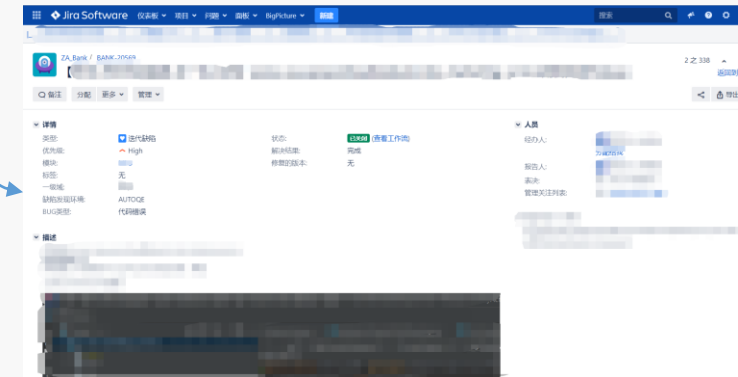
针对不同的关注重点，提供指标自配功能，所有复合指标均可自由定制

【风险预测】

实时监测质效心跳，实现质量风险预警，质量异常推送



故障台



JIRA

- 分析维度：迭代、业务线、一级域
- 常用角色：PM、一级域leader、一线工程师

质效管理平台

【趋势观测】

支持数据**环比**、数据**同期比**，可实时观察研发全过程**走向及趋势**

【追根溯源】

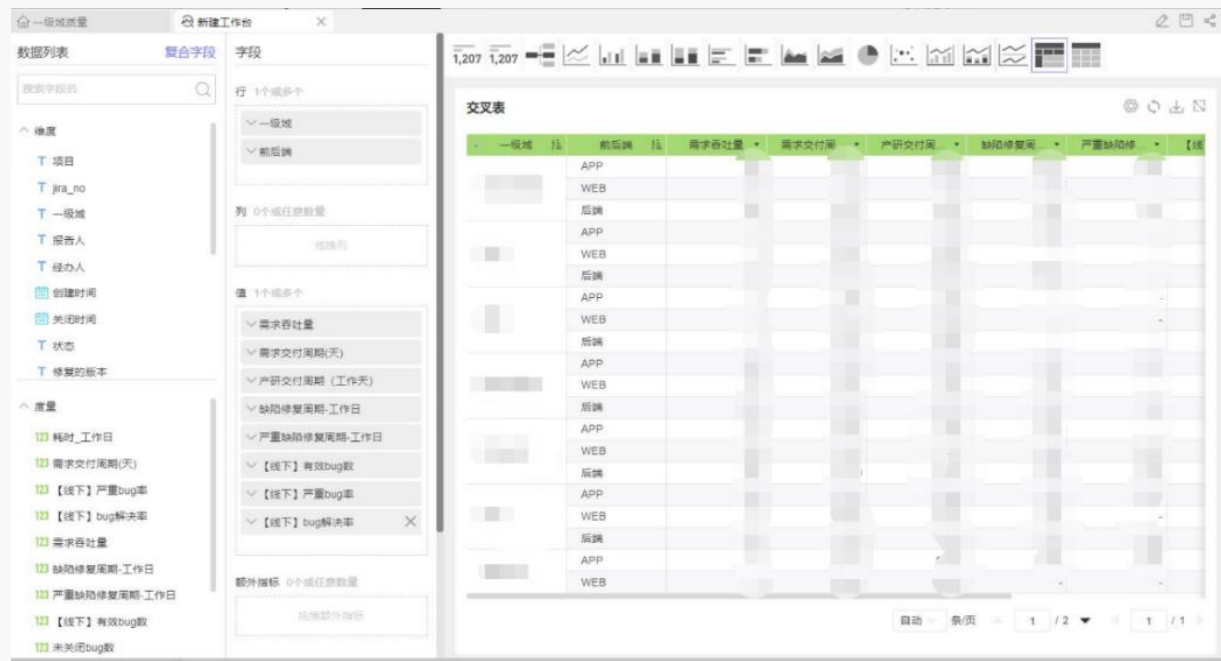
支持数据**下钻**，实现**一键追溯**原始数据，精准定位**异常数据**

【指标自助】

针对不同的关注重点，提供**指标自配**功能，所有复合指标均可自由定制

【风险预测】

实时监测质效心跳，实现质量**风险预警**，质量异常推送



- 常用场景：多业务线，可根据不同业务线的成熟度进行指标的裁剪优化

质效管理平台

【趋势观测】

支持数据环比、数据同期比，可实时观察研发全过程走向及趋势

【追根溯源】

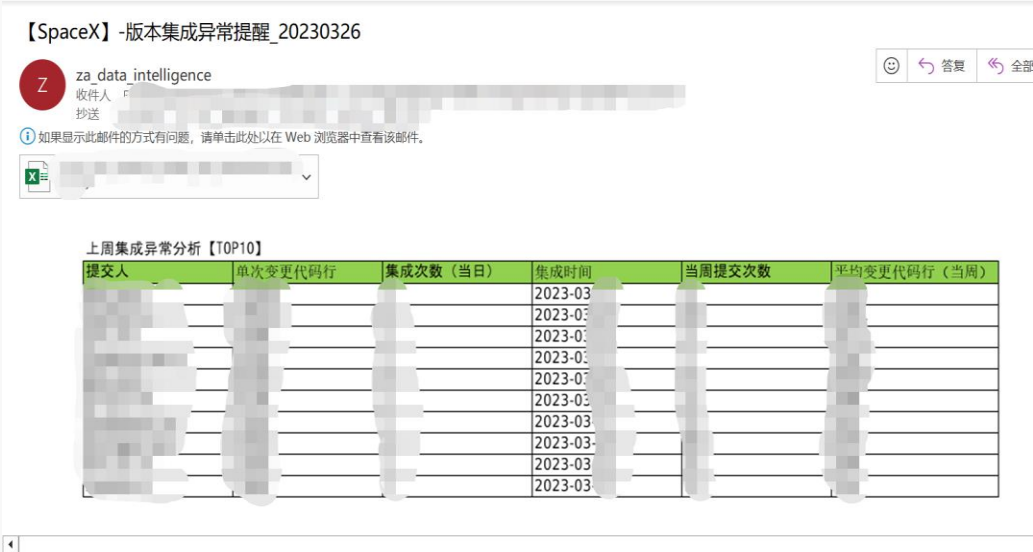
支持数据下钻，实现一键追溯原始数据，精准定位异常数据

【指标自助】

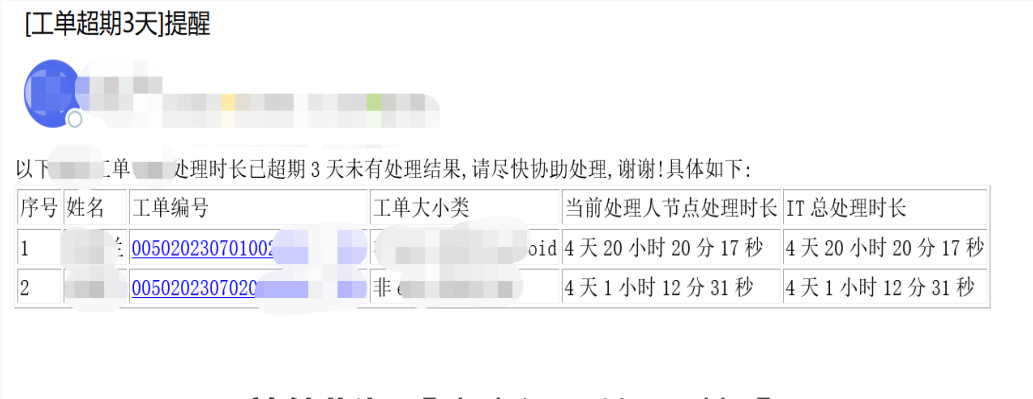
针对不同的关注重点，提供指标自配功能，所有复合指标均可自由定制

【风险预测】

实时监测质效心跳，实现质量风险预警，质量异常推送



质量指标【集成异常】



效能指标【生产问题处理时长】

◆ 应用场景一：过程风险预警

◆ 应用场景二：质效简报

◆ 应用场景三：智能诊断

快速分诊

系统自动化对指标数据扫描检查

高健康度

中健康度

低健康度

诊断建议

经验沉淀

轻量建议

团队复盘

数据分析

人工诊断

定制改进方案

改进效果评估

健康度分级、健康度检查

指标项	指标定义	健康度分级			自动检查频率
		高	中	低	
需求交付周期	需求从等待KT至需求发布上线的时间周期	< 10个工作日	> = 10个工作日 < = 15个工作日	> 15个工作日	迭代/次
产研交付周期	研发启动需求开发至需求发布上线的时间周期	< 8个工作日	> = 8个工作日 < = 12个工作日	> 12个工作日	迭代/次
集成异常	1) 单次/人 提交峰值超过1K行 2) 单日/人 集成次数超过3次	代码行 < 集成次数=1次	待评估	代码行 > 1K 集成次数 > 3次	天/次
线上逃逸率	生产环境出现的问题数(含故障)/线上线上有效问题总数	< 1%	> = 1% < = 3%	> 3%	迭代/次 月度/次
线上缺陷密度	统计周期内, 生产环境出现的问题数(含故障)/交付上线的需求数	待评估	待评估	> 5%	迭代/次 月度/次

- 心跳异常预警
- 问题升级

◆ 应用场景一：过程风险预警

◆ 应用场景二：质效简报

◆ 应用场景三：智能诊断

提示词(示例)

你是一位质量诊断工程师，你需要对以下指标进行健康度扫描，并输出健康度报告。

步骤一：

对“单次变更代码行”、“提交次数”数据进行扫描，记录如下2类数据：

- 1、每个提交人每次提交大于或等于1000行；
- 2、每个提交人每天提交次数累计大于或等于3次。

步骤二：

进行判断，判断原则如下：

- 1、如果单次变更代码行大于或等于1000行，请判断为：异常。
- 2、如果每个提交人每天提交次数累计大于或等于3次，请判断为：异常

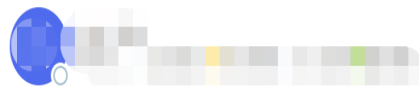
步骤三：

输出判断结果，“异常”或者“正常”。

请注意，你只需要输出“异常”或者“正常”，不需要输出思考过程。



[工单超期3天]提醒



以下工单处理时长已超期 3 天未有处理结果, 请尽快协助处理, 谢谢!具体如下:

序号	姓名	工单编号	工单大小类	当前处理人节点处理时长	IT 总处理时长
1		005020230701002	oid	4 天 20 小时 20 分 17 秒	4 天 20 小时 20 分 17 秒
2		0050202307020	非	4 天 1 小时 12 分 31 秒	4 天 1 小时 12 分 31 秒

【SpaceX】-版本集成异常提醒_20230326



上周集成异常分析【TOP10】

提交人	单次变更代码行	集成次数 (当日)	集成时间	当周提交次数	平均变更代码行 (当周)
			2023-03		
			2023-03		
			2023-03		
			2023-03		
			2023-03		
			2023-03		
			2023-03		
			2023-03		
			2023-03		

健康度级别	规则	上升机制	
		异常推送范围	推送方式
高	健康度分级表	异常处理人、QA、PM	邮件推送 群提醒
中		直属leader/ 所在一级域架构师、QA、PM	
低		业务线研发总监、运维总监、QA、PM	

◆ 应用场景一：过程风险预警

◆ 应用场景二：质效简报

◆ 应用场景三：智能诊断

- 全局诊断
- 数据聚合
- 趋势环比、同期比

质效概述简报

提示词(示例)

你是一位软件质量诊断工程师，需要你结合质效度量数据，输出质效概述简报。简报的时间范围是自然月。简报的内容包括三个部分：“故障数”、“事件数”、“引入原因”。

三个部分的计算规则如下。

- 1、故障数：从输入的数据中取“故障数”的总和，与上一个自然月对比，输出环比提升的数量或者环比下降的数量。输出故障数量最大的3个一级域。
- 2、事件数：从输入的数据中取“事件数”的总和，与上一个自然月对比，输出环比提升的数量或者环比下降的数量。
- 3、引入原因：按照“引入原因”的类型进行聚合，从数量最大的到数量最小的进行排序，并输出。

【线上质量整体概述】

【生产问题】

- ◆ Q1故障总数 **12**，环比上季度 **上升** (2022.Q4: **8**)，环比同期 **下降** (2022.Q1:10)
 - 【系统可用性】Q1系统可用性**99.9%**(达到目标值**99.9%**)
 - 【监控告警】Q1未监控故障数 **1**

【生产问题】

- ◆ Q1生产问题共 **12**
 - 引入原因：【研发】程序缺陷 **8**、【研发】技术方案 **2**、【其他】流程规范类 **2** 【发布】版本发布 **0** 【发布】变更操作失误 **0**
 - TOP3一级域：API **3**、后端-客户及 **3**、后端-支 **3**

【迭代质量分析】

◆ 【交付效率】

- 项目完成率 Q1大部分迭代稳定处于 **90%**左右，0307迭代略低 **85%**；
- 需求交付周期(自然日) 处于 **3-5**天区间
- 产研交付周期(工作日) 自2022年11月起稳定保持在 **1-2**天左右(2022.10前: **2-3**天)

◆ 【交付质量】

- 需求偏差率(移) **10%**
- 需求偏差率(移+) 处于 **10%**左右，与2022年持平
- KT一次通过率 处于 **90%**，与2022年持平
- 需求变更率(内容变更) Q1大部分迭代处于 **10%**左右，0214迭代略高 **15%**(变更需求 **5**，发布需求数 **10**)
- 测试驳回(冒烟不通过) 0214迭代 **1**，0307迭代 **1**，其他迭代 **0**情况
- 严重bug率 Q1三周迭代均控制在 **1%**以下

- 共性问题的下钻分析
- 生产问题的根因分析

◆ 应用场景一：过程风险预警

◆ 应用场景二：质效简报

◆ 应用场景三：智能诊断

提示词(示例)

你是一位质量诊断工程师，请根据历史事件的描述，对故障台中新的“事件描述”内容进行判断。
判断的方法如下：

- 事件类型：是否为“故障”或者“事件”；
- 事件原因：是否为“程序缺陷”；
- 故障跟因：是否包含“慢sql”或“报文积压”或“交易超时”；
- 故障细节：是否包含“慢sql”或“报文积压”或“交易超时”。

事件类型、事件原因、故障跟因需这3个需要同时判断。

判断结束后，请输出对应结论。

如果你认为新的事件与历史事件中的某次事件是同一根因，结论请回复历史事件的“事件名称”和“事件单号”，并标记为“历史故障根因”

诊断卡片

参考案例



下钻分析思路

- ◆ 需求交付周期过长，产研交付周期平稳
 - 考虑前置环节受阻等问题导致等待时间过长(需求评审质量、需求重要程度、资源排期等)
- ◆ 产研交付周期过长
 - 需求颗粒度过粗
 - 效率、质量的问题(研发过程中反复测试驳回)
- ◆ 缺陷修复周期过长
 - 代码耦合性、一致性、架构设计、技术债等问题，导致定位问题更难，修复成本更高
 - 效率问题



成果展示

投入成本

10-12人日/月 → 4人日/月

节省67%

数据准确性

90%+- → 100%

提升10%

报告时效性

每月15日 → 每月5日

提速67%

- 数据采集处理: 4-5 人日/月
- 数据复核: 1 人日/月
- 下钻分析: 4-5 人日/月
- 人工诊断(个例、低健康度): 1 人日/月
- 诊断报告: 1.5 人日/月
- 诊断报告: 1.5 人日/月
- 质量会议: 0.5 人日/人月
- 质量会议: 0.5 人日/人月

业务线投入 过程改进小组(虚拟)

【业务价值】



合作方满意度: 树立QA团队口碑、打造影响力



成本控制: 为企业大大节省了质量运营成本



产品品牌形象: 通过质效全过程的心跳监控, 帮助提前识别质效风险, 确保产品交付质量, 提升客户整体满意度

未来

V1.0

看不见 摸不着 不知好坏

风险不可控 问题难拦截

V2.0

可度量 可分析 驱动改进

事后可追溯 下次可杜绝

V3.0

自分析 自诊断 实现自愈

质量可预测 上线有保障





微信官方公众号：壹佰案例
关注查看更多年度实践案例