**ICS**

**中国建筑业协会团体标准 团体标准**

**P**  **T/CCIAT xxxx—20xx**

**工程测控云边端系统架构技术规程**

Technical Specification for Engineering Measurement and Control Cloud Edge System Architecture

**（征求意见稿）**

**20xx— xx—xx 发布　　　20xx—xx —xx 实施**

**中国建筑业协会 发布**

中国建筑业协会团体标准

**工程测控云边端系统架构技术规程**

Technical Specification for Engineering Measurement and Control Cloud Edge System Architecture

**T/CCIAT xxxx—20xx**

批准部门：中国建筑业协会

施行日期：20xx年x月x日

中国建筑工业出版社

**20xx 北京**

前  言

根据中国建筑业协会《关于印发<第八批中国建筑业协会团体标准编制工作计划>的通知》（建协函[2023]54号）的要求，标准编制组经过深入调查研究，认真总结工程实践经验，参考了国内外相关技术标准，进行了必要的理论研究和验证试验，并在广泛征求意见的基础上，制订本标准。

本规程共主要技术内容：1.总则、2.术语、3.云边端一体化管理机制的规定、4.系统架构、5.业务流程模型、6.总体要求。

本规程由中国建筑业协会负责管理，由中建工程产业技术研究院有限公司负责具体技术内容的解释。请各单位在执行过程中，总结实践经验，积累资料，随时将有关意见和建议反馈给中建工程产业技术研究院有限公司（地址：北京顺义林河大街15号；邮政编码：101300）。

本标准主编单位：中建工程产业技术研究院有限公司

中国建筑第三工程局有限公司

本标准参编单位：×××、×××

本标准主要起草人员：×××、×××

本标准主要审查人员：×××、×××

**目 次**

1 总则 1

2 术语 2

3 基本规定 5

4 云边端一体化管理机制的规定 7

4.1 一般规定 7

4.2 工作流程的标准规定 7

4.3 协同机制的标准规定 10

4.4 各方职责的标准规定 11

4.5 边端轻量化算法部署规定 13

5 系统架构 16

5.1 一般规定 16

5.2 框架结构规定 16

5.3 协同架构设计 21

5.4 系统运维管理 25

5.5 系统安全管理 27

5.6 系统数据管理 28

6 业务流程模型 32

6.1 一般规定 32

6.2 数据采集与处理流程 33

6.3 控制指令下发与执行流程 34

6.4 系统故障处理与恢复流程 36

6.5 数据存储与管理流程 38

6.6 用户交互流程 40

6.7 实施步骤 41

7 总体要求 44

7.1 一般规定 44

7.2 数据源采集、存储与传输要求 45

7.3 备份和恢复要求 45

7.4 故障恢复与迁移要求 46

7.5 兼容性要求 46

7.6 可扩缩性要求 47

本规程用词说明 48

引用标准名录 49

**Contents**

**1 General Provisions 1**

**2 Terminology 2**

**3 Basic regulations 5**

**4 Regulations on the Integrated Management Mechanism of Cloud Edge and End 7**

4.1 General provisions 7

4.2 Standard provisions for workflow 7

4.3 Standard provisions for colaborative mechanisms 10

4.4 Standard provisions for responsibilities of all parties 11

4.5 Deployment regulations for edge lightweight algorithms 13

**5 System Architecture 16**

5.1 General provisions 16

5.2 Framework Structure Regulations 16

5.3 Collaborative Architecture Design 21

5.4 System operation and maintenance management 25

5.5 System Security Management 27

5.6 System Data anagement 28

**6 Business Process Models 32**

6.1 General provisions 32

6.2 Data collection and processing process 33

6.3 Control command issuance and execution process 34

6.4 System Fault Handling and Recovery Process 36

6.5 Data Storage and Management Process 38

6.6 User interaction process 40

6.7 mplementation steps 41

**7 Verall requirements 44**

7.1 General provisions 44

7.2 Data source storage and distribution requirements 45

7.3 Backup and Recovery Requirements 45

7.4 Fault recovery and migration requirements 46

7.5 Compatibility requirements 46

7.6 Scalability requirements 47

**Explanation of Vocabulary in This Regulation 48**

**List of reference standards 48**

# 1 总则

### **1.0.1** 为规范工程测控数据采集，确保数据采集时在管理机制、系统架构、业务流程模型、数据管理等方面具有统一性，减少工程测控数据的孤岛效应，制定本规程。

### **1.0.2** 本规程适用于指导工程测控云边端系统架构的规划、设计、实施、运行和维护。

### **1.0.3** 本规程规定了工程测控云边端系统架构的通用技术要求，当本规程与国家法律、行政法规的规定相抵触时，应按国家法律、行政法规的规定执行。

# 2 术语

### **2.0.1** 云平台 cloud platform

基于云计算技术构建的一种基础设施和服务集合体。

整合了大量的计算资源（包括服务器、存储设备、网络设备等）、软件资源（如操作系统、数据库管理系统、中间件等）以及应用服务资源，通过虚拟化技术将这些资源进行池化管理，以按需分配的方式向用户或其他系统提供弹性的计算能力、存储能力、网络能力以及各种软件应用服务。

### **2.0.2** 中心云 central cloud

云边端架构中位于核心位置、具备强大计算和存储能力的云计算资源集中地，与边缘云计算相对。

### **2.0.3** 边缘云 edge cloud

部署在靠近数据源或用户端的分布式云计算资源，并能够支持与中心云计算协作，在靠近事物、数据源头的网络边缘侧，提供可弹性扩展的云服务能力，具有低延时、大带宽、多连接等服务特点。

### **2.0.4** 边缘基础设施 edge cloud infrastructure

网络边缘侧部署边缘云的基础设施，包括但不限于：分布式IDC、运营商通信网络边缘基础设施、边缘侧客户节点（如边缘网关、家庭网关等）等边缘设备及对应的网络环境。

### **2.0.5** 边缘云节点 edge cloud node

基于边缘云基础设施，部署了边缘云计算平台，具有进行计算、存储、网络、加速等数据处理能力的节点资源。

### **2.0.6** 端 terminal

直接与物理世界交互的设备或终端，数据采集和指令执行的最前端。包括各种类型的各种物联网设备，如手机、传感器（如温度传感器、压力传感器、位移传感器等）、摄像头等和执行器（如电机、阀门、继电器等）等。

### **2.0.7** 云边协同 cloud-edge collaboration

云端与边缘端之间的数据交互、任务分配与协同工作机制。

### **2.0.8** 边边协同 edge-edge collaboration

多个边缘计算节点之间相互通信、共享数据与资源，进行联合数据处理与决策优化的协同模式。

### **2.0.9** 工程测控 engineering Measurement and Control

工程测控是在工程全生命周期内，针对工程相关物理量、参数等开展测量、监测、控制与数据处理工作。它通过各类传感器采集数据，如工程结构应力应变、温湿度、位移等，经传输、存储后，依控制策略调控工程系统，实时掌握工程状况，为工程安全、质量、优化与决策提供数据支撑与控制手段，保障工程正常运行并预警异常。

### **2.0.10** 物联网 internet of things

指通过各种信息传感器、射频识别技术、全球定位系统、红外感应器、激光扫描器等各种装置与技术，实时采集任何需要监控、连接、互动的物体或过程的声、光、热、电、力学、化学、生物、位置等各种需要的信息。

# 3 基本规定

### **3.0.1** 规程涵盖了云计算平台、边缘计算节点和终端设备之间的协同工作，还涉及数据采集、处理、传输、存储及安全控制等多个方面。

### **3.0.2** 规程的适用领域包括但不限于工业自动化、智能制造、智慧城市等。

### **3.0.3** 系统架构设计应遵循开放性、可扩展性、安全性、可靠性等基本原则，以适应不断变化的工程测控需求与技术发展。

### **3.0.4** 系统架构设计原则应遵循下列规定：

**1** 云端：作为核心中枢，负责海量数据存储、全局分析、模型训练（如BIM优化、工程仿真）、跨项目协同及长期决策支持。

**2** 边缘层：部署于工地现场或区域服务器，处理实时数据（如传感器监测、施工机械状态）、本地化计算（如结构安全预警）及低延迟响应。

**3** 终端层：包括物联网设备（如传感器、无人机、智能机械）、移动终端（如工人手持设备），负责原始数据采集与指令执行。

**4** 云-边-端三层需动态协同，云端侧重宏观优化，边缘层解决实时性问题，终端保证数据精准采集。

### **3.0.5** 系统数据设计应符合全生命周期管理，具体符合下列规定：

1 采集终端设备需满足工程级精度（如毫米级位移监测）与抗干扰能力（如高温、高湿环境）。

2 应根据数据类型选择适应的协议，实时控制指令（如塔吊防碰撞）采用低时延5G/TSN，非关键数据（如环境监测）可异步传输。

3 云端采用分布式数据库管理长期数据，边缘层缓存短期高频数据。

4 工程应用应建立统一数据模型，支持多参与方（设计、施工、监理）按权限访问。

# 4 云边端一体化管理机制的规定

## 4.1 一般规定

### **4.1.1** 云边端一体化管理机制旨在确保云计算、边缘计算和终端设备之间的顺畅协同，实现高效的数据处理、资源分配和服务提供。

### **4.1.2** 云端构建任务调度与管理平台，对整个工程测控系统的任务进行统一规划与分配。

### **4.1.3** 边端连接各类传感器与数据采集设备，负责数据的采集、预处理、传输、与其他边端设备数据共享与任务协作。

### **4.1.4** 终端设备直接负责数据的采集，将其转换为数字型号。

### **4.1.5** 采用加密技术、访问控制技术、入侵检测与防御技术等，保障系统免受外部攻击与内部数据泄露风险。

## 4.2 工作流程的标准规定

### **4.2.1** 数据采集应符合下列规定：

**1** 边缘端设备依据工程测控需求确定采样频率、精度等参数，启动数据采集任务。

**2** 在采集过程中，对传感器传来的原始模拟信号进行模数转换，将其变为数字信号后，按照预定义的数据格式进行初步整理。

### **4.2.2** 边缘处理应符合下列规定：

**1** 边缘节点接收数据后，根据预设的算法进行实时处理或预处理。

**2** 将采集到的数据进行进一步的清洗和筛选，去除噪声数据以及与当前测控任务关联度较低的数据，以减轻后续处理压力并提高数据质量。

**3** 可进行数据融合处理，将来自不同传感器或不同数据源但相互关联的数据进行整合，以获取更全面准确的信息。

### **4.2.3** 数据传输应符合下列规定：

**1** 边缘端在完成数据采集与初步处理后，根据云端设定的数据上传策略，选择合适的通信协议（如 MQTT、HTTP 等）与云端建立连接。

**2** 应将处理后的数据进行压缩，以减少传输带宽占用，再使用加密算法（如 AES、RSA 等）对数据进行加密，保障数据在传输过程中的安全性。

**3** 应按照设定的时间间隔或事件触发条件（如数据量达到阈值）将数据上传至云端指定的接收接口。

### **4.2.4** 云端处理应符合下列规定：

**1** 云端接收到边缘端上传的数据后，首先进行解密与解压缩操作，恢复数据原始格式。

**2** 根据数据类型与业务需求，将数据存储至相应的数据库或数据仓库中。针对存储的数据，开展深度分析工作，如运用数据挖掘算法发现数据中的潜在模式与关联关系，通过机器学习模型进行预测性分析等。

**3** 基于海量数据进行模型训练，不断优化模型参数，以提升模型的准确性与泛化能力。

**4** 根据分析结果与模型输出，结合工程整体目标与预设策略，生成控制指令或优化方案。

### **4.2.5** 指令下发应符合下列规定：

**1** 云端生成控制指令或优化方案，按照预定的通信协议进行封装与加密处理，通过安全可靠的网络通道，将指令下发至对应的边缘端设备。

**2** 边缘端在接收到指令后，先进行解密与验证操作，检查指令的完整性、合法性以及来源的可靠性；验证通过后，对指令进行解析，提取出具体的控制参数与操作要求，并执行相应的控制动作，如调整设备的运行参数、启动或停止特定设备等。

### **4.2.6** 监控与优化应符合下列规定：

**1** 监控与优化流程贯穿整个工程测控云边端系统的运行过程，由监控系统进行实时监控，根据系统性能、资源利用率等指标进行动态优化。

**2** 在边缘端，一方面对自身设备的运行状态进行实时监控，包括设备的 CPU 使用率、内存占用、网络连接状况等；另一方面对数据采集和处理过程进行监控，统计数据采集的成功率、数据处理的耗时等指标。

**3** 在云端，对边缘端上传的数据质量进行监控，如数据的完整性、准确性、时效性等；同时，云端还对整个系统的资源利用情况进行监控，包括云端计算资源、存储资源以及边缘端资源的分配与使用情况，根据监控结果动态调整资源分配策略。

## 4.3 协同机制的标准规定

### **4.3.1** 云边数据协同应符合下列规定：

**1** 云端建立数据目录与元数据管理系统，对各类数据进行分类整理与标识，方便边缘端按需查询与获取。

**2** 边缘端根据自身的任务需求与数据处理状况，向云端发送数据请求，云端依据请求内容与权限设置，筛选并返回相应的数据。同时，边缘端在本地数据处理过程中，将产生的关键数据或处理结果数据按照规定的格式与频率上传至云端，实现数据的反向流动，以便云端进行全局数据整合与分析。

### **4.3.2** 云边任务协同应符合下列规定：

**1** 根据边缘端的计算资源、网络状况以及任务负载情况，将不同类型的任务（如数据处理任务、控制任务、模型训练任务等）合理地分配至边缘端执行。

**2** 在任务执行过程中，云端实时监控任务进度与边缘端资源使用情况，根据预设的规则与策略进行任务调整与资源优化。

### **4.3.3** 边边数据共享应符合下列规定：

**1** 边缘端之间建立对等的数据共享网络架构，基于分布式哈希表（DHT）或消息队列等技术实现数据的快速定位与传输。

**2** 在数据共享过程中，边缘端根据预先设定的数据共享策略与权限规则，确定可共享的数据范围与共享对象。

**3** 为保障数据共享的安全性，采用数据加密与身份认证技术，防止非法数据访问与数据泄露。

### **4.3.4** 边边任务协作应符合下列规定：

**1** 多个边缘端针对复杂的工程测控任务建立协作关系，通过任务分解与分配机制，明确各边缘端在任务中的角色与职责。

**2** 在任务执行过程中，边缘端之间通过消息传递或事件触发机制实时交互任务执行状态与数据信息，根据任务进展情况进行动态调整与协作配合。如当某个边缘端检测到异常数据时，及时通知其他相关边缘端协同进行进一步的监测与分析，共同完成对工程异常情况的快速定位与处理。

## 4.4 各方职责的标准规定

### **4.4.1** 云端职责应符合下列规定：

**1** 构建大规模数据存储基础设施，如数据中心、云数据库等，对工程测控全生命周期的数据进行持久化存储。

**2** 建立完善的数据管理机制，包括数据分类、索引、备份、恢复等操作，确保数据的安全性、完整性与可访问性。

**3** 运用大数据分析技术、数据挖掘算法以及机器学习模型等，对海量的工程测控数据进行深度分析。

**4** 挖掘数据中的潜在规律、趋势与异常模式，为工程的优化设计、故障预测、性能评估等提供数据支持与决策依据。

**5** 基于大规模数据开展模型训练工作，如神经网络模型、回归模型等，用于工程测控中的数据预测、状态评估、控制策略优化等任务。

**6** 不断根据新的数据样本与业务需求，对模型进行优化与更新，提高模型的准确性与适应性。

**7** 负责整个工程测控云边端系统的配置管理工作，包括云端与边缘端的系统参数设置、软件版本管理、网络配置等。

**8** 制定系统运行策略与规则，如数据上传下载策略、任务调度策略等，确保系统的稳定运行与高效协同。

**9** 制定全面的系统安全策略，涵盖数据安全、网络安全、设备安全等多个方面。

### **4.4.2** 边缘端职责应符合下列规定：

**1** 通过连接各类传感器与数据采集设备，按照预定的采样频率与精度要求，采集工程现场的原始数据。

**2** 对采集到的数据进行即时的预处理工作，包括数据清洗、格式转换、特征提取等操作，去除噪声数据与冗余信息，将原始数据转换为可供本地分析与传输的有效数据格式。

**3** 依据本地数据处理结果以及云端下发的控制策略与指令，在边缘端直接执行本地控制决策。

**4** 建立与云端及其他边缘端的稳定通信连接，遵循统一的通信协议与数据格式规范。

**5** 负责将本地处理后的数据上传至云端，接收云端下发的控制指令、模型参数更新等信息，并及时反馈本地的任务执行情况与设备状态信息。

**6** 与其他边缘端进行数据共享与任务协作，实现分布式测控任务的协同完成。

### **4.4.3** 终端职责应符合下列规定：

**1** 终端设备中的传感器负责直接感知工程对象的物理量、化学量或状态变化，并将其转换为电信号或数字信号。

**2** 终端设备中的执行器接收来自边缘端或云端的控制指令，并将其转换为实际的物理动作。

## 4.5 边端轻量化算法部署规定

### **4.5.1** 算法选择应符合下列规定：

**1** 根据边缘端的计算资源（如 CPU 性能、内存容量等），选择计算复杂度较低的算法。

**2** 选择内存占用量小的算法模型与数据结构，优化算法代码实现，减少不必要的内存分配与变量存储，提高边缘端内存资源的利用率。

**3** 考虑工程测控任务的实时性需求，选择能够快速响应与处理数据的算法。

**4** 依据工程测控所涉及的数据类型、分布特点等选择合适算法。

### **4.5.2** 算法部署应符合下列规定：

**1** 将选定的轻量化算法代码从开发环境或云端传输至边缘端设备。采用安全可靠的传输协议（如 SFTP、HTTPS 等），确保算法代码在传输过程中不被篡改或窃取。在传输前，对算法代码进行压缩处理，减少传输带宽占用与传输时间。

**2** 边缘端接收到算法代码后，进行解压与安装操作。根据边缘端的操作系统与运行环境，配置算法所需的依赖库、运行参数等。

**3** 在边缘端完成算法安装与配置后，进行算法测试与验证工作。使用模拟数据或真实的工程测控数据样本对算法进行功能测试，检查算法是否能够正确处理数据、输出符合预期的结果。

**4** 对算法的性能（如计算时间、内存占用等）进行评估，与设计要求进行对比验证。若发现算法存在问题或性能不达标，及时进行调试与优化，直至算法在边缘端能够稳定、高效地运行。

### **4.5.3** 算法优化应符合下列规定：

**1** 边缘端算法在运行过程中，依据本地持续采集的数据特性与变化趋势进行优化。当发现数据分布发生明显变化时，如均值或方差改变，相应调整算法内部参数或模型结构。

**2** 以算法在边缘端运行的性能指标为导向，如计算效率、内存使用峰值、数据处理准确率等。

**3** 充分考虑边缘端本地资源的动态变化情况，包括 CPU 负载、可用内存、网络带宽等。

### **4.5.4** 算法更新应符合下列规定：

**1** 云端基于对整个工程测控系统数据的全面分析、模型性能评估以及新的业务需求，决策是否需要对边缘端算法进行更新。

**2** 确定更新后，云端将新的算法代码、更新说明以及相关配置信息进行打包，按照安全可靠的推送机制（如基于消息队列的推送服务）向边缘端发送更新指令与数据。

**3** 边缘端接收到云端的算法更新指令后，首先对指令的合法性、完整性以及数据一致性进行验证。

**4** 验证通过后，边缘端按照预定的更新流程，暂停当前算法运行，备份旧算法相关数据与配置信息，然后接收并解压新算法代码，进行安装与配置。

**5** 在新算法安装与配置完成后，边缘端使用模拟数据与真实数据相结合的方式对新算法进行全面验证，包括功能正确性验证、性能指标评估以及与周边系统或设备的兼容性测试等。

# 5 系统架构

## 5.1 一般规定

### **5.1.1** 系统架构应包括资源层、服务层、应用层和用户层等内容。

### **5.1.2** 系统的架构设计包括云对边调度协同架构设计、云边协同架构设计、边边协同架构设计。

云对边调度协同架构设计旨在确保云计算资源与边缘计算资源之间的高效协同工作，以支持各种应用场景的需求。

云边协同架构设计是为了实现云端与边缘端的高效协同工作，提升数据处理能力和服务响应速度，以满足各种应用场景的需求。

边边协同架构设计旨在提高系统的整体性能、可靠性、可扩缩性和安全性。

### **5.1.3** 系统安全应覆盖架构的各个层级，包括边缘基础设施安全、运行在边缘上的应用安全、数据安全、边缘平台安全等。

### **5.1.4** 系统架构应满足全生命周期管理，包括系统运维、系统安全管理、系统数据管理等方面，提升系统的应用水平和能力。

## 5.2 框架结构规定

### **5.2.1** 资源层应符合下列要求：

**1** 资源层是系统架构的底层，负责资源的[存储](https://cloud.baidu.com/product/bos.html" \t "https://cloud.baidu.com/article/_blank)、访问和管理。资源层在系统架构中起着基础性的作用，确保各种资源得以安全、高效地共享和使用，并为上层应用和服务提供稳定的支持。

**2** 资源层主要涉及资源的存储方式、资源库结构以及资源访问控制等。

**3** 资源层的设计应注重资源的安全性和完整性，应采取必要的技术措施保障[其安全](https://cloud.baidu.com/solution/security/datacirculation.html" \t "https://cloud.baidu.com/article/_blank)。

**4** 要合理设计资源库结构，提高资源检索效率，满足系统对资源访问的性能要求。

**5** 资源层提供系统所需的基础设施组件和服务。包括日志记录、配置管理、安全性、消息队列、缓存等。

**6** 资源层应能提供不同应用场景所需的算力资源，包括计算资源、存储资源和网络资源等；

**7** 资源层应实现资源的虚拟化，包括计算虚拟化、存储虚拟化和网络虚拟化。例如，计算虚拟化功能可以包括边缘容器、边缘虚拟机、边缘GPU等；存储虚拟化功能可以包括本地存储、后端iSCSI存储、客户端存储、共享卷等；网络虚拟化功能可以包括VPC网络、弹性网卡、负载均衡能力、IPv6地址、边缘互联、vSwitch、SR-IOV网卡、vNIC、网络隔离与地址复用、MTU可调整、VLAN透传、组播等。

**8** 资源层需要具备资源管理功能，包括对机房、网络交换机、服务器等硬件资源的管理，以及对实例、镜像等软件资源的管理，并具备协调资源的功能。

**9** 资源层应具备虚机和容器的镜像管理功能，包括容器应用安装、配置、部署状态查询、启停以及容器应用市场等应用部署能力。

**10** 资源层应具备边缘自治能力，包括协议转换、异构计算、流量卸载、能力开放等，同时应支持对海量的边缘节点统一管理和终端服务精细化控制的服务协同能力。

**11** 资源层应具备边缘AI功能，包括在边缘侧部署AI模型和支持边缘云上运行的AI服务的查看、停止等管理功能。

**12** 资源层应支持边缘与云端数据协同能力，包括边缘数据同步传输到中心云的数据通道，以及中心云下发数据到边缘云和终端设备的数据通道。

### **5.2.2** 服务层应符合下列要求：

**1** 服务层是系统架构的核心层，负责实现系统的业务逻辑和提供必要的服务，起到承上启下的作用。

**2** 服务层应提供清晰、明确的接口定义，包括方法名、参数列表和返回值类型等。接口应具有一定的通用性和灵活性，以适应不同业务场景的需求。

**3** 服务层负责与数据库或其他数据存储进行交互，这些操作应确保数据的完整性和一致性，并遵循业务规则的要求。

**4** 服务层应提供服务的调用机制，使得表示层或其他层可以方便地调用服务层提供的方法。同时，服务层还应支持服务的定位、查找和调用等功能，以满足分布式系统或服务化架构的需求。

**5** 服务层应提供必要的安全性保障措施，包括身份验证、授权验证、数据加密等。这些措施可以确保服务的安全性和数据的保密性，防止未经授权的访问和数据泄露。

**6** 服务层需要合理设计中心系统的云服务，充分整合现有系统的功能，以灵活调配资源，提高支持各项业务的能力和效益。

**7** 服务层应包括信息收发、数据处理、存储管理、轨道计算、控制计算、事后处理、模拟仿真和综合显示等核心业务服务。

**8** 服务层应采用模块化设计，以支持系统的动态可重构性，能够动态加载、卸载模块。

**9** 服务层应具备边缘管理服务能力，包括边缘应用编排模板操作、应用操作、应用镜像操作等。

**10** 服务层宜支持通过web界面对平台服务、虚拟资源、物理主机的可视化监控，并提供操作日志查询与导出功能，支持告警管理。

**11** 服务层应在边缘节点与远程管理系统/中心云断开服务后，仍可以自主对外提供边缘服务。

### **5.2.3** 应用层应符合下列要求：

**1** 应用层是系统架构的中间层，负责处理业务逻辑和数据操作。

**2** 应用层包含系统的核心功能模块，如数据处理、业务规则制定等。

**3** 应用层的设计应注重模块化，将各个功能模块相互独立，便于维护和扩展。

**4** 应用层还需要与用户层和数据库服务器层进行通信，确保数据和指令的正确传输。

**5** 应用层应支持远程运维管理或监控等智能化工业应用，实现远程人机交互功能。

**6** 随着接入设备的增多，应用层需要具备多集群管理能力，通过多层级的分级自治管理，实现一次部署下发，全量集群实现部署的功能。

**7** 应用层应包含数据采集、监控和告警等功能，实现对多维数据的汇集、存储、查询以及自定义告警。

### **5.2.4** 用户层应符合下列要求：

**1** 用户层是系统架构的最外层，直接面向用户提供人机交互界面。

**2** 用户层的主要目标是简化用户操作，使用户能够方便地访问系统功能。

**3** 用户层的设计应注重用户体验，确保界面友好、操作简单。同时，要根据用户需求进行功能设计，确保系统能够满足用户实际需要。

**4** 用户层应支持多种终端设备的访问，包括PC、移动设备和其他智能终端，以满足不同用户的需求。

**5** 用户层应能够实时访问和展示系统中的数据，包括实时监控数据、历史数据和分析结果，确保用户能够及时获取所需信息。

**6** 用户层应具备任务管理功能，允许用户创建、修改和删除测控任务，并能够查看任务的执行状态。

**7** 用户层应提供数据处理和分析工具，支持用户对采集的数据进行处理、分析和可视化展示。

**8** 用户层应具备告警功能，能够根据设定的条件触发告警，并通过多种方式（如短信、邮件等）通知用户。

**9** 用用户层应实现用户权限管理，确保不同用户根据其角色和权限访问相应的功能和数据。

**10** 用户层应记录用户操作日志，支持审计功能，以便于追踪和分析用户行为。

**11** 用户层应确保数据传输和存储的安全性，包括身份认证、数据加密等措施，以防止未授权访问和数据泄露。

## 5.3 协同架构设计

### **5.3.1** 云对边调度协同架构设计应符合下列要求：

**1** 该架构应具备可靠性，以应对可能出现的硬件故障、网络中断或软件错误等各种问题。

**2** 该架构应支持动态的任务调度和资源分配，以应对不同应用场景的实时性、带宽和计算需求。

**3** 在任务调度过程中，应优先选择网络传输时延较小、带宽占用合理的路径，以确保任务的高效传输和快速处理。

**4** 根据任务需求和资源使用情况，动态调整云计算资源和边缘计算资源的分配，以确保资源的充分利用和高效协同。

**5** 在云对边协同过程中，涉及的数据传输和存储应采用加密技术，确保数据的机密性和完整性。

**6** 云对边调度协同架构应具备良好的可扩缩性，以适应未来可能出现的更大规模、更复杂的应用场景。

**7** 架构应支持多种不同的云计算平台和边缘计算设备，以便在不同场景下实现无缝对接和协同工作。

### **5.3.2** 云边协同架构设计应符合下列要求：

**1** 云边协同架构应包含云端、边缘端以及连接两者的通信网络。云端负责全局的资源管理和任务调度，边缘端负责就近处理数据和提供服务，通信网络则保证两者之间的信息传输。

**2** 云端应具备强大的计算、存储和网络资源，能够动态管理并分配这些资源给边缘端。云端应根据任务的需求和边缘端的负载情况，智能调度任务到合适的边缘端执行。云端应对边缘端上传的数据进行深度分析，提取有价值的信息，并为边缘端提供决策支持。

**3** 边缘端应具备数据采集、预处理和实时分析能力，能够就近处理大部分数据，减轻云端负担。边缘端应根据业务需求，提供低延迟、高可靠性的服务，如实时响应、本地缓存等。边缘端应合理利用本地资源，通过资源调度和负载均衡，优化性能并降低能耗。

**4** 通信网络应保证云端和边缘端之间的低延迟通信，确保数据的实时传输和处理。通信网络应具备足够的带宽，以支持大量数据的传输和共享。通信网络应采用加密和安全认证等措施，确保数据传输的安全性。

**5** 该架构应具备高可靠性和容错能力，能够在故障发生时快速恢复服务。

**6** 该架构应采取有效的数据加密和访问控制策略，确保数据的安全性。

**7** 该架构应支持弹性扩缩，能够根据业务需求动态调整云端和边缘端的资源规模。应支持多种应用场景和业务需求，能够灵活配置和调整各组件的功能和性能。

### **5.3.3** 边边协同架构设计应符合下列要求：

**1** 该架构应具备可靠性，确保在部分节点故障时，系统仍能正常运行。

**2** 该架构应易于扩缩，以适应不断变化的业务需求和设备数量。

**3** 架构应提供安全保障措施，对传输和存储的数据进行加密处理，确保数据的安全性，防止数据泄露、篡改和非法访问。

**4** 应采用标准、高效的通信协议，确保数据在节点间可靠传输。

## 5.4 系统运维管理

### **5.4.1** 应具备故障处理和应急响应的能力，以应对可能出现的突发情况。运维人员需要持续监控系统运行状态，及时发现并解决故障和问题，确保系统稳定运行。

### **5.4.2** 采取必要的安全防护措施，设置和管理系统的安全策略，防止恶意攻击、数据泄露和系统瘫痪等安全问题。

### **5.4.3** 可利用自动化工具和技术，实现运维任务的自动化，减少手动操作的错误率。例如，使用自动化脚本进行批量操作、使用监控工具实现实时性能监控等。

### **5.4.4** 能够实现云对边的远程的运维和管理，包括对边缘节点的基础设施进行管控和运维、虚拟化能力的监控、应用实例（或算力）的生命周期管理、监控报警、日志收集和上报等，以及中心到边缘节点的集中管控通道的安全、高可用等。

### **5.4.5** 跟踪系统运维管理的最新技术和趋势，持续学习和更新系统运维管理技术。

### **5.4.6** 运维管理需要规范对云计算系统或云服务资源及活动的监控告警，包括资源监控、告警处置等标准。

### **5.4.7** 应规范云资源管理能力建设，包括计算、存储、网络、设备、数据、服务、应用等物理和虚拟资源，以及资源灾备建设等标准。

### **5.4.8** 需要规范对云计算系统的故障处理能力，指导实施故障处理活动和运维能力建设，包括故障处理流程、处理方法、应急方案等标准。

### **5.4.9** 该架构应具备可靠性，确保在部分节点故障时，系统仍能正常运行。

### **5.4.10** 应具有信息采集和监控的手段，诊断和分析问题的方法，并在运维服务提供过程中熟练应用发现和解决问题的技术和方案以确保运维服务的可用性，并保留应用记录。

### **5.4.11** 应遵循信息技术运维服务通用规范，包括服务目录的建立、服务级别协议的签订、SLA考核自评估机制的建立等。

### **5.4.12** 应具备部署日志审计系统，记录相关安全事件，确保日志留存时间不少于180天，并部署运维堡垒机、数据库审计系统、网络审计系统等，对运维操作行为进行管控和审计。

### **5.4.13** 该架构应具备可靠性，确保在部分节点故障时，系统仍能正常运行。

### **5.4.14** 应建立稳定、安全的云边运维通道，实现中心云能够远程对边缘节点进行监控、告警、配置等操作，边缘平台定期对设备进行主动探测，确保设备连接正常。

### **5.4.15** 应具备实现云边不同地理位置和网络环境的集群统一安全连接，实现多集群的统一管理。

### **5.4.16** 宜通过云边容器网络互联互通、跨云边集群服务发现、服务智能路由、流量管理等，实现服务统一治理。

### **5.4.17** 宜实现对云、边、端分布式资源统一监控、运维，实现统一视角运维能力，最大程度简化用户操作。

## 5.5 系统安全管理

### **5.5.1** 系统安全管理宜基于终端、网络、用户行为等多维风险数据，并结合位置、用户、时间等信息进行决策，实时动态调整授权或安全策略。

### **5.5.2** 制定详细的安全管理制度，对系统安全策略、安全配置、日志管理和日常操作流程等方面进行规范。以此确保系统的安全性和稳定性，并减少因人为操作失误而导致的安全问题。

### **5.5.3** 指定专人对系统进行管理，并划分系统管理员角色。明确各个角色的权限、责任和风险，确保每个管理员都清楚自己的职责和权限范围。权限设定应遵循最小授权原则，即只授予完成任务所需的最小权限。

### **5.5.4** 依据操作手册对系统进行维护，并详细记录操作日志。包括重要的日常操作、运行维护记录、参数的设置和修改等内容。

### **5.5.5** 根据业务需求和系统安全分析，确定合适的访问控制策略。包括根据不同用户或系统访问其特定的资源或执行其特定的操作，以及其相应级别的访问权限。

### **5.5.6** 定期对系统进行漏洞扫描，以发现潜在的安全风险，一旦发现漏洞，应及时进行修补。

### **5.5.7** 定期对运行日志进行分析，以便及时发现异常行为或潜在的安全风险。

### **5.5.8** 应具备防御物理安全威胁能力、防止USIM卡被非法拔出或替换能力、防终端用户伪造能力、防业务数据监听能力、防认证信息伪造能力、防终端软件被破解/篡改能力。

### **5.5.9** 应具备设备安全能力，包括标识安全、身份标识与鉴别、访问控制安全、通信安全、设备数据安全等；平台安全能力，包括身份认证和账户安全、授权与访问控制安全、支援资源安全、存储资源安全、网络资源安全等；应用安全能力，包括身份鉴别、访问控制、web应用安全、资源控制安全等。

### **5.5.10** 应具备基础设施安全能力，包括物理环境安全、设备安全、网络安全等；平台安全能力，包括身份认证和账户安全、授权与访问控制安全、支援资源安全、存储资源安全、网络资源安全等；应用安全能力，包括身份鉴别、访问控制、web应用安全、资源控制安全等。

### **5.5.11** 应具备身份认证、数据备份与恢复、隐私保护、入侵检测等边云协同安全能力。

### **5.5.12** 提供基础的安全防护功能，包括防欺骗、ACL访问控制、账号口令核验、异常告警、日志安全处理等能力。

## 5.6 系统数据管理

### **5.6.1** 必须确保系统数据是完整且准确的。同时，需要定期进行数据清理和校验过程，以识别和纠正任何潜在的数据问题。

### **5.6.2** 数据安全是系统数据管理的核心要求。必须实施适当的安全措施，如数据加密、访问控制、审计日志等，以防止数据泄露、篡改或破坏。此外，还需要定期进行安全漏洞评估和渗透测试，以发现和解决潜在的安全风险。

### **5.6.3** 为了防止数据丢失或损坏，系统必须必须构建高可靠性的数据存储架构、实施有效的数据备份和恢复策略，包括冗余存储机制、定期备份数据、存储备份数据在安全的地点、以及制定详细的恢复计划，以便在发生数据丢失或灾难性事件时能够迅速恢复数据。

### **5.6.4** 应制定数据应急管理计划，保证在故障发生时系统能够迅速恢复至正常状态、避免数据丢失和功能失效。

### **5.6.5** 可以支持多样化的数据格式处理，遵循通用的接口和通信协议标准，以满足不同应用场景的需求。

### **5.6.6** 系统应确保用户能够方便、高效地访问所需的数据。这要求有清晰的数据目录结构、易于使用的数据查询工具以及高效的数据处理性能。

### **5.6.7** 系统数据管理必须遵守相关的法律法规和行业标准，如数据保护法规、隐私政策等，以确保系统能够处理和管理敏感数据，确保数据的合法使用和共享。

### **5.6.8** 系统数据管理应实现数据的全生命周期管理，从数据源的全面感知采集到数据的汇聚集成、大规模存储、智能分析等协同体系，有效提升数据应用水平和能力。

### **5.6.9** 在数据接入方面，应能处理海量多源异构数据，实现异构设备和协议的数据管理平台设备接入（边缘异构协议适配等）、数据处理（实时智能分析等）、数据传输（云边网络不稳定导致数据丢包等）、数据质量（完整性、分布式传输防篡改等）。

### **5.6.10** 应实现生产/现场数据与企业经营管理、服务等数据的深度融合，打破现有各个系统数据割裂状态，实现业务数据共享流通。

### **5.6.11** 宜采用校验技术、密码技术等措施进行安全存储，并实施数据容灾备份和存储介质安全管理，定期开展数据恢复测试。

### **5.6.12** 宜建立数据质量管理机构和机制，定义数据质量管理的角色和责任，建立数据质量管理方法，识别数据生命周期各个阶段的数据质量因素，构建数据质量评估框架。

### **5.6.13** 宜明确数据安全的内外部监管和管理需求，制定数据安全管理的目标、方针和策略，并持续改进和优化，做到数据防泄露、防篡改和防损毁。

### **5.6.14** 应定义明确的服务水平目标，包括响应时间、可用性、数据完整性等关键指标，并建立监测和报告机制，对数据库系统进行定期性能监测，并及时向相关人员报告任何异常情况。

### **5.6.15** 应建立数据安全的决策机制，界定部门和角色职责和权限，使数据安全任务有的放矢。规划数据安全技术架构，保护计算单元、存储设备、操作系统、应用程序和网络边界各层免受恶意软件、黑客入侵和内部人员窃取等威胁。

# 6 业务流程模型

## 6.1 一般规定

### **6.1.1** 业务需求应符合下列规定：

**1** 业务流程模型设计应始终围绕业务需求进行，确保模型能够满足实际业务的需要。

**2** 模型设计前，需充分理解业务需求，明确业务目标、业务流程、数据流向等关键信息，作为模型设计的输入。

### **6.1.2** 标准化与模块化应符合下列规定：

**1** 业务流程模型应采用标准化的设计方法和模块化的结构，以提高系统的可维护性、可扩展性和可重用性。

**2** 制定统一的业务流程模型设计规范和标准，确保模型的一致性和可理解性。同时，将业务流程划分为独立的模块，明确模块之间的接口和交互方式。

### **6.1.3** 灵活性与适应性应符合下列规定：

**1** 业务流程模型应具有一定的灵活性和适应性，能够应对业务变化和技术发展的挑战。

**2** 模型设计应充分考虑业务变化的可能性，设计可配置的参数和可扩展的接口。需采用先进的技术架构和设计方法，确保模型适应新技术和新业务的发展。

### **6.1.4** 可靠性与鲁棒性应符合下列规定：

**1** 业务流程模型应具有较高的可靠性和鲁棒性，能够确保系统在异常情况下仍能稳定运行并正确处理数据。

**2** 模型设计应充分考虑各种异常情况的处理机制，同时，需采用容错设计、备份机制等技术手段。

### **6.1.5** 用户友好性应符合下列规定：

**1**业务流程模型应设计得简单明了、易于理解，方便用户进行操作和管理。

**2**在模型设计时，应考虑用户的使用习惯和需求，设计直观易懂的图形化界面和操作流程，并提供帮助文档和在线支持服务。

## 6.2 数据采集与处理流程

### **6.2.1** 数据源应符合下列规定：

**1** 各种传感器、测量仪表、设备接口等数据源需具有其特定的数据格式和输出方式。

**2** 数据源输出的数据格式，需要转换为系统可识别的标准格式。

**3** 数据精度需要根据实际需求进行选择和调整。

**4** 数据更新频率需根据业务需求和数据的重要性，设置合理的数据更新频率。

### **6.2.2** 数据采集策略应符合下列规定：

**1** 根据业务需求确定需要采集的数据类型、范围、频率等。

**2** 对于重要数据，需要设置较高的采集频率和优先级，以确保数据的完整性和准确性。

**3** 在满足业务需求的前提下，需要考虑系统的处理能力、存储能力等因素，避免采集过多数据导致系统性能下降。

**4** 需采取必要的加密和备份措施，确保数据在采集和传输过程中的安全性和完整性。

### **6.2.3** 数据处理与分析应符合下列规定：

**1** 通过数据过滤、去重、异常值处理等步骤对数据进行清洗，消除数据中的噪声、错误和冗余信息，提高数据质量。

**2** 将原始数据转换为系统可识别的标准格式数据和标准单位。

**3** 对于大量的数据，可进行压缩以减少存储空间和传输带宽的需求。压缩算法需要根据数据的特性和需求进行选择。

**4** 可利用统计方法、机器学习等技术对处理后的数据进行深入分析和挖掘，提取有价值的信息和知识。

## 6.3 控制指令下发与执行流程

### **6.3.1** 指令生成与下发应符合下列规定：

**1** 根据业务需求，确定需要执行的控制操作，如设备启动、停止、参数调整等。

**2** 根据设备或系统的通信协议和控制接口，编写相应的控制指令。指令需要包含明确的操作命令、参数设置等信息。

**3** 在指令下发之前，需要对指令进行验证，确保指令的正确性和有效性。

**4** 选择通信协议：根据设备或系统的通信协议，选择合适的通信方式

**5** 建立与远程设备或系统的通信连接，确保指令能够准确传输。

**6** 将验证后的指令通过选定的通信协议传输给远程设备或系统。

7在传输过程中，需要确保指令的完整性和安全性。

### **6.3.2** 终端设备执行与反馈应符合下列规定：

**1** 远程设备或系统接收到指令后，首先进行指令的解析和验证。

**2** 根据解析后的指令，远程设备或系统执行相应的操作。

**3** 在执行过程中，设备或系统需要实时监控执行状态，确保操作按照预期进行。

4设备或系统在执行完指令后，需要生成相应的反馈信息，包括执行状态、执行结果等。

**5** 将反馈信息通过相同的通信协议回传给系统。在传输过程中，同样需要确保信息的完整性和安全性。

### **6.3.3** 指令执行效果评估应符合下列规定：

**1** 系统接收到终端设备的反馈信息后，需进行解析和验证，确保信息的准确性和有效性。

**2** 根据反馈信息，系统可对指令的执行效果进行评估。评估内容包括执行状态、执行结果是否符合预期等。如果执行结果不符合预期，系统需要进行相应的处理，如重新下发指令、调整参数设置等。

**3** 系统需要记录每次指令下发和执行的详细信息，包括下发时间、执行时间、执行结果等。根据需要，系统可以生成相应的报告。

## 6.4 系统故障处理与恢复流程

### **6.4.1**在工程测控云边端系统中，应根据系统需求，制定系统故障处理与恢复流程，保系统稳定运行、及时应对和处理各种故障。

### **6.4.2** 故障检测应符合下列规定：

**1** 实时监控：系统通过实时监控各组件的状态和性能数据，检测潜在的故障或异常现象。这包括硬件故障、软件错误、网络问题等多种可能的情况。

**2** 日志分析：系统记录和分析各种日志信息，如操作日志、错误日志、系统日志等，以发现可能的故障源。

**3** 告警机制：当检测到故障或异常时，系统触发告警机制，向管理人员发送告警信息，以便及时处理。

### **6.4.3** 故障诊断涉应符合下列规定：

**1** 初步判断：根据告警信息和系统状态，初步判断故障类型和可能的原因。

**2** 深入分析：利用专业的故障诊断工具和方法，对故障进行深入分析，确定故障的确切位置和原因。

**3** 生成诊断报告：将故障诊断的结果整理成报告，详细说明故障类型、原因、影响范围等，为后续的故障处理提供依据。

### **6.4.4** 故障隔离与恢复应符合下列规定：

**1** 为防止故障扩散，可先关闭或隔离故障设备或组件，确保其他部分不受影响。

**2** 在网络或系统中切断故障的传播路径，防止故障影响扩大。

**3** 对于某些临时性的故障，可以通过重启设备或组件来恢复其正常运行。

**4** 如果故障设备或组件无法立即修复，可以切换到备份设备或组件，确保系统继续运行。

**5** 根据故障诊断报告，对故障设备进行修复或更换，以恢复其正常功能。

### **6.4.5** 故障记录与分析应符合下列规定：

**1** 详细记录故障发生的时间、地点、类型、原因、处理过程等信息。

**2** 保存与故障相关的日志文件，以便后续分析和审计。

**3** 对故障原因进行深入分析，找出根本原因和潜在风险。

**4** 根据故障处理过程和分析结果，需提出改进措施和优化建议。

**5** 根据分析结果和改进建议，需完善系统的故障处理机制和应急预案，提高系统的可靠性和稳定性。

## 6.5 数据存储与管理流程

### **6.5.1**存储到数据库中的数据应通过数据存储与管理流程，进行有效的管理和维护，提供数据查询、统计和分析等功能。

### **5.5.2** 设计原则应符合下列规定：

**1** 安全性原则：确保数据的存储和访问过程均符合安全标准，防止数据泄露、篡改和非法访问。

**2** 可靠性原则：保证数据存储的持久性和稳定性，避免因硬件故障、软件错误或人为操作导致的数据丢失或损坏。

**3** 可扩展性原则：设计灵活可扩展的数据存储架构，以满足系统业务发展的需求，便于后期扩容和升级。

**4** 可管理性原则：提供完善的数据管理工具和方法，方便用户对数据进行备份、恢复、归档等操作。

### **6.5.3** 功能要求应符合下列规定：

**1** 数据存储：将处理后的数据存储到数据库中，支持多种数据格式和存储方式，确保数据的完整性和准确性。

**2** 数据管理：提供数据备份、恢复、归档等管理功能，保障数据的安全性和可追溯性。

**3** 数据查询：提供高效的数据查询功能，支持多种查询方式和查询条件，方便用户快速获取所需数据。

**4** 数据分析：提供数据分析和统计功能，支持对海量数据进行挖掘和分析。

### **6.5.4** 实施策略应符合下列规定：

**1** 根据业务需求和技术要求，选择合适的数据库系统。

**2** 根据业务需求和数据特点，设计合理的数据库表结构、索引和视图等，提高数据查询和处理的效率。

**3** 制定完善的数据备份和恢复策略，定期备份数据并存储在安全可靠的地方，确保数据的安全性。同时，提供数据恢复功能，以便在数据丢失或损坏时能够及时恢复数据。

**4** 通过分区、压缩、加密等技术手段优化数据存储性能，减少存储空间占用和提高查询性能。

**5** 加强数据安全管理：采用身份认证、访问控制、数据加密等安全措施，确保数据在存储和访问过程中的安全性。

## 6.6 用户交互流程

### **6.6.1** 一般设计原则应符合下列规定：

**1** 用户交互界面应设计得直观易懂，操作流程应简单明了，降低用户的学习成本和提高使用效率。

**2** 系统应保持一致的交互风格和操作流程，减少用户的认知负担和误操作的可能性。

**3** 用户交互流程应具有一定的灵活性，以适应不同用户的需求和习惯。

**4** 用户交互过程应确保数据的安全性和用户身份的真实性，防止未授权访问和数据泄露。

### **6.6.2** 功能要求应符合下列规定：

**1** 系统应提供用户登录功能，要求用户输入正确的用户名和密码进行身份验证，应确保只有合法用户能够访问系统。

**2** 系统应提供直观易懂的用户界面，展示实时数据、设备状态、历史记录等信息，方便用户查看和了解系统状态。

**3** 用户应能够通过系统下发操作指令，如控制设备开关、调整参数等，系统应能够准确执行并反馈执行结果。

**4** 统应提供系统配置和管理功能，允许用户修改系统参数、添加或删除设备等，以满足不同用户的需求。

**5** 系统应提供详细的帮助文档和在线支持服务。

### **6.6.3** 实施策略应符合下列规定：

**1** 采用图形化界面设计，减少文字输入和复杂操作，提高用户的使用体验。

**2** 简化操作流程，减少不必要的步骤和点击次数，提高用户的工作效率。

**3** 支持多种交互方式，如鼠标、键盘、触摸屏等，适应不同用户的使用习惯。

**4** 采用加密技术对用户输入的数据进行加密处理，确保数据在传输过程中的安全性。同时，对用户身份进行验证和授权管理，防止未授权访问和数据泄露。

**5** 为用户提供详细的帮助文档和在线支持服务，解答用户在使用过程中遇到的问题和困惑。

## 6.7 实施步骤

### **6.7.1** 实施准备应符合下列规定：

**1** 项目规划：明确项目目标、范围、时间表和预算，制定详细的项目计划。

**2** 需求分析：深入了解业务需求，明确系统应具备的功能和性能要求。

**3** 技术选型：根据项目需求和技术发展趋势，选择合适的技术方案和工具。

**4** 资源准备：准备必要的硬件、软件、网络等资源，确保项目顺利进行。

### **6.7.2** 系统设计应符合下列规定：

**1** 架构设计：设计合理的系统架构，明确各个模块的功能和接口。

**2** 数据库设计：根据业务需求设计数据库表结构、索引和视图等。

**3** 界面设计：设计直观易懂的用户界面，提高用户的使用体验。

**4** 安全设计：设计系统的安全策略，确保数据的安全性和用户身份的真实性。

### **6.7.3** 系统开发应符合下列规定：

**1** 编码实现：按照系统设计和数据库设计进行编码实现。

**2** 单元测试：对各个模块进行单元测试，确保模块功能的正确性。

**3** 集成测试：将各个模块集成在一起进行测试，确保系统整体功能的正确性。

### **6.7.4** 系统部署与测试应符合下列规定：

**1** 环境搭建：搭建系统所需的硬件、软件和网络环境。

**2** 系统部署：将开发好的系统部署到生产环境中。

**3** 系统测试：对系统进行全面的测试，包括功能测试、性能测试、安全测试等。

**4** 问题修复：对测试中发现的问题进行修复，确保系统的稳定性和可靠性。

### **6.7.5** 用户培训与上线应符合下列规定：

**1** 用户培训：对用户进行系统的操作培训，确保用户能够熟练使用系统。

**2** 系统上线：在系统测试通过并经过用户确认后，正式上线运行。

**3** 后期维护：提供系统的后期维护和升级服务，确保系统的持续稳定运行。

### **6.7.6** 总结与评估应符合下列规定：

**1** 项目总结：对整个项目进行总结，评估项目的成功与不足。

**2** 经验总结：总结项目过程中的经验教训，为今后的项目提供借鉴。

**3** 持续改进：根据用户反馈和项目总结，对系统进行持续改进和优化。

通过以上实施步骤，可以确保工程测控云边端系统架构的顺利实施和成功部署。在实际操作中，应根据项目具体情况进行调整和完善。

# 7 总体要求

## 7.1 一般规定

### **7.1.1** 系统应能兼容各种工程测控传感器和仪器，如温度传感器、压力传感器、位移传感器、流量计等不同类型、不同品牌、不同通信协议的设备。

### **7.1.2** 系统应具备良好的易用性，具体要求如下：

**1** 系统应支持多种工程测控设备常用的通信协议，确保不同厂商、不同类型的设备之间能够相互通信。

**2** 系统应具备统一的数据格式，使不同设备采集的数据能够被系统正确识别和处理。

**3** 提供开放的接口，便于与其他工程管理系统、数据分析平台等进行交互。

### **7.1.3** 系统应具备良好的可靠性，具体要求如下：

**1** 在在终端、边缘端和云端设备中应采用冗余设计，确保在部分设备出现故障时，系统仍能正常运行。

**2** 云边端系统应构建具有冗余链路的网络，同时须采用网络容错技术，如多路径传输、链路聚合等，提高网络的可靠性。

**3** 云端、边缘端和终端设备之间应建立有效的数据同步机制，确保数据一致性。

**4** 云边端系统应有合理的冲突处理机制。

## 7.2 数据源采集、存储与传输要求

### **7.2.1** 云边端系统须满足不同传感器的精度与频率要求。

### **7.2.2** 云边端系统应具备稳定、可靠的数据传输通道，确保从终端到边缘端、从边缘端到云端的数据传输过程中，数据不会丢失或出错。

### **7.2.3** 对于一些对时间敏感的工程测控数据（如紧急报警信号、实时控制指令），须采用低延迟的通信技术（如边缘计算减少传输路径），满足实时性要求。

### **7.2.4** 系统应采用云边端分布式存储方式，边缘端存储实时性强、本地处理频繁的数据，云端存储海量的历史数据和备份数据。

## 7.3 备份和恢复要求

### **7.3.1** 云边端系统应建立数据备份机制，定期备份数据，并能够在数据丢失或损坏时快速恢复。

### **7.3.2** 备份过程中必须确保数据的全面性与准确性，涵盖所有关键数据项与配置信息，以保障数据完整性，避免任何形式的数据丢失或损坏。

### **7.3.3** 备份数据应实施加密存储，以防范未经授权的访问和数据泄露风险。同时，备份存储环境需实施严格的安全管控措施，包括物理访问限制、网络安全防护等。

### **7.3.4** 系统应具备快速响应和准确恢复的能力，以最小化数据丢失和服务中断时间，确保业务快速恢复运行。

### **7.3.5** 随着业务规模的扩大和需求的变化，备份与恢复方案应具备高度的可扩展性和灵活性。系统应能够适应不同规模的数据量，支持多样化的备份策略和恢复需求。

### **7.3.6** 系统应支持多种备份方式，包括全量备份、增量备份、差异备份等，以满足不同场景下的备份需求。同时，应提供定时备份和手动备份等多种操作模式，以满足用户的个性化需求。

## 7.4 故障恢复与迁移要求

### **7.4.1** 系统具备强大的故障恢复能力，当系统出现故障时，能够快速恢复正常运行。

### **7.4.2** 系统应具备高效的故障检测与定位机制，以便在故障发生时能够迅速定位问题根源，为故障的快速恢复提供有力支持。

### **7.4.3** 在进行云边端系统迁移前，应对现有系统进行深入评估，包括数据量、应用类型、网络状况等方面。基于评估结果，制定详细的迁移计划，明确迁移目标、步骤和数据迁移过程中应确保数据的完整性和安全性。采用可靠的数据迁移工具和方法，对数据进行精确传输和验证。

### **7.4.4** 应确保原有应用程序的稳定性和可用性不受影响。通过合理的迁移策略和技术手段，避免数据丢失和功能失效。

## 7.5 兼容性要求

### **7.5.1** 云边端系统需灵活适应中央处理器（CPU）、图形处理器（GPU）以及专用集成电路（ASIC）等多种计算架构，以优化系统性能并降低能耗。

### **7.5.2** 系统需支持多样化的数据格式处理，包括但不限于图像、视频、音频、文本等多媒体数据以及结构化数据如数据库记录等，以满足不同应用场景的需求。

### **7.5.3** 云边端系统需遵循通用的接口和通信协议标准，如HTTP、MQTT、WebSocket等，以确保各类设备和应用程序能够顺畅通信与高效协作。

## 7.6 可扩缩性要求

### **7.6.1** 系统应能方便地接入新的工程测控终端设备，随着工程规模的扩大或新的监测需求的出现，能够快速将新设备集成到系统中。

### **7.6.2** 边缘端和云端的计算、存储和网络资源要能够方便地进行扩展。

### **7.6.3** 系统软件应具备良好的扩展性，能够方便地添加新的功能模块。

### **7.6.4** 系统应支持与其他相关系统的集成扩展，如与企业资源计划（ERP）系统、地理信息系统（GIS）等进行集成。

# 本规程用词说明

**1** 为便于在执行本规程条文时区别对待，对要求严格程度不同的用词说明如下：

**1）**表示很严格，非这样做不可的：

正面词采用“必须”，反面词采用“严禁”；

**2）**表示严格，在正常情况下均应这样做的：

正面词采用“应”，反面词采用“不应”或“不得”；

**3）**表示允许稍有选择，在条件许可时首先应这样做的：

正面词采用“宜”，反面词采用“不宜”；

**4）**表示有选择，在一定条件下可以这样做的，采用“可”。

**2** 条文中指明应按其他有关标准执行的写法为：“应符合……的规定”或“应按……执行”。

# 引用标准名录

《信息技术云计算边缘云通用技术要求》GB/T 44271-2024

《信息技术服务质量评价指标体系》GB/T 33850-2017

《信息技术服务分类与代码》GB/T 29264-2012

《信息技术服务智能运维通用要求》GB/T 43208.1-2023

《信息安全技术网络安全服务能力要求》GB/T 32914-2023

《数据资产管理要求》GB/T 40685-2021

《云资源监控通用要求》GB/T 37736-2019

《云服务级别协议基本要求》GB/T 36325-2018

《公共基础服务业云边协同技术标准算力协同》T/CI 096-2023

《公共基础服务业云边协同技术标准总则》T/CI 093-2023

中国建筑业协会团体标准

**工程测控云边端系统架构技术规程**

Technical Specification for Engineering Measurement and Control Cloud Edge System Architecture

**条文说明**

**制定说明**

《工程测控云边端系统架构技术规程》（T/CCIAT xxxx— 20xx），经中国建筑业协会××××年××月××日以第××号公告批准发布。

本规程制订过程中，编制组进行了云边端一体化管理机制、系统架构、业务流程的调查研究，总结了实践经验，同时参考了国外先进技术法规、技术标准，通过项目实践取得了本规程重要技术参考内容。

为便于广大设计、施工、科研、学校等单位有关人员在使用本规程时能正确理解和执行条文规定，编制组按章、节、条顺序编制了本规程的条文说明，对条文规定的目的、依据以及执行中需注意的有关事项进行了说明。但是，本条文说明不具备与标准正文同等的法律效力，仅供使用者作为理解和把握标准规定的参考。

**目 次**

2 术语 56

3 基本规定 58

4 云边端一体化管理机制规定 59

4.1 工作流程的标准规定 59

4.2 协同机制的标准规定 60

4.4 各方职责的标准规定 61

4.5 边端轻量化算法部署规定 61

6 业务流程模型 64

6.1一般规定 64

6.2数据采集与处理流程 64

6.3控制指令下发与执行流程 65

6.5数据存储与管理流程 65

6.6用户交互流程 66

7 总体要求 67

7.1 一般规定 67

7.2 数据源采集、存储与传输要求 67

7.4 故障恢复与迁移要求 68

7.6 可扩缩性要求 68

# 2 术语

### **2.0.1** 在工程测控云边端系统中，云负责核心管理和数据处理的角色。它负责接收来自“边”和“端”的数据，进行全局的数据分析和处理，提供基于全局数据的智能服务，如智能调度、运维、宏观决策等。同时，它还能够对“边”和“端”进行远程管理和控制。

### **2.0.2** 中心云负责处理全局性、复杂性和对计算资源需求较高的任务，例如对工程测控系统中来自多个边缘端的海量数据进行汇总、深度分析与挖掘，利用大规模数据训练复杂的模型（如预测性维护模型、系统优化模型等），并根据分析结果制定全局性的决策策略和控制指令。

### **2.0.6** 端负责全面感知和数据采集。它通过传感器或其他数据采集设备获取数据，并将数据上传至“边”或“云”进行处理。同时，它还能够接收来自“云”或“边”的指令和数据，进行相应的操作和控制。随着物联网技术的发展，端设备的种类越来越丰富，功能也日益强大，不仅具备基本的数据采集和执行功能，还可能集成了一定的本地处理能力和通信能力，如智能传感器能够在本地进行简单的数据处理和初步分析，然后再将结果上传，以减轻网络传输和云端处理的负担。

### **2.0.3**在工程测控领域，边缘云可对本地采集到的工程数据进行即时处理，如数据预处理、初步分析、本地决策控制等，过滤掉大量冗余数据后再将有价值的数据上传至中心云进行进一步的深度分析与全局优化。边缘云的存在有效缓解了中心云的计算压力，提高了整个系统的处理效率和可靠性。

### **2.0.7**云端计算与存储能力强，处理复杂分析、训练与全局决策；边缘端靠近数据源与终端，有低延迟处理、实时响应与本地缓存能力，可预处理数据。

云端与边缘交互数据、分配任务协同工作，边缘端预处理并上传关键数据，云端分析优化后下发策略，以满足工程测控实时、可靠、高效与智能需求。

### **2.0.8**在工程测控云边端系统中，工程规模大或分布广，各边缘节点负责不同区域或子系统测控任务，它们相互通信、共享数据与资源，联合处理数据与优化决策。

# 3 基本规定

### **3.0.3** 界定规程的应用边界，以便相关人员判断在具体项目中是否需要遵循本规程。阐述系统架构设计的基础准则，为后续具体设计要求提供理念依据，如开放性利于系统集成其他技术与设备，可扩展性方便系统功能与规模的扩充等。

# 4 云边端一体化管理机制规定

## 4.1 工作流程的标准规定

### **4.2.1-2** 在工程测控云边端系统中，此流程确保了从源头获取的数据具有较高质量，为后续的数据处理与分析奠定基础。合理设置采样频率与精度可在满足测控需求的同时，避免数据冗余与资源浪费；数据清洗与校验能有效减少错误数据对系统决策的干扰。

### **4.2.3-3** 数据传输流程中的通信协议选择需综合考虑网络环境、数据量与实时性要求等因素。压缩与加密操作是为了提高传输效率与保障数据机密性，防止数据在传输途中被窃取或篡改。

### **4.2.5-2** 边缘端应及时将指令执行结果与执行状态信息及时反馈给云端，以便云端进行后续的监控与决策调整。

### **4.2.6-2** 当边缘端发现设备运行异常时，如 CPU 长时间高负载或网络连接中断，自动启动本地的故障诊断与修复机制，尝试重启相关服务或切换网络连接方式；若发现数据采集失败率过高或数据处理时间过长，及时调整数据采集参数或优化边缘处理算法。

### **4.2.6-3** 当云端发现大量边缘端上传的数据存在质量问题，分析原因并向边缘端推送数据质量优化策略，如更新数据清洗规则；当在边缘端任务繁重且资源紧张时，从云端调配部分闲置资源给边缘端使用，以实现系统资源的优化配置，确保系统始终处于高效稳定的运行状态。

### **4.3.1-2** 云边数据协同旨在实现数据资源的合理分配与高效利用。云端的数据管理系统提高了数据的可发现性与可获取性，边缘端的主动请求与数据上传机制则保证了数据在云边之间的动态交互与实时更新，满足不同层面的测控数据需求。

## 4.2 协同机制的标准规定

### **4.3.2-2** 当边缘端任务执行出现延迟或资源紧张时，云端可将部分任务转移至其他空闲边缘端或在云端进行处理，确保系统整体任务的按时完成与高效运行。

云边任务协同能够充分发挥云端的全局统筹能力与边缘端的本地化优势。通过合理的任务分配与动态调整，提高了系统的任务处理效率与资源利用率，避免了任务集中导致的局部过载或性能瓶颈问题。

### **4.3.3-3** 边边数据共享有助于拓展边缘端的数据视野，提升分布式测控任务的协同效果。数据共享网络架构与技术选型需考虑边缘端的分布式特性与网络环境复杂性，加密与认证技术则是保障数据共享安全的必要手段。

### **4.3.4-2** 边边任务协作能够充分发挥多个边缘端的协同优势，提高复杂任务的处理效率与准确性。任务分解与分配机制以及实时交互机制是实现边缘端高效协作的关键要素，有助于构建分布式、智能化的工程测控网络。

## 4.4 各方职责的标准规定

### **4.4.1-10** 云端作为工程测控云边端系统的核心枢纽，承担着多项关键职责。其强大的存储与计算能力使其能够处理复杂的数据与模型任务，系统配置与安全管理职责则确保了整个系统的有序运行与数据安全防护。

### **4.4.3-2** 终端作为工程测控云边端系统的最前端，其数据采集的准确性与指令执行的有效性是整个系统实现精确测控的基础。不同类型的传感器与执行器在各类工程应用中发挥着关键作用，需根据工程需求合理选型与配置。

## 4.5 边端轻量化算法部署规定

### **4.5.1-1** 边缘端设备通常计算能力和存储资源相对有限，若选用过于复杂的算法，可能会导致设备运行缓慢甚至出现故障。选择计算复杂度适配的算法能够保障边缘端在处理数据时的高效性与稳定性，确保整个工程测控系统的实时性要求得以满足。

### **4.5.1-2** 边缘端设备通常计算能力和存储资源相对有限，若选用过于复杂的算法，可能会导致设备运行缓慢甚至出现故障。选择计算复杂度适配的算法能够保障边缘端在处理数据时的高效性与稳定性，确保整个工程测控系统的实时性要求得以满足。

### **4.5.1-4** 不同的工程测控任务会产生不同类型和分布特征的数据。选择与数据特性匹配的算法能够更精准地从数据中提取有价值的信息，提高数据分析的准确性，从而为工程测控的决策和控制提供更可靠的依据，避免因算法选择不当而导致数据处理结果偏差较大，影响整个工程测控的质量。

### **4.5.2-1** 算法代码在传输过程中面临安全风险，如被恶意篡改可能会导致边缘端运行错误的算法，影响工程测控结果甚至引发系统故障；同时，传输带宽在一些边缘端网络环境中可能较为有限，压缩代码可提高传输效率，减少对网络资源的占用，确保算法代码能够快速、安全地传输到边缘端设备。

### **4.5.3-1** 工程测控过程中，数据的分布特征可能会随着时间、环境等因素发生变化。基于数据驱动的优化能够使算法自动适应这些变化，保持对数据处理的有效性。通过动态调整算法参数或模型结构，能够提高算法对数据的适应性和处理精度，确保在不同数据条件下都能获取高质量的处理结果，为工程测控提供更可靠的数据支持。

### **4.5.3-2** 若算法计算时间过长影响实时性，通过算法简化、代码优化或采用近似计算方法等手段降低计算复杂度；若内存使用超出预期，优化数据存储结构与算法逻辑，减少内存占用。

### **4.5.3-3** 边缘端的资源使用情况会随着系统运行状态和外部环境而动态变化。本地资源感知优化能够使算法根据当前资源状况灵活调整自身运行模式，充分利用有限资源并避免因资源不足导致的系统故障。在资源充裕时提升算法复杂度，可以进一步挖掘数据价值，提高工程测控的精度和效果；在资源紧张时则优先保障系统的基本运行功能，确保工程测控的连续性。

### **4.5.4-4** 在更新过程中，实时监控更新进度与状态，若出现异常（如代码安装失败、配置参数不匹配等），及时向云端反馈错误信息，并根据预设的回滚策略恢复至旧算法版本，保障边缘端业务的连续性与稳定性。

# 6 业务流程模型

## 6.1一般规定

### **6.1.4-2** 异常情况如数据丢失、网络中断、设备故障等，通过异常处理机制、容错设计与备份机制保证系统的可靠性。

### **6.1.5-2** 在工程测控云边端系统架构中，业务流程模型的关键组成部分是确保系统高效、稳定、可靠运行的基础。下节将详细阐述业务流程模型的关键组成部分，包括数据采集与处理流程、控制指令下发与执行流程、故障处理与恢复流程、数据存储与管理流程以及用户交互流程。

## 6.2数据采集与处理流程

### **6.2.1** 在工程测控云边端系统中，数据源是数据的起始点，它决定了数据的质量和可用性。

### **6.2.1-1** 常见传感器（如位移传感器、温湿度传感器、压力传感器、角度传感器、加速度传感器等），测量仪表（如流量计、压力表等），设备接口（PLC,DCS,SCADA等）

### **6.2.1-2** 数据源输出的数据格式可能各不相同，如模拟信号、数字信号、文本、视频等，为了统一管理和处理，需要将这些数据转换为系统可识别的标准格式。

### **6.2.1-4** 对于实时性要求高的应用，需要设置较高的更新频率；对于实时性要求不高的应用，可以适当降低更新频率以节省资源。

### **6.2.2-4** 采集策略可以包括实时采集、定时采集或事件触发采集等不同的方式。实时采集适用于对实时性要求高的应用；定时采集则适用于对实时性要求不高的应用；事件触发采集则是当满足特定条件时触发数据采集。

### **6.2.3-2** 便于后续的分析和挖掘。例如，将模拟信号转换为数字信号、将不同单位的数据转换为统一单位等。

### **6.2.3-4** 数据分析与深度挖掘可以帮助了解设备的运行状态、性能趋势、潜在问题等，为决策提供支持。

## 6.3控制指令下发与执行流程

### **6.3.3-3** 通过指令下发与执行流程，系统能够实现对远程设备或系统的有效控制，并根据执行结果进行相应的处理和记录。有助于确保系统的稳定性和可靠性，提高设备的运行效率。

## 6.5数据存储与管理流程

### **6.5.4-5** 数据存储与管理流程是工程测控云边端系统架构中的重要组成部分。通过设计合理的数据库系统、实施数据备份和恢复策略、优化数据存储性能以及加强数据安全管理等措施，可以确保数据的安全性、可靠性和可用性。在实际应用中，应根据业务需求和技术要求，灵活调整数据存储与管理流程的设计和实施策略，以满足系统的稳定性和高效性要求。

## 6.6用户交互流程

**6.6** 本部分将详细阐述用户交互流程的设计原则、功能要求以及实施策略，以确保用户能够方便、快捷、准确地与系统进行交互。通过设计直观易懂的用户界面、优化操作流程、提供多种交互方式以及加强数据安全性等措施，可以确保用户能够方便、快捷、准确地与系统进行交互。同时，提供详细的帮助文档和在线支持服务也可以帮助用户更好地使用系统并解决问题。

# 7 总体要求

## 7.1 一般规定

### **7.1.2-3** 接口应遵循行业标准或通用的接口规范，如 RESTful API、OPC UA 等，使系统能够与外部系统进行数据共享和业务协作。例如，通过 RESTful API 接口，将工程测控系统中的设备运行数据提供给企业的设备管理系统，实现设备的远程监控和维护。

### **7.1.3-4** 在数据可能出现冲突的情况下（如多个终端同时对同一数据进行修改），要有合理的冲突处理机制。可以采用分布式锁、冲突检测与协商等方式，解决数据一致性问题。例如，在多人协同的工程设计系统中，当多个设计师同时对一个设计文件进行修改时，系统要能够检测到冲突，并通过协商或优先级规则来确定最终的数据版本。

## 7.2 数据源采集、存储与传输要求

### **7.2.1** 根据工程测控的具体需求，能够精准地采集数据，满足不同传感器的精度要求。同时，可以灵活设置数据采集频率，对于一些关键参数或变化快速的物理量（如振动数据），能够实现高频率采集，而对于相对稳定的参数（如环境温度），可以适当降低采集频率，以减少数据传输和存储的压力。

### **7.2.4** 这种架构可以减少数据传输延迟，同时满足长期数据存储和大数据分析的需求。例如，边缘端存储最近几分钟的生产设备运行参数，用于本地实时监控和快速故障诊断，云端存储数天甚至数月的历史数据，用于设备性能分析和趋势预测。

## 7.4 故障恢复与迁移要求

### **7.4.1** 这包括硬件故障、软件故障、网络故障等各种情况。可以采用故障检测和诊断技术，及时发现故障原因，并通过自动恢复机制或人工干预进行修复。例如，当边缘端设备出现故障时，系统能够自动切换到其他正常的边缘设备继续提供服务。

## 7.6 可扩缩性要求

### **7.6.2** 边缘端可以通过添加新的计算节点或存储设备来满足不断增长的数据处理和存储需求；云端可以根据数据量和业务规模的增长，灵活地增加服务器集群、存储容量等资源。