ICS XXX
CCS XXX

# T/HEBQIA

团　　体　　标　　准

T/HEBQIA××××—2025

# 有源配电网短路参数数据库建设规范

（征求意见稿）

2025-××-××发布　　2025-××-××实施

河北省质量信息协会发布

# 目　　次

# 前　言

本文件依据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。

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专利的责任。

本文件由上海浦源科技有限公司提出。

本文件由河北省质量信息协会归口。

本文件起草单位：上海浦源科技有限公司、XXX。

本文件主要起草人：XXX。

# 有源配电网短路参数数据库建设规范

## 1 适用范围

本文件规定了有源配电网短路参数数据库的设计原则、结构设计、数据采集与存储、数据处理与分析、数据安全与权限管理、接口规范以及系统运维等内容。

本文件适用于电力企业或使用单位建设、管理和应用有源配电网短路参数数据库的相关工作，重点规范10kV～35kV配电网短路参数的数据采集、存储、处理、分析和应用。

## 2 规范性引用文件

下列文件中的条款通过本标准的引用而成为本标准的条款。凡是注日期的引用文件，其随后所有的修改单（不包括勘误的内容）或修订版均不适用于本规范。凡是不注日期的引用文件，其最新版本适用于本规范。

GB/T 22081 网络安全技术 信息安全控制

GB/T 29246 信息安全技术 信息安全管理体系 概述和词汇

GB/T 31497 网络安全技术 信息安全管理 监视、测量、分析和评价

GB/T 32914 信息安全技术 网络安全服务能力要求

GB/T 32916 信息安全技术 信息安全控制评估指南

GB/T 35290 信息安全技术 射频识别（RFID）系统安全技术规范

GB/T 42560 系统与软件工程 开发运维一体化 能力成熟度模型

GB/T 42566 系统与软件工程 功能规模测量 MkII功能点分析方法

GB/T 42588 系统与软件工程 功能规模测量 NESMA方法

GB/T 43206 信息安全技术 信息系统密码应用测评要求

GB/T 43207 信息安全技术 信息系统密码应用设计指南

GB/T 43269 信息安全技术 网络安全应急能力评估准则

GB/T 43435 信息安全技术 移动互联网应用程序（App）软件开发工具包（SDK）安全要求

GB/T 43445 信息安全技术 移动智能终端预置应用软件基本安全要求

GB/T 43557 信息安全技术 网络安全信息报送指南

GB/T 43577.1 信息安全技术 电子发现 第1部分：概述和概念

GB/T 43578 信息安全技术 通用密码服务接口规范

GB/T 44659.1 新能源场站及接入系统短路电流计算 第1部分：风力发电

GB/T 44659.2 新能源场站及接入系统短路电流计算 第2部分：光伏发电

GB/T 44659.3 新能源场站及接入系统短路电流计算 第3部分：储能电站

DL/T 721 配电自动化终端技术规范

DL/T 860.904 电力自动化通信网络和系统 第90-4部分：网络工程指南

DL/T 1529 配电自动化终端设备检测规程

DL/T 1883 配电网运行控制技术导则

DL/T 1941 可再生能源发电站电力监控系统网络安全防护技术规范

DL/T 2608 配电自动化终端运维技术规范
DL/T 2769 配电自动化终端检测平台技术规范
DL/T 2810 配电网智能分布式终端互操作技术规范
DL/T 5226 发电厂电力网络计算机监控系统设计技术规程
DL/T 5729 配电网规划设计技术导则

## 3 术语和定义

下列术语和定义适用于本文件。

3.1

**短路参数 short-circuit parameters**

表征电力系统短路特性的物理量，包括短路阻抗、短路容量、短路电流等。

3.2

**短路参数数据库 short-circuit parameter database**

用于存储、管理和分析配电网短路参数数据的软件系统，通常包括数据采集、存储、处理、分析和应用等功能模块。

3.3

**有源配电网 active distribution network**

含有分布式发电、储能、可控负荷等主动参与电网运行的电力设备，并具备双向潮流、自动控制和优化运行等特性的配电网络。

3.4

**数据完整性 data integrity**

确保数据在存储、传输和处理过程中保持完整、准确、一致，不被非授权修改或破坏的特性。

3.5

**数据可用性 data availability**

数据能够被授权用户在需要时按照期望方式正常访问和使用的特性。

3.6

**数据一致性 data consistency**

确保数据在不同存储位置、不同处理环节之间保持逻辑统一和相互协调的特性。

3.7

**数据保密性 data confidentiality**

确保数据只能被授权用户访问，不被非授权用户或实体获取或泄露的特性。

3.8

**数据备份 data backup**

将数据副本保存到其他存储介质上，以便在原始数据丢失、损坏时可以进行恢复的过程。

3.9

**数据恢复 data recovery**

将备份的数据副本重新导入系统，恢复到原始状态的过程。

3.10

**数据接口 data interface**

用于不同系统、模块之间交换数据的规范化连接点，包括接口定义、数据格式、通信协议等。

## 4 总则

4.1 有源配电网短路参数数据库建设应坚持"统一规划、分级管理、安全可靠、开放共享"的原则，建立标准化、规范化的数据管理体系。
4.2 数据库设计应具备良好的可扩展性和兼容性，满足有源配电网分布式能源接入比例不断提高、运行方式日益复杂的发展需求。
4.3 数据库建设应采用国产化数据库平台和自主可控技术，保障系统信息安全和稳定运行。
4.4 数据库应支持多源异构数据的采集、存储和处理，建立统一的数据标准和规范，实现数据资源的高效管理和共享利用。
4.5 数据库应建立完善的数据质量管理机制，确保短路参数数据的准确性、完整性、一致性和可用性。
4.6 数据库应与调度自动化系统、配电自动化系统、保护信息管理系统、配电网管理系统等实现数据共享和协同应用，为配电网规划、设计、建设、运行、保护与控制提供数据支撑。
4.7 数据库建设应符合国家和电力行业关于信息安全等级保护的相关要求，建立完善的安全防护和权限管理体系。

## 5 数据库总体设计

### 5.1 设计原则

5.1.1 数据库设计应遵循以下原则：

a) 安全性原则：保障数据安全，防止非授权访问和数据泄露，这是电力系统信息化建设的首要考虑因素；
b) 可靠性原则：确保系统稳定运行，具备容错和恢复能力，保证短路参数数据的连续性和一致性；
c) 实用性原则：满足用户实际需求，便于使用和维护，确保系统功能符合配电网运行管理实际工作需要；
d) 扩展性原则：适应业务发展和技术演进，便于功能扩展和系统升级，适应未来配电网结构和运行模式的变化；
e) 标准化原则：采用标准化数据结构和接口，便于系统集成和数据共享，实现与其他电力系统的无缝对接；

f） 经济性原则：合理设计系统规模和功能，降低建设和运维成本，实现投入产出的最优化。

5.1.2 数据库设计应满足以下技术要求：

a） 采用先进成熟的数据库技术和架构,如分布式数据库、关系型数据库与时序数据库结合等模式，确保系统稳定性和技术先进性；

b） 满足大容量、高并发、实时处理等性能要求，能够处理大量的短路参数测量数据和扰动事件数据；

c） 支持分布式部署和异地容灾备份，保障在设备故障或自然灾害情况下的数据安全；

d） 具备多层次安全防护和权限管理能力，严格控制数据访问和操作权限；

e） 支持标准化数据接口和多种数据格式，便于与其他系统进行数据交换和共享。

## 5.2 总体架构

短路参数数据库总体架构应采用分层设计，主要包括以下层次：

a） 数据资源层是基础层,负责原始数据采集、存储和管理，直接与测量终端和其他数据源系统对接；

b） 数据处理层是核心层，负责数据预处理、质量评估和参数计算，将原始数据转化为有价值的短路参数信息；

c） 数据服务层是中间层，负责提供数据查询、分析和应用接口，为上层应用提供标准化服务；

d） 应用层是业务层，负责实现各类基于短路参数的应用功能，如保护整定、规划分析、异常诊断等；

e） 展现层是界面层，负责数据可视化展示和用户交互界面，提供直观的数据展示和操作方式；

f） 安全管理层是贯穿各层的横向层，负责系统安全防护和权限管理，保障系统和数据安全。

## 5.3 功能要求

5.3.1 数据采集功能应满足以下要求：

a） 支持从短路参数测量终端、配电自动化系统、调度自动化系统等多源系统采集数据，实现数据集成；

b） 支持定时采集、触发采集和手动采集等多种采集方式，灵活适应不同场景需求；

c） 支持数据采集策略配置，包括采集周期、采集条件等，便于优化数据采集效率；

d） 具备采集状态监测和异常告警功能，及时发现采集异常并预警；

e） 支持数据采集过程中的初步验证和筛选，提高采集数据质量。

5.3.2 数据存储功能应满足以下要求：

a） 支持结构化数据、半结构化数据和非结构化数据的存储，适应不同类型数据的特点；

b） 支持历史数据归档和在线数据快速访问，平衡存储效率和访问性能；

c） 支持数据压缩存储，提高存储效率，节约存储空间；

d） 支持数据分区、分表和索引优化，提高查询效率，尤其是对大表的处理；

e） 支持数据备份和恢复机制，确保数据安全和可恢复性。

5.3.3 数据处理功能应满足以下要求：

a） 支持数据清洗、过滤、转换等预处理功能，确保数据安全和可恢复性；

b） 支持数据质量评估和异常数据标记，识别数据问题；

c） 支持短路参数计算和验证，将原始测量数据转化为短路参数值；

d） 支持数据统计分析和趋势分析，揭示数据变化规律；

e） 支持基于规则的数据校正和补全，处理数据缺失和异常。

5.3.4 数据管理功能应满足以下要求：

a) 支持数据目录管理和元数据管理，处理数据缺失和异常；
b) 支持数据版本管理和变更记录，追踪数据变化历史；
c) 支持数据生命周期管理，规范数据从创建到归档的全过程；
d) 支持数据血缘关系管理，记录数据来源和转换过程；
e) 支持数据标准和规范管理，确保数据一致性和规范性。

5.3.5 数据服务功能应满足以下要求：

a) 提供标准化数据查询接口，便于应用系统获取数据；
b) 提供数据订阅和推送服务，支持数据的实时分发；
c) 提供数据分析和挖掘服务，支持深度数据价值发现；
d) 提供数据可视化展示服务，直观展现数据特征和规律；
e) 提供数据导入导出服务，支持与外部系统的数据交换。

5.3.6 系统管理功能应满足以下要求：

a) 提供用户管理和权限管理功能，控制系统访问和使用权限；
b) 提供系统配置和参数管理功能，灵活调整系统行为；
c) 提供系统运行监控和日志管理功能，掌握系统运行状态；
d) 提供系统备份和恢复功能，确保系统可靠性；
e) 提供系统维护和升级功能，保持系统技术先进性。

## 6 数据库结构设计

### 6.1 逻辑结构

6.1.1 短路参数数据库的逻辑结构应包括以下主要内容：

a) 基础数据：包括电网拓扑模型、设备参数、测量点信息等，构成系统的基础信息框架；
b) 测量数据：包括原始测量数据、扰动事件数据等，是短路参数计算的数据来源；
c) 参数数据：包括短路阻抗、短路容量等计算结果，是系统的主要数据产出；
d) 质量评估数据：包括数据质量指标、可信度评价结果等，用于评价数据可靠性；
e) 系统数据：包括用户信息、权限设置、系统配置等，支持系统运行和管理；
f) 应用数据：包括应用分析结果、报表数据等，服务于具体业务应用。

6.1.2 短路参数数据库的逻辑关系设计应遵循实体-关系模型（E-R 模型）理论，明确各实体间的关联关系，保持数据的一致性和完整性。

6.1.3 数据实体间的关系类型主要包括：

a) 一对一关系：如测量点与测量设备的关系，一个测量点对应一个测量设备，反之亦然；
b) 一对多关系：如配电线路与测量点的关系，一条配电线路可以包含多个测量点，但一个测量点只属于一条线路；
c) 多对多关系：如测量终端与短路参数的关系，一个测量终端可以测量多个短路参数，一个短路参数也可能由多个终端测量。

6.1.4 逻辑结构设计应考虑业务需求的变化和系统的扩展，留有适当的扩展空间。

### 6.2 物理结构

6.2.1 短路参数数据库的物理结构应根据数据特性、访问模式和性能要求进行设计，主要包括表空间设计、表结构设计、索引设计和分区设计等。

6.2.2 表结构设计应满足以下要求：

a) 表名应符合命名规范，具有描述性和唯一性，使用清晰的命名约定；
b) 字段设计应考虑数据类型、长度、约束条件等，选择合适的数据类型和长度，并设置必要的约束；
c) 主键设计应确保唯一性和最小性，选择最合适的列或列组合作为主键；
d) 外键设计应保证数据完整性和一致性，明确表间的引用关系；
e) 应适当进行表的规范化设计，避免数据冗余和异常，但也考虑查询效率，在必要时进行适当的反规范化。

6.2.3 索引设计应满足以下要求：

a) 为经常查询的字段建立适当的索引，提高查询效率；
b) 为主键和外键字段建立索引，优化关联查询性能；
c) 合理使用单列索引和复合索引，根据查询条件选择合适的索引类型；
d) 避免过多索引导致性能下降，考虑索引对写入操作的影响；
e) 定期维护和优化索引，如重建索引、更新统计信息等，保持索引的有效性。

6.2.4 分区设计应满足以下要求：

a) 大型表应根据数据特性进行适当分区，如时间分区、范围分区等；
b) 分区策略可采用范围分区、列表分区或哈希分区等，根据数据分布特点和查询需求选择合适的分区方式；
c) 分区设计应便于数据管理和查询优化，如便于历史数据归档和区域性数据查询；
d) 分区设计应考虑数据增长趋势和存储策略，预留足够的增长空间，并便于动态调整。

6.2.5 物理结构设计应考虑数据库性能和存储效率，进行适当的性能优化和存储优化。

### 6.3 数据模型

6.3.1 基础数据模型应包括以下主要实体：

a) 变电站：记录变电站基本信息，包括名称、位置、电压等级等；
b) 配电线路：记录配电线路基本信息，如线路名称、起止点、线路长度、导线类型等；
c) 设备信息：记录配电网设备基本信息，包括设备类型、规格参数、安装位置等；
d) 测量点：记录短路参数测量点信息，如点位编号、地理位置、测量参数类型等；
e) 测量终端：记录短路参数测量终端信息，包括终端编号、型号、安装时间等；
f) 拓扑结构：记录配电网拓扑连接关系，描述节点间的连接方式和网络结构。

6.3.2 测量数据模型应包括以下主要实体：

a) 原始波形数据：记录电压、电流原始波形数据，包括采样时间、采样值、通道信息等；
b) 扰动事件数据：记录系统扰动事件信息，如事件类型、发生时间、持续时间、触发条件等；
c) 测量过程数据：记录测量过程中的中间计算结果，如扰动前后的电压电流变化量、阻抗计算中间值等；
d) 测量状态数据：记录测量过程的状态信息，包括测量模式、采样状态、计算状态等。

6.3.3 参数数据模型应包括以下主要实体：

a) 短路阻抗数据：记录测量点的短路阻抗值，包括阻抗幅值、相角、测量时间等；
b) 短路容量数据：记录测量点的短路容量值，如三相短路容量、单相短路容量、测量时间等；

c) 时序参数数据：记录短路参数的时间序列变化，反映参数随时间的变化趋势；
d) 参数统计数据：记录短路参数的统计特征，如平均值、标准差、最大最小值等。

6.3.4 质量评估数据模型应包括以下主要实体：

a) 数据质量指标：记录数据质量评估指标值，如完整率、准确率、一致性系数等；
b) 参数可信度：记录短路参数的可信度等级，如 A、B、C、D 级可信度分类；
c) 异常数据标记：记录异常数据的标记信息，包括异常类型、异常原因、处理建议等；
d) 质量评估日志：记录质量评估过程信息，如评估时间、评估方法、评估结果等。

6.3.5 系统数据模型应包括以下主要实体：

a) 用户信息：记录系统用户基本信息，如用户名、密码、角色、联系方式等；
b) 角色权限：记录系统角色和权限设置，定义不同角色的操作权限范围；
c) 系统配置：记录系统配置参数，包括系统常量、运行参数、界面设置等；
d) 操作日志：记录系统操作历史记录，包括操作用户、操作时间、操作内容等。

6.3.6 数据模型设计应考虑以下优化策略：

a) 适当进行数据冗余设计，提高查询效率，如在相关表中复制常用字段；
b) 合理使用中间表和汇总表，优化复杂查询，预先计算和存储常用统计结果；
c) 设计历史数据归档策略，提高活跃数据访问效率，将历史数据移至专用存储区；
d) 根据业务需求设计特定的应用视图，简化复杂查询逻辑，提供便捷的数据访问方式；
e) 定期评估和优化数据模型设计，根据系统运行情况和用户反馈进行调整，保持模型的有效性。

## 7 数据采集与存储

### 7.1 数据来源

7.1.1 短路参数测量终端产生的数据应包括：

a) 电压、电流波形数据，记录系统扰动前后的详细波形变化；
b) 扰动事件数据，记录扰动类型、时间、特征等信息；
c) 计算的短路阻抗和短路容量值，即终端自身计算的参数结果；
d) 测量质量评估数据，反映测量数据的可靠性和准确性；
e) 终端状态和告警信息，反映设备运行状态和异常情况。

7.1.2 从配电自动化系统采集的数据应包括：

a) 配电网拓扑信息记录网络结构和连接关系，是理解短路参数分布的基础；
b) 配电设备运行状态记录各类设备的工作状态，帮助分析系统运行条件；
c) 开关操作记录提供网络结构变化信息，便于关联分析参数变化原因；
d) 故障录波信息记录系统故障时的波形数据，可用于交叉验证短路参数；
e) 系统运行方式信息记录网络运行配置，是参数分析的重要背景数据。

7.1.3 从调度自动化系统采集的数据应包括：

a) 变电站运行数据提供上级电网运行信息，反映电源侧条件；
b) 配电网运行方式记录整体网络运行架构，是系统级分析的基础；
c) 主网短路参数提供上级网络强度信息，是配电网短路参数的重要影响因素；
d) 系统运行参数记录频率、电压等关键指标，反映系统整体状态；
e) 负荷预测数据提供未来负荷变化趋势，用于短路参数变化预测分析。

7.1.4 从保护信息管理系统采集的数据应包括：

a) 保护装置配置信息记录保护设备的安装位置和类型；
b) 保护定值信息记录当前使用的保护整定参数，反映系统保护策略；
c) 故障录波数据记录故障时的详细波形，是验证短路参数的重要参考；
d) 保护动作记录提供故障响应信息，便于关联分析系统保护性能；
e) 保护装置状态信息记录设备运行状况，是系统可靠性分析的基础。

7.1.5 从电力物联网平台采集的数据应包括：

a) 分布式电源运行数据记录各类新能源的出力和运行模式，是分析其对短路参数影响的关键；
b) 配电网感知设备数据提供网络运行状态的广域感知信息；
c) 智能终端监测数据记录关键节点的运行参数，丰富系统监测数据；
d) 环境气象信息记录天气条件，便于分析外部因素影响；
e) 用电负荷数据记录系统负荷分布和变化，是理解短路参数变化的重要背景。

7.1.6 通过手工录入的数据应包括：

a) 配电网规划设计数据提供网络建设和改造信息，用于预测分析；
b) 设备台账信息记录详细的设备参数，是模型计算的基础数据；
c) 短路参数理论计算值提供理论参考，用于实测数据的比对验证；
d) 历史短路参数记录保存历史测试数据，用于长期趋势分析；
e) 特殊运行方式信息记录非常规运行配置，便于分析特殊情况下的系统特性。

## 7.2 采集频率

7.2.1 数据采集频率应根据数据类型、变化特性和应用需求确定，综合考虑数据价值和系统资源消耗。

7.2.2 实时数据采集频率应满足以下要求：

a) 短路参数测量结果：每次有效测量后实时采集，确保测量数据的及时性；
b) 测量终端状态数据：采集周期不超过 1 分钟，保证对设备状态的及时监控；
c) 系统运行方式变化：运行方式变化时实时采集，及时反映网络结构调整；
d) 分布式电源出力数据：采集周期不超过 5 分钟，确保能够监测到其出力变化对系统的影响。

7.2.3 周期性数据采集频率应满足以下要求：

a) 短路参数统计数据：每日至少采集 1 次，用于日常分析和趋势监测；
b) 配电网拓扑数据：每日至少更新 1 次，确保网络结构信息的准确性；
c) 设备参数数据：每周至少更新 1 次，反映设备特性的变化；
d) 运行管理数据：每月至少更新 1 次，满足管理统计和报表需求。

7.2.4 事件触发数据采集应满足以下要求：

a) 系统扰动事件：事件发生时触发采集，记录完整的扰动过程数据；
b) 短路参数异常：参数异常时触发采集，及时捕获异常状态信息；
c) 保护装置动作：动作发生时触发采集，记录故障响应过程；
d) 系统运行方式变化：变化发生时触发采集，记录网络结构调整信息。

7.2.5 数据采集频率应可配置，根据系统运行状况和应用需求进行动态调整。

## 7.3 存储格式

7.3.1 短路参数数据库的存储格式应标准化、规范化，便于数据共享和交换。

7.3.2 结构化数据存储格式应满足以下要求：

a) 采用标准的关系型数据库表结构，遵循数据库设计规范；
b) 字段类型和长度符合数据特性，选择合适的数据类型和长度；
c) 采用统一的编码标准，如 UTF-8 字符集，确保多语言支持；
d) 遵循数据库设计规范，如规范化设计、索引优化等，保证数据存储的效率和一致性。

7.3.3 半结构化数据存储格式应满足以下要求：

a) 采用 XML、JSON 等标准格式，便于跨平台数据交换；
b) 定义清晰的数据结构和标签，明确数据含义和组织方式；
c) 支持灵活的数据扩展，允许结构变化和字段添加；
d) 便于数据解析和处理，提供高效的数据访问和操作方法。

7.3.4 非结构化数据存储格式应满足以下要求：

a) 波形数据采用 COMTRADE 或自定义格式，确保电力行业兼容性；
b) 图像数据采用标准图像格式，如 JPEG、PNG 等，兼顾质量和存储效率；
c) 文档数据采用 PDF、DOC 等标准格式，确保可读性和兼容性；
d) 多媒体数据采用标准压缩格式，如 MP4、MP3 等，优化存储空间和传输效率。

7.3.5 数据存储格式详细规范见附录 C。

## 7.4 数据保存期限

7.4.1 数据保存期限应根据数据类型、价值和应用需求确定，综合考虑存储成本和数据价值。

7.4.2 原始数据保存期限应满足以下要求：

a) 电压、电流波形数据由于体积大，保存期限不少于 3 个月，满足短期分析需求；
b) 扰动事件数据作为重要参考信息，保存期限不少于 1 年，便于历史事件分析；
c) 测量过程数据主要用于短期问题排查，保存期限不少于 3 个月；
d) 测量终端状态数据用于设备健康监测，保存期限不少于 1 年，支持长期设备状态分析。

7.4.3 短路参数数据保存期限应满足以下要求：

a) 短路阻抗和短路容量值作为核心数据应永久保存，用于长期趋势分析和历史比对；
b) 参数质量评估数据保存不少于 3 年，用于数据质量长期评价；
c) 参数统计分析数据保存不少于 5 年，支持中长期系统特性研究；
d) 参数变化趋势数据保存不少于 5 年，用于识别系统长期演化规律。

7.4.4 系统运行数据保存期限应满足以下要求：

a) 配电网拓扑变化记录保存不少于 5 年，记录网络结构演变历史；
b) 运行方式变化记录保存不少于 3 年，反映系统运行策略变化；
c) 系统操作日志保存不少于 3 年，便于追踪历史操作和责任判定；
d) 系统告警记录保存不少于 1 年，用于系统健康状态分析。

7.4.5 超过保存期限的数据应进行归档处理，重要数据应保留离线备份。

# 8 数据处理与分析

## 8.1 数据预处理

8.1.1 数据清洗应满足以下要求：

a) 检测并处理缺失值、异常值和重复值，确保数据完整性和准确性；

b) 识别并修正数据格式错误，如时间格式不一致、小数点位置错误等；
c) 检查并修正数据逻辑错误，如参数值超出物理可能范围、相位关系不合理等；
d) 统一数据单位和精度，确保数据一致性；
e) 记录数据清洗过程和结果，保证数据处理的可追溯性，便于质量评估和问题排查。

8.1.2 数据过滤应满足以下要求：

a) 根据预设规则过滤无效数据，如信号质量差、异常波动的数据；
b) 根据时间、空间等条件筛选数据，满足特定分析需求；
c) 根据数据质量指标筛选高质量数据，优先使用可靠数据；
d) 根据应用需求提取关键数据，减少数据处理量；
e) 保留原始数据和过滤记录，确保数据处理的透明性和可追溯性，便于后续验证和审核。

8.1.3 数据转换应满足以下要求：

a) 进行数据格式转换，将不同来源的数据转换为统一格式，满足处理需求；
b) 进行数据单位转换，确保计量单位的一致性，避免单位混淆导致的错误；
c) 进行坐标系转换，使空间数据在统一的参考系下表示，便于空间分析；
d) 进行时区转换，保持时间数据的一致性，特别是跨时区数据的整合；
e) 记录数据转换规则和过程，确保转换操作的透明性和可追溯性。

8.1.4 数据集成应满足以下要求：

a) 识别不同来源的同一实体数据，确保转换操作的透明性和可追溯性；
b) 消除数据冗余和矛盾，解决不同来源数据的不一致问题；
c) 建立数据关联关系，明确数据间的逻辑和业务关联；
d) 形成统一的数据视图，为上层应用提供集成的数据访问接口；
e) 保持数据一致性和完整性，确保集成后数据的质量和可靠性。

8.1.5 数据压缩应满足以下要求：

a) 对长期存储的历史数据进行压缩，减少存储空间占用；
b) 采用无损或适当有损压缩算法，根据数据重要性选择适当的压缩方式；
c) 保证压缩后数据的可恢复性，必要时能还原原始数据；
d) 压缩比和压缩效率满足系统要求，在压缩率和性能间取得平衡；
e) 记录压缩方式和参数，便于数据解压和使用。

## 8.2 数据质量评估

8.2.1 数据质量评估指标应包括但不限于：

a) 完整性指标衡量数据记录的完整程度，如缺失率、有效记录比例等；
b) 准确性指标衡量数据值与真实值的接近程度，如误差率、准确性评分等；
c) 一致性指标衡量数据内部和外部的逻辑一致程度，如与规则符合率、跨表一致性等；
d) 时效性指标衡量数据的更新及时程度，如数据延迟时间、更新频率等；
e) 有效性指标衡量数据对应用的有用程度，如业务适用性、决策支持度等；
f) 唯一性指标衡量数据实体表示的唯一程度，如重复率、唯一标识完整性等；
g) 可理解性指标衡量数据含义的清晰程度，如标准符合度、注释完整性等；
h) 可验证性指标衡量数据内容可被验证的程度，如来源可追溯性、验证难度等。

8.2.2 数据质量评估方法应包括但不限于：

a) 规则检验法基于预定义规则进行校验，如数据范围检查、格式验证等，适用于已知规则的场景；

b) 统计分析法基于统计特征进行评估，如异常值检测、分布分析等，适用于大量数据的整体评价；
c) 对比验证法与参考数据进行对比，评估数据一致性，如与理论值比较、与历史数据比较等；
d) 专家评估法由领域专家进行评估，基于经验和专业知识判断数据质量，适用于复杂数据的质量评价；
e) 综合评分法综合多指标进行评分，形成整体质量评价，如加权评分、多维度评价等，提供全面的质量度量。

8.2.3 针对质量评估发现的问题，应采取以下处理措施：

a) 数据修正是对可修正的错误进行更正，如格式错误、计算错误等；
b) 数据补全针对缺失数据，通过插值、估算等方法进行合理补全；
c) 数据标记是对质量问题数据进行标记，便于使用者了解数据限制；
d) 数据过滤用于处理严重质量问题数据，将其从分析数据集中剔除；
e) 数据替换是用高质量数据替换问题数据，如用模型计算值替代异常测量值；
f) 源头改进是向数据源提供反馈，改进数据生产过程，从源头提高数据质量。

8.2.4 应定期生成数据质量评估报告，内容包括：

a) 质量评估范围和时间：明确报告的覆盖面和时间段，界定评估边界；
b) 质量问题统计和分类：提供问题的数量和类型分布，展示质量状况；
c) 质量问题原因分析：深入探讨问题根源，为改进提供依据；
d) 质量改进建议和措施：提出具体的改进方向和行动计划；
e) 质量趋势分析和预测：展示质量变化趋势，预判未来质量状况。

## 8.3 数据分析方法

8.3.1 短路参数数据的统计分析应包括：

a) 描述性统计分析基本统计特征，如均值、中位数、标准差等，提供数据的总体概况；
b) 分布特性分析研究数据的分布形态，如分布类型、偏度、峰度等，揭示数据的概率特性；
c) 相关性分析探索短路参数与影响因素的相关关系，如与负荷、分布式电源出力的相关性，识别关键影响因素；
d) 时序特性分析研究参数随时间的变化规律，如周期性、趋势性等，揭示时间维度的变化模式；
e) 空间特性分析研究参数的空间分布特征，如空间分布、聚类特征等，发现区域差异和集聚现象。

8.3.2 短路参数数据的趋势分析应包括：

a) 时间序列分析识别长期变化趋势，如线性趋势、非线性趋势等，揭示参数的演变方向；
b) 季节性分析识别周期性变化规律，如日内、周内、季节性变化等，了解参数的周期特性；
c) 突变点检测识别参数突变时点，发现系统重大变化或异常事件；
d) 趋势预测基于历史数据预测未来参数变化，为系统规划和运行提供参考；
e) 敏感性分析研究影响趋势的关键因素，识别哪些因素对趋势变化影响最大，为趋势控制提供依据。

8.3.3 短路参数数据的对比分析应包括：

a) 历史数据对比：将当前参数与历史参数值比较，评估变化幅度和方向；
b) 理论值对比：将实测值与理论计算值比较，验证测量准确性和模型有效性；
c) 不同区域对比：分析不同区域间的参数差异，发现区域特性和问题；
d) 不同运行方式对比：研究运行方式变化对参数的影响，为运行优化提供依据；
e) 分布式电源影响对比：分析分布式电源接入前后的参数变化，评估新能源的系统影响。

8.3.4 短路参数数据的关联分析应包括：

a) 系统拓扑关联：分析网络结构变化与短路参数的关系，理解拓扑调整的影响；
b) 设备参数关联：研究设备参数变化（如变压器抽头位置、线路阻抗）对短路参数的影响；
c) 负荷特性关联：分析负荷特性（如负荷类型、功率因数）与短路参数的关系；
d) 分布式电源关联：研究分布式电源特性（如类型、容量、控制模式）对短路参数的影响；
e) 故障特性关联：分析短路参数与系统故障特性（如故障类型、位置、影响范围）的关系，为故障预防和处理提供依据。

8.3.5 短路参数数据的模型分析应包括：

a) 回归分析：建立参数与影响因素的关系模型，如多元线性回归、非线性回归等，量化各因素的影响；
b) 聚类分析：识别具有相似特性的参数组，发现系统中的相似区域或节点；
c) 分类分析：对参数进行类别划分，如高中低短路容量区域划分，便于分区管理；
d) 异常检测：识别异常参数值，发现系统异常状态或测量问题；
e) 知识发现：挖掘参数变化的隐含规律，从大量数据中提取有价值的知识和模式，提升对系统特性的理解。

## 9 数据安全与权限管理

### 9.1 安全防护

9.1.1 物理安全防护应满足以下要求：

a) 服务器和存储设备应部署在安全机房，配备专业的防护设施；
b) 机房应具备访问控制、防火、防水、防雷等措施，全面防范物理风险；
c) 关键设备应采用冗余配置和不间断电源，确保系统连续运行；
d) 应建立完善的机房管理制度和应急预案，规范机房运行管理；
e) 遵循电力行业信息系统物理安全相关规定，符合行业标准和要求。

9.1.2 网络安全防护应满足以下要求：

a) 采用网络隔离和访问控制技术，如划分安全区域、部署安全网关等，控制网络访问路径；
b) 部署防火墙、入侵检测系统等安全设备，形成多层次防护体系；
c) 采用 VPN 等安全传输技术，保护数据传输安全；
d) 定期进行网络安全扫描和评估，发现并修复安全隐患；
e) 建立网络安全事件监测和响应机制，及时发现和处置安全事件。

9.1.3 系统安全防护应满足以下要求：

a) 操作系统和数据库系统应及时更新安全补丁，及时发现和处置安全事件；
b) 应用系统应进行安全性测试和加固，排除常见安全风险；
c) 部署防病毒软件和终端安全管理系统，防范恶意程序威胁；
d) 建立安全审计和日志管理机制，记录系统操作和安全事件；
e) 定期进行系统安全评估和改进，持续提升安全水平。

9.1.4 数据安全防护应满足以下要求：

a) 实施数据分级分类管理，根据数据敏感性和重要性进行分级保护；
b) 采用数据加密和脱敏技术，保护敏感数据安全；
c) 建立数据访问控制和审计机制，严格控制数据访问权限；

d) 实施数据备份和恢复管理，防止数据丢失；
e) 防范数据泄露和篡改风险，采取技术和管理措施保护数据完整性和保密性。

9.1.5 应用安全防护应满足以下要求：

a) 应用系统应采用安全开发方法，在设计和编码阶段注重安全性；
b) 实施代码安全审计和漏洞扫描，检查并修复应用安全问题；
c) 建立应用访问控制和认证机制，控制应用功能访问权限；
d) 防范常见 Web 安全威胁，如 SQL 注入、XSS 攻击等；
e) 定期进行应用安全评估和改进，持续提高应用安全水平。

## 9.2 权限管理

9.2.1 用户管理应满足以下要求：

a) 建立规范的用户账号管理制度，明确账号申请、审批、变更和注销流程；
b) 实行用户实名制和唯一性管理，确保账号与实际用户一一对应；
c) 定义清晰的用户分类和层次，如系统管理员、业务管理员、普通用户等；
d) 建立用户生命周期管理机制，从创建到注销的全过程管理；
e) 定期进行用户账号审核和清理，确保账号设置符合最小权限原则。

9.2.2 角色管理应满足以下要求：

a) 基于业务需求定义系统角色，如数据管理员、数据分析员、报表用户等；
b) 明确各角色的职责和权限范围，详细定义可访问的功能和数据；
c) 实行角色与用户的多对多关联，一个用户可拥有多个角色，一个角色可分配给多个用户；
d) 定期评估和优化角色设置，确保角色定义与业务需求一致；
e) 建立角色冲突和分离机制，避免权限过度集中，防范权限滥用风险。

9.2.3 权限分配应满足以下要求：

a) 按照最小授权原则分配权限，仅授予用户完成工作所需的最小权限；
b) 实行基于角色的访问控制（RBAC），通过角色分配权限，简化管理复杂度；
c) 细化到数据对象和操作类型的权限控制，实现精细化的权限管理；
d) 支持权限的临时授权和回收，满足临时业务需求；
e) 记录权限变更历史，便于审计和追溯。

9.2.4 访问控制应满足以下要求：

a) 实施多因素认证机制，如密码结合短信验证码、硬件令牌等，提高身份认证安全性；
b) 建立会话管理和超时控制，自动终止长时间无操作的会话；
c) 记录和审计用户访问行为，监控异常访问活动；
d) 实施异常访问检测和阻断，发现并阻止可疑访问尝试；
e) 定期评估访问控制有效性，确保控制措施的持续有效。

## 9.3 数据备份与恢复

9.3.1 数据备份策略应满足以下要求：

a) 定义不同类型数据的备份周期和保留期限，如关键数据每日备份、一般数据每周备份等；
b) 实施全量备份和增量备份相结合的方式，优化备份效率和存储空间；
c) 备份数据应存储在物理隔离的介质上，避免主存储系统故障连带影响备份数据；
d) 关键数据应保留多个备份副本，提高数据冗余度和可靠性；

e) 定期测试备份数据的可用性，确保备份数据可以成功恢复。

9.3.2 数据备份方式应满足以下要求：

a) 支持自动备份和手动备份，满足定期备份和特殊情况下的临时备份需求；
b) 支持热备份和冷备份，热备份适用于需要持续服务的系统，冷备份提供更高的数据隔离度；
c) 支持本地备份和远程备份，提供不同级别的灾难恢复能力；
d) 支持物理备份和逻辑备份，物理备份速度快，逻辑备份便于选择性恢复；
e) 采用数据压缩和加密技术，提高存储效率和备份数据安全性。

9.3.3 数据恢复机制应满足以下要求：

a) 制定详细的数据恢复流程和规程，明确恢复步骤和责任人；
b) 支持全量恢复和部分恢复，根据实际需求选择适当的恢复范围；
c) 支持时间点恢复（Point-in-Time Recovery），能够恢复到指定时间点的数据状态；
d) 恢复操作应有严格的授权和审核，防止未经授权的恢复操作造成数据混乱；
e) 记录恢复操作的全过程日志，便于后续审计和问题分析。

9.3.4 容灾备份应满足以下要求：

a) 建立同城灾备或异地灾备系统，提供不同级别的灾难恢复能力；
b) 实现数据的实时或准实时同步，最小化灾难情况下的数据丢失；
c) 定期进行灾备切换演练，验证灾备系统的可用性和切换的可靠性；
d) 制定完善的灾难恢复预案，明确灾难情况下的响应流程和恢复步骤；
e) 确保关键业务的连续性，在主系统故障时能够快速切换到备用系统，保持业务连续运行。

## 10 接口规范

### 10.1 内部接口

10.1.1 数据库系统内部模块间接口应满足以下要求：

a) 采用标准化接口定义，明确接口名称、功能和参数规范；
b) 明确接口参数和返回值，包括参数类型、格式、取值范围等；
c) 规范接口调用方式和流程，确定同步/异步调用、错误处理机制等；
d) 实现接口版本管理，确保接口向下兼容，支持系统平滑升级；
e) 记录接口调用日志，便于问题排查和性能分析。

10.1.2 数据库系统内部数据交换接口应满足以下要求：

a) 定义统一的数据交换格式，如 JSON、XML 等标准格式；
b) 支持批量数据交换和实时数据交换，满足不同场景需求；
c) 确保数据交换的完整性和一致性，包括事务处理、冲突解决等机制；
d) 提供数据转换和映射功能，处理不同模块间的数据格式差异；
e) 监控数据交换过程和状态，及时发现并处理异常情况。

10.1.3 数据库系统内部查询接口应满足以下要求：

a) 提供灵活的数据查询功能，支持多种查询条件和方式；
b) 支持多条件组合查询和模糊查询，满足复杂的查询需求；
c) 提供数据分页和排序功能，优化大量数据的展示和处理；
d) 优化查询性能和响应时间，如使用缓存、索引等技术；

e) 限制复杂查询的资源消耗，防止单个查询影响系统整体性能。

## 10.2 外部接口

10.2.1 数据采集接口应满足以下要求：

a) 支持从各类数据源采集数据，如测量终端、自动化系统等；
b) 定义标准化的数据采集协议，便于不同系统集成；
c) 提供数据转换和预处理功能，处理原始数据格式和质量问题；
d) 支持数据采集状态监控，实时掌握采集过程和结果；
e) 实现数据采集的安全认证和加密，保护数据传输安全。

10.2.2 数据共享接口应满足以下要求：

a) 提供标准化的数据共享服务，便于其他系统调用；
b) 支持多种数据格式和交换方式，适应不同系统的集成需求；
c) 实现数据共享的权限控制，防止敏感数据泄露；
d) 提供数据共享记录和审计，监控数据访问和使用情况；
e) 满足实时性和可靠性要求，确保共享数据的及时性和完整性。

10.2.3 应用接口应满足以下要求：

a) 提供面向业务应用的标准接口，支持各类应用系统集成；
b) 支持 RESTful、SOAP 等标准接口方式，便于不同技术平台的应用对接；
c) 提供接口说明文档和调用示例，降低应用开发的难度；
d) 实现接口安全认证和访问控制，防止非授权访问；
e) 记录接口调用日志和异常情况，便于问题排查和安全监控。

## 10.3 标准接口规范

10.3.1 接口协议应满足以下要求：

a) 采用 HTTP/HTTPS、WebService 等标准协议，确保广泛兼容性；
b) 明确协议版本和兼容性要求，便于系统演化和升级；
c) 规范请求和响应格式，包括参数格式、返回值结构等；
d) 定义错误码和异常处理机制，确保接口调用的可靠性；
e) 确保通信安全和数据完整性，如使用 HTTPS、数据签名等技术。

10.3.2 数据格式应满足以下要求：

a) 采用 XML、JSON 等标准数据格式，确保跨平台兼容性；
b) 定义清晰的数据结构和字段，包括字段名称、类型、含义等；
c) 规范数据类型和编码标准，如日期格式、字符编码等；
d) 支持数据版本管理和兼容性，确保接口升级时的向后兼容；
e) 优化数据体积和传输效率，如使用压缩、精简冗余字段等技术。

10.3.3 接口文档应满足以下要求：

a) 提供完整、清晰的接口说明，包括接口目的、使用场景等；
b) 描述接口功能、参数和返回值，详细说明每个字段的含义和格式；
c) 提供接口调用示例和测试方法，便于开发人员理解和使用；
d) 说明错误处理和异常情况，包括可能的错误码和处理建议；
e) 保持文档的及时更新和版本控制，确保文档与实际接口一致。

# 11 系统运维

## 11.1 运行维护

11.1.1 日常运维应满足以下要求：

a) 制定系统运行监控计划，明确监控内容、频率和责任人；
b) 定期检查系统运行状态，包括服务器、数据库、应用服务等各个环节；
c) 及时处理系统告警和异常，保证问题的快速响应和解决；
d) 定期进行数据备份和验证，确保数据安全可靠；
e) 记录运维操作和系统变更，建立完整的运维记录，便于追溯和分析。

11.1.2 定期维护应满足以下要求：

a) 定期进行数据库优化和整理，如索引重建、统计信息更新等，保持数据库高效运行；
b) 定期清理临时文件和日志，防止磁盘空间耗尽；
c) 定期检查和更新系统补丁，修复已知安全漏洞；
d) 定期检查和优化系统配置，根据运行情况调整参数设置；
e) 定期进行系统安全检查，发现并消除安全隐患。

11.1.3 版本升级应满足以下要求：

a) 制定详细的升级方案和回退计划，明确升级步骤、风险点和回退机制；
b) 进行充分的升级前测试，验证新版本的功能和兼容性；
c) 选择适当的升级时间窗口，最小化对业务的影响；
d) 记录升级过程和结果，包括版本变更、配置修改等信息；
e) 升级后进行系统验证和性能评估，确认系统正常运行并达到预期效果。

## 11.2 性能监控

11.2.1 性能监控指标应包括但不限于：

a) 系统资源指标：监控基础硬件资源，如 CPU、内存、磁盘、网络等的使用率和负载情况；
b) 数据库指标：监控数据库系统性能，如连接数、事务量、缓存命中率等关键参数；
c) 应用性能指标：监控应用系统运行状态，如响应时间、吞吐量、并发数等用户体验相关指标；
d) 数据处理指标：监控数据处理过程，如数据处理速率、队列长度等，反映数据流转效率；
e) 用户体验指标：监控最终用户感知，如页面加载时间、操作响应时间等，直接反映系统使用体验。

11.2.2 性能监控方式应满足以下要求：

a) 采用自动化监控工具如 Zabbix、Prometheus 等，实现系统性能的自动监测；
b) 实现实时监控和历史数据分析，既能及时发现问题，又能分析长期趋势；
c) 设置合理的告警阈值和规则，在性能异常时触发告警；
d) 提供可视化监控界面，直观展示系统性能状态；
e) 支持监控数据的统计分析和报表生成，便于性能分析和决策支持。

11.2.3 性能优化应满足以下要求：

a) 定期分析性能瓶颈和问题，识别系统中的性能短板；
b) 优化数据库结构和索引，提高数据库查询和处理效率；
c) 优化查询语句和存储过程，提高数据访问性能；
d) 调整系统配置参数，如缓存大小、连接池设置等，优化系统运行环境；

e) 合理分配和扩展系统资源，如增加服务器、调整负载均衡等，提高系统整体处理能力。

## 11.3 故障处理

11.3.1 故障分类应包括但不限于：

a) 硬件故障：服务器、存储、网络设备等物理设备的故障，如硬盘故障、网络中断等；
b) 软件故障：操作系统、数据库、应用系统等软件层面的故障，如系统崩溃、服务不可用等；
c) 数据故障：数据本身的问题，如数据损坏、丢失、不一致等；
d) 安全故障：系统安全方面的问题，如安全漏洞、攻击、病毒感染等；
e) 操作故障：人为因素导致的问题，如误操作、违规操作等。

11.3.2 故障响应应满足以下要求：

a) 建立故障报告和响应机制，确保故障能够及时报告和处理；
b) 明确故障处理责任和流程，包括报告、响应、处理、恢复各环节的责任人和操作流程；
c) 根据故障影响程度确定响应优先级，优先处理影响大的关键故障；
d) 提供多种故障报告渠道，如系统告警、电话、邮件等，确保故障信息及时传达；
e) 记录故障响应过程和结果，便于后续分析和评估。

11.3.3 故障处理应满足以下要求：

a) 快速定位故障原因和范围，明确故障的具体表现、发生位置和影响范围；
b) 采取有效措施恢复系统功能，如系统重启、数据恢复、切换备用系统等；
c) 最小化故障影响和恢复时间，尽可能减少对业务的中断；
d) 按照预定流程进行故障升级，对于无法快速解决的复杂故障及时上报和寻求支持；
e) 记录故障处理全过程，包括故障现象、原因分析、处理措施和结果等。

11.3.4 故障分析与预防应满足以下要求：

a) 对故障进行深入分析和根本原因调查，找出故障的真正根源，而不仅是表面现象；
b) 总结故障经验和教训，从故障中学习，提高系统和运维能力；
c) 制定并实施故障预防措施，针对发现的问题进行系统性改进；
d) 更新和完善故障处理流程，根据实际情况不断优化应对策略；
e) 定期进行故障演练和应急响应训练，提高团队处理突发事件的能力。

# 附 录 A
（规范性附录）
短路参数数据库表结构设计

## A.1 基础信息表结构

### A.1.1 配电网拓扑信息表

配电网拓扑信息表用于存储配电网的拓扑结构信息，包括线路、节点、开关等基础信息。主要字段如下：

表 A.1 配电网拓扑信息表

| 字段名 | 数据类型 | 描述 | 备注 |
|---|---|---|---|
| TOPO_ID | VARCHAR(32) | 拓扑 ID | 主键 |
| PARENT_ID | VARCHAR(32) | 父级 ID | 可为空 |
| TOPO_TYPE | VARCHAR(16) | 拓扑类型 | 变电站/线路/节点/开关等 |
| TOPO_CODE | VARCHAR(32) | 拓扑编码 | 唯一编码 |
| TOPO_NAME | VARCHAR(64) | 拓扑名称 | |
| VOLTAGE_LEVEL | INT | 电压等级 | 单位：kV |
| LONGITUDE | DOUBLE | 经度 | |
| LATITUDE | DOUBLE | 纬度 | |
| STATUS | INT | 状态 | 0-停用，1-启用 |
| CREATE_TIME | DATETIME | 创建时间 | |
| UPDATE_TIME | DATETIME | 更新时间 | |
| REMARK | VARCHAR(256) | 备注 | 可为空 |

### A.1.2 设备参数信息表

设备参数信息表用于存储各类设备的参数信息，包括线路参数、变压器参数等。主要字段如下：

表 A.2 设备参数信息表

| 字段名 | 数据类型 | 描述 | 备注 |
|---|---|---|---|
| DEVICE_ID | VARCHAR(32) | 设备ID | 主键 |
| DEVICE_TYPE | VARCHAR(16) | 设备类型 | 线路/变压器/开关等 |
| DEVICE_CODE | VARCHAR(32) | 设备编码 | 唯一编码 |
| DEVICE_NAME | VARCHAR(64) | 设备名称 | |
| TOPO_ID | VARCHAR(32) | 所属拓扑ID | 外键 |
| PARAM_R | DOUBLE | 电阻值 | 单位：Ω |
| PARAM_X | DOUBLE | 电抗值 | 单位：Ω |
| RATED_CURRENT | DOUBLE | 额定电流 | 单位：A |
| RATED_CAPACITY | DOUBLE | 额定容量 | 单位：MVA |
| STATUS | INT | 状态 | 0-停用，1-启用 |
| CREATE_TIME | DATETIME | 创建时间 | |

| | | | |
|---|---|---|---|
| UPDATE_TIME | DATETIME | 更新时间 | |
| REMARK | VARCHAR(256) | 备注 | 可为空 |

### A.1.3 分布式电源信息表

分布式电源信息表用于存储分布式电源的基本信息和参数。主要字段如下：

**表 A.3 分布式电源信息表**

| 字段名 | 数据类型 | 描述 | 备注 |
|---|---|---|---|
| DG_ID | VARCHAR(32) | 分布式电源 ID | 主键 |
| DG_TYPE | VARCHAR(16) | 电源类型 | 光伏/风电/储能等 |
| DG_CODE | VARCHAR(32) | 电源编码 | 唯一编码 |
| DG_NAME | VARCHAR(64) | 电源名称 | |
| NODE_ID | VARCHAR(32) | 接入节点 ID | 外键 |
| RATED_CAPACITY | DOUBLE | 额定容量 | 单位：kW |
| RATED_VOLTAGE | DOUBLE | 额定电压 | 单位：kV |
| CONTROL_MODE | VARCHAR(16) | 控制方式 | PQ/PV/V/f 等 |
| INTERFACE_TYPE | VARCHAR(16) | 接口类型 | 同步/异步/逆变器等 |
| STATUS | INT | 状态 | 0-停用，1-启用 |
| CREATE_TIME | DATETIME | 创建时间 | |
| UPDATE_TIME | DATETIME | 更新时间 | |
| REMARK | VARCHAR(256) | 备注 | 可为空 |

## A.2 短路参数数据表结构

### A.2.1 短路参数测量数据表

短路参数测量数据表用于存储各监测点的短路参数测量原始数据。主要字段如下：

**表 A.4 短路参数测量数据表**

| 字段名 | 数据类型 | 描述 | 备注 |
|---|---|---|---|
| MEASURE_ID | VARCHAR(32) | 测量 ID | 主键 |
| NODE_ID | VARCHAR(32) | 节点 ID | 外键 |
| TERMINAL_ID | VARCHAR(32) | 终端 ID | 监测终端标识 |
| MEASURE_TIME | DATETIME | 测量时间 | 精确到毫秒 |
| FAULT_TYPE | VARCHAR(16) | 故障类型 | 三相/单相/两相等 |
| MEASURE_TYPE | VARCHAR(16) | 测量类型 | 非故障扰动/人工扰动等 |
| Z_REAL | DOUBLE | 阻抗实部 | 单位：Ω |
| Z_IMAG | DOUBLE | 阻抗虚部 | 单位：Ω |
| AMPLITUDE | DOUBLE | 阻抗幅值 | 单位：Ω |
| ANGLE | DOUBLE | 阻抗角度 | 单位：度 |

| | | | |
|---|---|---|---|
| QUALITY | INT | 数据质量 | 0-100 分 |
| CREATE_TIME | DATETIME | 创建时间 | |
| REMARK | VARCHAR(256) | 备注 | 可为空 |

### A.2.2 短路容量计算结果表

短路容量计算结果表用于存储基于短路阻抗计算得到的短路容量数据。主要字段如下：

表 A.5 短路容量计算结果表

| 字段名 | 数据类型 | 描述 | 备注 |
|---|---|---|---|
| RESULT_ID | VARCHAR(32) | 结果ID | 主键 |
| NODE_ID | VARCHAR(32) | 节点ID | 外键 |
| MEASURE_ID | VARCHAR(32) | 对应测量ID | 外键，可为空 |
| CALC_TIME | DATETIME | 计算时间 | |
| FAULT_TYPE | VARCHAR(16) | 故障类型 | 三相/单相/两相等 |
| SCC_VALUE | DOUBLE | 短路容量值 | 单位：MVA |
| SCC_CURRENT | DOUBLE | 短路电流 | 单位：kA |
| Z_VALUE | DOUBLE | 阻抗值 | 单位：Ω |
| CALC_METHOD | VARCHAR(32) | 计算方法 | 直接测量/推导计算等 |
| QUALITY | INT | 数据质量 | 0-100分 |
| CREATE_TIME | DATETIME | 创建时间 | |
| UPDATE_TIME | DATETIME | 更新时间 | |
| REMARK | VARCHAR(256) | 备注 | 可为空 |

### A.2.3 短路参数历史数据表

短路参数历史数据表用于存储历史短路参数数据，支持长期趋势分析。主要字段如下：

表 A.6

| 字段名 | 数据类型 | 描述 | 备注 |
|---|---|---|---|
| HISTORY_ID | VARCHAR(32) | 历史数据ID | 主键 |
| NODE_ID | VARCHAR(32) | 节点ID | 外键 |
| STAT_TIME | DATETIME | 统计时间 | |
| TIME_TYPE | VARCHAR(16) | 时间类型 | 小时/日/月/年 |
| SCC_AVG | DOUBLE | 短路容量平均值 | 单位：MVA |
| SCC_MAX | DOUBLE | 短路容量最大值 | 单位：MVA |
| SCC_MIN | DOUBLE | 短路容量最小值 | 单位：MVA |
| SCC_STD | DOUBLE | 短路容量标准差 | 单位：MVA |
| SAMPLE_COUNT | INT | 样本数量 | |
| QUALITY_AVG | DOUBLE | 平均数据质量 | 0-100分 |
| CREATE_TIME | DATETIME | 创建时间 | |
| UPDATE_TIME | DATETIME | 更新时间 | |
| REMARK | VARCHAR(256) | 备注 | 可为空 |

## A.3 运行状态数据表结构

### A.3.1 配电网运行状态表

配电网运行状态表用于存储配电网运行状态数据，作为短路参数分析的辅助数据。主要字段如下：

**表 A.7　配电网运行状态表**

| 字段名 | 数据类型 | 描述 | 备注 |
|---|---|---|---|
| STATUS_ID | VARCHAR(32) | 状态ID | 主键 |
| TOPO_ID | VARCHAR(32) | 拓扑ID | 外键 |
| STATUS_TIME | DATETIME | 状态时间 | |
| SWITCH_STATUS | VARCHAR(16) | 开关状态 | 开/闭/异常等 |
| VOLTAGE | DOUBLE | 电压值 | 单位：kV |
| CURRENT | DOUBLE | 电流值 | 单位：A |
| ACTIVE_POWER | DOUBLE | 有功功率 | 单位：kW |
| REACTIVE_POWER | DOUBLE | 无功功率 | 单位：kVar |
| POWER_FACTOR | DOUBLE | 功率因数 | |
| QUALITY | INT | 数据质量 | 0-100分 |
| CREATE_TIME | DATETIME | 创建时间 | |
| REMARK | VARCHAR(256) | 备注 | 可为空 |

### A.3.2　分布式电源运行状态表

分布式电源运行状态表用于存储分布式电源的运行状态数据。主要字段如下：

**表 A.8　分布式电源运行状态表**

| 字段名 | 数据类型 | 描述 | 备注 |
|---|---|---|---|
| DG_STATUS_ID | VARCHAR(32) | 状态ID | 主键 |
| DG_ID | VARCHAR(32) | 分布式电源ID | 外键 |
| STATUS_TIME | DATETIME | 状态时间 | |
| OPERATION_MODE | VARCHAR(16) | 运行模式 | 并网/孤岛等 |
| OUTPUT_POWER | DOUBLE | 输出功率 | 单位：kW |
| OUTPUT_REACTIVE | DOUBLE | 输出无功 | 单位：kVar |
| OUTPUT_VOLTAGE | DOUBLE | 输出电压 | 单位：kV |
| OUTPUT_CURRENT | DOUBLE | 输出电流 | 单位：A |
| SOC | DOUBLE | 储能荷电状态 | 百分比，仅储能适用 |
| QUALITY | INT | 数据质量 | 0-100分 |
| CREATE_TIME | DATETIME | 创建时间 | |
| REMARK | VARCHAR(256) | 备注 | 可为空 |

# 附 录 B
（规范性附录）
短路参数数据库索引设计方案

## B.1 主键索引

主键索引是数据库表的基础索引，用于唯一标识表中的记录。在短路参数数据库中，所有表都建立了主键索引以确保数据的唯一性和完整性：

配电网拓扑信息表：PRIMARY KEY (TOPO_ID)
设备参数信息表：PRIMARY KEY (DEVICE_ID)
分布式电源信息表：PRIMARY KEY (DG_ID)
短路参数测量数据表：PRIMARY KEY (MEASURE_ID)
短路容量计算结果表：PRIMARY KEY (RESULT_ID)
短路参数历史数据表：PRIMARY KEY (HISTORY_ID)
配电网运行状态表：PRIMARY KEY (STATUS_ID)
分布式电源运行状态表：PRIMARY KEY (DG_STATUS_ID)

## B.2 外键索引

外键索引用于建立表之间的关联关系，提高关联查询性能：
设备参数信息表：INDEX idx_device_topo (TOPO_ID)
分布式电源信息表：INDEX idx_dg_node (NODE_ID)
短路参数测量数据表：INDEX idx_measure_node (NODE_ID)
短路容量计算结果表：INDEX idx_result_node (NODE_ID)
INDEX idx_result_measure (MEASURE_ID)
短路参数历史数据表：INDEX idx_history_node (NODE_ID)
配电网运行状态表：INDEX idx_status_topo (TOPO_ID)
分布式电源运行状态表：INDEX idx_dgstatus_dg (DG_ID)

## B.3 时间索引

时间索引用于提高基于时间的查询效率，特别是对时序数据的查询：
短路参数测量数据表：INDEX idx_measure_time (MEASURE_TIME)
短路容量计算结果表：INDEX idx_calc_time (CALC_TIME)
短路参数历史数据表：INDEX idx_stat_time (STAT_TIME)
INDEX idx_node_time (NODE_ID, STAT_TIME)
配电网运行状态表：INDEX idx_status_time (STATUS_TIME)
分布式电源运行状态表：INDEX idx_dgstatus_time (STATUS_TIME)

## B.4 组合索引

组合索引用于优化多条件查询和统计分析：
短路参数测量数据表：INDEX idx_node_time_type (NODE_ID, MEASURE_TIME, FAULT_TYPE)
INDEX idx_quality_time (QUALITY, MEASURE_TIME)
短路容量计算结果表：INDEX idx_node_type_time (NODE_ID, FAULT_TYPE, CALC_TIME)
INDEX idx_quality_method (QUALITY, CALC_METHOD)
短路参数历史数据表：INDEX idx_node_type_time (NODE_ID, TIME_TYPE, STAT_TIME)
配电网运行状态表：INDEX idx_topo_time (TOPO_ID, STATUS_TIME)
分布式电源运行状态表：INDEX idx_dg_time_mode (DG_ID, STATUS_TIME, OPERATION_MODE)

## B.5 性能优化建议

a) 对于大型历史数据表，考虑采用分区表技术，按时间范围分区，提高查询效率并简化数据管理；
b) 对于高频写入的表，如测量数据表，可以采用内存表或缓存技术，提高写入性能并减少对磁盘I/O的压力；
c) 定期进行索引维护，包括重建索引和更新统计信息，保持索引的高效性；
d) 对于不常用但需要保留的历史数据，考虑采用数据归档策略，降低活跃表的数据量，提升整体查询性能；
e) 根据实际查询模式，调整索引组合和覆盖面，针对高频查询场景进行重点优化。

# 附 录 C
（规范性附录）
短路参数数据质量评估方法

## C.1 数据质量评估指标

C.1.1 完整性指标用于评估数据的完整程度，主要包括：

a) 记录完整率：计算实际获取的数据记录数与应获取的数据记录数的比值，直接反映数据采集的成功程度；

b) 字段完整率：衡量非空字段数与总字段数的比值，检查单条记录内部的完整程度；

c) 时间序列连续性：评估时间序列数据的连续程度，尤其重要于趋势分析和预测；

d) 采样完整率：对比实际采样点数与理论采样点数，特别适用于波形数据的评估。

C.1.2 准确性指标用于评估数据的准确程度，主要包括：

a) 数值范围合理性：检查数据是否在预期的合理范围内，防止明显错误值；

b) 物理约束符合度：验证数据是否满足物理规律的约束条件，如功率关系、电压电流关系等；

c) 突变检测率：评估系统对异常突变数据的识别能力，防止异常数据影响分析结果；

d) 精度符合度：检查数据精度是否达到设计要求，满足应用所需的精确度。

C.1.3 一致性指标用于评估数据的一致程度，主要包括：

a) 数据重复率：检测重复数据的比例，避免数据冗余导致的统计偏差；

b) 关联一致性：评估关联数据之间的逻辑一致程度，如不同表中相关字段的一致性；

c) 时间趋势一致性：检验数据变化趋势的合理性和平滑性，识别异常波动；

d) 空间分布一致性：分析空间相近节点数据的相似程度，基于地理位置关系识别异常值。

C.1.4 时效性指标用于评估数据的时效程度，主要包括：

a) 数据延迟度：衡量数据从产生到可用的时间延迟，反映数据处理的实时性；

b) 更新及时率：评估数据更新的及时程度，尤其对于动态变化的参数；

c) 处理响应度：测量数据处理的响应速度，关系到系统对事件的反应时间；

d) 历史可追溯性：评价历史数据的可追溯程度，支持历史分析和回溯。

## C.2 数据质量评估方法

C.2.1 统计分析法是一种基于数据统计特性的质量评估方法，通过数理统计揭示数据特征和异常。

a) 描述性统计：计算均值、标准差、最大值、最小值等统计量，形成数据概貌；

b) 分布分析：研究数据的分布形态和特性，检测异常分布情况；

c) 相关性分析：探索数据之间的关联关系，发现变量间的内在联系；

d) 趋势分析：考察数据的变化趋势和规律，理解系统动态特性。

C.2.2 规则校验法是一种基于预定义规则的质量评估方法，通过规则判断数据是否符合预期。

a) 范围校验：检查数据是否在有效范围内，筛选出超限值；

b) 逻辑关系校验：验证数据是否满足已知的逻辑关系，如三相电压的相位关系；

c) 物理约束校验：评估数据是否符合物理定律，如能量守恒等基本原则；

d) 标准符合性校验：检查数据是否符合行业标准和规范要求。

C.2.3 模型检验法是一种基于理论模型的质量评估方法，通过模型预测与实际数据对比验证数据合理性。

a) 回归模型：检验建立变量间的回归关系，检验数据是否符合预期关系；
b) 时间序列模型：检验基于历史数据构建时序模型，检验新数据是否符合变化规律；
c) 网络拓扑模型：检验利用电网拓扑关系建立物理模型，验证数据的空间一致性；
d) 故障模型：检验根据故障特性构建专门模型，检验故障数据的合理性。

C.2.4 专家评估法是一种基于人工经验的质量评估方法，借助领域专家的知识和判断进行数据评价。

a) 经验规则评估：根据专家经验规则判断数据质量，利用隐性知识识别问题；
b) 对比分析评估：将数据与历史或参考数据对比，基于相似性评价质量；
c) 案例验证评估：通过典型案例验证数据的准确性和可用性，特别适用于特殊工况；
d) 综合判断评估：综合多种因素和视角进行整体评价，平衡各方面考量。

## C.3 数据质量评分方法

C.3.1 加权评分法是一种基于多指标综合的质量评分方法，通过为不同指标赋予权重计算总体评分。基于多个质量指标的加权计算，评分公式如下：

$$Q=\sum_{i=1}^{n} w_i \times q_i \tag{C.3-1}$$

式中：$Q$—总体质量评分；

$w_i$—第i个指标的权重，且 ；$\sum_{i=1}^{n} w_i = 1$

$q_i$—为第i个指标的评分。

C.3.2 层次分析法是一种基于层次结构的系统化评分方法，通过系统分解和综合判断进行质量评价。

a) 建立指标层次结构，将质量指标组织成多层次的结构体系；
b) 构造判断矩阵，通过两两比较确定各指标的相对重要性；
c) 计算权重向量，通过特征值方法求解判断矩阵的权重；
d) 一致性检验，确保判断矩阵的逻辑一致性；
e) 计算最终评分，综合各层各指标的权重和评分。

C.3.3 模糊综合评价法是一种基于模糊数学的质量评分方法，适合处理评价边界不明确的模糊情况。

a) 确定评价因素集和评价等级集，建立评价框架；
b) 建立隶属度函数，描述数据对各质量等级的归属程度；
c) 确定权重向量，反映各评价因素的相对重要性；
d) 计算模糊综合评价结果，得到对各等级的隶属度分布；
e) 将模糊评价结果转换为具体的评分值，便于使用和比较。

## C.4 数据质量分级标准

数据质量分级标准为质量评估结果提供了清晰的分类框架，如表C.1所示，将数据质量分为五个等级。

**表 C.1 数据质量分级表**

| 等级 | 分数范围 | 质量描述 | 可用性建议 |
|---|---|---|---|

| | | | |
|---|---|---|---|
| A级 | 90-100 | 优秀，完全可靠 | 可直接用于所有应用场景 |
| B级 | 75-89 | 良好，较为可靠 | 可用于大多数应用场景,关键应用需验证 |
| C级 | 60-74 | 一般，基本可靠 | 可用于一般应用，不建议用于关键决策 |
| D级 | 40-59 | 较差，可靠性不足 | 仅供参考，使用前须经校正或验证 |
| E级 | 0-39 | 低劣，不可靠 | 不建议使用，需重新采集或测量 |

# 附 录 D
（规范性附录）
短路参数数据库性能优化方案

## D.1 数据库服务器配置建议

### D.1.1 硬件配置建议

数据库服务器的硬件配置是系统性能的基础保障，应根据系统规模和数据量合理选择配置。对于不同规模的短路参数监测系统，建议的硬件配置如下：

表 D.1 不同规模系统的硬件配置建议

| 系统规模 | 处理器 | 内存 | 存储 | 网络 |
|---|---|---|---|---|
| 小型系统（<100 监测点） | 8 核以上，3.0GHz 以上 | 32GB 及以上 | SSD 500GB（系统和活跃数据），HDD 2TB（历史数据） | 千兆网卡 |
| 中型系统（100-500 监测点） | 16 核以上，3.0GHz 以上 | 64GB 及以上 | SSD 1TB（系统和活跃数据），HDD 8TB RAID5（数据归档） | 万兆网卡 |
| 大型系统（>500 监测点） | 32 核及以上，3.0GHz 以上 | 128GB 及以上 | SSD 2TB RAID10（系统和活跃数据），HDD 20TB RAID5（数据归档） | 万兆网卡冗余设计 |

### D.1.2 软件配置建议

合理的软件配置对于发挥硬件性能至关重要，短路参数数据库系统的软件配置建议如下：

表 D.2 数据库系统软件配置建议

| 配置类别 | 推荐配置 | 说明 |
|---|---|---|
| 操作系统 | Linux(CentOS/Ubuntu Server/RHEL) | 稳定性好、资源占用少、安全性高；关闭不必要服务，优化内核参数 |
| 数据库系统 | 关系型：MySQL 8.0+/PostgreSQL 13.0+/Oracle 19c+时序数据库：InfluxDB/TimescaleDB | 可结合使用关系型与时序数据库，发挥各自优势 |
| 中间件 | Redis 6.0+（缓存）；Kafka 2.8+/RabbitMQ 3.9+（消息队列）；Prometheus+Grafana（监控） | 提高系统吞吐量和可靠性，实时监控系统资源使用情况 |

## D.2 数据库参数优化建议

### D.2.1 MySQL参数优化

MySQL作为常用的关系型数据库，其参数配置对性能影响显著，应根据服务器资源和业务特点进行优化：

a) 缓冲池配置是MySQL性能优化的关键，建议将innodb_buffer_pool_size设置为服务器总内存的70%左右，为数据缓存提供充足空间；同时将innodb_buffer_pool_instances设置为8或等于CPU核心数，提高并发访问效率。对于大内存服务器，合理的缓冲池配置可显著减少磁盘I/O，提升查询性能。
b) 日志配置对数据写入和恢复性能有重要影响，建议将innodb_log_file_size设置为1GB左右，提供足够的日志空间；设置innodb_log_buffer_size为64MB，减少日志写入磁盘的频率。这些配置有助于提高事务处理性能，尤其是在大量数据写入场景下。
c) 并发配置方面，建议将innodb_thread_concurrency设置为0，让MySQL自动管理并发线程；max_connections设置为1000左右，支持更多并发连接。这些参数应根据系统实际访问模式进行调整，避免资源过度竞争或连接拒绝问题。
d) 查询缓存方面，对于MySQL 8.0之前的版本，由于查询缓存在高并发场景下可能成为瓶颈，建议设置query_cache_size和query_cache_type均为0，关闭查询缓存功能。MySQL 8.0已默认移除查询缓存功能。
e) 事务配置方面，在数据安全性与性能之间寻求平衡，可将innodb_flush_log_at_trx_commit设置为2，在性能和数据安全间取得平衡；sync_binlog设置为1000，减少binlog同步频率，提高写入性能。这些设置适合对性能要求较高而数据安全要求适中的场景。

### D.2.2 PostgreSQL参数优化

PostgreSQL具有良好的扩展性和高级特性，适合复杂数据处理，其参数优化建议如下：

a) 内存配置方面，建议将shared_buffers设置为服务器总内存的25%左右，为数据缓存提供空间；work_mem设置为64MB，提高复杂查询的排序和哈希操作性能；maintenance_work_mem设置为256MB，加速维护操作如索引创建。这些参数的合理配置可显著提升查询性能和系统响应时间。
b) 写入配置方面，建议将wal_buffers设置为16MB，提高WAL写入效率；checkpoint_timeout设置为15分钟，减少检查点频率；max_wal_size设置为2GB，允许更大的WAL文件空间，提高写入性能。这些设置有助于优化系统在大量写入场景下的表现，减少I/O压力。
c) 并发配置方面，建议将max_connections设置为200左右，支持合理的并发连接数；effective_cache_size设置为服务器总内存的70%左右，帮助查询优化器做出更好的索引使用决策。这些参数应根据系统访问模式和负载特点进行调整，避免资源浪费或竞争。
d) 查询优化方面，对于使用SSD存储的系统，建议将random_page_cost设置为1.1左右，反映随机I/O的实际成本；effective_io_concurrency设置为200左右，增加并行I/O操作。这些设置可帮助优化器生成更符合硬件特性的执行计划，提高查询效率。

## D.3 数据分区与归档策略

### D.3.1 数据分区策略

合理的数据分区策略可显著提高大数据量下的查询性能和管理效率，短路参数数据库的分区策略建议如下：

a) 时间分区是最常用且有效的分区方式，适合大多数时序数据。建议将实时数据按天分区，便于快速访问最新数据；短期历史数据（如1-3个月内）按周或月分区，兼顾查询效率和管理复杂度；长期历史数据按季度或年分区，简化存储管理并支持数据归档。时间分区使查询可以只扫描相关时间范围的分区，大幅提高查询效率。

b) 空间分区根据地理或拓扑结构进行数据划分，适合分布式系统。可按配电网区域分区，将同一区域的数据存储在一起；或按电压等级分区，将相同电压等级的数据集中管理；或按监测终端分区，便于终端级的数据分析。空间分区适合于地理位置查询和区域级分析，提高局部数据的访问效率。

c) 混合分区结合时间和空间维度的优势，适合复杂查询需求。可先按区域分区，再按时间分区，便于区域内的时序分析；或先按数据类型分区，再按时间分区，适合特定类型数据的纵向分析。混合分区能够更灵活地满足不同业务场景的查询需求，但管理复杂度也相应提高，需谨慎设计分区策略。

### D.3.2 数据归档策略

随着数据量的增长，有效的数据归档策略对于系统性能和存储成本管理至关重要：

a) 数据保留周期应根据数据价值和使用频率设定。建议将实时数据保留 30 天在高性能存储中，满足高频访问需求；短期历史数据保留 1 年内在在线存储中，支持常规分析和报表需求；长期历史数据（3 年及以上）转移到归档存储中，平衡成本和可访问性。不同类型的数据可能需要不同的保留策略，如关键节点的数据可能需要更长的保留周期。

b) 归档压缩策略应平衡存储效率和数据价值。对于关键数据，如主要节点的短路容量数据，应按原始精度归档，保证数据完整性；对于非关键数据，可采用降采样归档，如将秒级数据聚合为分钟级数据；对于长期历史数据，可进一步汇总为小时、日、月级统计数据，大幅减少存储需求同时保留趋势信息。

c) 归档实施建议包括使用自动化脚本定期执行归档任务，减少人工干预；归档前进行数据质量评估，确保归档数据的完整性和准确性；归档后进行数据完整性验证，确保归档过程没有导致数据损坏或丢失；保留详细的归档日志和元数据，记录归档内容、时间、位置等信息，便于日后检索和恢复。

## D.4 查询优化策略

### D.4.1 索引优化

索引是提高查询性能的基础手段，短路参数数据库的索引优化策略如下：

a) 合理设计索引是查询优化的基础，应针对频繁查询条件建立索引，如时间戳、节点 ID、测量类型等常用查询字段；注意避免过多索引导致写入性能下降，通常只为最常用的查询条件创建索引；定期维护和重建索引，特别是在大量数据变更后，确保索引效率。索引设计应结合实际查询模式，重点优化高频查询和性能敏感查询。

b) 索引使用策略对查询性能有直接影响。尽量使用覆盖索引，即索引包含查询所需的所有字段，避免回表操作；注意复合索引的列顺序，确保与查询条件匹配，遵循"最左前缀"原则；避免索引失效的常见陷阱，如在 WHERE 子句中对索引字段使用函数、类型转换或 NULL 值比较等。正确的索引使用策略可显著提高查询效率，减少不必要的全表扫描。

### D.4.2 SQL优化

优化SQL查询是提高数据库性能的重要手段，应遵循以下原则：

a) 查询语句优化是提高性能的关键。避免使用 SELECT *，只查询需要的列，减少数据传输量；合理使用 JOIN，避免过多表关联导致性能下降，必要时考虑分步查询；使用适当的过滤条件

减少数据处理量，尤其是时间范围和关键 ID 的过滤；避免在 WHERE 子句中使用函数，导致索引失效。

b) 分页查询是数据浏览的常用场景，需要特别优化。使用主键或索引列进行分页，确保高效定位；避免使用 OFFSET，特别是大偏移量时性能急剧下降，可使用 " HERE id > last_id LIMIT N " 代替；对大数据量分页查询使用延迟关联技术，先通过索引获取目标行的 ID，再关联获取完整数据。

### D.4.3 缓存策略

缓存是提高频繁访问数据响应速度的有效手段，短路参数数据库的缓存策略如下：

a) 数据库缓存是第一级缓存防线，应合理配置数据库缓冲池大小，确保热点数据留在内存中；利用数据库内部的查询计划缓存，避免重复的解析和优化开销。数据库缓存由数据库系统自动管理，但合理的参数配置可以显著提高缓存效率。

b) 应用缓存提供了更灵活的缓存机制，建议使用 Redis 缓存热点数据，如常用配置、计算结果和实时监测值；实现多级缓存机制，如本地内存缓存、分布式缓存和数据库缓存的组合使用；定期更新缓存数据，确保数据新鲜度，可采用设置合理的过期时间或主动刷新机制。应用缓存可以大幅减轻数据库负担，提高系统响应速度。

c) 缓存失效策略决定了缓存的有效性和资源利用率。可采用基于时间的失效策略，如设置不同类型数据的生存时间；或基于更新的失效策略，在数据变更时主动使相关缓存失效；或使用 LRU (Least Recently Used) 策略，自动淘汰最少使用的缓存项目，优化内存使用。合理的失效策略可以平衡数据新鲜度和缓存效率。

## D.5 读写分离与高可用方案

### D.5.1 读写分离方案

读写分离是提高数据库并发处理能力的有效架构，实施建议如下：

a) 架构设计以一主多从结构为基础，主库负责处理所有写操作，从库负责处理读操作，分散访问压力。建议配置 1 个主库和 2-3 个从库，满足基本读写分离需求；规模较大的系统可增加更多从库或引入分片技术。使用数据库中间件（如 ProxySQL、MyCat 等）实现自动路由，根据 SQL 类型自动将查询导向适当的库，简化应用开发。

b) 数据同步是读写分离的关键环节，需合理配置主从复制机制。针对 MySQL，可使用异步复制提高性能，或使用半同步复制提高数据安全性，根据业务对一致性的要求选择合适的复制方式；注意监控复制延迟，当延迟超过阈值时应触发告警或暂时禁用延迟较大的从库。复制延迟监控和管理对于保障数据一致性至关重要。

c) 业务适配是读写分离实施的实践考量。对于实时性要求高的查询，如监测点实时数据查询，应直接使用主库确保数据最新；对于报表分析等查询，可使用从库减轻主库压力；对于复制延迟导致的数据不一致问题，应在应用层面有应对策略，如重要操作后的查询强制走主库。业务层面的合理适配可以最大化读写分离的效益同时避免一致性问题。

### D.5.2 高可用方案

高可用方案是保障系统持续服务能力的关键，建议采取以下措施：

a) 主备切换是高可用的核心机制，应实现自动故障检测，通过心跳检测或监控工具及时发现主库故障；配置自动主备切换机制，在主库故障时自动将从库提升为新主库，最小化服务中断时间；

使用专业高可用工具如 MHA、MGR(MySQL Group Replication)或 Patroni 等实现可靠的故障转移。主备切换机制应定期测试，确保在真正需要时能够正常工作。

b) 数据备份是灾难恢复的基础，应建立多层次备份策略。定期执行全量备份，如每周一次，保存数据库完整状态；结合增量备份（如每日一次）和日志备份（如每小时或实时）减少数据丢失风险；实施跨区域备份，将备份数据存储在地理上隔离的位置，防范区域性灾难。备份策略应全面考虑恢复时间、存储成本和操作复杂度，找到最佳平衡点。

c) 灾难恢复计划是应对严重故障的最后防线。制定详细的灾难恢复计划，明确不同级别故障的应对流程和责任人；定期组织灾难恢复演练，验证恢复流程的有效性和团队的应对能力；建立恢复时间目标(RTO)和恢复点目标(RPO)，量化业务容忍的最大中断时间和数据丢失量，并设计相应的技术方案满足这些目标。完善的灾难恢复机制可以确保即使在最坏情况下，系统也能在可接受的时间内恢复服务。

# 附 录 E
（规范性附录）
短路参数数据库管理与维护指南

## E.1 数据库日常管理

### E.1.1 日常监控

数据库管理员应建立全面的日常监控体系，重点关注CPU使用率、内存使用率、磁盘I/O、网络I/O、数据库连接数以及查询执行情况等关键性能指标。监控工具可采用数据库自带的监控功能，也可集成Prometheus与Grafana等开源监控平台，或使用Zabbix、Nagios等专业监控软件。对于重要的生产环境，还可开发自定义脚本进行更精细化的监控。

告警机制是监控体系的重要组成部分，应为各项指标设置合理的告警阈值，并实施分级告警策略。例如，当CPU使用率超过85%时触发一级告警，超过95%时触发紧急告警。告警通知可通过短信、邮件、即时通讯工具等多种渠道发送给相关人员，确保问题能够得到及时处理。所有告警事件均应记录在案，便于后续分析和系统优化。

### E.1.2 日常备份

完善的备份策略是数据库安全的基础保障。根据数据重要性和业务需求，通常采用全量备份与增量备份相结合的方式，例如每周进行一次全量备份，每日进行一次增量备份，同时对数据库日志进行实时或每小时备份。备份工具可使用数据库自带的备份功能，也可选择专业的第三方备份软件，或开发自动化脚本实现定制化备份流程。

备份验证是确保备份数据可用性的关键环节。管理员应定期抽检备份数据，验证其完整性和可恢复性，可在测试环境中模拟数据恢复过程，确认恢复后的数据与原始数据一致。每次备份验证的结果都应详细记录，包括验证时间、验证方法、验证结果以及发现的问题及解决方案。

### E.1.3 日常优化

数据库管理员应定期对系统进行资源优化，包括检查CPU、内存、磁盘空间等资源的使用情况，调整资源分配，并评估是否需要进行硬件扩容。当监控到某项资源使用率持续超过80%时，应考虑增加相应资源或优化资源使用方式。

数据优化是提高数据库性能的重要手段，包括清理临时表和冗余数据、优化表结构、定期维护和重建索引等工作。例如，可设置自动清理脚本，定期删除超过保留期限的临时数据；对于频繁更新的表，定期进行碎片整理；对于变化较大的表，定期更新统计信息以优化查询计划。

查询优化是性能调优的核心，管理员应定期分析慢查询日志，找出执行时间较长的SQL语句，分析执行计划，进行针对性优化。对于频繁执行的查询，可考虑添加合适的索引、调整查询语句或重构数据模型来提高查询效率。同时，应定期更新数据库的统计信息，确保优化器能够生成最优的执行计划。

## E.2 数据库定期维护

### E.2.1 维护计划

数据库维护应建立分层次的维护计划，涵盖日、周、月、季度等不同时间周期。每天的维护工作主要包括监控系统运行状况、检查告警信息以及备份日志文件，确保系统正常运行。这些工作可以通过自动化脚本完成，但管理员应每日检查自动化任务的执行结果，及时处理异常情况。

每周维护内容更为全面，包括分析一周内的性能趋势、检查备份执行情况以及清理临时文件等。管理员应关注性能指标的变化趋势，发现潜在问题并及时处理。对于备份任务，应检查其完成情况和数据完整性，确保备份策略有效实施。

每月维护是系统健康状况的全面检查，包括系统完整性检查、安全漏洞扫描和修复以及更新统计信息等。完整性检查可以发现潜在的数据损坏或不一致问题；安全检查则有助于发现和修复系统漏洞；统计信息更新则为查询优化提供基础。

季度维护侧重于系统长期健康和发展，包括全面性能评估、容量规划评估以及系统参数优化等。通过分析系统长期运行数据，预测未来负载增长趋势，提前规划系统扩容或升级，并根据实际运行情况优化系统参数配置。

### E.2.2　数据归档与清理

数据归档是长期数据管理的重要环节，应建立规范的归档流程。首先明确归档的范围和条件，例如哪些数据需要归档、归档的时间节点等；然后执行数据归档操作，将数据从活动表移至归档表或归档存储；归档完成后，应验证归档数据的完整性和一致性；最后，在确认归档数据可用的情况下，清理原表中的历史数据，释放存储空间。

数据清理策略应根据实际业务需求制定，可基于时间（如保留最近一年的数据）、基于容量（如当表大小超过阈值时清理）或基于业务需求（如根据数据使用频率决定保留策略）。无论采用何种策略，都应确保清理前数据已妥善归档，且清理不会影响业务运行。

操作建议方面，归档和清理前应进行完整备份，以防操作失误导致数据丢失；选择系统负载较低的时间段执行大规模数据操作，减少对业务的影响；对于大量数据的处理，应采用分批执行的方式，避免长时间锁表或系统资源耗尽；所有操作应详细记录，包括操作内容、操作时间、操作结果等，便于跟踪和审计。

### E.2.3　版本升级与补丁管理

数据库版本升级前应进行全面的升级评估，包括分析新版本的功能特性和已修复的问题，评估升级可能带来的风险和收益，以及制定详细的升级计划。评估应覆盖兼容性、性能影响、安全增强等多个方面，确保升级能够真正带来价值。

升级测试是确保升级成功的关键环节。应在测试环境中先行验证升级过程，测试关键业务功能在新版本下的运行情况，以及进行性能和兼容性测试。测试环境应尽可能模拟生产环境，确保测试结果具有参考价值。测试中发现的问题应记录并解决，必要时调整升级计划。

升级实施应按照既定计划进行。首先，在升级前进行完整备份，确保万一升级失败可以回退；然后，按计划执行升级，严格遵循操作步骤；升级完成后，应进行全面验证，确认系统功能和性能正常；同时，应制定详细的回滚计划，明确在何种情况下需要回滚以及具体的回滚步骤。整个升级过程应有专人负责，确保各环节顺利进行。

## E.3　安全管理

### E.3.1　访问控制

数据库访问控制应遵循最小权限原则，即用户只能获得完成其工作所需的最小权限集。管理员应定期审核用户权限，确保权限分配合理且及时更新。当用户离职或岗位变动时，应及时调整或禁用其账号，防止未授权访问。

权限设置应实行细粒度控制，根据用户角色和职责分配不同级别的权限。可采用角色化权限管理模式，将相似职责的用户归入同一角色，统一管理权限。对于数据修改、结构变更等敏感操作，应建立审批机制，确保操作经过适当授权。

密码策略是访问控制的基础，应实施强密码策略，要求密码包含大小写字母、数字和特殊字符，并定期更换。系统应强制密码复杂度检查，防止简单密码的使用。同时，应明确禁止密码共享行为，确保每个用户账号只由授权人员使用，提高账号行为的可追溯性。

#### E.3.2 安全审计

安全审计应覆盖数据库操作的各个方面，包括用户登录和退出活动、权限变更操作、数据修改操作、系统配置变更以及数据库结构变更等。这些审计信息对于安全事件的追查和系统问题的排查都具有重要价值。

审计实施应开启数据库自带的审计功能，捕获关键操作日志。审计日志应集中存储在专用的安全日志服务器上，便于统一管理和分析。管理员应定期分析审计日志，识别异常行为和潜在安全风险。对于可疑的异常行为，应配置自动告警机制，确保及时发现和处理。

定期生成安全审计报告是审计工作的重要输出。报告应包括用户活动统计、异常行为分析、权限变更记录等内容，帮助管理层了解系统安全状况。通过分析长期审计数据，可以发现安全趋势和模式，为制定更有针对性的安全策略提供依据。

#### E.3.3 安全加固

系统安全加固是提高数据库安全性的基础工作，包括关闭不必要的数据库服务和功能、限制数据库服务器的网络访问、定期安装安全补丁等。应采用"默认拒绝"策略，只开放必要的服务和端口，减少攻击面。

数据加固主要保护数据本身的安全性，包括对敏感数据进行加密存储、使用SSL/TLS加密数据传输以及对敏感数据进行脱敏处理等。数据加密应采用业界认可的加密算法和密钥管理方案，确保加密的有效性和安全性。

防入侵措施是抵御外部攻击的重要手段，包括部署防火墙限制网络访问、使用入侵检测系统监控异常行为以及定期进行安全漏洞扫描等。应建立多层次的安全防护体系，形成纵深防御，提高系统整体安全性。这些措施应与数据库本身的安全机制结合，形成完整的安全防护体系。

### E.4 应急处理机制

#### E.4.1 常见故障及处理

性能故障是数据库常见的问题类型，表现为系统响应缓慢、CPU使用率高或内存不足等。当发生此类故障时，应首先终止导致资源耗尽的异常查询，然后分析慢查询日志识别性能瓶颈，优化问题SQL语句或调整数据库参数配置。对于反复出现的性能问题，应考虑从索引优化、查询重写或硬件升级等方面进行根本性解决。

连接故障通常表现为应用无法连接数据库或连接数耗尽。处理此类故障时，应首先检查网络连接状态，确认数据库服务是否正常运行；然后检查连接池配置，调整最大连接数和连接超时参数；必要时重

启数据库服务恢复连接能力。长期解决方案包括优化应用程序的连接管理策略，避免连接泄漏和连接数激增。

数据故障是最严重的故障类型，包括数据错误、数据丢失或表损坏等。处理此类故障时，应首先确定故障范围和原因，然后使用数据库恢复工具修复损坏的数据结构；如无法修复，则需从最近的备份中恢复数据，并重新应用事务日志追上最新状态。为减少数据丢失风险，应建立完善的数据备份和验证机制，确保能在最短时间内恢复业务数据。

### E.4.2 应急预案

服务中断应急预案是保障系统可用性的重要工具。当发生服务中断时，应快速定位原因，区分是网络故障、硬件故障还是软件故障；根据故障类型采取相应措施，如尝试重启服务、切换到备用系统或调整系统配置；同时，通知相关人员包括技术团队、业务部门和管理层，确保各方了解情况并协调应对。预案应详细描述各种中断场景的处理流程，明确责任人和联系方式，确保在紧急情况下能够快速响应。

数据丢失应急预案是数据库最关键的应急措施之一。当发现数据丢失时，应立即确定丢失范围，包括影响的表、记录数量和时间范围；停止相关业务操作，防止问题扩大；根据备份策略，从最近的备份中恢复数据，并应用事务日志至故障前状态；恢复完成后，验证数据完整性和一致性，确保业务可以正常运行。预案中应详细说明不同级别数据丢失的恢复路径和预期恢复时间。

安全事件应急预案用于应对数据库安全威胁。当发现安全事件时，应立即隔离受影响的系统，切断外部连接，防止攻击扩散；保留所有相关日志和证据，支持后续调查；分析入侵途径和影响范围，评估数据泄露风险；修复安全漏洞，加强安全防护措施；必要时，通知相关监管机构和受影响的用户。预案应包括安全事件分级标准、应对流程和责任分工，确保安全事件得到及时有效处理。

### E.4.3 应急演练

应急演练是检验应急预案有效性的重要手段。应制定定期演练计划，覆盖不同类型的应急场景，如服务中断、数据丢失、安全事件等。演练计划应明确演练目标、范围、参与人员、时间安排以及评估标准，确保演练的针对性和有效性。

演练执行应尽可能模拟真实的应急环境，测试预案的可执行性和有效性。参与人员应严格按照应急预案执行操作，记录操作过程、遇到的问题和解决方法。演练可分为桌面演练和实战演练两种形式，前者重点检验预案的完整性和逻辑性，后者重点测试实际操作的可行性和效果。

演练评估是提升应急能力的关键环节。演练结束后，应对演练过程和结果进行全面评估，分析预案执行过程中的问题和不足，如响应速度不够快、责任分工不够明确、恢复步骤不够详细等。根据评估结果，优化和完善应急预案，提高预案的实用性和有效性。评估结果和改进措施应形成书面报告，并在下次演练中验证改进效果。