

团 体 标 准

T/ISC XXXX—XXXX

应用可观测性平台通用能力技术要求

(征求意见稿)

XXXX - XX - XX 发布

XXXX - XX - XX 实施

目 录

前 言	II
1 范围	1
2 规范性引用文件	1
3 术语和定义	1
4 通用能力	3
5 安全能力	15
6 AI 能力.....	16
7 可观测平台自运维能力	16
8 可扩展能力	17
9 标准化互操作能力	18
附录 A（规范性附录/资料性附录） XXX	19

前 言

本标准按照GB/T 1.1-2020给出的规则起草。

请注意本文件的某些内容可能涉及专利。本文件的发布机构不承担识别这些专利的责任。

本标准由中国互联网协会归口。

本标准主要起草单位：……

本标准主要起草人：……

应用可观测性平台通用能力技术要求

1 范围

本规范旨在定义应用可观测性平台通用能力技术要求，帮助企业构建具备高度可观测性的应用程序平台。可观测性是指对应用程序和基础设施的实时监控、诊断和分析能力，以便及时发现和解决潜在的问题，提高应用程序的稳定性、健壮性和可靠性。

该规范适用于各种类型和规模的应用架构，无论是单体、分布式还是云原生架构。它适用于各种行业和领域，包括但不限于软件开发、云计算、网络服务、物联网、金融科技、电子商务等。通过遵循本规范所定义的技术要求，企业能够建立健壮、高效、安全和易于管理的可观测性平台，提供优质的用户体验，快速响应业务需求，并为持续创新和发展打下坚实的基础。

2 规范性引用文件

下列文件对于本文件的引用是必不可少的。凡是注日期的引用文件，仅注日期的版本适用于本文件。凡是不注日期的引用文件，其最新版本（包括所有的修改单）适用于本文件。

本规范涉及以下引用文件和标准，这些文件对于理解和实施本规范非常重要。在使用本规范时，应参考这些文件的最新版本，并确保遵守其规定。

这些引用文件和标准提供了在构建可观测性平台过程中的指导和参考，应结合本规范进行理解和实施。随着技术的发展和标准的更新，建议定期查阅最新版本，并根据实际情况进行适当的调整和应用。

3 术语和定义

下列术语和定义适用于本文件。

在本规范中，以下术语和定义适用于理解和实施应用可观测性平台通用能力技术要求。

3.1

可观测性（Observability）

可观测性是指应用程序和基础设施的实时监控、诊断和分析能力，包括收集、存储、可视化和分析相关数据以了解系统状态和行为。

可观测性要素（日志、指标和追踪）

可观测性要素包含Metrics、Trace和Log，这三种要素形成一个立体化的架构，是复杂应用程序和基础设施部署和安全稳定运行的支柱。

可观测性平台

可观测性平台是一个集成现有指标数据的工具，它能够为应用程序和基础设施组件添加新的监测数据。该平台的主要功能是持续地识别和收集性能数据，并提取关键信息。

3.2

监控（Monitoring）

监控是指对应用程序和基础设施的指标、事件和日志进行实时的收集、记录和分析，以便获取关于系统性能和行为的信息。

APM :Dynatrace/Datadog/Grafana/ VictoriaMetrics/

3.3

诊断 (Diagnostics)

诊断指通过监控数据和日志信息，分析应用程序和基础设施的问题，以确定故障原因并提供解决方案的过程。

3.4

日志管理 (Log Management)

日志管理指对应用程序和基础设施产生的日志进行收集、存储、检索和分析的过程，以帮助故障排查、性能优化和安全审计等活动。

3.5

性能优化 (Performance Optimization)

性能优化指通过监控和分析应用程序的性能指标，找到瓶颈并进行优化的过程，以提高应用程序的响应速度、吞吐量和资源利用率。

3.6

故障排查 (Troubleshooting)

故障排查指通过使用各种工具和技术，定位并解决应用程序和基础设施中出现的故障或问题的过程。

3.7

安全性 (Security)

安全性指在应用可观测性平台中保护应用程序和数据免受未经授权访问、数据泄露、恶意攻击和其他安全威胁的能力。

3.8

AI 赋能 (AI Empowerment)

AI 赋能指将人工智能技术应用于可观测性平台，包括数据分析、机器学习和智能决策等，以提升监控、诊断和决策能力。

3.9

开放性 (Openness)

开放性指可观测性平台的接口和标准采用开放的、公开的协议和格式，以实现与其他系统和工具的互操作性和集成性。

3.10

自动化 (Automation)

自动化是指利用自动化工具和技术来简化配置、部署、扩展和管理可观测性平台的过程，提高效率 and 一致性。

3.11

可扩展性 (Scalability)

可扩展性是指可观测性平台能够适应不断增长的需求，具备在应用程序和数据规模增长时进行水平和垂直扩展的能力。

3.12

标准化 (Standardization)

标准化是指制定一致的数据格式、协议和模型，以促进行业间的一致性和互操作性，降低集成和开发的复杂性。

4 通用能力

4.1 数据模型

4.1.1 指标模型

- 1、应支持定义统一的指标模型，包括指标的名称、描述、单位等属性。
- 2、应支持设立指标分类和层级结构，方便指标的组织管理。
- 3、宜支持常见的指标类型，如计数器、计时器、分布等。（可选）

4.1.2 日志模型

- 1、应支持定义统一的日志模型，包括日志的级别、标签、时间戳、应用、IP等属性。
- 2、应支持设立日志的结构和格式，以便日志的解析和分析。
- 3、应支持日志的过滤和检索，方便用户根据关键字和条件查询日志数据。

4.1.3 追踪模型

- 1、应支持定义统一的追踪模型，包括追踪的标识、开始时间、结束时间等属性。
- 2、应支持设立追踪的关联关系和层级结构。
- 3、宜提供追踪的可视化和分析。（可选）

4.1.4 剖析模型

- 1、应支持定义统一的剖析模型，包括剖析的目标、参数、结果等属性。
- 2、宜支持设立剖析的类型和方法，如性能剖析、代码剖析等。（可选）
- 3、宜能够提供剖析数据的可视化和分析功能。（可选）

4.1.5 元数据模型

- 1、应支持定义统一的元数据模型，包括数据集的描述、结构、来源等属性。
- 2、应支持设立元数据的分类和层级结构。
- 3、宜支持元数据的关联和查询。（可选）

4.1.6 事件模型

- 1、应支持定义统一的事件模型，包括事件的类型、关键字段、时间戳等属性。
- 2、宜支持设立事件的触发条件和处理流程。（可选）
- 3、宜支持提供事件数据的聚合和分析功能。（可选）

4.2 数据采集和存储

4.2.1 数据采集

- 1、应能够提供多种数据采集方式和协议，例如Agent、API接口、日志收集器、网络流量探针等。
- 2、应支持实时数据采集，能够及时收集应用程序和基础设施产生的指标、事件和日志数据。

- 3、应支持被采集应用程序的稳定运行，减少对应用性能的影响。
- 4、应提供统一的元数据配置方式，应用关键信息，指标、链路、日志、网络流量数据宜包含在内。
- 5、宜提供灵活的数据采集策略和配置，允许按需采集特定数据，减少不必要的数据收集和存储。（可选）
- 6、宜支持数据采集的自动发现，能够自动识别和添加新的数据源。（可选）
- 7、宜提供数据采集性能容量评估机制，支持数据采集限流，在突然流量过大时，宜针对监控指标进行限流。（可选）

4.2.2 数据采集协议和格式

- 1、应支持常见的数据采集协议，如OpenTelemetry、Prometheus、StatsD等。
- 2、应能够定义数据采集格式，如JSON、Protobuf等，以适应不同数据类型和传输需求。
- 3、应能够提供适配器和插件，宜支持第三方数据采集工具的集成，如Fluentd、Logstash等。

4.2.3 数据采集接口和集成工具

- 1、应支持提供统一的数据采集接口。
- 2、应支持设立开发者文档和示例。
- 3、宜支持集成工具和SDK能力，简化数据采集的配置和部署过程。（可选）

4.2.4 自定义数据采集

- 1、应支持用户自定义数据采集，允许用户定义和采集特定业务指标和事件。
- 2、宜支持灵活的数据采集配置，包括数据源、采集频率、过滤条件等。（可选）
- 3、宜支持自定义数据采集插件和脚本，以满足特定数据采集需求。（可选）
- 4、宜支持自定义数据采集粒度及采样率，以满足无损观测和降低采集成本。（可选）

4.2.5 数据标签和元数据

- 1、应支持定义数据标签和元数据的命名规则和规范。
- 2、应支持提供元数据管理功能，允许用户定义和维护数据标签和属性。
- 3、宜支持数据标签的查询和过滤。（可选）
- 4、宜支持数据标签的全链路染色，允许对指标、日志、追踪等模型染色标。（可选）

4.2.6 数据质量和校验

- 1、应支持实施数据质量校验机制，检测和过滤低质量的数据。
- 2、应提供数据异常检测和清洗功能，确保采集到的数据准确和可靠。
- 3、宜支持数据质量评估和报告，帮助用户了解数据的完整性和准确性。（可选）

4.2.7 数据采集性能和容错性

- 1、应支持设立数据采集的容错和重试机制，确保数据采集的可靠性。
- 2、应能够提供数据采集的监控和告警功能，及时发现和解决采集异常和故障。
- 3、宜能够优化数据采集的性能和效率，减少对系统资源的影响。（可选）

4.2.8 数据传输和通信

- 1、应提供安全可靠的数据传输通道，保证数据在传输过程中的完整性和保密性。
- 2、应支持加密和认证机制，保护敏感数据的安全性。
- 3、宜支持数据压缩和优化，减少数据传输的带宽消耗和延迟。（可选）

4.2.9 数据处理和转换

- 1、应能够提供数据处理和转换的功能。
- 2、应提供灵活的数据转换和映射规则，允许根据特定的业务需求和数据模型进行数据处理和转换数据存储和管理。
- 3、宜提供可靠的数据存储和管理机制，能够处理大规模的数据并保证数据的可靠性和持久性。（可选）
- 4、宜支持高吞吐量和低延迟的数据存储。（可选）
- 5、宜提供灵活的数据存储方案，以满足不同类型和规模的数据存储需求，支持根据业务量对存储节点进行灵活扩展。（可选）
- 6、宜支持数据的索引和查询，提供高效的数据检索和访问能力。（可选）
- 7、宜提供数据备份和恢复功能，保障数据的安全性和可用性。（可选）
- 8、宜支持数据聚合和汇总。（可选）

4.2.10 数据存储引擎

- 1、应提供适合不同数据类型和查询需求的存储引擎选择。
- 2、应能够设立存储引擎的部署和管理规范，确保数据存储的稳定性和可靠性。
- 3、应支持多种数据存储引擎，如关系型数据库、时序数据库、分布式存储系统等。

4.2.11 数据存储结构

- 1、应支持设计合适的数据存储结构，以满足不同类型的数据存储需求。
- 2、宜支持定义数据表、索引、分区等存储结构的规范和最佳实践。（可选）
- 3、宜支持提供数据存储结构的管理工具和接口。（可选）

4.2.12 数据保留和清理策略

- 1、应支持设立数据保留和清理策略，定义数据的保存期限和清理规则。
- 2、宜支持自动化的数据清理和归档功能。（可选）
- 3、宜支持提供数据保留和清理的监控和报告。（可选）

4.2.13 数据备份和恢复

- 1、应支持设计数据备份和恢复策略，确保数据的安全性和可恢复性。
- 2、应支持定义数据备份的频率、存储位置和恢复过程。
- 3、宜支持提供数据备份和恢复的工具和接口。（可选）

4.2.14 数据安全和合规性

- 1、应支持设立数据安全和合规性规范，确保数据在存储过程中的安全性和合规性。
- 2、应支持数据加密和访问控制。

3、宜遵守相关的数据隐私法规和合规标准，如GDPR、HIPAA等。（可选）

4.2.15 数据存储性能和扩展性

- 1、应支持设计可扩展的存储架构，支持数据量和用户规模的增长。
- 2、宜支持提供监控和调优工具。（可选）
- 3、宜能够优化数据存储的性能和响应时间，提高数据查询和分析的效率。（可选）

4.2.16 数据质量和一致性

- 1、应提供数据质量监测和评估功能，能够检测和报告数据的准确性、完整性和一致性。
- 2、应提供数据一致性保证机制，确保在分布式环境下的数据一致性和同步。
- 3、宜支持数据质量的自动化检测和修复，能够自动发现和修复数据质量问题。（可选）

4.2.17 数据生命周期管理

- 1、应提供数据生命周期管理功能，支持数据的归档、压缩、转储和删除等操作，以最优化存储资源的使用和成本。
- 2、宜具备数据保留策略和合规性要求的支持。（可选）

4.3 数据分析能力

- 1、应实现跨多个层次的端到端数据联动。
- 2、应对不同类型数据的全面分析能力。
- 3、应支持多维度数据分析，能够对数据进行多维度的聚合、过滤和分组，以获取更全面和深入的数据洞察。
- 4、应针对不同业务日志需实现数据的快速聚合。
- 5、应支持数据关联，应用的监测数据涉及多个维度（机房、网络、机器、应用），能自动生成观测拓扑。
- 6、宜提供适用全部类型的可观测性产品和技术方案。（可选）
- 7、宜针对不同类型的用户提供不同层级的可观测性解决方案。（可选）
- 8、宜提供一体化可观测性解决方案，针对指标、日志、链路的关联分析等数据应用提供可行路径。
- 9、宜提供强大的数据查询功能，支持灵活的查询语言和查询接口。（可选）
- 10、宜提供数据筛选和过滤功能。（可选）
- 11、宜支持复杂的数据计算和统计，包括求和、平均值、最大值、最小值、百分比等常用的统计计算操作。（可选）

4.4 指标体系

4.4.1 指标定义和命名规范

- 1、应支持定义指标的概念和含义。
- 2、应支持设立指标命名规范，包括命名约定、命名空间等。
- 3、应支持提供指标的文档和描述。

4.4.2 指标类型和分类

- 1、应支持定义不同指标的类型，如计数型指标、时长型指标、比率型指标等。

- 2、宜支持设立指标的分类和层级结构，可针对多层数据进行合成、聚合分析等。（可选）
- 3、宜支持提供指标类型和分类的文档和示例。（可选）

4.4.3 指标计算和聚合

- 1、应定义指标的计算方法和公式。
- 2、应设立指标的聚合方式，支持求和、平均值、最大值、最小值等常用聚合操作。支持提供指标计算和聚合的工具和函数。

4.4.4 指标关联和分析

- 1、应定义指标之间的关联关系。
- 2、应提供指标关联分析的功能，如相关性分析、趋势分析等。
- 3、宜支持指标的多维度分析。（可选）

4.4.5 指标可视化和报告

- 1、应支持提供指标的可视化展示，如图表、仪表盘等，方便用户直观了解数据。
- 2、宜支持设立指标报告的生成和分享功能。（可选）
- 3、宜支持指标数据的导出和导入。（可选）
- 4、宜支持报告指标语义统一，方便用于用户理解和解释。（可选）

4.4.6 指标监控和警报

- 1、应支持设计指标监控和警报机制，及时发现指标异常和趋势变化。
- 2、应支持提供指标监控和警报的设置和配置，允许用户定义监控规则和阈值。
- 3、宜支持指标监控的可视化和实时反馈。（可选）

4.5 拓扑体系

4.5.1 拓扑可视化

- 1、应定义拓扑层次结构划分标准。
- 2、宜支持联动应用系统自动发现应用拓扑，以应用为基本单元，绘制全局拓扑关系。（可选）
- 3、宜支持准实时显示系统故障，全面分析应用健康状况和性能指标。（可选）
- 4、宜支持选择拓扑历史快照进行可视化展示。（可选）

4.6 巡检体系

4.6.1 监控巡检

- 1、应可配置巡检逻辑，允许用户自定义监控频率、性能阈值、巡检范围等
- 2、宜对应用状态以及CPU、内存、网络等性能容量信息巡检。（可选）
- 3、宜能够提供巡检仪表盘，展示应用关键运行状态。（可选）
- 4、宜能够提供巡检结果，以邮件、短信等方式通知。（可选）

4.7 可视化能力

4.7.1 数据可视化

- 1、应提供丰富的数据可视化方式，包括图表、仪表盘、报表等。

- 2、宜支持多种图表类型，如折线图、柱状图、饼图、热力图等。（可选）
- 3、宜具备灵活的可视化配置选项，允许用户自定义图表样式、颜色、坐标轴等，以适应特定的数据展示需求。（可选）
- 4、宜提供交互式的可视化功能。（可选）

4.7.2 实时数据监控和报警

- 1、应提供实时数据监控的可视化界面，能够实时显示应用程序和基础设施的状态和性能指标。
- 2、应支持实时数据的动态更新和展示。
- 3、宜具备灵活的告警规则配置，能够根据定义的阈值和条件触发实时告警，及时通知相关人员。（可选）
- 4、宜提供告警通知的可视化和报告。（可选）

4.7.3 数据报告和导出

- 1、应提供报告生成和导出功能。
- 2、宜具备灵活的报告模板和样式配置。（可选）
- 3、宜支持报告的自动化生成和定时发送。（可选）
- 4、宜提供报告的导出选项，支持常见的文件格式。（可选）

4.7.4 数据探索 and 发现

- 1、应提供数据探索工具和功能。
- 2、宜支持数据挖掘和机器学习算法的集成。（可选）
- 3、宜提供数据关联和相关性分析。（可选）

4.8 可观测阈值

4.8.1 时序数据预警

- 1、应能够设定合适的时序数据预警阈值，用于监测关键指标和性能参数的异常情况。
- 2、宜能提供预警规则和配置的灵活性，以满足不同指标和业务场景的需求。（可选）
- 3、宜支持预警通知和报警机制，及时向相关人员发送警报信息。（可选）

4.8.2 智能监控

- 1、应能够提供自动化的异常检测算法和模型，以识别潜在的问题和异常行为。
- 2、宜能够结合机器学习和人工智能技术，实现智能监控和异常检测功能。（可选）

4.8.3 容量规划和预测

- 1、应提供容量规划工具和功能，可根据历史数据和趋势分析预测未来的容量需求。
- 2、宜支持容量规划的可视化展示。（可选）
- 3、宜提供容量规划的报告和建议，帮助用户制定合理的容量扩展和升级计划。（可选）
- 4、宜具备容量调整和优化的功能，能够根据实时需求和优先级进行容量的动态调整和管理。（可选）

4.8.4 性能优化报告和总结

- 1、应提供性能优化报告的生成和导出功能。
- 2、宜具备报告模板和样式的配置选项，支持定制化的报告格式和布局。（可选）
- 3、宜支持性能优化经验和最佳实践的知识库和共享。（可选）

4.9 监控标准化

4.9.1 时序数据库监控

- 1、应支持可提供统一的基于时序数据库监控指标采集，结合大盘实现对系统基础资源、业务指标的监控。
- 2、宜支持可提供统一的基于时序数据库监控指标，结合应用可观测能力，提供多维度指标告警能力。（可选）

4.9.2 数据格式标准化

- 1、应支持定义和采用统一的数据格式标准，确保数据在不同系统和组件之间的互操作性。
- 2、应支持提供数据格式转换和映射功能。
- 3、应支持常见的数据格式标准，如JSON、XML、CSV等，以满足不同数据交换和集成的需求。

4.9.3 数据命名和命名空间管理

- 1、应需设立统一的数据命名规范和命名空间管理策略，确保数据的唯一性和一致性。
- 2、应提供数据命名空间的注册和管理功能。
- 3、宜支持数据命名的自动化和规范化。（可选）

4.9.4 数据词汇和术语统一

- 1、应支持确立统一的数据词汇和术语表。
- 2、宜支持提供数据词汇和术语的管理和维护功能。（可选）
- 3、宜支持数据词汇和术语的自动化解析和标注。（可选）

4.9.5 数据质量和校验

- 1、应实施数据质量管理策略，包括数据完整性、准确性、一致性和可靠性等方面的校验。
- 2、宜提供数据质量评估和监控功能，及时发现和处理数据质量问题。（可选）
- 3、宜支持数据质量报告和数据修复的功能。（可选）

4.9.6 数据元数据管理

- 1、应支持管理和维护数据元数据，包括数据的描述、定义、结构和关系等信息。
- 2、宜支持提供数据元数据的搜索和查询功能。（可选）
- 3、宜支持数据元数据的血缘分析和影响分析。（可选）

4.9.7 数据安全和隐私保护

- 1、应支持制定数据安全和隐私保护策略。
- 2、应提供数据访问控制和权限管理功能，限制对敏感数据的访问和操作权限。
- 3、宜遵循相关法规和合规性要求，如GDPR、HIPAA等，保护用户隐私和数据安全。（可选）

4.10 自监控能力

- 1、宜具备自监控能力，对于可观测性工具，能够确保运行及性能上的自主监测。（可选）
- 2、宜具备平台自身的审计能力，能对使用用户的关键操作进行审计及查看分析。（可选）

4.11 实时监控和告警

4.11.1 监控

- 1、应能发现潜在问题。
- 2、宜能实时监控内存、CPU、网络情况、慢SQL、慢堆栈、JVM内存、线程等指标能及时发现性能瓶颈。（可选）
- 3、宜能通过实时监控应用程序的错误信息、日志等，可及时发现和解决应用程序的错误和异常，确保应用程序的稳定运行。（可选）

4.11.2 实时监控仪表盘

- 1、应能够提供实时监控仪表盘，展示关键指标的实时数值和趋势。
- 2、宜需要具备可配置的仪表盘布局和样式，允许用户自定义关注的指标和展示方式。（可选）
- 3、宜能够支持实时刷新和动态更新。（可选）

4.11.3 实时指标监测

- 1、应能够实时监测关键指标，包括性能指标、资源利用率、错误率等，以及自定义的应用程序指标。
- 2、宜能提供多维度的指标监测，支持对不同维度的指标进行聚合、分组和对比分析。（可选）
- 3、宜具备灵活的指标阈值配置，能够设置告警触发的条件和级别。（可选）

4.11.4 实时日志监控

- 1、应支持实时收集和监控应用程序和系统的日志数据。
- 2、宜提供日志过滤和搜索功能，以使用户快速定位关键日志和事件。（可选）
- 3、宜支持自定义日志查询和分析，允许用户根据特定的关键字和条件进行日志分析和筛选。（可选）
- 4、宜具备日志告警功能，能够根据定义的规则和条件触发实时告警通知。（可选）

4.11.5 告警配置和管理

- 1、应提供灵活的告警规则配置，能够根据指标的阈值、趋势和时间条件定义告警规则。
- 2、宜支持多种告警通知方式，包括邮件、短信、即时通知等，以确保及时通知相关人员。（可选）
- 3、宜具备告警级别和优先级的定义，以区分不同严重程度的告警情况。（可选）
- 4、宜提供告警事件的管理和跟踪功能。（可选）

4.11.6 告警通知和响应

- 1、应支持实时发送告警通知，及时通知相关人员或团队。
- 2、宜支持告警通知的自定义模板和格式，以适应不同通知渠道和需求。（可选）
- 3、宜能提供告警事件的确认和处理流程，确保告警的及时响应和解决。（可选）
- 4、宜具备告警事件的历史记录和报告，以便进行告警分析和审计。（可选）

4.11.7 告警可视化和分析

- 1、应提供告警仪表盘和报表，以展示告警的统计信息和趋势。
- 2、宜支持告警数据的分析和比较，帮助用户了解告警的分布、频率和紧急程度。（可选）
- 3、宜提供告警事件的关联和根因分析，以便快速定位和解决问题。（可选）

4.12 故障诊断与排查

4.12.1 诊断能力

- 1、系统宜具备针对CPU代码级的剖析能力。（可选）
- 2、宜提供针对代码级内存的诊断能力。（可选）
- 3、宜提供线程剖析诊断能力。（可选）

4.12.2 故障定位和分析

- 1、应提供强大的故障定位工具，能够快速定位和分析应用程序和基础设施的故障原因。
- 2、宜支持故障定位的可视化界面，展示故障相关的指标、日志和事件，说明故障范围，展示故障爆炸半径。（可选）
- 3、宜提供故障定位的自动化和智能化功能，利用机器学习或数据分析技术帮助用户快速定位故障点。（可选）
- 4、宜需具备故障定位的历史记录和跟踪功能。（可选）

4.12.3 根因分析和追溯

- 1、应要提供根因分析工具和功能，能够追溯故障的根本原因和影响因素。
- 2、宜支持故障根因的自动化识别和推理，以便快速找出导致故障的关键因素。（可选）
- 3、宜提供根因分析的可视化展示，例如故障树、流程图等。（可选）
- 4、宜需具备故障根因的追溯和溯源功能。（可选）
- 5、宜需提供故障模拟工具，能够快速发起故障注入模拟测试。（可选）
- 6、宜需具备故障历史现场的保留功能，例如故障历史库、根因库、应急方案库等。（可选）

4.12.4 故障日志和事件检索

- 1、应能够提供快速的故障日志和事件检索功能，以使用户根据关键字和条件查找相关日志和事件。
- 2、宜能够支持日志和事件的过滤和筛选，以排除无关的日志信息，聚焦于故障相关的数据。（可选）
- 3、宜要具备日志和事件的可视化展示，能够以图表或图形的形式呈现故障相关的日志和事件。（可选）

4.12.5 问题追踪和协作

- 1、应提供问题追踪工具和功能，能够记录和跟踪故障处理过程中的问题和任务。
- 2、宜支持故障处理团队的协作和沟通，能够分配任务、共享知识和进行实时讨论。（可选）
- 3、宜具备问题状态和优先级的管理，以及问题解决进度的跟踪和报告。（可选）

4.12.6 故障预测和自动化修复

- 1、应能够利用机器学习和数据分析技术，提供故障预测和自动化修复的功能。
- 2、宜支持基于历史数据和模型训练，能够预测潜在的故障和问题，并提供相应的预防措施。（可选）
- 3、宜支持自动化修复和恢复功能，能够自动识别和修复常见的故障情况。（可选）

4.12.7 故障分析报告和总结

- 1、宜提供故障分析报告的生成和导出功能。（可选）
- 2、宜具备报告模板和样式的配置选项，支持定制化的报告格式和布局。（可选）
- 3、宜支持故障案例的知识库和共享，以便提供参考和共享经验。（可选）

4.13 性能优化与容量规划

4.13.1 性能监测和分析

- 1、应提供实时性能监测功能，能够实时监测应用程序和基础设施的性能指标。
- 2、宜支持多维度的性能分析，能够对不同维度的性能指标进行聚合、分组和对比分析。（可选）
- 3、宜支持提供性能趋势分析，能够识别和预测性能变化的趋势，以及性能瓶颈的潜在问题。（可选）
- 4、宜具备性能异常检测和警报功能，能够自动识别性能异常并及时通知相关人员。（可选）

4.13.2 瓶颈识别和优化

- 1、应提供瓶颈识别工具和功能，能够帮助用户快速定位性能瓶颈和瓶颈原因。
- 2、宜支持性能瓶颈的自动化识别和分析，利用机器学习或数据分析技术提供智能化的瓶颈分析结果。（可选）
- 3、宜提供性能优化建议和推荐。（可选）
- 4、宜具备性能优化的历史记录和跟踪功能。（可选）

4.13.3 资源规划和管理

- 1、应支持资源利用率的实时监测和报告，以使用户了解资源的使用情况和趋势。
- 2、宜提供容量规划和预测功能，基于历史数据和趋势分析预测未来的资源需求，并进行相应的资源规划。（可选）
- 3、宜具备资源分配和调整的功能，能够根据实时需求和优先级进行资源的动态分配和管理。（可选）
- 4、宜提供资源利用率的自动化调整和优化，以最大程度地提高资源的利用效率。（可选）

4.14 多环境适配

应用可观性应支持多种环境下的可观测：

4.14.1 可支持多云环境

- 1、应支持主流云服务提供商的环境，例如Amazon Web Services (AWS)、Microsoft Azure、Google Cloud Platform (GCP)、华为云、阿里云、腾讯云等。

2、宜具备与不同云服务商的API和服务集成能力，以便在多云环境中无缝迁移和纳采应用程序。
(可选)

3、宜提供统一的控制面板或管理平台。(可选)

4.14.2 可支持混合云环境

1、应支持混合云架构，即支持将应用程序部署在私有云和公共云之间的混合环境中。

2、宜具备与私有云平台 and 公共云平台的集成能力，实现资源的统一管理和协同工作。(可选)

3、宜提供一致的监控和管理接口。(可选)

4.14.3 ARM/X86 环境部署

1、应支持不同处理器架构的环境部署，包括ARM和X86架构。

2、宜具备对ARM架构和x86架构的应用程序的自动适配和优化能力。(可选)

4.14.4 终端兼容

1、应支持适配多种操作系统，如Windows、Linux、Ubuntu、EulerOS、Kylin等操作系统。

2、宜支持适配多种浏览器，如IE、Chrome、Edge等。(可选)

4.14.5 云平台兼容

1、应支持阿里云、华为云、微软云等云平台。

2、宜支持公有云、私有云、自建云等云平台。(可选)

3、宜支持多种混合云平台。(可选)

4、应支持主流的云平台虚拟机部署。(可选)

5、宜支持主流云平台Kubernetes容器化管理平台部署。(可选)

4.14.6 多环境配置管理

1、宜提供灵活的配置管理功能，能够根据不同环境的要求进行配置参数的调整。(可选)

2、宜具备配置版本控制和回滚功能。(可选)

1、宜支持不同环境变量中的动态配置的管理。(可选)

4.14.7 灰度发布和回滚

1、应能提供灰度发布和回滚功能。

2、应具备灰度发布策略的配置和管理能力。(可选)

3、应支持快速回滚到上一个稳定版本的功能。(可选)

4.15 集成能力

4.15.1 数据源和数据集成

1、应支持与各种数据源的集成，包括数据库、消息队列、文件系统等。

2、应具备数据采集和数据转换的能力，能够将不同数据源的数据统一收集和处理。

3、宜提供数据集成和数据流的管理功能。(可选)

4.15.2 同业产品、开源产品的集成能力

- 1、宜提供标准化统一接口，具备与同业产品（如监控系统、日志管理系统）的集成能力，能够接收和处理其生成的数据和事件。（可选）
- 2、宜支持与开源产品（如 Prometheus、Grafana）的集成。（可选）
- 3、宜支持提供与第三方API和服务的集成接口。（可选）

4.15.3 容器化和微服务架构的深入集成

- 1、宜支持与容器化平台（如 Docker、Kubernetes）的集成。（可选）
- 2、宜具备与微服务架构的集成能力，能够监控和管理分布式的微服务组件。（可选）
- 3、宜提供容器化和微服务架构的可观测性解决方案。（可选）
- 4、宜支持与无服务（Serverless）架构平台集成，能够适应不断变化的应用部署。（可选）

4.15.4 第三方服务集成

- 1、应支持与身份验证和授权服务的集成，以确保对第三方服务的安全访问和操作。
- 2、宜具备与第三方服务（如云服务、API服务）的集成能力，以便实现功能扩展和增强。（可选）
- 3、宜提供第三方服务的监控和性能分析功能。（可选）

4.15.5 自定义集成接口和插件

- 1、宜提供自定义集成接口和插件机制。（可选）
- 2、宜具备开发者友好的集成文档和工具。（可选）
- 3、宜支持第三方开发者社区的集成插件和扩展。（可选）

4.16 应用管理架构

- 1、应从人员、技术、组织管理等层面明确主题，确保可观测性技术能够应用落地、产生应用价值。

4.17 应用性能观测

应支持服务器与操作系统性能指标的可观测性，包括但不限于：

- 1、应支持CPU、内存、磁盘状态、磁盘IO速率、内存Swap情况等。
- 2、应支持数据链路状态、链路节点吞吐量情况、链路响应耗时情况、链路访问错误率等。
- 3、应支持网络流量性能指标。
- 4、宜支持异常性能告警与分析，其中异常性能包含错误请求及慢请求，其中慢请求的阈值提供默认值，可自定义进行设置。（可选）
- 5、宜支持用户体验性能分析。（可选）
- 6、宜支持业务性能分析。（可选）

4.18 可持续性

应用可观测性技术标准需要具有良好的可持续改进性，标准需定期审查和更新，以反映最新技术和最佳实践，确保企业始终能够获得最先进的应用可观测性解决方案。

4.18.1 定期检查和更新

- 1、宜建立定期审查和更新机制，对应用可观测性平台的技术要求进行评估和更新。（可选）
- 2、宜跟踪行业发展和最佳实践，及时调整和更新技术要求。（可选）

4.18.2 技术演进和创新

- 1、宜持续关注新的技术趋势和创新。（可选）

- 2、宜推动技术创新和实验，探索新的可观测性技术和解决方案。（可选）
- 3、宜与技术社区和合作伙伴合作，促进技术共享和开放创新。（可选）

4.19 持续改进和优化

- 1、应设立持续改进的机制，定期评估应用可观测性平台的性能和效果。
- 2、宜收集用户反馈和需求，针对问题和改进点进行优化和调整。（可选）
- 3、宜要推动自动化和智能化的改进，提高应用可观测性平台的效率和可靠性。（可选）

4.20 整体性能和可扩展性

- 1、应需要关注应用可观测性平台的整体性能和可扩展性，确保其能够满足日益增长的数据量和用户需求。
- 2、宜进行性能测试和负载测试，发现瓶颈并进行优化。（可选）
- 3、宜支持水平扩展和垂直扩展。（可选）

5 安全能力

5.1 数据加密

应在采集层、数据层、网络传输层对数据进行各种加密和脱敏等处理，来确保用户的数据安全具体要求如下：

- 1、应支持数据加密。
- 2、应支持数据内容的加密。
- 3、应支持数据脱敏处理。
- 4、应支持数据安全管控，如权限管控、身份验证、审计管理等，其中审计管理包括可观测行为记录、用户权限和资源分布。
- 5、应实施数据加密机制，保护敏感数据在存储和传输过程中的安全性。
- 6、应采用适当的加密算法和密钥管理策略，确保数据的机密性和完整性。
- 7、应支持端到端的数据加密，包括数据的采集、传输和存储环节。

5.2 数据脱敏

- 1、应能够针对敏感数据采用数据脱敏技术，保护用户隐私和敏感信息。
- 2、应需要确保脱敏后的数据仍能保持可观测性，以满足业务分析和监控的需求。
- 3、宜能够提供多种脱敏方式，如替换、屏蔽、泛化等，根据不同数据类型和需求进行选择。（可选）

5.3 数据安全管控

- 1、应需要设立数据安全管控策略和流程，确保数据的安全性和合规性。
- 2、应支持实施数据访问控制和权限管理。
- 3、应能够进行数据备份和恢复的规划。
- 4、应能够监测和记录数据的访问日志。

5.4 安全漏洞管理

- 1、应需要定期进行安全漏洞扫描和评估，发现和修复系统中的安全漏洞。
- 2、宜需要设立漏洞管理流程，及时修复已知漏洞，并防止新漏洞的出现。（可选）
- 3、宜需要关注安全漏洞的公开信息和最新补丁，及时更新和升级系统和组件。（可选）

5.5 威胁监测和防御

- 1、应支持部署威胁检测和防御系统，监测和阻止潜在的安全威胁。
- 2、宜需要实施入侵检测和防御措施。（可选）
- 3、宜能够加强网络安全和数据传输的加密。（可选）

5.6 日志审计

- 1、应能够记录系统操作日志和安全事件日志。
- 2、应能够实施日志的安全存储和访问控制，防止篡改和未经授权的访问。
- 3、宜能够分析日志数据，发现潜在的安全问题和异常行为。（可选）

6 AI 能力

应用可观测性相关产品/解决方案可以集成机器学习和人工智能技术，对大量数据进行分析和预测，提供更加智能化的服务。

6.1 智能分析

- 1、应支持趋势预测，预测可能会发生的事件。
- 2、宜支持根因分析，找到异常发生的根本原因。（可选）
- 3、宜支持推荐解决/优化方案，找到原因后需要能够提供临时解决方案来解决问题。（可选）

6.2 智能决策

- 1、宜支持智能决策，根据分析得出结论并提供应对建议后，决策是否执行该建议。（可选）
- 2、宜支持智能变更，依托于基础设施的自动化对智能决策结果进行变更动作的执行。（可选）
- 3、宜支持智能决策将智能分析的结果确定下来并执行，可释放大量人力，降本增效。（可选）

6.3 智能展示

- 1、应支持立体化观测，即对一个小目标的全方位微观观测。
- 2、宜支持智慧大屏，即对一个大系统宏观展示。（可选）
- 3、宜支持智能客服，可以将关键事件主动通知到目标接受人，或根据需要响应返回内容。（可选）

6.4 自动化

宜支持通过自动化监测、诊断和修复流程，减少人工干预和提高系统的自适应性。（可选）

7 可观测平台自运维能力

7.1 自动化部署和配置

- 1、应提供自动化的应用程序部署和配置功能，能够自动完成环境搭建、依赖安装、配置文件修改等操作。
- 2、应支持基础设施的自动化配置和编排。

7.2 自动化运维和管理

- 1、应提供自动化的运维和管理功能，包括自动化的监控、日志收集、备份、扩展和故障恢复等操作。

作。

- 2、宜支持自动化的任务调度和定时作业。(可选)
- 3、宜具备自动化的故障检测和恢复能力。(可选)

7.3 智能化问题诊断和处理

- 1、宜基于机器学习和数据分析技术，提供智能化的问题诊断和处理功能。(可选)
- 2、宜能够自动识别和分类常见的问题类型，并提供相应的解决方案和建议。(可选)
- 3、宜具备智能化的异常检测和预测能力，能够提前发现潜在的问题并采取预防措施。(可选)

7.4 自动化扩展和负载均衡

- 1、应提供自动化的扩展和负载均衡功能，能够根据实时负载情况自动调整应用程序和基础设施的规模。
- 2、宜具备智能化的负载均衡策略和算法，能够根据负载情况动态调整负载分配和请求路由。(可选)
- 3、宜支持自动化的资源调度和负载优化。(可选)

7.5 自动化报告和决策支持

- 1、宜提供自动化报告生成和导出功能，能够根据预定义的指标和模板生成定期报告。(可选)
- 2、宜支持自动化的数据分析和决策支持，能够根据数据模型和算法生成洞察报告和建议。(可选)
- 3、宜具备智能化的数据可视化和交互功能。(可选)

8 可扩展能力

增加描述

8.1 水平扩展

- 1、应支持设计可扩展的架构和组件，根据数据量和用户规模的变化，提供水平扩展能力。
- 2、宜支持实施负载均衡机制，将请求和数据分发到多个节点，提高系统的稳定性。(可选)

8.2 弹性扩展

- 1、应支持弹性伸缩，根据业务需求和负载变化自动调整系统的规模和容量。
- 2、宜能够设立弹性伸缩策略和规则，根据预设条件自动扩展或收缩系统资源。(可选)
- 3、宜能够提供自动化的伸缩功能，减少人工干预和响应时间。(可选)

8.3 垂直扩展

- 1、应支持垂直扩展，通过增加单个节点的资源（如 CPU、内存、存储）来提升系统性能。
- 2、宜支持设计可扩展的数据库和存储方案。(可选)

8.4 分布式结构

- 1、应能够采用分布式架构，将系统功能和数据分布到多个节点和区域。
- 2、应能够提供分布式数据存储和处理能力，以实现数据的可靠性和高可用性。
- 3、应能够设计弹性和容错性，保证系统在节点故障或网络中断的情况下仍能正常运行。

8.5 开放接口和插件机制

- 1、应能够提供开放的接口和 API。
- 2、宜能够支持自定义插件和扩展机制。(可选)
- 3、宜能够提供开发者文档和示例。(可选)

9 标准化互操作能力

9.1 API 标准

9.1.1 API 命名和规范

- 1、应支持定义统一的 API 命名规则，包括命名风格、命名空间等。
- 2、应支持设立 API 的版本管理机制，确保 API 的向后兼容性和升级策略。
- 3、应能够提供 API 文档和示例，明确 API 的用途、参数、返回值等。

9.1.2 认证和权限管理

- 1、应支持常见的 API 认证机制，如 API 密钥、OAuth 等。
- 2、应能够设立权限管理策略，控制 API 的访问权限和操作权限。
- 3、应能够提供访问令牌管理和刷新机制，确保 API 的安全性和合规性。

9.1.3 请求和响应格式

- 1、应支持定义统一的请求格式，包括 HTTP 方法、路径、参数、请求体等。
- 2、应能够设立一致的响应格式，包括状态码、响应头、响应体等。
- 3、应支持常见的数据格式，如 JSON、XML 等，以满足不同数据交换需求。

9.1.4 错误处理和异常管理

- 1、应支持定义错误码和错误消息的规范，方便错误处理和问题定位。
- 2、应能够提供异常处理机制，捕获和处理 API 调用过程中的异常情况。
- 3、应能够提供合适的错误日志和错误报告，帮助用户快速定位和解决问题。

9.1.5 API 版本管理

- 1、应支持设立 API 版本管理策略，支持多版本 API 的并存和切换。
- 2、应能够提供 API 版本的文档和变更记录，帮助用户了解和适应新版本的 API。
- 3、宜支持向后兼容性和退化策略，以减少对旧版本 API 的依赖和影响。(可选)

9.1.6 API 性能和限流控制

- 1、宜支持设计高性能的 API 接口，减少响应时间和提高并发能力。(可选)
- 2、宜能够提供 API 请求的限流和配额控制，防止滥用和资源过载。(可选)
- 3、宜支持监控和记录 API 调用的性能指标，以便进行性能分析和优化。(可选)

9.2 页面展示规则

9.2.1 仪表盘设计

- 1、宜支持设计直观清晰的仪表盘布局。(可选)
- 2、宜支持提供灵活的仪表盘配置，允许用户自定义添加、删除和调整组件。(可选)
- 3、宜支持可视化组件的自由拖拽和调整大小，满足用户的个性化展示需求。(可选)

9.2.2 报表生成

- 1、应支持提供报表生成功能，允许用户根据特定时间范围和数据维度生成报表。
- 2、宜支持设立报表模板和样式，方便用户快速生成规范化的报表。(可选)
- 3、宜支持导出报表的多种格式，如 PDF、Excel 等。(可选)

9.2.3 图表展示

- 1、宜支持提供多样化的图表类型，如折线图、柱状图、饼图等。(可选)
- 2、宜支持图表的配置和定制，包括颜色、图例、坐标轴等属性。(可选)
- 3、宜支持提供交互式图表功能，允许用户进行缩放、选择和筛选操作。(可选)

9.2.4 呈现规则和样式

- 1、宜支持设立一致的呈现规则和样式。(可选)
- 2、宜支持提供主题和样式定制。(可选)
- 3、宜支持响应式设计，使页面在不同设备和分辨率下呈现良好。(可选)

9.2.5 数据过滤和搜索

- 1、应支持提供数据过滤和搜索功能。
- 2、应支持设立过滤和搜索规则，允许用户自定义过滤器和搜索条件。
- 3、宜支持快速定位和跳转到特定数据和时间范围。(可选)
- 4、宜支持观测数据的搜索、快速访问。(可选)

9.2.6 用户权限和个性化配置

- 1、应支持设计灵活的用户权限管理机制，例如多租户管理，确保数据及访问权限隔离。
- 2、宜支持提供个性化配置功能。(可选)
- 3、宜支持用户偏好和保存设置。(可选)

附录 A
(规范性附录/资料性附录)
XXX
