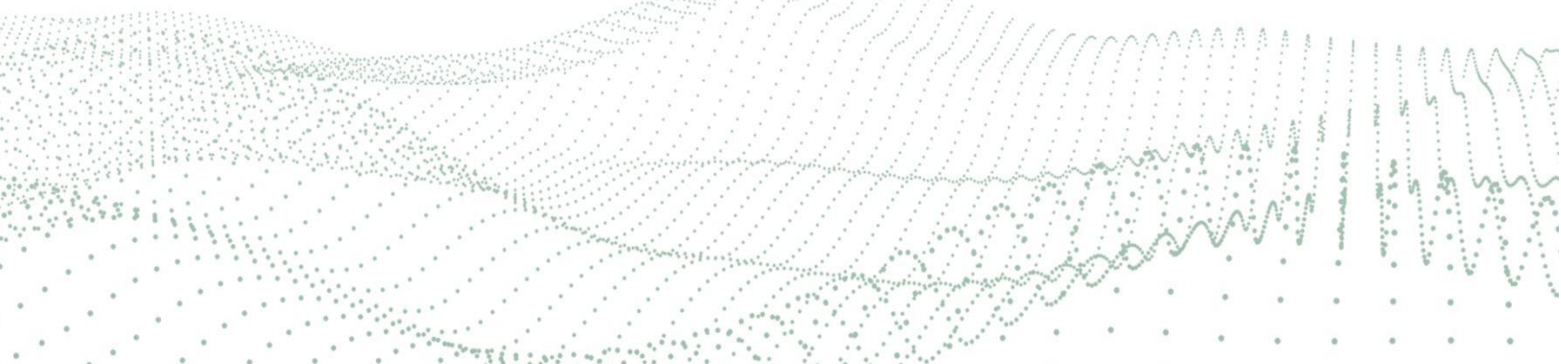


分布式数据库运维管理 实践指南 (1.0)

CCSA TC601 大数据技术标准推进委员会

2024 年 12 月



版 权 声 明

本报告版权属于 CCSA TC601 大数据技术标准推进委员会，并受法律保护。转载、摘编或利用其它方式使用本报告文字或者观点的，应注明“来源：CCSA TC601 大数据技术标准推进委员会”。违反上述声明者，本推委会将追究其相关法律责任。

编写委员会

本报告的撰写得到了数据库领域多家企业与专家的支持和帮助，主要参与单位与人员如下。

❖ 主要编写单位（排名不分先后）：

大数据技术标准推进委员会、云和恩墨（北京）信息技术有限公司、天翼数字生活科技有限公司、北京新数科技有限公司、浪潮软件科技有限公司、天翼云科技有限公司、中电科金仓(北京)科技股份有限公司、中移（杭州）信息技术有限公司、中国民航信息网络股份有限公司、中国联合网络通信有限公司软件研究院、华青融天（北京）软件股份有限公司、贵州省农村信用社联合社。

❖ 编写组主要成员（排名不分先后）：

刘蔚、刘思源、齐丹阳、马嘉慧、马鹏玮、姜春宇、李轶楠、成思敏、刘正超、黄浩、张峰、杜逢山、孙悦、刘俊、胡一鸣、王建荣、陈思源、严安林、仲文迪、杨昕、沈勇、梁策、张承基、邓宇城、盛付苗、包彤、李友福、庞潇。

前 言

随着开源、分布式数据库的大量上线，数据库运维管理在诸多方面面临新挑战。应用侧厂商在遴选服务商时缺乏可参考标准，企业内部也需明确运维管理团队能力情况。近年来，数据库引发的危机事件频发，多因制度流程未规范执行、人员操作不当、运维管理工具未充分发挥价值等因素导致。

不论是对数据库的架构改造，还是数据库上云搬迁，亦或是数据库产品变更，都涉及数据库及应用系统的迁移。目前，针对此领域的指导性工程方法、权威交付流程和专业评价体系的发展仍在起步阶段。

本指南首先介绍了分布式数据库运维管理过程中的需求和痛点，然后梳理了运维管理过程中常见的工具，并对企业运维能力进行多维度建议。最后，指南总结了运维管理的策略和典型问题，并分享行业经典案例。

本指南的编写得到了行业内数据库厂商，数据库专业服务公司以及数据库开源社区的大力支持，在此表示由衷感谢！由于时间仓促，水平所限，错误和不足之处在所难免，欢迎各位读者批评指正，意见建议请发送至 liuwei11@caict.ac.cn。

目 录

一、分布式数据库运维管理概述.....	1
（一）本指南内容概述.....	2
（二）本指南的价值.....	2
（三）本指南的适用范围.....	2
（四）本指南的指导对象.....	3
二、分布式数据库日常运维.....	4
（一）性能调优.....	4
（二）巡检与监报告警.....	6
（三）备份和恢复策略.....	9
（四）安全与防护.....	12
（五）升级.....	14
（六）日志管理.....	15
（七）智能运维.....	19
三、分布式数据库运维应急保障.....	22
（一）故障处理.....	22
（二）高可用与容灾管理.....	23
四、分布式数据库运维体系建设.....	25
（一）组织建设.....	25
（二）能力培养.....	26
（三）运维流程优化.....	27
五、分布式数据库运维管理标准能力模型.....	29
（一）数据库集中运维管理平台.....	29
（二）SQL 质量管理平台.....	30
（三）数据库运维管理成熟度模型.....	31
六、分布式数据库运维管理的未来趋势.....	32
（一）云计算与数据库运维.....	32

(二) 人工智能在运维管理中的应用.....	33
(三) DevOps 与数据库运维.....	33
(四) 异构数据库运维.....	34
七、总结.....	35
(一) 典型问题发现总结.....	35
(二) 对企业分布式数据库运维管理的建议.....	36
附录：分布式数据库运维管理案例分析.....	38
(一) 天翼云在线扩容实践案例.....	38
(二) 中移杭研分布式 MySQL 集群升级实践.....	40
(三) 某股份制银行分布式核心系统改造与运维监控实践.....	42
(四) 某证券公司集中运维管理实践.....	44
(五) 某科技馆一体化运维监控系统实践.....	46
(六) 某云项目数据库集中运维管控一体化平台案例实践.....	48
(七) 浪潮云运维管理实践案例.....	50

图 目 录

图 1	分布式数据库运维管理能力框架	1
图 2	数据库集中运维管理平台标准框架	29
图 3	SQL 质量管理平台标准框架	30
图 4	数据库运维管理能力成熟度模型标准框架	31
图 5	数据库扩容流程图	39
图 6	滚动升级副本流程示意图	41
图 7	更新复制拓扑示意图	41
图 8	主库切换流程示意图	42
图 9	银行核心系统架构图	43
图 10	访问关系拓扑图	44
图 11	某证券 DBaaS 平台架构	46
图 12	集中运维管控一体化平台架构	49

表 目 录

表 1	性能调优工具分类	5
表 2	监控工具分类	8
表 3	备份恢复工具分类	10
表 4	常见数据库安全审计工具	13
表 5	常见数据库日志分析工具	19
表 6	常见数据库智能化运维工具	22
表 7	数据库管理平台测试通过情况	29
表 8	SQL 质量管理平台测试通过情况	30
表 9	数据库运维管理能力成熟度模型测试通过情况	31

一、分布式数据库运维管理概述

随着开源、分布式数据库的大量上线，数据库运维管理在数据库设计、运维工具使用及人员组织架构完善等方面遇到新挑战。近年来，数据库引发的服务中断、数据泄露等危机事件频发，往往是由于数据库使用方内部的制度流程未规范执行、人员未进行充分培训导致的操作不当、运维管理工具未能充分发挥价值等因素导致。



图 1 分布式数据库运维管理能力框架

分布式数据库运维管理可分为日常运维、运维应急保障以及运维体系建设三个维度如上所示。分布式数据库在运维管理上提供了更高的灵活性和可扩展性，但同时也带来了更复杂的架构和更多的管理挑战，涉及数据一致性和同步、故障恢复和高可用性、性能监控和优化、安全性和合规性、扩展性等多个方面。

（一）本指南内容概述

本指南旨在为数据库管理人员提供全面、系统化指导，详细介绍了分布式数据库的运维管理理念、流程、技术及最佳实践。内容涵盖了从数据库的规划、部署、监控、优化到故障排查和安全保障的各个方面。此外，本指南还探讨了自动化运维工具的应用，以及在云环境下分布式数据库的特殊运维挑战。

（二）本指南的价值

本指南的价值在于为分布式数据库的运维管理提供了一系列经过验证的最佳实践和方法论。它不仅帮助读者理解分布式数据库的复杂性，还提供了实际操作的指导，以确保数据库系统的稳定性、可用性、安全性和性能。通过本指南，读者可以学习如何有效地监控和管理分布式数据库，如何应对分布式数据库特有的挑战，以及如何通过自动化和智能化工具简化日常运维任务、提升运维效率。此外，本指南还有助于企业构建和优化分布式数据库的运维管理体系，从而支持业务的快速发展和创新。

（三）本指南的适用范围

本指南适用于所有使用或计划部署分布式数据库的组织和企业。无论是金融、电信、互联网、制造业还是政府部门，都可以从本指南中获得宝贵的知识和指导。此外，本指南也适合数据库管理员、系统架构师、

IT 经理以及对分布式数据库运维管理感兴趣的技术人员阅读和参考。它旨在帮助这些专业人员提升他们在分布式数据库运维管理方面的专业技能，以及在实际工作中解决具体问题的能力。

（四）本指南的指导对象

本指南对分布式数据库应用系统的运维管理痛点做分析，并对运维管理的过程与步骤做分析、总结，为企业、单位的数据库应用系统运维工作提供理论指导依据。本指南的主要指导对象是数据库运维管理人员，包括但不限于：

- 数据库管理员：负责日常数据库的维护、备份和恢复工作。
- 系统架构师：设计和优化数据库系统架构，确保系统满足业务需求。
- IT 经理：负责 IT 团队的管理和项目决策，需要了解数据库运维的复杂性和挑战。
- 技术支持工程师：提供数据库相关的技术支持和咨询服务。
- 技术爱好者和学生：对数据库运维管理感兴趣，希望了解行业实践和提升个人技能。

本指南以实用性和可操作性为原则，结合理论与实践，旨在帮助上述人员提升分布式数据库的运维管理能力。

二、分布式数据库日常运维

(一) 性能调优

1. 分布式数据库性能调优的需求

性能调优追求查询优化、负载均衡、数据分布、资源调度和事务管理，以提升系统效率和响应速度，同时保证数据安全与性能之间的平衡。

2. 分布式数据库性能调优的痛点

- **数据一致性与事务管理**: 保证数据一致性同时需提高并发处理能力。
- **系统架构与配置**: 多节点配置调整增加了调优的难度和工作量。
- **查询优化与索引维护**: 多节点数据分布使查询优化索引维护更复杂。
- **负载均衡与资源分配**: 存在负载不均衡和资源分配不合理问题。
- **安全与性能平衡**: 需在安全和性能间找到平衡。

3. 分布式数据库 SQL 优化注意事项与实践建议

- **NDV (Number of Distinct Values) 与数据分布**: 高 NDV 意味着数据分布均匀，索引效果佳。

- **驱动表 (Drive Table) 的选择确定**: 当 SQL 通过谓词过滤返回结果数较少时，建议在驱动表谓词条件创建索引并且建议在被驱动表链接列中创建索引，提升业务 SQL 性能。

- **执行计划 (Execution Plan)**: 根据真实的执行计划每个算子的执行消耗分析 SQL 性能主要矛盾，判别是否 SQL 缺失索引或需要等价改写使得 SQL 可以通过理想的执行计划。

● **表连接 (Table Join)**：数据库优化器根据表的大小和查询条件等信息，选择一个最低成本的表连接方式来进行表连接操作。

● **分区表与分区索引**：建议根据数据量和业务需求合理规划分区，选择适当的分区键和策略。

● **分布式查询或分布式事务**：分布式事务中，跨节点交互会带来额外开销。通过表组特性，可以聚合数据副本，减少跨节点交互，提升性能。

4. 数据库性能调优工具

性能调优工具能够自动收集性能数据并输出格式化的性能数据汇总报告，能够自动对性能问题进行分析并执行优化，具备 SQL 审核平台，形成从项目设计、开发到上线运行全生命周期的 SQL 优化的平台支撑。

表 1 性能调优工具分类

工具分类	功能分类	工具示例
实时性能监控	核心指标监控	MySQL Performance Monitor, Oracle Enterprise Manager
	查询性能监控	
	连接状态监控	
	锁等待监控	
查询分析	SQL 解析	Percona Toolkit, slowlog (MySQL), pgBadger (PostgreSQL)
	执行计划分析	
	慢查询分析	
	索引分析	
数据库负载分析	负载分布	/
	热点分析	
	负载均衡	

资源利用分析	资源利用率	SolarWinds Database Performance Analyzer
	资源瓶颈	
	资源分配	
集群分析	节点状态监控	/
	数据分布	
	故障切换	
性能调优	参数调优	Oracle SQL Tuning Advisor
	配置建议	
	索引优化	

（二）巡检与监控告警

1. 分布式数据库巡检与监控告警的需求

对分布式数据库实施全面、精准的巡检与监控告警至关重要。需求聚焦于实时采集并分析各数据库节点的性能指标，以及业务相关指标，确保数据库运行高效且满足业务需求。同时，巡检需覆盖数据库的备份恢复策略、数据分片与复制状态、安全配置与权限管理等关键领域，监控告警系统则需具备智能预警功能，以最小化故障对业务的影响。

2. 分布式数据库巡检与监控告警面临的痛点

- **技术复杂性与兼容性：**架构各异导致巡检与监控面临兼容性问题。
- **数据海量且分散：**高效整合分析数据与数据实时性和准确性的矛盾。
- **告警精准度低：**缺乏对机制逻辑的理解导致产生大量误报和漏报。
- **响应速度慢：**从告警触发到问题解决存在较长的时延。
- **多云与混合云环境：**在多云、混合云环境下统一监控和管理。

3. 分布式数据库巡检与监控告警的解决方案

(1) **技术架构优化与兼容性:** 采用微服务架构和容器化技术, 实现监控组件的灵活部署和快速迭代, 提高系统的可扩展性和可维护性。通过统一的 API 接口和插件机制, 支持多种分布式数据库类型和版本, 解决技术兼容性问题, 实现跨平台、跨数据库的统一监控。

(2) **智能监控与告警系统:** 通过对历史数据的深度学习, 构建预测模型, 提前识别潜在故障, 减少误报和漏报。结合分布式数据库的业务逻辑机制, 实现智能化的告警分级和自动处置策略, 缩短故障响应时间。

(3) **数据采集与传输优化:** 利用云计算和边缘计算技术, 优化数据采集和传输效率, 减少网络延迟, 确保监控数据的实时性和准确性。采用分布式存储和计算技术, 提高数据处理能力, 满足海量数据的分析需求。

(4) **多云与混合云环境统一管理:** 通过云原生监控工具, 实现对多云、混合云环境下分布式数据库的统一管理和监控。利用云平台的弹性伸缩能力, 根据业务需求动态调整监控资源, 降低运维成本。

(5) **运维自动化与智能化:** 利用自动化脚本和机器人流程自动化(RPA)技术, 实现巡检任务的自动化执行和告警处理的自动化响应。通过构建运维知识库和智能推荐系统, 提高运维人员的工作效率和问题解决能力。

4. 分布式数据库监控工具

监控工具应具备监控主机系统状态、数据库运行状态、空间使用情况、数据库报错以及性能展示的能力; 具备监控界面、邮件、短信、声音告警

等告警能力；具备监控模板功能，能够通过模板为不同的数据库配置不同监控指标；具备发现问题、报警下钻的功能，能够通过下钻的方式快速定位问题相关信息，辅助问题分析；能够提供 SQL 辅助优化分析的能力，能够对监控发现的低效 SQL 进行下钻，展示 SQL 执行计划、相关元数据等信息以便辅助技术人员对慢 SQL 的分析；能够提供接口供其他监控平台集成调用，从而实现专业化监控平台与一体化监控平台的集成；

建议该工具具备（1）基于监控数据进行风险趋势预测告警的能力；（2）对发现问题、报警自动分析，提供智能故障解决建议的能力；（3）将故障解决建议与故障处理系统联动，实现告警自动化处理的能力。

表 2 监控工具分类

工具分类	功能分类	工具示例
性能监控	响应时间监测	Prometheus、Grafana、Navicat Monitor、Zabbix
	吞吐量评估	
	资源利用率分析	
可用性监控	节点状态监测	MHA (MySQL Master High Availability)
	数据复制和同步监测	
	故障自动检测和恢复	
安全性监控	访问控制监测	SkyWalking、ELK Stack
	数据加密监测	
	漏洞扫描和安全更新	
应用到数据库监控分析	访问关系拓扑图	/
	实时指标监控	
	智能告警	

	链路追踪	
管理和维护	监控工具选择配置	MySQLTuner-perl、DBA Dash
	性能优化和调优	
	指标监控	
	异常检测	
	预警通知	

（三）备份和恢复策略

1. 分布式备份和恢复策略的需求

分布式数据库备份恢复策略需确保数据在分布式环境下的完整性、一致性和可恢复性，同时提供灵活、快速的恢复机制，以应对各种突发情况，如硬件故障、数据损坏、人为误操作等。

2. 分布式备份和恢复策略面临的痛点

- **数据一致性问题：** 备份过程中的数据一致性问题导致数据丢失。
- **备份窗口与性能影响：** 备份质量与对业务运行的影响的权衡。
- **恢复速度与效率：** 分布式数据库的恢复过程往往复杂且耗时。
- **存储成本与管理复杂性：** 数据安全性与存储成本和管理难度的矛盾。
- **跨地域与多云平台：** 实现跨地域、跨云平台的分布式备份和恢复。

3. 分布式备份和恢复策略的解决方案

（1）**一致性备份技术：** 采用分布式快照、日志复制等一致性备份技术，确保备份过程中各节点数据的一致性。通过定期的全量备份和增量备份相结合的方式，减少备份数据冗余，提高备份效率。

(2) **备份窗口优化**: 利用低负载时段进行备份操作, 减少对业务运行的影响。采用并行备份和压缩技术, 缩短备份时间, 提高备份效率。

(3) **快速恢复机制**: 构建基于时间点恢复 (PITR) 和即时恢复 (ITR) 的快速恢复机制, 提供灵活的恢复选项。通过预置恢复脚本和自动化恢复流程, 简化恢复操作, 提高恢复速度和效率。

(4) **存储优化与管理**: 采用云存储、分级存储等策略, 降低备份数据的存储成本。通过备份数据去重、压缩等技术, 减少存储空间的占用。建立备份数据的生命周期管理机制, 定期清理过期备份, 降低管理复杂性。

(5) **跨地域与多云平台支持**: 利用云原生技术和分布式存储解决方案, 实现跨地域、跨云平台的分布式备份和恢复。

4. 数据库备份与恢复工具

在使用备份与恢复工具时, 通常以实例为单位。即只需对实例进行设置备份或恢复, 工具将自动对每个实例节点设置备份或恢复。同时, 工具将确保在所有实例节点上对应的备份或恢复操作一致。

备份恢复工具能够提供对备份和恢复的监控能力, 建议支持自动生成备份策略、自动改善备份策略、提示备份恢复风险等智能化能力。

表 3 备份恢复工具分类

功能分类	功能描述	工具示例
备份功能	支持全量备份 支持增量备份 支持差异备份	mysqldump、 mysqlpump、Percona XtraBackup、MySQL

	支持物理备份 支持逻辑备份 支持自动备份 支持手动备份	Enterprise Backup
恢复功能	支持按时间点恢复 支持完全恢复 支持部分恢复 支持快速恢复	mysqldump 恢复、 Percona XtraBackup 恢复
其它功能	支持查看备份进度 支持备份验证 支持备份删除与保留策略 支持跨平台备份与恢复	/

分布式数据库备份与恢复工具的使用对于确保数据的安全性和可恢复性至关重要。为了让用户更好的使用工具，以下是几点使用建议：

（1）定期备份：定期对数据库进行数据备份，确保数据的完整性和安全性。备份频率应根据业务需求和数据变化量进行设定。

（2）选择合适的备份方式：根据业务需求和数据重要性选择合适的备份方式。对于关键业务数据，建议采用物理备份和逻辑备份相结合的方式，以确保数据的全面恢复能力。

（3）备份数据存储：备份数据应存储在安全可靠的位置，防止本地故障导致数据丢失。云存储、本地备份服务器或跨地域备份中心都是常见的选择。

(4) **备份验证**: 定期对备份数据进行验证, 确保备份数据的完整性和可用性。在验证过程中, 可以模拟数据恢复过程, 检查数据是否能够成功恢复。

(5) **制定恢复计划**: 在备份的基础上, 制定详细的恢复计划。明确在数据丢失或系统故障时的恢复步骤和恢复时间。

(6) **使用自动化工具**: 考虑使用自动化备份和恢复工具, 提高备份和恢复的效率和准确性。

(四) 安全与防护

1. 数据库安全防护的需求

数据库安全防护需求包括确保数据的机密性、完整性和可用性。这涉及到强化身份验证、实施数据加密、细粒度访问控制、审计监控、更新和补丁管理, 以及备份和恢复策略, 以防止数据泄露、篡改和未授权访问。

2. 数据库安全防护的痛点

- **连接与访问控制**: 密码简单或权限设置宽松, 增加被破解风险。
- **数据传输与存储**: 传输加密不足被截获篡改, 缺少完整性保护机制。
- **配置错误和软件漏洞**: 不当的配置可能导致系统暴露给攻击者。
- **维护和更新**: 系统维护工作被忽视或延迟导致系统的脆弱性增加。
- **认证和授权**: 缺少基于角色的认证授权机制导致权限管理混乱。
- **审计和溯源**: 审计数据不完整导致难以发现阻止未经授权的操作。
- **隐私保护**: 共享数据时未进行充分的匿名处理导致隐私泄露。

3. 数据库安全防护的解决方案

- (1) **强化身份验证：**使用强密码和多因素认证，限制访问权限。
- (2) **数据加密技术：**采用数据加密技术，对存储传输进行加密保护。
- (3) **定期更新和补丁管理：**及时更新数据库软件，修复安全漏洞。
- (4) **配置静态和动态数据加密：**配置数据库加密和 SSL/TLS 加密。
- (5) **审计和监控：**定期审计数据库用户和权限，监控异常活动。
- (6) **备份和恢复策略：**定期备份数据库，并确保备份文件的安全性。
- (7) **使用防火墙和网络安全措施：**配置防火墙，限制访问 IP 地址。

4. 安全审计工具

分布式数据库安全审计工具是一种专门设计用于监控和记录分布式数据库系统中各类安全事件和操作的的工具。

安全审计工具应支持安全审计功能在线和离线的启用和关闭、审计策略配置等功能；应具备安全追溯和风险分析能力；建议工具支持智能安全检测、自动化分析和自适应的拦截能力

表 4 常见数据库安全审计工具

功能分类	功能描述	工具示例
用户行为审计	监控和记录用户对数据库的各种操作行为,包括登录、查询、数据修改等。	SQLMap、Yearning
风险告警	基于预设的安全规则,对异常行为进行实时告警。	McAfee mysql-audit
数据库配置审计	审计数据库配置和权限,	DAFT

	确保配置安全。	
--	---------	--

在使用分布式数据库安全审计工具时，以下是一些使用建议，以确保工具的安全性和有效性：

（1）设置审计规则：开启安全审计后，需使用安全审计专用账号，审计管理员，配置相应的审计规则。审计规则配置原则应按最小需求配置，尽量做到规则少、记录少，避免因审计影响数据库性能，避免占用过多磁盘等。

（2）管理审计规则：通过可视化页面，提供审计规则的展示、管理能力，包括但不限于审计规则的创建、修改、删除、禁用、启用等操作。需注意无效审计规则，即对于用户、对象审计，删除表或者用户，系统会残留审计配置，需要在可视化页面提示无效的审计规则，并支持一键清理无效审计规则的能力。

（3）查看审计日志记录：配置审计规则后，执行 SQL 语句，满足审计规则的操作将会被记录在审计日志中，由 Agent 定期自动采集并记录在配置库中，可通过控制台页面查看审计信息。

（五）升级

1. 分布式数据库升级的需求

为了满足日益增长的数据、支持更复杂的业务查询和分析功能、提高系统的可用性和稳定性，用户也同样需要对分布式数据库进行升级，以确保分布式数据库性能提升和优化。

2. 分布式数据库升级的痛点

- **数据一致性挑战：**在多节点升级中保持数据迁移和同步的一致性。

- **多节点升级效率**: 大集群或节点数量很大需要提升多节点升级效率。
- **业务连续**: 负载均衡和流量调度机制不足导致业务中断或性能下降。
- **安全合规**: 需要确保数据的安全性, 防止数据泄露或被恶意攻击。
- **数据迁移与同步复杂性**: 数据迁移、数据同步增加了迁移复杂度。

3. 分布式数据库升级的解决方案

(1) **数据一致性与完整性**: 新建实例并同步数据, 使用数据迁移工具, 集成控制台实现数据同步, 自动化脚本和工具快速完成数据同步。

(2) **升级效率提升**: 支持多节点并行批量升级或回退, 以提升效率。

(3) **业务连续性**: 灰度发布和小版本升级, 大版本升级时暂停业务。

(4) **安全与合规性**: 支持数据传输和存储加密, 访问控制技术, 以及安全漏洞扫描与恢复技术, 加强防火墙和入侵防御系统配置。

3. 升级工具及建议

升级过程具备多种数据库架构的标准升级规范与文档, 具备数据库升级的审批和测试流程; 具备详尽的数据库升级调研表, 包含但不限于应用维度、业务维度、环境维度、数据库维度、资源维度; 具备版本评估机制, 对升级需求进行必要性和风险性评估。

升级工具平台能够实现升级的自动化, 建议支持自动感知升级需求、判定升级风险、提供数据库自动化升级能力。

(六) 日志管理

1. 分布式数据库日志管理的需求

分布式数据库日志管理需求包括确保日志的一致性和可维护性，收集和整合多源日志，高效存储与安全保护日志数据，实时分析和可视化日志以快速定位问题，以及制定日志清理和权限管理策略，确保合规性。这要求一个综合的日志管理策略，以应对海量日志数据和系统的复杂性。

2. 分布式数据库日志管理的痛点

- **海量日志处理：**收集、存储和分析海量日志数据的挑战。
- **系统复杂性：**在多节点环境中整合和分析日志，增加系统复杂性。
- **安全性问题：**保护日志数据安全，防止敏感信息泄露。
- **实时性与整合：**确保日志收集的实时性，整合不同源的日志。
- **存储容量规划：**合理规划存储容量，避免资源浪费。
- **合规性挑战：**满足法律法规和行业标准的合规性要求。

3. 分布式数据库日志管理的解决方案

(1) **日志生成：**采用标准化日志框架，定义清晰的日志级别和关键信息记录标准。

(2) **日志收集：**使用高效的收集工具如 Fluentd、Logstash，确保实时性和多源日志整合。

(3) **日志存储：**选择 Elasticsearch 等分布式搜索引擎存储日志，规划存储容量，实施数据加密和访问控制。

(4) **日志分析：**利用 Kibana 等工具进行实时分析和可视化，定义关键分析维度，建立告警机制。

(5) **日志可视化**: 构建日志可视化平台, 提供直观的查询和分析界面, 支持多维度展示。

(6) **日志管理策略**: 制定日志清理和权限管理策略, 确保日志管理合规性。

(7) **最佳实践和优化**: 优化日志生成和收集过程, 结合日志管理系统进行性能监控, 根据反馈持续改进日志管理策略。

4. 日志分析工具

(1) 日志采集:

1) 来源确定:

明确需要收集的日志来源, 在分布式数据库环境中, 每个节点、每个分区、每个副本都可能产生日志。

2) 采集方案:

代理方式: 在数据库节点和日志存储系统之间设置代理程序, 代理程序负责收集各个节点的日志, 并将其传输到集中的日志存储系统。

插件方式: 在数据库系统内部安装日志收集插件, 插件直接将数据库产生的日志发送到指定的存储位置。

系统日志采集: 利用操作系统的日志收集机制, 将数据库产生的日志作为系统日志的一部分进行收集。

(2) 日志存储:

1) 介质选择:

关系型数据库: 可以使用传统的关系型数据库来存储日志, 如 MySQL、Oracle 等。

非关系型数据库: 像 MongoDB、Cassandra 等非关系型数据库非常适合存储分布式数据库日志。

分布式文件系统: Hadoop 的 HDFS 等分布式文件系统也是日志存储的一种选择。

2) 策略制定:

分区存储: 根据日志的产生时间、节点位置、业务类型等因素对日志进行分区存储。

压缩存储: 对日志数据进行压缩可以节省存储空间, 降低存储成本。常见的压缩算法如 GZIP、Snappy 等可以在不影响数据读取性能的前提下, 对日志数据进行有效的压缩。

(3) 日志分析:

1) 实时分析:

流式处理: 利用流式计算框架, 如 Apache Flink、Apache Spark Streaming 等, 对实时产生的日志进行快速分析和处理。

增量分析: 采用增量分析的方式, 只分析新产生的日志或者最近一段时间内的日志。

2) 离线分析:

批量处理: 将一段时间内的日志收集起来, 进行批量处理和分析。

数据挖掘和机器学习: 利用数据挖掘和机器学习算法对日志数据进行分析, 发现隐藏在日志中的模式和异常。

(4) 日志管理与维护:

日志清理: 根据要求, 定期清理过期的日志数据, 以释放存储空间。

日志备份: 为了防止日志数据的丢失, 需要定期对日志进行备份。

系统优化: 根据分析结果, 对日志收集、存储和分析系统进行优化。

表 5 常见数据库日志分析工具

功能分类	功能描述	工具示例
数据收集	从不同来源收集日志数据, 包括数据库操作日志、系统日志等。	Fluentd、Logstash
日志解析	解析原始日志文本, 提取有用信息, 将结构化或半结构化的文本文件转换为可分析的数据。	Logstash
日志存储与管理	对日志数据进行有效存储和管理, 提高日志分析效率, 降低存储成本。	Graylog

(七) 智能运维

1. 智能化运维需求

分布式数据库智能化运维工具利用 AI、大数据和机器学习技术, 实现数据库的自动化监控、预警、诊断、优化和管理。

2. 工具的能力

1) 实时监控与预警

智能化运维工具应具备强大的实时监控能力，能够全面监控分布式数据库的各项性能指标，以及数据查询响应时间等关键业务指标。

2) 智能预警与预测

基于历史数据和实时数据，智能化运维工具应能够建立精准的预测模型，对数据库的运行状态进行预测。当预测到可能出现故障或性能瓶颈时，工具应提前发出预警信号，为运维人员预留足够的处理时间。

3) 故障智能诊断与定位

当数据库出现故障时，智能化运维工具应能够自动收集故障现场的数据，运用机器学习算法对故障进行快速定位和分析，提供可能的故障原因和解决方案建议。

4) 性能智能优化

智能化运维工具对现代数据库管理至关重要，它们通过分析运行数据来识别性能瓶颈和优化点，如不合理的索引和低效 SQL 语句。

5) 自动化管理

智能化运维工具应具备自动化管理功能，包括自动备份、扩容和缩容，自动执行备份保障数据安全，根据流量和数据增长自动扩容以维持性能。

6) 跨平台与兼容性

随着多元化技术栈的并存，智能化运维工具需要具备强大的跨平台、跨数据库兼容性。

7) 定制化与个性化服务

企业运维需求多样化，智能化运维工具通过高度可配置，以适应企业特定需求，利用 AI 算法理解业务逻辑，自动调整运维策略。

8) 数据安全与隐私保护

智能化运维工具需要集成先进的加密技术、访问控制机制以及数据脱敏技术，确保数据在采集、存储、处理、传输等各个环节中的安全。

3. 工具的使用建议

在使用分布式数据库智能化运维工具时，企业应遵循以下建议，以确保工具的有效利用和运维工作的顺利进行：

1) 合理配置与监控

在使用智能化运维工具时，企业应合理配置监控指标和预警阈值，确保工具能够全面、准确地监控分布式数据库的运行状态。同时，企业还应定期检查和更新监控配置，以适应分布式数据库和业务的变化。

2) 充分利用智能诊断与优化功能

智能化运维工具提供的智能诊断与优化功能是企业提高运维效率的重要手段。企业应充分利用这些功能，对分布式数据库进行定期的诊断和优化，发现并解决性能瓶颈和潜在问题。

3) 加强数据安全与隐私保护

在使用智能化运维工具时，企业应高度重视数据安全与隐私保护。企业应确保工具具备先进的数据加密、访问控制以及数据脱敏技术，以防止

数据泄露和滥用。

表 6 常见数据库智能化运维工具

功能分类	功能描述	工具示例
自动化部署与配置管理	实现数据库的自动化安装、配置和版本控制，确保环境一致性和可重复性。	Ansible、Terraform、Chef、Puppet
自动化备份与恢复	建立自动化的数据库备份和恢复机制，确保数据的安全性和可用性。	SQL Server 自带备份工具、Redgate Backup and Restore
持续集成与持续部署 (CI/CD)	将数据库变更纳入 CI/CD 流程，实现自动化测试和部署。	Jenkins、Ansible

三、分布式数据库运维应急保障

(一) 故障处理

1. 分布式数据库故障处理的需求

分布式数据库故障处理需求包括快速恢复、数据一致性、自动化故障管理、容灾部署和故障预警。

2. 分布式数据库故障处理的痛点

- **数据一致性：**多节点间的数据同步可能导致一致性问题。
- **故障定位与恢复：**分布式系统的复杂性使得故障定位和恢复困难。
- **分布式事务管理：**分布式事务管理器可能增加复杂性和性能开销。

- **节点故障与负载均衡：**故障恢复消耗资源，负载均衡影响性能。
- **安全性与权限控制：**加密和权限控制增加系统复杂性和性能开销。

3. 分布式数据库故障处理的解决方案

使用如 MHA (MySQL Master High Availability) 这样的高可用解决方案，实现自动故障检测和转移。MHA 能够在主数据库发生故障时自动将备数据库提升为主数据库，实现快速恢复。通过 MHA 等工具实现自动故障转移和恢复，减少手动干预和资源消耗。同时，使用如 ProxySQL 这样的数据库代理来实现负载均衡，提高系统性能。

4. 故障处理工具及建议

建议故障处理工具支持主动发现、预测故障并提供调整建议，生成清晰、完整的故障分析数据包，支持提供故障发生时最小化业务影响的自动诊断、快速自治处理能力。完善的故障处理流程至少包括故障需求分析、故障数据收集、诊断分析、解决方案、方案验证、回退方案、故障处理实施、跟踪保障、故障总结、知识库完善等阶段。

(二) 高可用与容灾管理

1. 高可用与容灾管理的需求

分布式数据库高可用与容灾管理痛点涉包括分布式事务的一致性、节点故障切换与恢复、负载均衡与资源调度、系统升级与维护、数据备份与恢复、跨地域容灾部署、故障检测与预警以及应急响应与恢复计划等。

2. 高可用与容灾管理的痛点

- **一致性维护困难:** 多个节点间的协调同步增加系统的复杂性和开销。
- **故障切换挑战:** 可能会遇到数据不一致、服务中断等问题。
- **负载均衡复杂性:** 需考虑数据分布、访问模式、节点性能等因素。
- **系统升级风险:** 系统升级过程中可能影响业务正常运行。
- **数据备份与恢复问题:** 可能会遇到数据丢失、恢复时间长等问题。
- **跨地域部署成本:** 跨地域容灾部署会增加系统的复杂性和成本。
- **故障检测难度:** 分布式数据库的多节点特性使故障检测变得困难。
- **应急响应计划设计:** 需要充分考虑各种可能的故障场景和恢复策略。

3. 高可用与容灾管理的解决方案

针对上述痛点，解决方案包括：

(1) **一致性解决方案:** 采用两阶段提交 (2PC)、三阶段提交 (3PC)、补偿事务 (TCC) 等策略，以及利用本地消息表和事务消息保证一致性。

(2) **故障切换机制:** 实现自动故障检测、主节点选举、请求重新路由和数据恢复流程。

(3) **负载均衡策略:** 采用读写分离、数据分片、动态节点扩缩容、智能请求路由和负载均衡算法。

(4) **系统升级方法:** 使用滚动升级和蓝绿部署，确保升级过程中服务可用性。

(5) **数据备份与恢复策略:** 实施全量备份、增量备份、日志备份和多副本机制，优化备份存储和管理。

(6) **跨地域容灾部署**: 在多地数据中心架构中实现数据的跨地域复制和冗余存储。

(7) **故障检测与预警系统**: 部署监控代理和日志收集系统, 利用智能分析技术识别故障模式, 建立预警通知机制。

(8) **应急响应与恢复计划**: 制定详细的应急预案, 包括故障分类、响应流程、恢复策略和定期演练培训, 提升应对突发事件的能力。

4. 容灾建设及灾难恢复工具平台

建议工具平台支持自动部署容灾、对容灾进行监报告警、智能诊断容灾状态并进行容灾切换等自动化、智能化能力, 确保在同机房或同城场景下, $RT0 \leq 3$ 分钟, RPO 为 0; 在不同城的场景下, $RT0 \leq 5$ 分钟, RPO 为 0。建议具备完善的容灾体系、通用的容灾方案、容灾部署文档、容灾切换文档, 支持对容灾实施、容灾切换中每个流程执行时间、质量的定量和定性指标, 能够对容灾实施、容灾切换的效果与质量进行量化分析。

四、分布式数据库运维体系建设

(一) 组织建设

组织建设能力是指具备建立健全标准化数据库运维管理体系的能力, 具体包括团队建设和制度建设, 能够满足组织对数据库运维管理工作的基本要求。

组织建设的过程描述具体如下:

(1) 拥有专门的数据库运维服务团队，设置明确的团队工作目标、清晰的运维人员岗位职责、分工和考核指标；

(2) 拥有完整的数据库人员等级技能定义，以及完善的人员晋级流程、制度；

(3) 拥有完整的数据库运维制度，如标准运维制度、流程等。

建议持续优化组织架构、团队目标、岗位职能和人员配置,以实现整体业务目标的优化，持续优化标准化运维体系相关的制度、流程、管理规范 and 考核要求，具备行业领先优势，确保实现业务目标。

(二) 能力培养

能力培养是指为实现运维团队目标，有计划地组织运维人员进行学习和培训，以提高运维团队的专业技能和标准化运维的能力，使运维人员胜任本职工作。

能力培养的过程描述具体如下：

(1) 建立针对人员培训相关的管理制度，能够全面评估内部人员的专业技能水平程度以及内部人员的技术成长需求，制定全面和持续的培训教学计划和任务，包括但不限于时间计划、教学教案、培训目标等方面的制定；

(2) 能够根据培训教学大纲在确定的场所或远程教学方式，开展教学任务；

(3) 培训完成后，根据培训教学任务，分阶段、分专业对培训完

成质量进行评估，进一步完善培训体系，提升培训效果。

建议具备可量化的培训管理制度和梯队人员培养计划，拥有完善的教学目标，每年结合人员能力、岗位需求、组织发展等制定当年人员培训计划并执行。

（三）运维流程优化

数据库运维流程的持续改进旨在达成如下目标：

（1）提升系统稳定性：数据库作为信息系统的核心，系统稳定性直接关系到业务连续性。运维流程的改进可以通过降低故障率、减少宕机时间，确保数据库的稳定运行。

（2）提高运维效率：通过流程优化，减少人为操作和冗余流程，提高工作效率，降低运维团队的劳动强度。

（3）减少人为错误：在运维过程中，误操作可能引发数据丢失、系统宕机等严重后果。优化流程有助于引入更多的自动化与监控措施，从而降低人为错误风险。

（4）支持业务快速增长：随着业务的发展，数据量增加、用户访问增多，数据库的扩展需求也随之增加。改进运维流程，可以支持数据库架构的快速扩展。

（5）缩短问题响应时间：改进流程中的监控和告警机制，使团队能够在问题发生时快速响应和定位，从而减少故障对业务的影响。

2. 流程评估与痛点分析

在实施持续改进之前，需要先对现有运维流程进行评估，识别出流程中的瓶颈和改进需求。这一过程可以通过以下方法展开：

（1）数据指标分析

通过数据库系统中的关键性能指标（KPI），如响应时间、CPU 和内存占用、故障次数等，识别出数据库当前的瓶颈。例如，某些查询的响应时间较长、CPU 利用率经常过高等都可能是流程中存在问题的征兆。

（2）故障回顾与根因分析

回顾以往的故障案例，找出引发故障的根因，分析哪些流程步骤可能不存在不合理之处或需要改进的地方。例如，如果数据库备份恢复的时间过长，可以考虑优化备份流程或增加自动恢复手段。

（3）流程可视化

将现有的数据库运维流程绘制成流程图，直观地展示每个操作的顺序和依赖关系。这有助于发现不必要的冗余步骤或优化的可能性。例如，对于周期性任务，可以考虑是否通过合并步骤来减少重复操作。

（4）人员反馈收集

通过调研运维团队成员，收集他们的反馈，了解在日常工作中有哪些痛点，尤其是哪些重复性、耗时或容易出错的工作可以优化。

（5）业务部门和客户反馈

收集用户对数据库响应速度、故障恢复时间等方面的反馈，了解从用户角度有哪些急需改进的地方，从而制定更加合理的流程优化方案。

五、分布式数据库运维管理标准能力模型

(一) 数据库集中运维管理平台

《数据库集中运维管理平台技术要求》包含平台基础能力、平台资源管理能力、安装部署能力、性能分析及优化能力、巡检健康检查能力、高可用能力、运维管理能力、备份和恢复能力、数据库安全管理能力、查询语句执行能力、平台租户管理能力、兼容能力等十二大能力域，为构建功能全面、性能稳定、高可用的数据库管理平台提供技术参考。

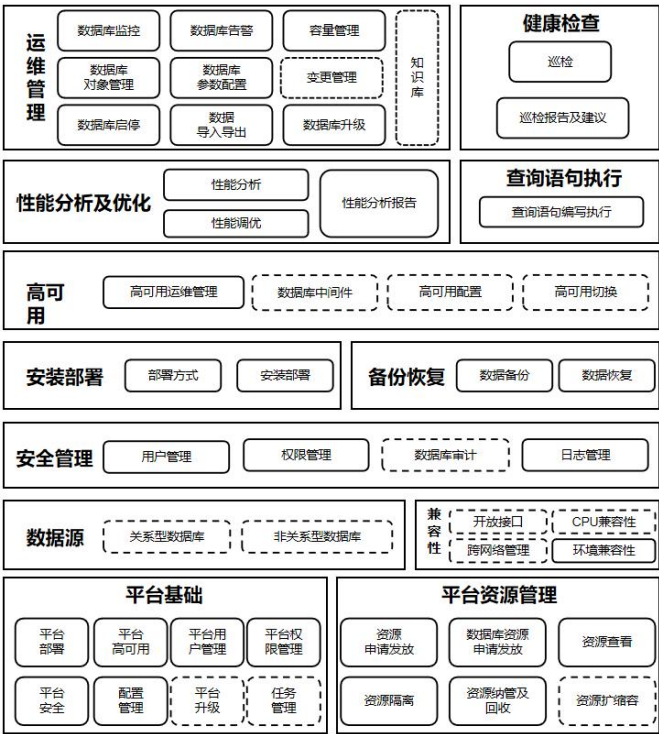


图 2 数据库集中运维管理平台标准框架

截至 2024 年 12 月，已有 5 家企业通过该标准测试，通过情况如下：

表 7 数据库管理平台测试通过情况

公司名称	产品信息
------	------

杭州云猿生数据有限公司	KubeBlocks Enterprise 信创云数据库平台
联想（北京）有限公司	数据库管理平台 V1.4
阿里云计算有限公司	DMS 数据管理平台
云和恩墨（北京）信息技术有限公司	数据库云管平台 zCloud
上海爱可生信息技术股份有限公司	云树数据库集群管理平台软件【CTREE DMP】

（二）SQL 质量管理平台

《大数据 结构化查询语言（SQL）质量管理平台能力分级要求》包含 SQL 采集域、SQL 审核域、SQL 查询域、SQL 优化域、集成兼容域等六大能力域，帮助数据库服务商及用户方评估自身 SQL 质量管理平台的技术能力。

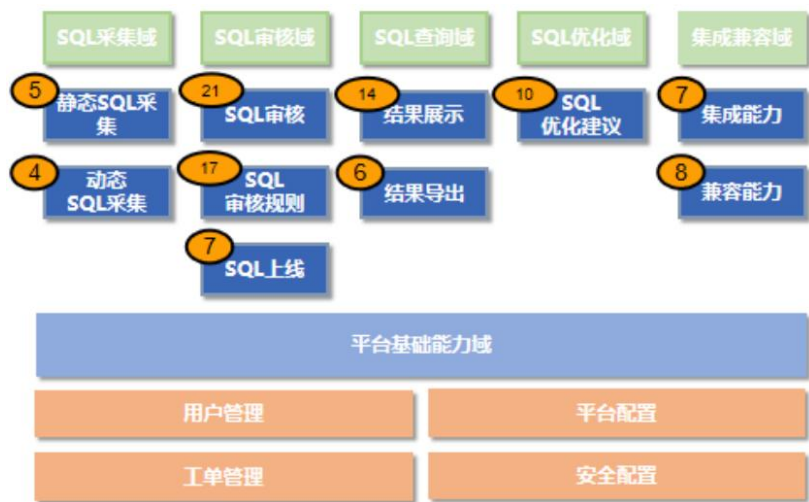


图 3 SQL 质量管理平台标准框架

截至 2024 年 12 月，已有 3 家企业通过该标准测试，通过情况如下：

表 8 SQL 质量管理平台测试通过情况

公司名称	通过等级
联想（北京）有限公司	第三级（先进级）
云和恩墨（北京）信息技术有限公司	第三级（先进级）
上海爱可生信息技术股份有限公司	第三级（先进级）

（三）数据库运维管理成熟度模型

《数据库运维管理能力成熟度模型》囊括组织建设、制度流程、人员能力和技术工具等维度共计 800 余条详细要求，能力域根据业务类型划分为运维体系建设能力、规划实施能力、基础运维能力、数据库诊断优化能力、监控巡检能力、故障处理能力、数据库高可用与容灾能力、安全与合规能力八大能力域进行评估。评估结果由低到高分为初始级、可重复级、稳健级、量化管理级、优化级五个等级。

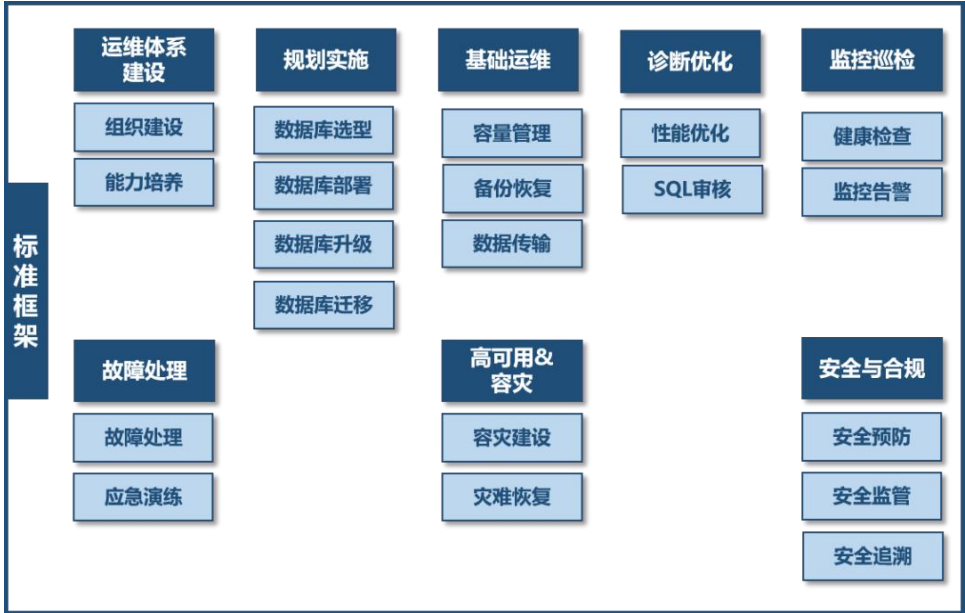


图 4 数据库运维管理能力成熟度模型标准框架

截至 2024 年 12 月，已有 5 家企业通过该标准测试，通过情况如下：

表 9 数据库运维管理能力成熟度模型测试通过情况

公司名称	通过等级
中国移动通信集团北京有限公司	第四级（量化管理级）
中国移动通信集团陕西有限公司	第四级（量化管理级）
中国移动通信集团湖北有限公司	第四级（量化管理级）

上证所信息网络有限公司	第四级（量化管理级）
中国移动通信集团浙江有限公司	第四级（量化管理级）

六、分布式数据库运维管理的未来趋势

（一）云计算与数据库运维

云计算与数据库运维在 2024 年以后的未来趋势均表现出强劲的增长潜力和广阔的发展空间。随着技术的不断进步和应用需求的不断增长，这两个领域将持续创新和发展，为数字化转型和可持续发展提供有力支撑。

云原生数据库的兴起：随着云计算技术的不断发展，云原生数据库应运而生。云数据库以云化运行环境为前提，在设计上通常结合分布式技术并普遍采用计算——存储分离、日志即数据的设计思想，能够灵活调动资源进行扩缩容，实现资源池化、弹性变配、集约运维等能力。

多云和混合云部署：云计算的发展促使分布式数据库向跨云和多云部署发展，以提供更高的灵活性和可扩展性。这种部署模式允许企业根据业务需求和成本效益在不同的云服务提供商之间灵活迁移数据和应用，同时降低对单一云服务提供商的依赖。

自动化和自服务化运维：云计算推动了自动化和自服务化运维的发展。数据库即服务（DBaaS）模式使得企业可以通过网络访问和使用数据库服务，无需关注底层的硬件和软件安装、配置、维护等工作，从而减轻企业的运维负担。

（二）人工智能在运维管理中的应用

人工智能在运维管理中的应用涵盖了智能监控、故障诊断、告警管理、资源管理、安全防控以及运维管理与决策支持等多个方面。展望未来，分布式数据库智能化运维的发展将呈现出更为多元与深入的趋势，具体可从以下几个方面进行扩展阐述：

技术融合与创新：人工智能技术正使运维系统能自动分析日志、预测故障和优化资源。大数据分析技术帮助从海量数据中提取决策支持信息。云计算提供弹性算力，支持资源按需调配，降低成本，提升响应速度。

定制化与个性化服务：企业运维需求差异大，智能化运维工具需提供高度可配置的服务，根据企业需求调整功能。AI 算法帮助理解业务逻辑，自动调整运维策略。

安全与隐私保护：数据安全和隐私保护是智能化运维工具的关键责任。面对严格的法律法规，工具需集成加密、访问控制和数据脱敏技术，确保数据全流程安全。

（三）DevOps 与数据库运维

DevOps 与数据库运维的趋势正在朝着更加高效、智能和协同的方向发展。通过采用新技术、优化流程和提高团队协作能力，企业可以不断提高软件开发和运维的效率和质量，为业务发展提供有力支持。

一体化 DevOps 平台的兴起：这些平台通常集成了代码管理、持续集成/持续部署、自动化测试、监控和日志分析等功能，为开发、运维和测

试团队提供了一个统一的协作环境。数据库运维也可以被纳入这一体化平台中，实现数据库自动化部署和监控等功能，提高运维效率和可靠性。

DevOps 文化的推广：强调协作、自动化和持续改进。这种文化也可以被应用到数据库运维中，促进开发、运维和数据库管理员之间的协作和沟通，共同提高数据库的质量和性能。

智能化运维工具的应用：随着 AI 和 ML 技术的发展，智能化运维工具在 DevOps 和数据库运维中得到了广泛应用。这些工具可以自动分析监控数据等信息，快速准确地诊断故障，并提供智能化的运维建议和决策支持。

（四）异构数据库运维

异构数据库系统的发展趋势包括技术融合与创新、数据实时性与智能性、数据整合与互操作性、安全与隐私保护以及市场与应用拓展等方面

技术融合与创新：通过集成大数据处理框架，异构数据库系统能够实现对海量数据的快速分析和处理。异构数据库系统将更多地应用在云计算环境中，使得异构数据库能灵活应对数据量动态变化，同时降低运维成本。

数据实时性与智能性：随着实时数据分析需求的增加，异构数据库系统需要支持数据的实时处理。这包括实时数据摄入、实时查询和实时分析等功能，以满足企业对数据时效性的高要求。

数据整合与互操作性：随着数据量的增加和查询复杂度的提高，异构数据库系统需要支持跨数据库查询和优化。异构数据库系统需要提高数据整合能力，将分布在多个数据库中的数据整合，以便进行查询和分析。

七、总结

在本《分布式数据库运维管理实践指南》中，我们深入探讨了分布式数据库运维管理的各个方面，从基础概念到高级实践，从性能优化到故障排除。以下是对本指南内容的总结和对企业数据库运维管理的建议。

（一）典型问题发现总结

数据一致性问题：在分布式数据库中，数据一致性是一个核心挑战。解决数据同步复制问题需要精细的配置和对一致性级别的深入理解。

网络延迟和分区容错：网络问题导致的延迟和分区是分布式数据库运维中的常见问题，需要高可用架构和有效的分区容错策略来解决。

事务管理复杂性：分布式事务的管理比传统数据库更为复杂，尤其是在跨多个节点时。合理的隔离级别和锁策略利于维护数据完整性。

性能优化难题：随着数据量的增长，性能优化成为分布式数据库运维中的一个关键问题。

故障恢复和高可用性：故障恢复能力是衡量分布式数据库稳定性的重要指标，自动化故障检测和恢复机制对于减少系统停机时间非常有效。

安全和审计挑战：在分布式环境中，数据的安全和审计变得更加复杂。我们总结了实施数据加密、访问控制和审计跟踪的最佳实践。

运维管理复杂性：随着分布式系统的扩展，运维管理的复杂性也随之增加。自动化运维工具和流程对于简化管理、提高效率至关重要。

(二) 对企业分布式数据库运维管理的建议

通常一个完整全面的集中统一数据库管理平台会为用户提供从资源管理、安装部署到各类监控、巡检、以及优化、故障诊断等多方面的能力支持，基本涵盖数据库运维管理的各个方面，建议在具备如下能力：

1. 资源管理

- (1) 能够屏蔽不同架构/地域/类型数据库和主机资源。
- (2) 管理不同环境中的数据库与主机，整合资源、统一管理。
- (3) 能够实现资源统一集中管理，统一分配，利用率高且易管理。
- (4) 具备资源弹性扩展、收缩和回收的能力。

2. 安装部署

- (1) 按不同的数据库类型、场景选择模板，实现数据库逻辑隔离。
- (2) 使用人员无需关心软硬件的搭建、配置和资源占用，一键部署。

3. 监控告警

- (1) 可对数据库进行实时监控告警。
- (2) 提供不同维度的数据库运行指标监控，快速发现数据库各种异常并实时告警，减少或规避故障的发生，降低故障带来的业务影响。
- (3) 能够提供查看监控历史信息。

4. 智能健康巡检

能够通过对数据库运行状态数据的自动判断，以及指标关联分析，结合算法知识库，全面检查数据库存在的健康隐患，帮助用户更好的了解数据库运行状态，并提出改进建议。

5. 性能管理

能够通过对数据库性能指标和相关对象的分析，快速定位数据库性能问题，提升数据库运行稳定性和可靠性，帮助运维管理人员快速解决问题。

6. 一站式智能诊断、处理

- (1) 具备问题生命周期管理能力。
- (2) 生成专家诊断树，继承专家经验，不断迭代，简化人工分析。
- (3) 支持自定义诊断树。

7. 数据库容量管理

- (1) 能够辅助用户进行高效容量规划，减少前期的硬件投入。
- (2) 在业务快速增长，数据库容量发生变化的情况下，能够自动扩容或手动一键扩容的方式进行快速扩容。

8. 可用管理

- (1) 能够通过一键部署高可用数据库架构。
- (2) 能够实时探测和自动故障诊断，快速发现业务节点异常状态，保障数据完整性、一致性同时，快速进行高可用切换，保障业务的连续性。
- (3) 自定义高可用策略，满足不同场景的高可用及容灾管理需求。

9. SQL 审核管理

(1) 实现 SQL 审核需求，提供标准的工单流程，对用户发起的 SQL 变更进行合规性审核，避免危险变更影响到数据库运行的安全性和稳定性。

(2) SQL 审核功能通过系统自动审核 ->DBA 复核 ->开发人员修改提交等流程步骤，使得变更 SQL 得到流程化妥善处理。

附录：分布式数据库运维管理案例分析

本指南梳理了部分国内数据库厂商和数据库专业服务商主导或参与的典型运维案例。案例介绍框架分为案例背景、技术工具、制度流程、实施方案、效果评估五部分。

(一) 天翼云在线扩容实践案例

1. 案例背景

为了迎接业务的快速增长，保证业务的连续性和用户体验，天翼云 TeleDB 决定对分布式数据库进行在线扩容。

2. 技术工具

天翼云 TeleDB 通过集成在线扩容能力，能够按需扩充集群，保证集群可以支撑到 PB 级别的存储。同时结合业务历史数据不常被访问的特点，可以将历史数据自动转移到廉价存储设备上，兼顾性能和成本。

3. 制度流程



图 5 数据库扩容流程图

(1) 扩容前准备

- 性能评估：对当前的数据库性能和容量需求进行全面评估。
- 资源规划：根据评估结果，规划所需的额外资源。
- 数据拆分策略：制定数据拆分策略。

(2) 在线扩容实施

- 读写分离：通过读写分离技术降低主节点的压力。
- 水平扩容：在不停机的前提下，将数据从现有节点迁移到新节点上。
- 渐进式数据迁移：新请求根据数据迁移的进度被路由分到相应节点。
- 无中断配置更新：通过动态更改配置将请求和数据迁移到新节点。

(3) 扩容后验证与优化

- 性能监控：扩容期间密切监控系统性能。
- 容量规划：对预期的负载增长进行提前规划和测试。
- 数据一致性校验：在扩容完成后，进行数据一致性校验。

- 性能优化：根据扩容后的实际情况，对数据库进行性能优化。

4. 实施方案

在分布式数据库中，当对其进行扩容并增加节点数后，数据分布的计算逻辑就会导致已经存在的数据无法正常访问。为处理这一变化，天翼云 TeleDB 通常采取哈希槽方式和一致性哈希两种策略来确保数据的均匀分布和高效路由：

5. 效果评估

本次评估的主要目的是验证分布式数据库在线扩容能力的实际效果，包括性能指标、容量指标、稳定性指标和用户体验等。

- 性能指标：包括吞吐量、响应时间、延迟等关键性能指标。
- 容量指标：包括存储容量、节点数量等。
- 稳定性指标：包括系统稳定性、数据一致性、容错性等。
- 用户体验：包括系统可用性和故障恢复时间等。

（二）中移杭研分布式 MySQL 集群升级实践

1. 案例背景

随着 MySQL 5.7 的生命周期即将结束，为获得最新安全补丁、错误修复，并使用例如在线 DDL、隐藏索引和压缩的 binlogs 等新功能，故计划将生产 MySQL 升级至 8.0 版本。

2. 技术工具

技术工具包含 gh-ost、orchestrator 及内部自动化系统，工具及自

动化系统需处理 5.7 和 8.0 混合版本。

3. 制度流程

升级过程需遵守：

- （1）能够升级每个独立集群，遵守 SLO 和 SLA 指标。
- （2）升级过程需能够回滚至 5.7 版本而不中断服务。

4. 实施方案

第一步：滚动升级 replica 副本

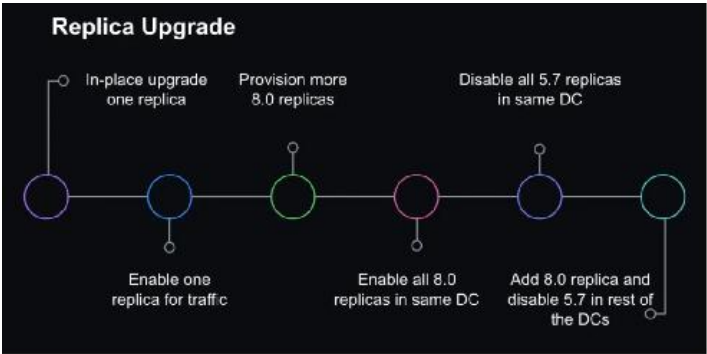


图 6 滚动升级副本流程示意图

第二步：更新 replication 复制拓扑

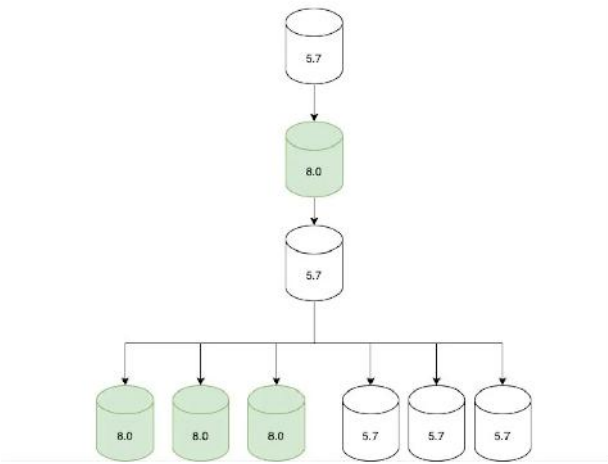


图 7 更新复制拓扑示意图

第三步：把 MySQL 8.0 切换为主库

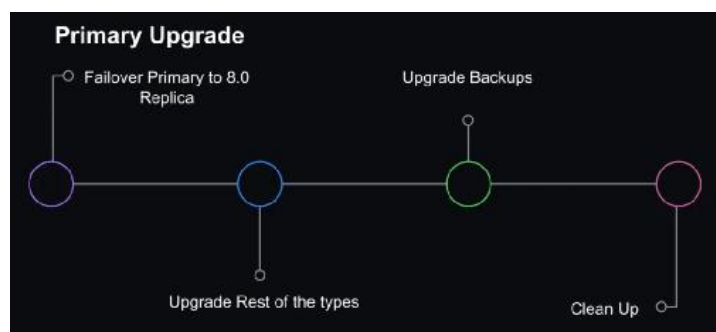


图 8 主库切换流程示意图

第四步：内部实例升级

将用于备份或非生产工作负载的 MySQL 在后续升级，以保持一致性。

第五步：清理

确认集群不需要回滚并成功升级到 8.0 后，删除 5.7 节点。

5. 效果评估

从计划升级开始，最终升级完成用时半年，成功将线上数据库集群及自动化工具升级至 8.0 版本。期间处理了例如 Replication 复制延迟、Java 客户端缓存兼容性问题，成功升级集群，促使对数据库自动化维护工具及能力的提升，流程的优化，并解决了旧版本遗留的安全风险，也为业务需求的支撑提供了更高性能。

（三）某股份制银行分布式核心系统改造与运维监控实践

1. 背景介绍

某股份制银行作为国内大型商业银行响应国家关于金融科技自主可控的号召，对其核心业务系统进行全面改造，将原有的 IBM AS/400 主机

+DB2 数据库架构替换为国产服务器+金篆信科分布式数据库 GoldenDB。

2. 实施方案

(1) 系统架构

该银行分布式核心系统采用两地三中心部署模式，系统架构包括应用集群、数据库负载均衡、计算节点集群、数据节点集群等多个部分。

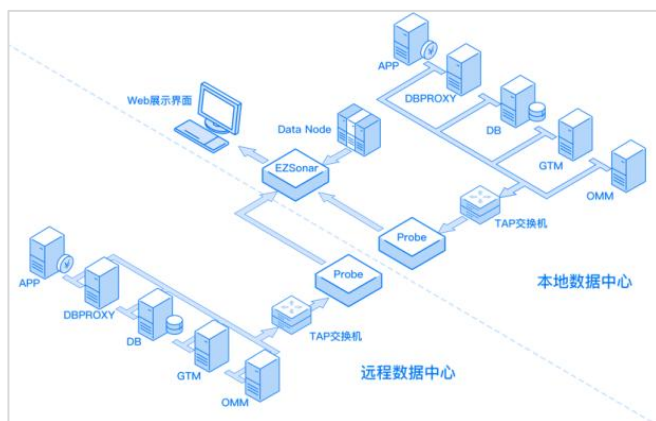


图 9 银行核心系统架构图

(2) 数据库选型

选择了金篆信科分布式数据库 GoldenDB 作为新系统的数据库支撑。

(3) 运维监控工具选择

为了确保新系统的稳定运行，该银行选择了融天 EZSonar 分布式数据库运维监控解决方案。

3. 运维监控实践

(1) 访问关系拓扑图

融天 EZSonar 提供了从应用集群到数据库各节点的访问关系拓扑图。运维人员可以通过拓扑图全面、实时地掌握整个分布式数据库的运行状况。

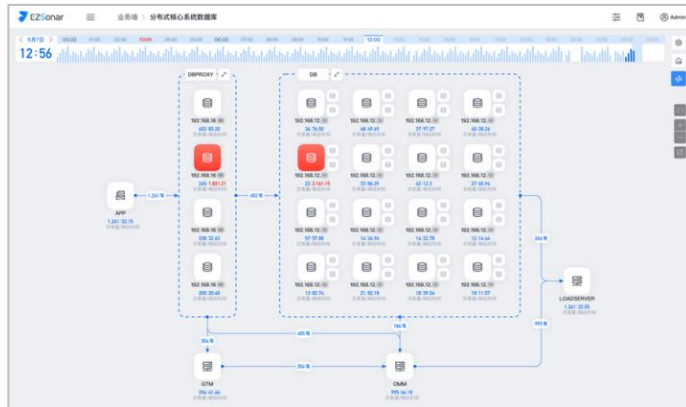


图 10 访问关系拓扑图

(2) 实时指标监控

融天 EZSonar 能够实时监控并展示每个节点的关键性能指标,如请求数量、平均响应时间、响应率、成功率、返回结果等。

(3) 智能告警

融天 EZSonar 基于预置的告警规则,对访问关系拓扑图中的各节点进行实时监测。

(4) 链路追踪

融天 EZSonar 提供了从应用到数据库节点的链路追踪功能。

4. 改造效果

系统稳定运行: 该银行分布式核心系统自改造完成以来,已稳定运行超过 3 年。

性能卓越: 系统承载了 3 亿用户、15 亿账户,日均交易 3 亿笔,平均交易时延仅为 50ms。

(四) 某证券公司数据库集中运维管理实践

1. 案例背景

某证券总部设于深圳，在全国设立了数十家分公司、拥有近 400 家营业网点。随着信创工作的开展，该证券公司新引进了多款数据库，但缺乏统一的监控及维护管理平台，数据库运维自动化程度较低。

2. 技术工具

引入云和恩墨基于鲲鹏的 zCloud，与现网系统对接，打造全栈信创数据库 DBaaS 平台。DBaaS 平台统一纳管数百套各类国产信创、商业和开源数据库，上线功能模块包括数据库巡检、数据库监控告警、数据库性能分析、数据库故障一键诊断等涵盖数据库运维全生命周期的多个功能模块。

- 基于鲲鹏架构，实现数据库安装部署、监控、灾备等自动化能力。
- 对接 ITIL 申请数据库安装部署功能和用户创建及授权功能
- 对接 IaaS 云管平台，申请虚拟机资源自动完成数据库安装部署
- 对接 CMDB 提供数据库基本信息
- 对接 IAM 实现 SSO 用户权限

3. 实施方案

该项目建设伊始，云和恩墨采用敏捷开发管理模式，与客户协同推进，让 zCloud 顺利与客户的 ITIL、IaaS、CMDB 等模块对接，全面实现对客户多种数据库的统一纳管，同时 zCloud 提供的数据库巡检、监控告警等模块能够充分满足客户智能运维需求，实现了流程管理统一化、资产管理精细化的一体化运维管理，建立了符合金融机构的 IT 运维服务流程。

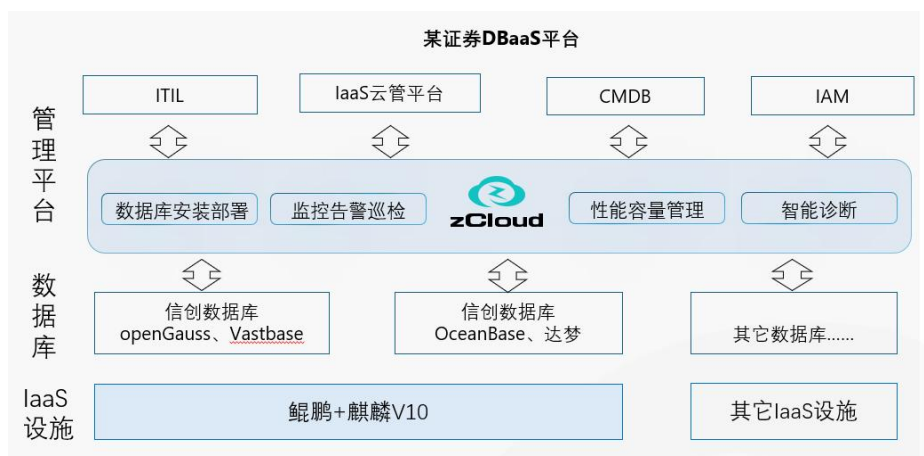


图 11 某证券 DBaaS 平台架构

4. 效果评估

- 平台显著提升了客户的国产数据库运维能力，大幅降低人员技能要求，运维效率和安全性提升。
- 集成最佳实践和专家经验，数据库主动巡检，提前规避 80% 问题
- 三位一体，智能高效诊断，问题定位效率提升 10 倍：
- 通过 zCloud 平台让用户的数据库管理从数种工具频繁切换管理模式转向统一平台管理，资产和运维双视角展示，资源、状态一目了然。

（五）某科技馆一体化运维监控系统实践

1. 案例背景

某省科技馆信息化系统不断建设，其 IT 基础设施和智能化系统日益复杂，传统的运维管理模式已难以满足科技馆对系统稳定性、安全性和高效运维的需求。因此，该科技馆决定实施一体化运维监控项目，以提升运维效率，确保各系统稳定运行，为游客提供更好的参观体验。

2. 技术工具

监控平台：采用夜莺作为核心监控工具，实现对科技馆内所有 IT 设备和智能化系统的实时监控，包括服务器、网络设备、存储设备、安防摄像头等。

日志管理：引入 ELK 进行日志收集、分析和告警，实现对系统异常行为的快速定位和响应。

自动化运维：利用 Puppet 进行配置管理和自动化部署，减少人为干预，提高运维效率。

可视化展示：基于一体化运维监控系统采集的各种指标，我们为业主定制化开发了适用于该科技馆的领导驾驶舱，便于运维人员直观了解系统状态。

3. 制度流程

运维规范：制定详细的运维操作手册和应急预案，确保运维活动遵循统一标准，提高应急响应能力。

定期巡检：建立定期巡检制度，对科技馆内所有设备和系统进行全面检查，及时发现并处理潜在问题。

故障处理流程：明确故障报告、分析、处理和反馈的流程，确保故障能够得到及时、有效的解决。

4. 实施方案

需求分析与规划：与科技馆各部门沟通，明确监控需求，制定项目计划。

环境搭建与配置：部署监控平台、日志管理系统、自动化工具等，完成基础配置和测试。

数据接入与整合：将科技馆内所有设备和系统的数据接入监控平台，

进行统一管理和展示。

培训与推广：对运维团队进行系统使用培训，逐步推广至科技馆各部门。

持续优化：根据运行反馈，不断调整监控策略，优化告警规则，提升系统效能。

5. 效果评估

系统稳定性提升：通过实时监控和自动化运维，科技馆内各系统的稳定性显著提升，故障率大幅下降。

运维效率提高：自动化工具和可视化展示的应用，使运维人员能够更快速、准确地定位和处理问题，运维效率显著提高。

资源成本节约：通过精准的资源监控和优化配置，有效降低了科技馆的运维成本。

（六）某云项目数据库集中运维管控一体化平台案例实践

1 概述

电科金仓数据库集中运维管控一体化平台支持金仓数据库的全生命周期管理。该平台集成了多维度监控指标、健康评价、性能分析和故障诊断模型，提供自动化运维功能，包括租户管理、安装、纳管、扩缩容、备份恢复、升级、监控告警、健康检查、自动巡检和故障分析等。

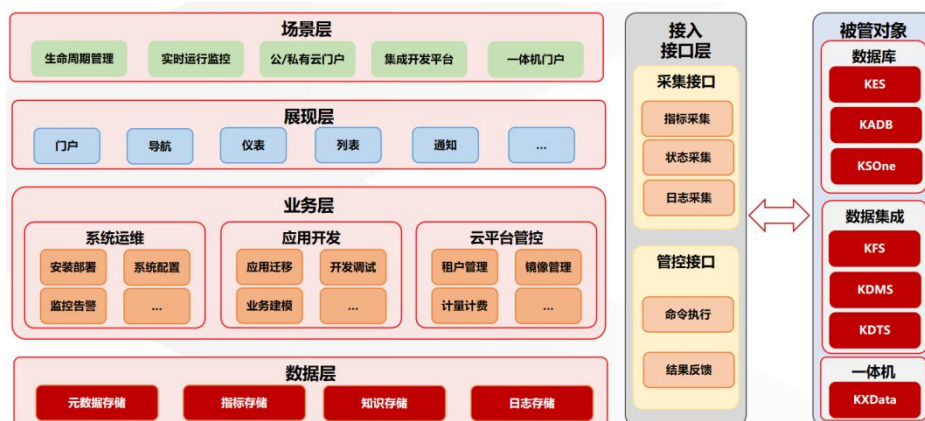


图 12 集中运维管控一体化平台架构

2. 案例背景

在广东省某云项目中，中电科金仓（北京）科技股份有限公司（以下简称“电科金仓”）成功部署了 KES 数据库服务，为用户单位提供强大的数据库支持。然而，在用户的云管平台（RDS）环境中，数据库的稳定运行对业务至关重要，但随之也带来了若干挑战：

数据复杂度：大规模的数据存储和处理增加了实时监控的难度。

性能调优：缺乏有效的性能指标跟踪会导致优化策略制定困难。

成本控制：不适当的监控解决方案可能会带来高昂的成本。

服务器中断：意外的数据库中断会影响日常业务，可能造成损失。

数据维护：高可用性和数据复制策略不足可能导致业务风险。

3. 技术工具

为了应对上述挑战，电科金仓引入了其自主研发的数据库监控系统——KMonitor。该系统通过在数据库服务中部署探针，实现了对运行状态的实时监控，能够及时、准确地发送预警信息，预防潜在故障风险。

4. 制度流程

为确保数据库的高效运作，电科金仓建立了完善的故障风险管控策略。

这不仅涉及到技术层面的监控与报警设置，还包括一系列操作规程。

5. 实施方案

1. 资源监控：持续监测云主机资源使用情况及实例存活状态，提供给运维人员详细的主机使用信息，以便快速定位问题根源，减少误报。

2. 性能洞察：实时展示数据库性能指标，使管理员能够深入理解关键性能参数，通过自定义仪表盘直观地查看性能数据，确保服务质量。

3. 异常警报：自动触发异常情况下的警报，帮助降低平均修复时间（MTTR），提高问题解决效率。

6. 效果评估

目前，电科金仓的监控解决方案已满足管理和运营需求，功能仍在迭代更新中。通过这一系列措施，可以显著提升数据库服务的质量，增强系统的稳定性和响应速度，最终为客户带来更好的体验。

（七）浪潮云运维管理实践案例

1. 案例背景

随着业务的快速发展，浪潮云在高并发、跨地域容灾和动态扩展等方面面临严峻的分布式数据库运维挑战。为此，浪潮云信息技术股份公司（以下简称“浪潮云”）采用分布式数据库架构，大幅提升数据管理和处理能力，同时针对节点扩容、缩容、故障恢复和性能优化等复杂运维任务，制定了系统化管理实践，以保障数据库服务的高效稳定运行。

2. 技术工具

浪潮云结合业务需求，构建了一套基于多种技术工具的运维体系，涵盖分布式数据库、运维调度、业务支撑、运营管理和云监控五大系统，全面提升 IT 服务管理能力。

a) 分布式数据库系统：浪潮自主研发的 HTAP 分布式数据库以 100% 自主知识产权为基础，专为高并发、高可用场景设计。

b) 运维调度系统：基于 ITIL 核心实践，运维调度系统为客户和 IT 服务提供者搭建高效协作的纽带。

c) 业务支撑系统：业务支撑系统覆盖云服务全生命周期管理，包括服务申请、审批、实施等关键流程，满足客户从资源上线到运营管理的需求。

d) 运营管理系统：运营管理系统整合用户、商品和订单等全量数据，为“一朵云”运营提供多维度数据分析与决策支持。

e) 云监控系统：云监控系统以云平台 SLA 为核心，保障基础设施、云平台和服务产品的稳定运行

3. 制度流程

为实现高效管理，浪潮云制定了分布式数据库的标准化运维流程：

a) 节点管理流程：严格审批节点初始化和集群加入，确保规范操作。

b) 备份与恢复流程：每天执行全量备份，每小时执行增量备份。

c) 变更管理流程：所有数据库配置和版本升级需提前 7 天申请。

d) 故障处理流程：建立 7 × 24 小时值班制度，按故障级别响应。

e) 定期巡检制度：每月进行全面巡检，生成报告并优化性能瓶颈。

4. 实施方案

实际实施中，浪潮云遵循“分阶段、可持续”的策略，具体步骤如下：

a) 需求分析与规划：与业务部门深入沟通，明确数据库容量、事务并发量、故障容忍度等关键指标，制定运维目标。

b) 环境搭建：基于 Docker 和 Kubernetes 搭建分布式数据库测试环境，通过模拟真实负载，验证架构设计的可行性。

c) 自动化部署：实现数据库集群自动化部署，避免人为误操作。

d) 监控与优化：上线后，利用 Prometheus 采集实时指标，结合 SQL 分析工具定位热点查询，优化索引和表结构。

e) 培训与知识分享：定期为运维团队和开发人员开展培训，确保全员掌握分布式数据库运维的核心知识。

浪潮云在为某省级政务云平台实施分布式数据库方案时，仅用 3 天便完成了 100 节点的部署，单集群支持千万级别的并发处理能力。

5. 效果评估

通过上述实践，浪潮云显著提升了分布式数据库运维管理能力：

a) 稳定性增强：节点宕机恢复时间从 3 小时缩短至 10 分钟，全年数据库可用性达到 99.99%。

b) 性能提升：分布式数据库的吞吐能力相比传统架构提升了 5 倍，事务响应时间稳定在 50 毫秒以内。

c) 数据安全保障：通过完善的备份和恢复机制，成功避免了多次误操作导致的数据丢失。

d) 用户满意度提升：客户反馈系统运行更加流畅，业务高峰期无明显性能波动，极大提升了客户体验。