

企业核心交易数据库系统性能评估与优化实践

王 勇 任 毅 孔泽腾

国网数字科技控股有限公司 北京 100010

摘 要：在数字化转型的浪潮中，数据已然成为企业最为关键的资产之一。随着数据量的爆炸式增长以及业务复杂度的不断攀升，企业数据库正面临着前所未有的性能瓶颈。通过数据库性能评估，对数据库进行全方位的精准“体检”，帮助企业及时发现数据库中潜在的问题，进而为企业的业务决策提供有力的数据支持。本文基于某大型企业核心交易数据库性能评估与优化实践，总结提炼数据库性能评估方法、评估指标、性能优化策略、效果评估与持续改进方法。

关键词：数据库；性能评估；性能优化

一、性能评估方法

（一）性能监测工具

利用数据库监控工具定期检查数据库的运行状态，包括DB Time、活动会话数、I/O流量等。通过持续监控及时发现数据库系统性能瓶颈，避免性能问题积累。

（二）AWR报告分析

查询AWR快照获取数据库性能指标数据，AWR报告的生成时间重点覆盖核心业务的负载高峰或性能问题时间段。

（三）历史数据分析

基于数据库运行的历史数据进行分析，通过比较数据库不同时间段的性能表现，发现潜在的运行趋势或规律，有助于预测未来性能的变化趋势，提前采取优化措施。

二、性能评估指标

数据库性能评估指标包括6个层面：数据库负载、数据库架构、SQL文本、数据库参数配置、硬件资源状态、性能监控指标。

1.数据库负载：包括DB Time、活动会话数、I/O流

量、会话连接数、读写分离5个指标维度。

2.数据库架构：包括数据库部署架构、表结构设计、表空间管理、索引策略、存储方式管理5个指标维度。

3.SQL文本：指业务高峰时段执行频繁的SQL文本，包括合理使用索引、使用绑定变量、减少全表扫描、调整分区键列条件值类型等指标。

4.数据库参数配置：包括自适应游标共享特性、LGWR自适应工作方式、UNDO表空间自动调整功能等12项参数配置。

5.硬件资源状态：包括CPU、内存、I/O等待时间、TPS等4个指标维度。

6.性能监控：包括索引叶子层高、insert类型语句执行时长、高并发会话3个指标。

三、性能评估过程

（一）数据采集

通过数据库工具获取数据库运行状态，包括CPU、内存、网络等资源消耗、I/O等待、TPS等信息。

通过采集不同时段的AWR报告获取数据库负载、响应时间、性能指标等信息。

采集业务高峰时段的SQL语句，数据量较大的数据库表、索引，数据库参数设置等信息。

（二）数据预处理

数据库预处理是保障数据库性能优化工作顺利进行的重要环节，即在采集的原始数据的基础上进行数据审查和过滤，识别和删除冗余的、不完整或异常的数据记录，包括去除重复记录、识别缺失值并填补或排除等。目标是保证数据的准确性、完整性，形成高质量的初始

作者简介：

- 1.王勇（1982年-），男，汉族，北京市，硕士，高级，研究方向：人工智能与机器学习；
- 2.任毅（1984年-），男，汉族，北京市，硕士，高级，研究方向：数据科学与大数据技术；
- 3.孔泽腾（1990年-），男，汉族，河北省，本科，中级，研究方向：计算机科学与技术。

数据集。

(三) 数据分析

1. 数据库负载

从业务高峰时段的DB Time、活动会话数、I/O流量、会话连接数、读写分离5个指标维度进行分析,判断业务高峰时段的数据库负载是否处于可承载范围内。

(1) 会话连接数

核心数据库会话连接方式采用独享模式,设置允许连接的最大会话数为6000。业务高峰时段2个RAC节点的会话连接数稳定在2000以内,属于正常范围。

(2) 高并发读写

结合业务高峰时段的AWR报告和SQL文本,从DB Time、活动会话、I/O流量等指标进行分析,发现数据库存在高并发读写冲突问题。

核心数据库承载核心业务交易类数据写入,同时进行业务批量对账等数据读取操作。每日业务批量对账操作的活动会话处于150-200之间,产生了大量的数据读取操作,消耗了较多的CPU资源,业务高峰时段与交易类数据写入并行时,对数据库产生较大负载,容易导致并发会话数过高,进而产生索引分裂等问题,影响核心业务稳定运行。

2. 数据库架构

从数据库部署架构、表结构设计、表空间管理、索引策略、存储方式管理5个指标维度进行分析。

部署架构方面。采用Oracle双节点RAC集群部署,具备高可用性、高扩展性和负载均衡能力。

表结构设计方面。核心数据库针对数据量较大的表通过创建分区表的方式提升查询性能和可维护性。分区表共计95张,其中范围分区表92个,HASH分区表2个,列表分区表1个,分区类型符合规范。最大单表数据量达到2.3TB。

索引策略方面。核心数据库根据表的数据特点选择合适的索引类型,符合规范。目前数据量超过100GB的索引有9个。

存储方式管理方面。使用自动存储管理ASM进行统一管理,具备高可用性、数据冗余能力和资源动态调配能力。

3. SQL文本分析

选取业务高峰时段执行频繁的SQL文本,分析可能导致数据库执行效率降低的SQL语句,定位性能问题并提出优化建议措施。

4. 数据库参数

分析自适应游标共享特性、LGWR自适应工作方式、UNDO表空间自动调整功能等12项指标,判断是否存在影响数据库性能和稳定性的参数配置。

5. 硬件资源

主要从CPU、内存、I/O等待时间、TPS等4个指标进行分析。业务高峰时段的CPU利用率、内存使用率、TPS每秒事务数均处于正常水位,数据库的I/O等待时间峰值为10毫秒。

6. 性能监控

在现有监测指标的基础上,分析可能造成索引分裂、insert类型语句执行时间过长、高并发会话等问题的监测能力。

四、性能优化策略

(一) 数据库负载

1. 数据读写分离

将业务批量对账任务从目前的核心数据库迁移至运营管理库,实现读写分离,核心数据库、运营管理库分别承载交易类数据写入和运营类数据查询操作,从而降低业务批量对账任务对核心数据库的性能影响。

2. 引入缓存技术

针对高并发、重复性较高的业务运营场景,建议部署Redis缓存数据库与运营管理库(Oracle)结合使用,共同承载运营类数据查询任务。通过将查询结果缓存至Redis中,便于下次查询时直接从缓存中获取数据,以提升运营数据访问效率,降低对运营管理库的依赖。

为最大限度地发挥缓存技术优势,应当定期监控、动态调整缓存策略,包括设置合理的缓存大小限制、设定缓存数据过期时间等。

3. 数据库拆分

将核心数据库的非核心业务进行剥离拆分,实现核心业务的独立部署,降低业务相互影响。

(二) 数据库架构

1. 高水位治理

核心数据库的部分表已处于高水位。建议增加高水位治理频率,采取历史数据归档、无效数据清理、索引重建等措施收缩表空间,提升数据库性能。

2. 索引优化

平衡索引覆盖度。新建索引需要平衡索引的数量与存储成本,避免过多的索引或索引数据量过大增加写入类操作的资源消耗。

定期进行索引优化。建议结合业务使用需求，针对数据量较大的索引进行深入分析，定期开展索引碎片整理，按需进行索引合并，及时清除冗余索引等。

（三）SQL 文本分析

分析出8个影响核心数据库性能的SQL文本。建议通过SQL语法优化、创建适合索引、调整分区键列条件值类型、合理使用绑定变量、优化原有索引等方式，进一步提升SQL执行效率。

1. 合理构建SQL语句

仔细分析和优化SQL语句语法，避免不必要的复杂查询和嵌套，识别并消除不必要的子查询或循环引用，避免部分SQL语句占用过多的CPU资源。

2. 新建合适的索引

使用简洁的查询逻辑和适当的索引，根据查询条件和数据表结构，选择适当的索引类型，以确保其效率。

3. 避免全表扫描

分区键列CREATE_TIME传入的条件值为TIMESTAMP类型，发生隐式转换，导致需要扫描全部分区。建议调整对应的条件值为DATE类型。

4. 合理使用绑定变量

部分键列上的数据存在分布倾斜，应避免在该列的条件值上使用绑定变量。

（四）数据库参数配置

核心数据库12项参数配置的建议值如下：

1. 数据库参数optimizer_extended_cursor_sharing，参数含义为关闭自适应游标共享特性，建议值设置为NONE。

2. 数据库参数optimizer_adaptive_cursor_sharing，参数含义为关闭自适应游标共享特性，建议值设置为FALSE。

3. 数据库参数serial_direct_read，参数含义为避免产生direct path read等待事件，建议值设置为NEVER。

4. 数据库参数use_adaptive_log_file_sync，参数含义为关闭LGWR自适应工作方式，建议值设置为FALSE。

5. 数据库参数undo_autotune，参数含义为关闭UNDO表空间的自动调整功能，建议值设置为FALSE。

6. 数据库参数optimizer_null_aware_antijoin，参数含义为易引发bug，建议值设置为FALSE。

7. 数据库参数row_cr，参数含义为行级优化功能，建议值设置为FALSE。

8. 数据库参数deferred_segment_creation，参数含义为关闭延迟段创建特性，建议值设置为FALSE。

9. 数据库参数partition_large_extents，参数含义为避免初始占用空间过大，建议值设置为FALSE。

10. 数据库参数gc_undo_affinity，参数含义为关闭RAC的DRM（dynamic remastering）特性，建议值设置为FALSE。

11. 数据库参数enable_ddl_logging，参数含义为记录ddl操作日志，建议值设置为TRUE。

12. 数据库参数resource_limit，参数含义为是否允许资源控制，建议值设置为TRUE。

（五）性能监控

1. 索引叶子层高

建议监控索引的层高blevel是否大于4。索引层高超过阈值，表明索引分裂问题较为严重，需要进行及时处置，同时考虑开展数据库高水位治理等工作。

2. 语句执行时长

结合历史数据分析，核心数据库日常insert类型的语句执行时间一般不超过1秒。建议监控执行时长超过1秒的insert类型语句。

3. 活动会话数

结合历史数据分析，核心数据库日常的活动会话数位于200以内。建议设置活动会话数监控阈值为200，便于及时发现数据库高并发问题。

五、效果评估与持续改进

（一）效果评估

1. 高水位治理和索引优化

通过历史数据归档、无效数据清理、索引重建等方式，治理数据库表4张、分区8个、索引17个，表空间数据量下降48%，索引数据量下降53%。

2. 核心数据库读写分离

将业务批量对账任务迁移至运营管理库后，核心数据库的IO响应峰值下降29%，业务高峰时段的DB Time值（数据库实例处理所有用户请求所消耗的总时间，包括CPU时间和非空闲等待时间）下降34%。

3. SQL文本优化

完成8个执行频率最高的SQL文本优化。逻辑读方面，7个SQL基本下降到原来的0.1%，1个SQL下降到原来的10%。8个SQL各自的累计CPU使用占比下降81%~90%。8个SQL各自的单次执行时间下降90%~98%。

（二）持续改进

数据库性能评估与优化是一个持续的过程。建立持续改进机制，实现数据分析与性能优化的动态管理，是

保障数据库高效、稳定运行，形成闭环管理的重要环节。

1. 强化业务协同参与

数据库性能优化工作的最终目标是提升业务运行质效。做好性能分析评估的一个重要前提，是结合业务活动特性、业务数据、用户行为、研发实现逻辑等开展分析，以更加准确地定位性能问题。性能优化措施的实施，需要结合业务发展需求（如高水位治理），涉及业务运行逻辑、业务行为调整（如读写分离），离不开研发代码优化（如SQL优化）。需要加强运行单位、研发单位、业务单位的协同协作。

2. 完善性能监控系统

完善数据库实时性能监控工具，用于跟踪关键性能指标的变化确实，依据业务活动特性设置合适的预警阈值，自动触发预警通知，便于尽快开展数据库性能分析。

3. 建立定期分析机制

除了技术层面的评估，业务反馈、用户体验是衡量

优化效果的重要指标之一。需要持续关注业务发展和用户需求变化，确保优化措施能够持续有效地提升数据库运行质效。

参考文献

- [1] 吴华锋. 面向多维数据分析的数据库存储与查询优化[J]. 信息记录材料, 2025, 26(06): 158-160. DOI: 10.16009/j.cnki.cn13-1295/tq.2025.06.044.
- [2] 尚思圆. 大数据下数据库优化器算法的性能优化研究[J]. 石油化工设计, 2025, 42(02): 39-41+46+75.
- [3] 刘康. 面向云计算的数据库优化策略与性能分析[J]. 软件, 2024, 45(08): 132-134.
- [4] 荣垂田, 田浩辉, 杜方. 多目标深度强化学习驱动的数据库系统参数优化技术[J/OL]. 软件学报, 2025, (12): 5512-5536[2025-11-12]. <https://doi.org/10.13328/j.cnki.jos.007405>.