
云计算开源产业联盟技术文件

Q/KXY MI002—2022

云原生能力成熟度模型 第 5 部分：中间件

Cloud native capability maturity model part 5: middleware

2022-9-26 发布

2022-11-1 实施

云计算开源产业联盟 发布

目次

前言..... III

引言..... IV

云原生能力成熟度模型 第 5 部分：中间件..... 1

 1 范围..... 1

 2 规范性引用文件..... 1

 3 术语和定义..... 1

 3.1..... 1

 3.2..... 1

 云原生 cloud native..... 1

 3.3..... 1

 3.4..... 1

 云原生平台 cloud native platform..... 1

 3.5..... 2

 4 缩略语..... 2

 5 云原生中间件成熟度模型..... 2

 5.1 概述..... 2

 5.2 云原生中间件综合成熟度评估方法..... 2

 6 弹性..... 3

 7 可观测性..... 3

 7.1 日志..... 3

 7.2 监控..... 4

 7.3 链路追踪..... 4

 8 可移植性..... 4

 8.1 制品形态..... 5

 8.2 部署与卸载..... 5

 8.3 环境兼容..... 5

 9 可维护性..... 5

 9.1 版本管理..... 6

 9.2 配置管理..... 6

 9.3 状态管理..... 6

 9.4 客户端..... 7

 10 高可用性..... 7

10.1 架构设计 7

10.2 容灾 7

10.3 数据可靠性 7

10.4 故障处理 8

 11 安全性 8

 11.1 权限控制 8

 11.2 资源隔离 9

 11.3 安全增强 9

 12 开放性 9

 12.1 能力扩展 9

 12.2 开放式 API 9

前 言

本文件按照GB/T 1.1-2020《标准化工作导则 第1部分：标准化文件的结构和起草规则》的规定起草。

本文件是云原生能力成熟度系列标准之一。该标准的结构和名称预计如下：

- 第1部分：技术架构；
- 第2部分：业务应用；
- 第3部分：架构安全；
- 第4部分：电信行业IT业务系统；
- 第5部分：中间件。

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。

本文件起草单位：中国信息通信研究院、网易数帆、腾讯云、华为云、谐云、中移云能、博云、金蝶云、浪潮云。

本技术报告主要起草人：周丹颖、裴明明、张晓宇、黄珂、朱玉华、陈幼安、郭旻、王伟、林琳、赵山。

引 言

云原生技术持续演进，行业应用不断深化，对云原生平台的功能丰富度提出更高要求，但平台力量始终是有限的，需要中间件、SaaS等第三方生态服务补足能力空缺。在此背景下，为提升中间件服务的云原生能力，规范中间件服务与云原生平台的融合对接方式，建立针对中间件的云原生能力成熟度模型具有相当的必要性，有助于推动中间件服务深度云原生化转型。《云原生能力成熟度模型》系列标准拟由5部分构成：

- 第1部分：技术架构。目的在于规范云原生技术架构和服务能力，指导服务提供商和用户建设云原生技术平台。
- 第2部分：业务应用。目的在于指导用户基于云原生的业务应用系统建设路径，帮助用户提升云原生应用水平。
- 第3部分：安全架构。目的在于规范云原生安全架构和服务能力，指导服务提供商和用户提高云原生平台和应用的安全水平。
- 第4部分：电信行业IT业务系统。目的在于指导电信行业用户基于云原生的IT业务系统的建设路径，帮助电信行业用户提升云原生应用水平。
- 第5部分：中间件。目的在于提升中间件服务的云原生能力，指导中间件服务与云原生平台的规范化对接。

云原生能力成熟度模型

第5部分：中间件

1 范围

本文件规定了中间件服务的云原生能力成熟度评估方法与指标体系，从弹性、可观测性、可移植性、可维护性、高可用性、安全性、开放性七个方面考察中间件服务自身的云原生化程度以及中间件服务与云原生平台的融合对接能力。

本文件适用于衡量中间件的云原生化程度，也适用于为中间件的云原生化转型提供参考方向和指引路径。

2 规范性引用文件

下列文件对于本文件的应用是必不可少的。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。

GB/T 32399-2015 信息技术 云计算 参考架构

GB/T 33847-2017 信息技术 中间件术语

3 术语和定义

下列术语和定义适用于本文件。

3.1

中间件 middleware

位于系统软件之上，用于支持分布式应用软件，连接不同软件实体的支撑软件。

[GB/T 33847—2017, 2.1]

3.2

云原生 cloud native

一种面向云计算环境的应用设计理念、技术栈的集合。

3.3

云原生中间件 cloud native middleware

具备云原生特性且适用于云原生场景的中间件。

3.4

云原生平台 cloud native platform

基于云原生技术，如容器、微服务、服务网格、服务器无感知等，构建的提供应用运行所需服务的平台。

3.5

活动 activity

一组特定任务的合集。

[GB/T 32399—2015, 3.2.1]

3.6

角色 role

一组服务于共同目的的活动（3.5）的集合。

[GB/T 32399—2015, 3.2.7]

4 缩略语

API：应用程序接口（Application Programming Interface）

CPU：中央处理器（Central Processing Unit）

CRD：自定义资源定义（CustomResourceDefinition）

HTTP：超文本传输协议（Hyper Text Transfer Protocol）

SDK：软件开发工具包（Software Development Kit）

RPM：红帽软件包管理器（Red-Hat Package Manager）

5 云原生中间件成熟度模型

5.1 概述

表1 云原生中间件成熟度模型图

能力域	弹性	可观测性	可移植性	可维护性	高可用性	安全性	开放性
能力项	弹性伸缩	日志	制品形态	版本管理	架构设计	权限控制	能力扩展
		监控	部署与卸载	配置管理	容灾	资源隔离	开放式API
		链路追踪	环境兼容	状态管理	数据可靠性	安全增强	
				客户端	故障处理		

5.2 云原生中间件综合成熟度评估方法

云原生中间件成熟度共分为五级，具体要求如下：

级别	英文	中文	定义
1级	Initial Level	初始级	开始云原生化尝试，可部署在云原生环境
2级	Fundamental Level	基础级	初步云原生化，可运行在云原生平台上并执行简单的管理操作
3级	Comprehensive Level	全面级	基本云原生化，利用云原生技术对自身架构进行部分改造，与云原生平台适配度高
4级	Excellent Level	优秀级	深度云原生化，利用云原生技术对自身架构做深度改造，与云原生平台能力深度融合与协作
5级	Fabulous Level	卓越级	全面云原生化，完全基于云原生技术构建，可灵活交付并完全融入到任意云原生平台，可实现智能化运维与治理

6 弹性

级别	弹性伸缩
1	支持在部署前配置副本数与规格，部署后不具备伸缩能力。
2	支持手动横向扩缩容副本，且提供负载自动均衡能力，数据在扩缩容后可自动均衡分布； 支持手动纵向提高规格； 弹性伸缩过程中业务需暂停。
3	在上一级基础上， 支持定时横向自动扩缩容策略； 支持弹性伸缩对业务的影响控制在分钟级。
4	在上一级基础上， 支持根据监控指标横向自动扩缩容副本，如负载情况、性能指标等； 支持手动纵向降低规格； 支持执行减少副本数或降低规格配置时的错误检查，并可自动调整为合理配置并完成缩容； 支持弹性伸缩操作后可实现秒级流量接入。
5	在上一级基础上， 支持通过丰富的事件源触发横向自动扩缩容与集群数据自动迁移，如HTTP/HTTPS、消息队列、日志等事件； 支持通过丰富的事件源自动执行纵向规格调整； 支持弹性伸缩对业务无感知。

7 可观测性

7.1 日志

级别	日志
1	支持记录系统日志。
2	在上一级基础上，

	支持日志分割； 支持对日志进行分级，如Info、Warn、Error； 支持标准化错误码，同类错误对应同一种错误码； 支持日志的轮转和保留策略。
3	在上一级基础上， 支持日志内容涵盖多个维度，如管控、运行、访问等； 支持日志中详细记录请求处理流程，且请求日志有关联 ID（适用于非高吞吐类中间件）。
4	在上一级基础上， 支持根据分布式链路规范生成日志，如OpenTelemetry，可灵活对接到依据规范建设的日志服务。
5	

7.2 监控

级别	监控
1	支持监控指标涵盖集群健康状况。
2	在上一级基础上， 支持以监控面板形式将监控数据对接到云原生平台； 支持监控指标涵盖集群的基础指标，如使用、运行情况。
3	在上一级基础上， 支持适配云原生主流开源规范，如 Prometheus，可将监控数据对接到云原生平台； 支持监控指标涵盖集群、业务的关键性能指标； 支持监控指标的多维度聚合。
4	在上一级基础上， 支持监控指标涵盖集群全面指标，包括集群级别、节点级别、资源级别等维度； 支持内置逻辑对监控指标的正常、异常数值范围进行判定； 支持用户以模板形式配置所需监控指标； 支持云原生可观测数据规范，如OpenTelemetry，可灵活对接到依据规范建设的监测服务。
5	

7.3 链路追踪

级别	链路追踪
1	
2	
3	支持作为调用链的一环，接入应用的分布式调用链中。
4	在上一级基础上， 支持云原生可观测数据规范，如OpenTelemetry，可灵活接入到应用的分布式调用链中并提供基础数据。
5	在上一级基础上， 支持中间件自身提供内部的详细链路追踪，如各组件间的调用链路、存储与计算间的调用链路等。

8 可移植性

8.1 制品形态

级别	制品形态
1	支持提供官方软件包，如rpm包、压缩包、二进制包等。
2	在上一级基础上， 支持提供容器镜像作为制品。
3	同上
4	在上一级基础上， 支持提供基于云原生平台执行自动化部署、编排、运维等操作的制品包，如Operator、Helm chart包； 支持制品包中根据云原生平台要求描述所需信息，如产品名、类型、版本号、功能、参数值推荐范围、依赖组件、使用注意事项等（参照 云原生平台中间件管理能力要求 6.2.1制品包规范）。
5	在上一级基础上， 支持适配云原生开源规范，灵活生成制品包。

8.2 部署与卸载

级别	部署与卸载
1	支持手动部署各个组件； 支持手动添加配置文件。
2	在上一级基础上， 支持通过脚本半自动化部署。
3	在上一级基础上， 支持基于 Helm Chart 或同类型编排能力进行部署和卸载。
4	在上一级基础上， 支持中间件实例及相关资源由 Operator 创建； 支持中间件Operator自动实现实例资源的删除与清理； 支持灵活与制品库、部署流水线等集成，实现流程化部署。
5	

8.3 环境兼容

级别	环境兼容
1	
2	
3	支持主流 Kubernetes 发行版本，兼容不同版本的 Kubernetes 集群。
4	在上一级基础上， 支持兼容异构服务器，如X86、Arm等； 支持兼容异构硬件，如鲲鹏、海光、申威、龙芯、飞腾、倚天等； 支持兼容多操作系统，如linux、uos、麒麟、龙蜥等。
5	在上一级基础上， 支持兼容在同集群中混合部署的异构服务器、硬件、多操作系统。

9 可维护性

9.1 版本管理

级别	版本管理
1	版本更新需要通过卸载旧版本安装新版本的方式实现。
2	支持在当前版本基础上直接进行更新，无需卸载旧版本。
3	在上一级基础上， 支持自定义更新策略，如滚动、蓝绿等，且更新过程中服务基本不受影响； 支持更新过程中或更新完成后执行版本回退。
4	在上一级基础上， 支持灵活与制品库、部署流水线等集成，实现流程化版本更新。
5	在上一级基础上， 支持基于流量的灰度，能够主动将流量分发到未处于更新中的实例上，更新过程中服务不中断。

9.2 配置管理

级别	配置管理
1	支持静态配置。
2	在上一级基础上， 支持动态的简单配置管理，配置参数以功能相关为主； 支持配置修改，但无法热更新； 支持集群配置变更自动同步。
3	在上一级基础上， 支持动态的基础配置管理，配置参数涵盖功能、性能、安全等多方面； 支持部分配置热更新； 支持配置参数的模板管理。
4	在上一级基础上， 支持配置参数的多维度管理，如集群、节点等维度。
5	在上一级基础上， 支持所有配置的热更新、接口更新。

9.3 状态管理

级别	状态管理
1	支持简单的集群状态管理能力，可区分可用和不可用两种状态。
2	在上一级基础上， 支持针对集群、节点等关键资源进行细致的状态管理，状态有多种； 支持提供状态暴露接口；
3	在上一级基础上， 支持针对各类资源都有状态管理，状态维度丰富，每种状态有详细的衡量方式，能反馈集群详细运行状况，并且提供上述所有资源状态的暴露接口； 支持提供非健康状态下的配套运维手段，如启动、停止、重启等操作。
4	
5	

9.4 客户端

级别	客户端
1	支持开源SDK接入。
2	在上一级基础上， 支持提供 SDK，有业务封装，接入简单； 支持通过客户端运维中间件，可实现如集群管理、状态管理、配置更新等能力。
3	在上一级基础上， 支持提供适配多语言的SDK； 支持SDK的灾备、故障处理等高级容灾能力。
4	
5	

10 高可用性

10.1 架构设计

级别	架构设计
1	支持单点模式。
2	在上一级基础上， 支持多副本模式，数据实时同步、节点间选主； 支持依赖解耦，通过声明式引入依赖。
3	在上一级基础上， 支持集群模式，支持数据分片； 支持配置解耦，配置修改不影响代码变更。
4	在上一级基础上， 支持数据解耦，计算与存储分离。
5	在上一级基础上， 支持采用Serverless形态托管。

10.2 容灾

级别	容灾
1	支持单节点数据持久化。
2	支持多计算节点容灾。
3	在上一级基础上， 支持同城多机房容灾。
4	在上一级基础上， 支持异地容灾。
5	在上一级基础上， 支持异地多活。

10.3 数据可靠性

级别	数据可靠性
1	支持单副本持久化数据，数据弱一致性。
2	支持双副本持久化数据，数据弱一致性。
3	在上一级基础上， 支持多副本（至少三个）持久化数据，数据弱一致性。
4	在上一级基础上， 支持多副本持久化数据，动态管理同步列表； 支持集群内数据强一致性，集群间数据弱一致性。
5	在上一级基础上， 支持实时持久化数据，数据强一致性，断电不丢数据。

10.4 故障处理

级别	故障处理
1	支持定位中间件自身故障。
2	在上一级基础上， 支持提供对中间件自身故障的修复建议。
3	在上一级基础上， 支持提供对中间件自身故障修复的详细方案与步骤； 支持单节点宕机故障转移或自愈，自愈后中间件可正常提供服务，且数据一致。
4	在上一级基础上， 支持网络分区故障自愈，自愈后中间件可正常提供服务，且数据一致； 支持少数节点宕机故障转移或自愈，自愈后中间件可正常提供服务，且数据一致。
5	在上一级基础上， 支持智能化自动化应对各类故障，自愈期间中间件不中断提供服务，且数据一致； 支持中间件的依赖服务故障不影响业务连续性。

11 安全性

11.1 权限控制

级别	权限控制
1	支持提供管理员和用户两种权限。
2	在上一级基础上， 支持提供多维度权限限制，如读、写、管理等； 支持身份验证、授权； 支持对接云原生平台的权限体系，如通过接口暴露等方式。
3	在上一级基础上， 支持基于角色的访问控制； 支持从横向和纵向两个维度执行权限控制，并且提供自由组合模式。
4	在上一级基础上， 支持授权粒度精细化，如基于进出数据、数据字典、方法调用等控制点。
5	

11.2 资源隔离

级别	资源隔离
1	支持粗粒度的物理资源隔离，如物理机、网络等维度。
2	同上
3	在上一级基础上， 支持细粒度的逻辑资源隔离，如 namespace、访问权限等维度。
4	在上一级基础上， 支持细粒度的物理资源隔离，如 cpu、存储等维度。
5	

11.3 安全增强

级别	安全增强
1	支持用户名密码静态认证模式，并支持一定的密码复杂度要求。
2	在上一级基础上， 支持用户名密码动态认证模式，如 token； 支持提供数据传输的完整性保护，如 TLS 非对称密钥加密传输。
3	在上一级基础上， 支持数据传输的端到端机密性保护，并支持国际通用加密算法和国密算法； 支持保障对外暴露的API安全。
4	在上一级基础上， 支持与第三方安全管控平台联动以增强安全能力。
5	

12 开放性

12.1 能力扩展

级别	能力扩展
1	无法实现能力扩展。
2	支持通过侵入式编程方式实现能力扩展。
3	支持通过插件实现能力扩展，扩展组件需要重启服务。
4	同上
5	在上一级基础上， 支持热加载能力扩展组件。

12.2 开放式 API

级别	开放式 API
1	支持提供状态查询的开放式API。
2	在上一级基础上， 支持提供中间件相关的全维度查询类开放式API。
3	在上一级基础上，

	支持提供中间件配置变更的声明式API； 支持提供的API具有统一的风格规范。
4	在上一级基础上， 支持提供部分运维操作的开放式API；
5	在上一级基础上， 支持提供完整的生命周期管控、运维操作等开放式API，三方平台可接入中间件集群所有能力； 支持开放式API提供细粒度权限划分。