

基于 SQL 优化的数据库审核系统在生产中的应用

韦峻峰, 王 骞

中国联合网络通信有限公司河南省分公司 河南

【摘要】随着公司业务发展,业务需求变化和数据量变化呈现爆炸式增长,随业务多样性和数据量变化,给数据库性能带来前所未有的压力,因此数据库性能带来的故障越来越受到关注。生产系统变更涉及到性能优化需要修改设计、修改代码、耗时较长,解决问题牵涉的人员较多、流程偏于复杂、时效性较差,给系统问题解决带来一定的影响,如何在系统设计、开发和测试阶段,及时发现问题,解决问题,简化流程。为此,开发基于 SQL 优化的数据库审核系统,有效地协助开发人员、DBA 审核发现问题,快速精准地定位 SQL 的问题点和快速解决 SQL 问题,在保证高质量的上线 SQL 代码,SQL 质量管控和预测 SQL 性能方面,极大的简化 DBA 的工作和提高运维效率。有效避免性能问题产生故障。

【关键词】SQL; 数据库; 审核

Application of database audit system based on SQL optimization in production

Junfeng Wei, Qian Wang

China United Telecommunications Co. Ltd. Henan Branch, Henan

【Abstract】With the development of a company's services, service requirements and data volume change rapidly increase. As service diversity and data volume change, database performance is under unprecedented pressure. Therefore, database performance faults are more and more concerned. Production system change involves the performance optimization need to modify the design, modify the code, takes longer, solve the problem involving more people and process more complicated, poor timeliness, brings the certain effect to the system problem solving, how to in system design, development and testing phase, timely find problems and solve the problem, simplify the process. To this end, the development of database audit system based on SQL optimization, effectively assist developers and DBA audit to find problems, quickly and accurately locate SQL problems and quickly solve SQL problems, in the assurance of high quality online SQL code, SQL quality control and prediction of SQL performance. Greatly simplifies DBA's work and improves operation and maintenance efficiency. Effectively avoid performance problems.

【Keywords】SQL; Database; Audit

1 背景

目前我公司内部数据库规模较大,但在运维方面更多还较为依靠人工方式进行,随着业务逻辑的变化越来越复杂,在数据库 SQL 性能优化、巡检及日常运维方面对人员技术水平的依赖程度越来越高,如何降低运维工作对人员技能水平的依赖。如何为业务开发人员赋能,将数据库 SQL 开发规范以

规则方式进行配置实现在线审核,减少不规范 SQL 上线风险;如何使业务及开发人员可以实时掌握生产数据库系统所存在的性能问题 SQL,提供 SQL 优化建议,这些都成为了摆在我们面前的问题。

2 概述

因此,针对人工审核工作量大、规范落地和监督约束无保证、性能隐患易带入生产环境、审查结

果无法可视化等多种弊端提出了集 SQL 审核、优化、性能管理分析及日常运维能力为一体化的、专业化的、基于 SQL 优化为核心的数据库审核管理方案。该方案可以结合现有的数据库运维平台系统，基于以下两点开展。

2.1 SQL 优化审核

我们将 SQL 优化审核功能分为三个部分：SQL 上线前审核，SQL 运行中审核，SQL 结束后审核。

上线前审核：

包括 SQL 规则审核：平台可支持基于 SQL 开发规范配置 SQL 审核规则，建立相应规则库，可针对 SQL 语法进行规则性审核；敏感语句审核管理：平台可支持对敏感类语句进行分类审核管理，如删表、删库等操作语句。

运行中审核：

包括针对 DQL SQL 告警及优化：平台可支持对 DQL 类 SQL 进行告警配置和优化，在发现相关 DQL 类语句占用性能高时，可实时进行告警，同时平台提供相应 SQL 的优化建议；同时平台可支持对 DML、DDL 变更类 SQL 进行告警和影响度分析，在发现相应 SQL 运行时，及时感知风险。

结束后审核：

包括历史 SQL 查询：通过选择特定时间，可以查询该时间段内运行 SQL 情况，并形成相应报表。

2.2 性能分析

我们将平台与数据库运维管理平台系统结合使用，可支持对数据库的 TOP SQL、TOP 事件、TOP 会议进行实时在线分析，展现内容包含访问 SQL 内容、SQL ID、访问时长、访问次数等信息，让运维人员更聚焦于 TOP 问题。历史性能分析：平台与数据库运维管理平台系统结合使用，可支持对数据库历史性能 SQL 进行分析，并同时提供数据库对象信息、参数信息以及相关 SQL 的执行计划、建表语句、索引等信息，提升运维可视化分析能力。

3 系统架构

整个方案平台采用 B/S 架构,支持无代理部署方式，通过 JDBC 收集被监控端数据库的性能数据，再经过后台统计、规则审核，性能分析等处理，通过前端 WEB 界面对数据进行分析与展示，目前支持主流的 ORACLE、MYSQL、SQLServer 数据库。

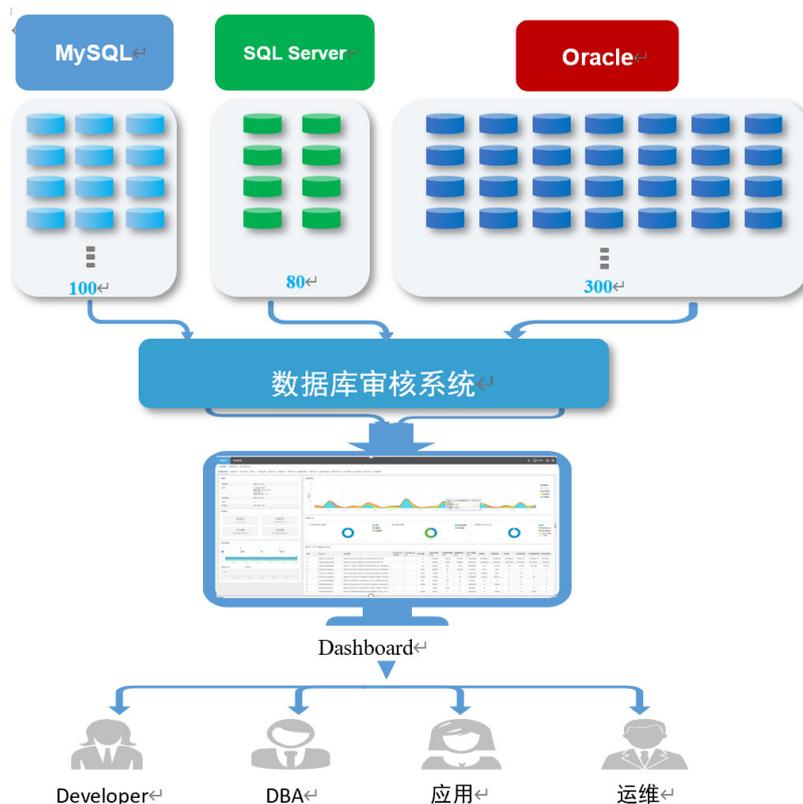


图 1 架构图

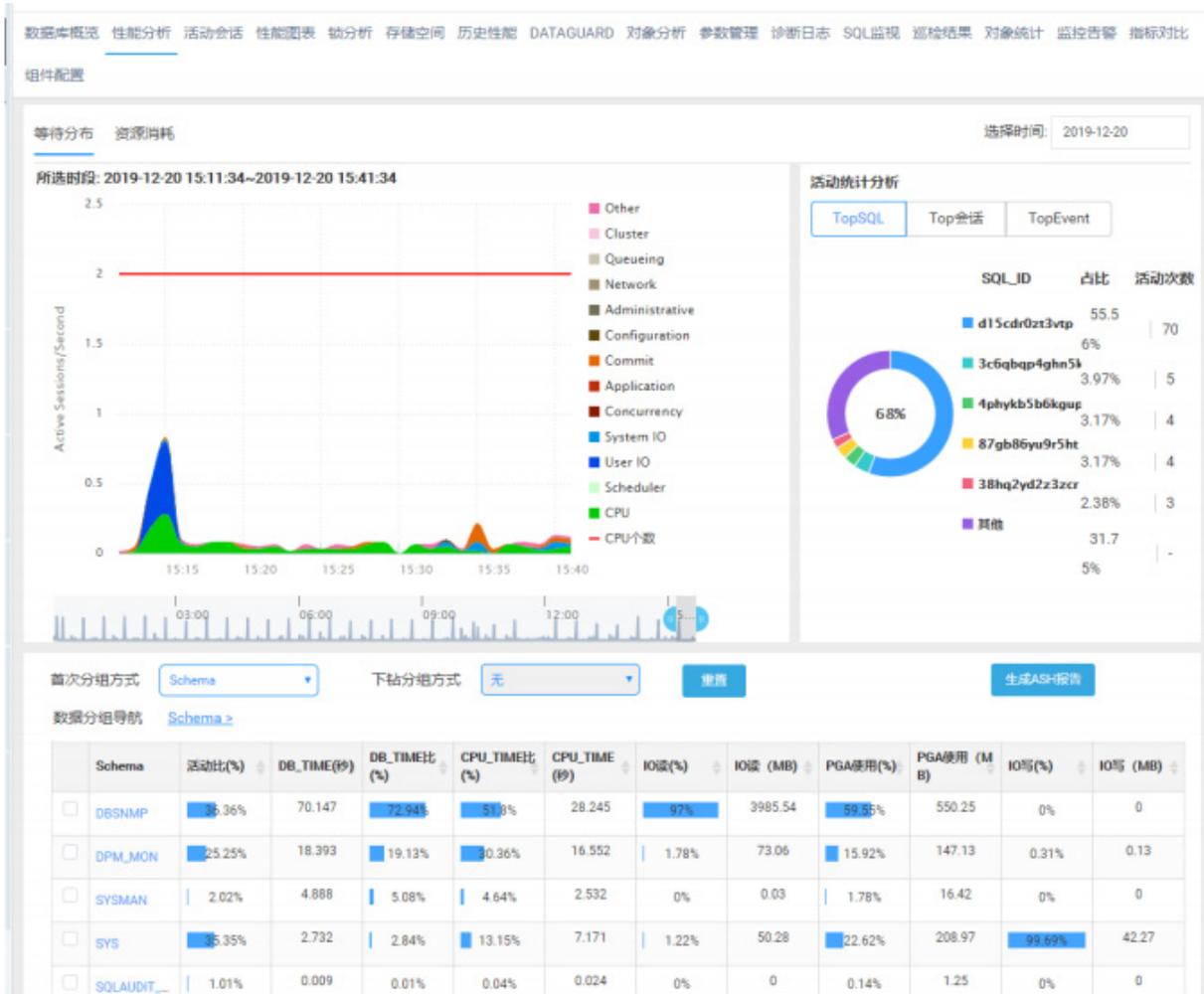


图2 性能分析面板

根据现场需求, 选择适合的基于 SQL 优化的数据库审核系统产品架构. 基于 SQL 优化的数据库审核系统总体来说可以分为两种架构:

3.1 单节点部署的架构, 即可以把所有的基于 SQL 优化的数据库审核系统相关组件都安装部署在同一个服务器上;

3.2 多个节点部署的架构, 自然就是说可以将所有的基于 SQL 优化的数据库审核系统相关组件分散部署在多个不同的服务器主机上。

对比以上两种不同的产品架构, 很显然可以得出以下结论: 选择单节点部署基于 SQL 优化的数据库审核系统可以实现快速部署、节省相关资源, 而且可以减少后续维护成本, 但此种架构的不足之处也显而易见, 那就是对单台服务器的配置要求较高, 且容灾能力相对较差一些; 后面一种基于 SQL 优化

的数据库审核系统架构总体原理是一致的, 分散部署基于 SQL 优化的数据库审核系统各个组件到不同的服务器资源上, 尤其是将 MYSQL 数据库单独部署, 可以提高系统的整体稳定性, 但是虽然对单个服务器资源的要求相对较低, 但整体资源需求比单节点更高一些, 且增加了后续的维护成本。结合公司实际情况和功能复杂情况, 最终在公司内部资源池上进行单节点部署。

4 方案能力

4.1 性能分析

展示和分析数据库总体性能负载情况, 通过详细统计分析 ASH 数据信息, 从等待分布和资源消耗维度综合展现数据库性能负载情况; 活动统计分析 & 资源消耗统计分析支持通过 TopSQL、Top会话、TopEvent 的数据汇总统计展示; TopSQL、

Top 会话、TopEvent 支持下钻到详细数据进行深度分析;支持生成 ASH 报告。展示对应的 SQL 详情模块,支持提供 SQL 基本信息、SQL 语句详情、执行计划、对象与绑定变量和优化建议功能模块 tab 页面。

对性能数据进行分组分析:模块跟随繁忙度 AAS 趋势图的时间段选择进行变化统计。首次分组方式可选: Schema、Module、Program、Machine、SQL,默认首次分组方式为 Schema;下钻分组方式:从下方列表选择一行数据,解锁下钻分组方式下拉框,此时的下拉框支持选择除了首次分组方式可选已选的选择项,其他均可选择。如首次分组方式为 Schema,则下钻分组方式可选 Module、Program、Machine、SQL。

4.2 性能图表

用来展示数据库主机和 db 的资源总体消耗情况,从总体上了解和判断当前数据库是否繁忙,支持历史趋势判断性能总体走势。可以查看数据库指定时间段内的数据库和数据库主机性能图表数据。默认展示数据库活动的性能图表、包括 DB 等待情况、执行情况、逻辑 IO 情况、物理 IO 情况等图表的曲线信息。主机的 CPU、内存、磁盘和网络等图表的曲线信息。

4.3 历史性能分析

具备 AWR 快照在线分析功能,在线对历史 AWR 数据进行分析。支持用户资源消耗趋势走向,有效的发现用户性能突变情况。

可以将数据库按多用户、多租户维度分析对应用户或租户的物理读、逻辑读、CPU 耗时、执行次数等性能消耗分析。默认显示所有用户的 TOP SQL 的 buffer gets、Executions、Disk reads、Cpu time、Elapsed time、IOWait time、CLwait time 等指标图表信息。

4.4 SQL 监视能力

通过监控数据库 SQL 语句,获取 SQL 语句更多信息,可进一步查看 SQL 的执行计划 等信息,支持 html 格式查看 SQL 详细信息和时间插件回溯查看历史数据。

4.5 巡检管理

通过定义巡检指标项,将巡指标组合成模板并形成巡检任务、运行巡检任务形成巡检报告,通过

报告发现数据库潜在问题和风险。

在新增页面填写任务名称、选择数据库实例、数据库实例只能按照数据库类型分类选择,不可同时选择不同的数据库类型的数据库实例。选择巡检项,可通过巡检项名称、使用数据库进行巡检项的筛序。

5 应用效果

我们通过建设基于 SQL 优化的数据库审核系统,可以实现 SQL 规则审核、SQL 性能分析等核心能力,实现性能问题的运维前移解决,持续提升数据库性能稳定性。

降低 SQL 开发难度,规范 SQL 合规

对于业务及开发人员而言,SQL 审核、SQL 优化、SQL 性能可视化等功能大幅降低了相关技术壁垒,业务开发人员可以更加便利的对 SQL 开发规范进行执行,及时发现业务开发 SQL 过程中存在的风险和问题。

提升运维水平,降低高危操作风险

对运维人员而言,SQL 优化工作平台化大幅降低了相关运维技术能力对人员水平的依赖,相关工作效率得到提升;性能可视化能力也极大程度上降低了运维人员操作生产环境可能存在的误操作风险。

实现 SQL 审核的质量跟踪和运维能力前移

数据库巡检能力有效解决了运维人员重复性工作问题,可以将运维关注核心由数据库系统转向业务整体 SLA 质量,SQL 质量的后续跟踪在逐步推行的过程取得了一定效果,对于业务侧来说,他们通过 SQL 审核熟悉了规范,同时也提高了 SQL 开发经验,实现运维能力前移。

6 结语

基于 SQL 优化的数据库审核系统的使用,有效降低了 SQL 开发难度、更加规范了 SQL 合规性,大幅提升了运维水平、降低了高危操作风险,保证了生产业务系统的高性能和高可用。

参考文献

- [1] [美]理查德·尼米克(Richard Niemic) 著;董志平 刘永甫 吕学勇 译 Oracle Database 12cR2 性能调整与优化(第5版)。北京 清华出版社,2019:195-231。
- [2] 鸟哥.鸟哥的 Linux 私房菜:服务器架设篇(第3版).北京:

机械工业出版社, 2018: 30-172。

- [3] [美]Bob Bryla(OCP) 著 明道洋 译。Oracle Database 12c DBA 官方手册 (第 8 版)。北京 清华大学出版社, 2016:243-246。
- [4] [美]理查德·尼米克(Richard Niemic) 著; 董志平 刘永甫 吕学勇 译 Oracle Database 12cR2 性能调整与优化 (第 5 版)。北京 清华大学出版社, 2019: 641-706。
- [5] [美]Bob Bryla(OCP) 著 明道洋 译。Oracle Database 12c DBA 官方手册 (第 8 版)。北京 清华大学出版社, 2016:243-246。
- [6] [美]理查德·尼米克(Richard Niemic) 著; 董志平 刘永甫 吕学勇 译 Oracle Database 12cR2 性能调整与优

化 (第 5 版)。北京 清华大学出版社, 2019: 641-706

收稿日期: 2022 年 3 月 18 日

出刊日期: 2022 年 6 月 30 日

引用本文: 韦峻峰, 王骞, 基于 SQL 优化的数据库审核系统在生产中的应用[J]. 国际计算机科学进展, 2022, 2(1): 37-41.

DOI: 10.12208/j. aics.20220010

检索信息: RCCSE 权威核心学术期刊数据库、中国知网 (CNKI Scholar)、万方数据 (WANFANG DATA)、Google Scholar 等数据库收录期刊

版权声明: ©2022 作者与开放获取期刊研究中心 (OAJRC) 所有。本文章按照知识共享署名许可条款发表。<http://creativecommons.org/licenses/by/4.0/>



OPEN ACCESS