

ICS 点击此处添加 ICS 号

CCS 点击此处添加 CCS 号

T/CAICI

中国通信企业协会团体标准

T/CAICI XXXX—XXXX

算力集群多云调度技术规范

A technical norm for Computing First Network Cloud scheduling

（征求意见稿）

在提交反馈意见时，请将您知道的相关专利连同支持性文件一并附上。

XXXX – XX – XX 发布

XXXX – XX – XX 实施

中国通信企业协会 发布

目 次

前 言 II

引 言 III

1. 范围 1

2. 规范性引用文件 1

3. 术语和定义 1

4. 概述 1

5. 基础能力要求 2

6. 多云接入 3

7 控制管理 5

8 算力编排 7

9 可视化层 9

10 安全 10

参 考 文 献 11

索 引 12

算力集群多云调度技术规范

1. 范围

本标准规定了算力集群多云调度技术的技术架构、设计及部署等方面的技术要求。
本标准适用于指导算力集群多云系统设计和建成。

2. 规范性引用文件

下列文件中的内容通过文中的规范性引用而构成本文件必不可少的条款。其中，注日期的引用文件，仅该日期对应的版本适用于本文件；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。

3. 术语和定义

算力 Computing force

算力是设备、集群、平台处理数据的计算能力，是任意计算场景下任何信息数据处理量的总和。

算力服务 Computing force services

算力服务是以多样性算力为基础，以算力网络为链接，通过全新计算技术将异构算力统一输出，并与云、大数据、AI 等交叉融合，最终将算力、数据等资源统一封装，以服务形式（如 API）交付的新型服务类型。

算力调度 Computing force scheduling

算力调度是指基于对用户位置、业务需求、网络条件、算力资源信息等条件综合考虑，对泛在算力进行整体调度的行为，具体环节包含算力注册、算力感知等。

容器标准 Container Standard

容器标准指的是一系列规范和标准，用于定义如何创建、打包和部署容器。这些标准可以确保不同容器的可移植性和互操作性，使得容器可以在各种不同的环境中运行。

4. 概述

4.1 总体技术架构

算力集群多云调度的技术体系架构主要分为基础能力、多云接入、控制管理、编排能力、业务逻辑等五部分。本标准主要定义这五部分的相关功能及关键流程。



图 1.1 总体架构图

5. 基础能力要求

5.1 算力资源

5.1.1 集群管理

5.1.1.1 提供集群管理能力

- a) 系统应提供集群的创建能力，可将资源添加到某一集群中进行统一管理。
- b) 系统应提供查看、修改已创建的集群信息的能力，信息需包括集群当前已纳管的资源信息等。
- c) 系统应提供增加集群、删除集群及集群已经纳管的资源的能力。
- d) 系统应提供在集群下动态增加、移除节点的能力。

5.1.1.2 应提供资源管理能力

- a) 系统应具备管理员能够将算力资源注册到集群中进行纳管，并支持查看、修改资源信息和清除资源注册的能力。
- b) 系统应具备管理员能够将算力资源进行补充、扩容、缩容的能力。
- c) 系统应具备管理员能够将算力资源进行删除的能力，如算力资源已承载业务，需进行清理或迁出后才允许删除。

5.1.1.3 提供对集群的人工配置或自动化管理能力

- a) 系统应提供配置文件或操作命令，使得访问系统的人或程序可通过调整配置，达到算力资源异构化或差异化管理的能力。

5.1.2 数据存储

5.1.2.1 具备数据库管理能力；

- a) 对数据库数据，应具备能够确保数据至少保持两份一致性的拷贝的能力。
- b) 系统应具备经过用户认证后，对数据库进行添加、删除、查看的能力。
- c) 数据库应支持根据地域、规格等创建数据实例的能力；
- d) 应支持数据库只读实例的创建；
- e) 应支持数据库实例名称的修改和备注能力；
- f) 应支持对数据库实例规格和存储空间变更；

- g) 应提供数据库主备切换、备份的能力；
- h) 应具备数据库用户管理的能力，包括但不限于创建用户、查询用户列表、删除用户、重置密码、授权和取消授权等；

5.1.2.2 具备数据操作管理的能力；

- a) 应提供数据上传、下载、列举、拷贝、重命名、分享、搜索、删除的能力；
- b) 应提供数据的目录、数据索引的能力；
- c) 应提供数据误删恢复、限制访问、敏感数据保护、加密的能力；
- d) 应提供数据的跨区域复制、同区域复制、定时备份等能力；
- e) 应具备对数据库的人工配置或自动化管理能力。

5.1.3 网络资源

网络传输能力要求如下：

- a) 应具备按照 ipv4、ipv6 协议传输的能力；
- b) 应具备按照弹性 ip 协议传输的能力；
- c) 应支持申请、查询、删除、修改弹性 ip 基础信息、带宽的能力；
- d) 应具备通过专线传输的能力；
- e) 应支持创建、修改、删除、查询云专线的能力；
- f) 应具备在网关之间传输的能力；
- g) 应支持网关间配置、删除、修改、查询的能力；

6. 多云接入

6.1 云容器管理

多云容器平台通过集群联邦实现对多个云运营商的集群进行统一管理，支持集群跨地域统一部署、统一发布及统一运维。基于多集群与联邦技术，多云容器平台可以实现多个不同区域、不同云的容器集群管理，支持统一的全局应用管理。

a) 多云容器平台应支持跨集群的应用弹性伸缩策略，用以均衡各集群的应用实例分布，实现全局负载均衡。

b) 多云容器平台应支持集群节点的水平扩展，多云容器平台将根据应用的负载情况，轻松灵活的自动扩缩容应用所需的资源。

c) 多云容器平台应支持创建联邦服务。

d) 应支持跨集群的服务发现机制，能够基于服务就近访问原则实现业务的区域亲和，从而在业务进行多区域部署时，实现访问与服务的同地域优先，降低用户访问时的网络延迟。

e) 多云容器平台兼容 Kubernetes 社区最新 Federation 架构，提供原生 Kubernetes API 及 Karmada API。

f) 应支持 Kubernetes 命令行和 API 来部署容器应用，无需修改任何服务代码即可支持 Kubernetes 典型应用场景。

g) 多云容器平台应支持提供混合云、多云、多区域三种类型的组网方案，

h) 应支持将云上 Kubernetes 集群和自建的 Kubernetes 集群添加到多云容器平台中进行统一管理，混合云组网方案应支持以下能力：

- 云上统一管理：在云上实现资源、应用、网络等多云统一管理。
- 云上伸缩应用：当业务出现突发流量高峰时，实现云上资源的快速扩容。
- 云上数据无关：核心数据放在线下，公有云上运行与核心数据无关的应用。

- 业务迁移过渡：合理利用原有的物理计算资源，将业务逐步平稳切换到公有云上，实现资源的最大化利用。
- i) 多云方案应支持以下能力：
 - 统一管理：实现资源、应用、网络等多云统一管理。
 - 可扩展性：比单集群更高的横向扩展性，解决单集群管理容量受限的问题。
 - 智能路由：多集群分布在不同区域，实现地域亲和的请求转发，降低用户因跨区域访问造成的网络延迟。
 - 多云容灾：相同业务部署到不同云运营商的集群，实现多云容灾。
- j) 多区域组网方案应支持以下能力：
 - 统一管理：实现资源、应用、网络等多云统一管理。
 - 可扩展性：比单集群更高的横向扩展性，解决单集群管理容量受限的问题。
 - 智能路由：多集群分布在不同区域，实现地域亲和的请求转发，降低用户因跨区域访问造成的网络延迟。
 - 多区域容灾：相同业务部署到不同区域的集群，实现多云容灾。

6.2 云容器注册

集群建立联邦后，可以使用集群管理 API 来管理多个集群资源。应具备：

- a) 配置需要在集群中传播的 api 资源；
- b) 配置 api 资源如何分配到不同的集群；
- c) 对集群中 dns 记录注册；
- d) 在多个集群之间分散工作负载(容器)，以提升应用(服务)的可靠性；
- e) 在不同集群中，能更快速更容易地迁移应用(服务)；
- f) 跨集群的服务发现，服务可以就近访问，以降低延迟。

6.3 多云调度

6.3.1 多云调度应支持高效容灾管理

- a) 注册并更新应用的域名记录至全局 dns；
- b) 支持故障时访问流量自动切换到可用集群；
- c) 应用实例自动弹性到其他云上，实现业务的容灾管理；
- d) 支持应用部署在不同地域的数据中心的集群中，实现地域亲和的请求转发，有效降低业务时延；
- e) 支持跨云应用迁移能力，通过多云容器平台，可以对单云的资源进行一键式跨云迁移，实现应用的数据保护。

6.3.2 多云调度支持跨云弹性伸缩能力

- a) 灵活配置跨集群的应用弹性伸缩策略；
- b) 采集资源的使用情况（cpu 或者内存、自定义指标项），动态调整负载实例数，支持跨云应用容量治理能力；
- c) 结合服务网格技术打通多集群网络，实现多云/混合云应用流量治理，灵活应对业务流量突发，保障业务的稳定可靠。
- d) 多云调度结合服务网格技术打通多集群网络，实现多云/混合云应用流量治理，灵活应对业务流量突发，保障业务的稳定可靠；
- e) 支持跨云统一监控运维能力，提供多云应用统一视图，以应用为中心，拉通各个跨云集群进行一站式分析。

7 控制管理

7.1 算力接入

7.1.1 算力注册

7.1.1.1 概述

算力注册为底层算力服务厂商提供与多云调度系统的对接能力，其目的在于采集底层算力的基础信息、资源信息和服务信息等，以及子系统的 API 信息等，从而为任务、资源管理等多云调度能力提供支撑。

7.1.1.2 注册管理

为算力供应方提供向导式的注册能力，通过流程引导、信息填报等方式，采集算力资源的相关信息。

- a) 应支持基本信息填报，包括资源的系统类型、厂家类型、地理位置等信息；
- b) 应支持资源信息填报，包括提供的算力版本、异构算力信息、可用规格等；
- c) 应支持系统能力填报，包括提供的系统能力，如 IaaS、PaaS、SaaS 以及承载的应用服务系统、软件系统等；
- d) 应支持网络策略填报，包括多云调度系统与底层算力对接所需的网络策略信息，如 VPN 等。

7.1.1.3 API 管理

为算力供应方提供相关子系统的 API 注册能力，包括资源、服务、权限、计费等。

- a) 应为用户提供模板化 API 信息填报能力，包括基础接口、参数映射等；
- b) 应按照多云调度能力需求，将 API 等级分为必填、可选等；
- c) 应保证 API 接入系统的可扩展性，支持 API 版本升级等。

7.1.2 算力认证

算力认证应支持对算力供应方提供的基础信息进行校验，对算力资源的连通性、资源和服务的真实性和基础信息的合法性进行校验。包括：

- a) 应对多云调度系统与底层算力对接的连通性进行校验。
 - 应对多云调度系统与底层算力网络连通性进行校验；
 - 应根据用户录入 API 信息，对 API 接口的连通性进行校验。
- b) 应对用户填报数据、API 接口等进行校验。
 - 应支持按照规则、三方数据等对用户录入算力信息的校验能力；
 - 应支持对用户录入 API 信息，包括参数完整性、合法性、参数与系统需求符合度等校验能力。

7.1.3 算力标识

算力标识为各个底层算力资源分配标识，该标识应当满足唯一性、层次性、完整性需求。其中，唯一性指多云调度系统内各个算力的标识不重复，层次性指算力标识体系应当按照标识范围由大到小组织，完整性指算力标识内各个字段应当完整，且能够充分代表算力资源的基础特征。算力标识应包括资源标识和服务标识，包括：

- a) 应支持按照层级生成包括基本信息、算力类型等的标识能力，如国家-区域-省份-地市-算力类型-供应商-容量等级等；
- b) 应支持按照层级生成对算力服务的基本标识能力，如 IaaS/PaaS/SaaS-业务领域-服务名称-服务版

本等；

- c) 应支持将资源标识、服务标识进行组织，并定义编码规则从而生成针对算力的唯一标识，如将资源标识和服务标识按照层级生成树结构后按照深度优先形式进行编码；
- d) 应支持对资源标识的快速检索能力。

7.2 算力感知

7.2.1 感知接入

感知接入一方面通过 API 访问等形式，对底层算力集群、节点、容器、网络、存储以及算力承载的服务应用等进行感知监控、存储和分析，另一方面与上层业务逻辑层对接，感知多云调度系统的整体服务质量。包括：

- a) 应支持对底层算力资源的状态、静态配置和动态利用率等信息进行感知：
 - 应支持对算力节点的连通性、稳定性等的感知；
 - 应支持对算力节点的资源总量、异构资源占比、网络配置、存储总量等配置信息等的感知；
 - 应支持对算力节点的计算、存储、网络等资源动态利用率的感知。
- b) 应支持对底层算力承载的服务状态、服务占用资源量等的感知。
- c) 应支持对底层算力的服务服务质量感知，包括资源开通成功率、服务时延、服务稳定性的感知。

7.2.2 感知管理

感知管理应包括感知存储、感知分析、感知共享和感知配置能力，包括：

- a) 应支持对感知数据的存储和生命周期管理，支持结构化、非结构化等数据的存储以及按照时间、业务属性等不同生命周期策略。
- b) 应支持对感知数据的加工和分析处理能力，包括：
 - 应支持对原始感知数据的加工能力，包括非结构化数据的信息抽取、异常数据清洗、数据预处理等；
 - 应支持内嵌规则策略、算法模型等，支持异常检测等能力。
- c) 应支持以 API 接口、数据库开放等形式，向算力编排以及业务逻辑层开放感知数据接口。
- d) 应支持对算力感知的采集策略、采集周期、数据解析规则等的配置能力。

7.2.3 算力度量

算力度量应基于算力感知数据，对底层算力、节点、服务进行度量，通过规则定义、数据加权等，将感知数据转化为统一的度量体系和分级标准，从而为算力编排提供数据依据，包括：

- a) 应支持对硬件计算能力的度量，应支持对通用计算(Flops)、异构加速卡(GPU: Flops/Ops; FPGA: 逻辑门)等的计算能力以及能耗等的度量；
- b) 应支持通过 Benchmark、理论模型等，综合考虑计算、通信、存储等能力的算力节点度量；
- c) 应支持通过 Benchmark、理论模型等，综合考虑跨节点通信、服务平台等的集群算力度量；
- d) 应支持定义分级标准，如根据算力容量大小，将算力资源分为 I,II,III 等多个级别。

8 算力编排

8.1 资源管理能力

8.1.1 需求转换

需求转换面向用户业务需求，应具备从业务术语转换为资源需求的能力，包括：

- a) 对于用户通过表单等提供算力资源需求的情况，应具备对用户需求规范化的能力，转换为对算力、网络、存储、服务的需求；
- b) 对于用户通过业务术语提供需求的情况，应具备需求分析能力，将用户时延、成本、安全等转换为对算力、网络、存储、服务的需求。

8.1.2 资源编排

8.1.2.1 概述

资源编排应支持将需求转换等得到的大颗粒资源需求按照底层算力资源情况进行拆解，与资源编排模板匹配，生成满足用户业务需求的资源编排模板。

8.1.2.2 需求拆解

应支持对用户资源需求的拆分能力，包括：

- a) 算力拆解：应支持满足用户总算力需求情况下，将大颗粒算力需求拆解为多个计算节点或多个存储需求；
- b) 网络拆解：应支持根据用户需求，将用户带宽、时延需求拆解为具体的接入、核心、数据中心内等网络配置能力；
- c) 其他拆解：应支持根据用户需求和整体资源配置规则，将存储、内存、安全策略等进行拆分能力。

8.1.2.3 编排策略

应支持将拆解后的业务需求与编排系统模板进行匹配，生成可执行的编排策略。

- a) 模板匹配：应支持按照关键字、业务场景等进行模板检索和匹配能力；
- b) 模板管理：应支持编排模板的发布、版本、删除等管理能力。

8.1.3 资源调度

资源调度应支持根据资源编排策略，实现资源开通等调度能力，同时具备迁移、恢复等调度异常处理能力。

- a) 调度执行：应支持与底层算力资源对接，实现资源镜像下发、资源预留、资源开通、资源配置、扩缩容、资源销毁等调度能力。
- b) 调度管理：应支持在多任务情况下的调度队列管理能力，包括调度优先级等。
- c) 调度容错：应支持面向调度流程和资源的异常处理能力；
- d) 调度异常处理：应支持在调度失败、调度拥塞、调度冲突等情况下，进行算力节点和集群的重选、调度策略回退以及重启等能力；
- e) 资源异常处理：应支持在算力、网络、存储等资源异常情况下，自动进行资源销毁、应用迁移和恢复能力。

8.1.4 资源监控

资源监控应支持以算力感知能力为基础，对资源状态进行监控，同时与业务和调度系统对接，应具备：

- a)资源监控：应具备对已接入算力资源进行利用率、算力异常等的监控能力；
- b)业务对接：应支持与上层业务逻辑层对接，实现告警触发、事件报表等能力；
- c)调度对接：应支持与调度系统对接，对异常资源进行调度容错处理；
- d)监控管理：应具备对监控策略、时间间隔、触发策略等的配置能力。

8.1.5 资源运维

资源运维应支持对纳管的资源进行集群、节点、服务、安全等方面的管理，包括：

- a)集群运维：应支持对集群的剔除、下线、冻结、降低调度优先级等运维能力；
- b)节点运维：应支持对集群内算力节点以及相应虚拟机、容器等进行配置变更、下线、亲和度调整等操作；
- c)应用运维：应支持对集群内服务的下线、冻结、重启等运维能力；
- d)安全运维：应支持对算力节点接入的安全策略、算力集群内安全策略以及应用安全等运维能力。

8.2 任务管理能力

8.2.1 任务拆解

任务管理承接业务逻辑层提供的业务需求和服务质量需求，应支持：

- a)任务拆分：应支持按照任务间的依赖关系、子任务的数据需求等进行任务拆分，以满足数据传输、应用部署环境以及逻辑依赖关系等；
- b)资源映射：应支持将拆分后的子任务集与资源管理能力对接，进行算力资源开通或与已开通的算力资源进行映射。

8.2.2 任务调度

任务调度应支持按照任务拆分后的模板以及与底层资源的映射关系，实现应用的调度、管理、容错等，包括：

- a)调度执行：应支持与底层算力资源对接，实现任务镜像下发、环境部署、任务拉起、任务生命周期管理等。
- b)调度管理：应支持在多任务情况下的调度队列管理能力，包括调度优先级等。
- c)调度容错：应支持面向调度流程和任务的异常处理能力；
- d)调度异常处理：应支持在调度失败、调度拥塞、调度冲突等情况下，进行调度策略回退以及重启等能力；
- e)任务异常处理：应支持在任务异常情况下的任务迁移、任务重启等能力，同时支持异地双活等高可用能力；
- f)流量控制：应支持任务服务中负载均衡等流量控制能力。

8.2.3 任务监控

任务监控应支持以算力感知能力为基础，对任务的运行状态进行监控，同时与业务和调度系统对接，包括：

- a)任务监控：应具备对已调度任务的运行状态、资源占用情况、服务质量等的监控能力；
- b)业务对接：应支持与上层业务逻辑层对接，实现告警触发、事件报表等能力；
- c)调度对接：应支持与调度系统对接，对异常任务进行调度容错处理；
- d)监控管理：应具备对任务监控策略、时间间隔、触发策略等的配置能力。

9 可视化层

9.1 可视化能力

9.1.1 资源可视化

资源可视化能力要求如下：

- a) 应具备算力资源可视化的能力，包括但不限于资源统计、算网路径等；
- b) 应具备网络资源可视化的能力，包括但不限于网络信息、拓扑信息等；
- c) 应具备基础设施可视化的能力，包括但不限于基本状态、损耗情况等；
- d) 应具备人力资源可视化的能力；

9.1.2 运维可视化

运维可视化能力要求如下：

- a) 应具备算力资源使用情况报表可视化的能力；
- b) 应具备网络资源监控报表可视化的能力；
- c) 应具备故障分析可视化报表的能力；
- d) 应具备日志检索、统计、分析可视化的能力；

9.1.3 运营可视化

运营可视化能力要求如下：

- a) 应具备算力资源使用情况报表可视化的能力，包括但不限于调度平台整体情况、资源分配量等；
- b) 应具备使用方管理情况可视化的能力，包括但不限于人员管理、角色管理等；
- c) 应具备运营情况统计分析报表可视化的能力；
- d) 应具备产品管理中心可视化能力，包括但不限于产品展示、产品查询、产品推荐、订单查询（订单编号、订单类型、资源类型、订单状态等）等。

10 安全

10.1 资源鉴权

10.1.1 算力资源鉴权

应具备对算力资源鉴权的能力：

- a) 对算力资源应有单独的算力鉴权服务进行鉴权管理。
- b) 对算力资源的管理操作，包括资源的创建、修改、删除、查询等行为，均应在鉴权操作完成后才允许操作。如果鉴权失败，应禁止操作。
- c) 禁止操作其他用户下未授权算力资源。若进行操作，应提示无权限。
- d) 应提供用户对算力资源只开放部分操作的鉴权能力，允许授予创建、修改、删除、查询等操作部分权限，当用户对算力资源进行未授权操作时，应提示无权限。

10.1.2 网络资源鉴权

应具备对网络资源鉴权的能力：

- a) 对网络资源应有单独的网络鉴权服务进行鉴权管理。
- b) 对网络资源的管理操作，包括资源的创建、修改、删除、查询等行为，均应在鉴权操作完成后才允许操作，如果鉴权失败，应禁止操作。
- c) 禁止操作其他用户下未授权网络资源，若进行操作，应提示无权限。
- d) 应提供用户对网络资源只开放部分操作的鉴权能力，允许授予创建、修改、删除、查询等操作部分权限，当用户对网络资源进行未授权操作时，应提示无权限。

10.2 身份鉴权

身份鉴权能力要求如下：

- a) 应具备使用方身份鉴权的能力：

对于算力资源系统使用方，需具备用户身份认证功能，访问系统的人或程序必须要先向鉴权服务进行身份认证后才能访问系统，认证内容包括用户名、密码、验证码等，认证失败应显示认证失败的提示；认证通过后，访问系统的人或程序，可对分配的算力资源进行创建、修改、删除、查询等管理操作。操作执行前需对操作行为做权限认证，执行未授权操作时，应提示无权限。

- b) 应具备资源提供方身份鉴权的能力：

对于算力资源提供方，需具备提供方身份认证功能，访问系统的人或程序必须要先向鉴权服务进行身份认证后才能访问系统，认证内容包括用户名、密码、验证码等，认证失败应显示认证失败的提示；认证通过后，访问系统的人或程序，可对提供的算力资源进行创建、修改、删除、查询等管理操作。操作执行前需对操作行为做权限认证，执行未授权操作时，应提示无权限。

参 考 文 献

索 引