

时序数据库在航天型号试验数据管理中的应用

曹雪君,姚 娜,王天秀,童 心,范景怡

北京航天自动控制研究所,北京 100854



摘 要 通过分析航天领域现有型号试验数据的特点,发现传统关系型数据库已不能满足日益激增的海量试验数据的存储与管理需求。针对现有试验数据量级大、写入吞吐低、访问效率差等特点,提出了一种基于时序数据库的试验数据存储管理方法。通过在传统关系型数据库与时序数据库中对比型号试验数据的读写性能进行对比分析,发现基于时序数据库的数据写入与数据查询性能均明显优于关系型数据库,且具有快速存储、快速读取功能。将该方法应用于试验数据管理系统和数据处理显示与分析系统,能够满足系统对试验数据处理的实时写入、快速读取等需求。

关键词 时序数据库;试验数据;航天

中图分类号: TP392

文献标识码: A

文章编号: 1006-3242(2024)01-0051-07

Application of Time Series Database in the Management of Aerospace Project Test Data

CAO Xuejun, YAO Na, WANG Tianxiu, TONG Xin, FAN Jingyi

Beijing Aerospace Automatic Control Institute, Beijing 100854, China

Abstract The characteristics of existing project test data in the aerospace field are analyzed and then the results show that the traditional relational database can no longer meet the storage and management demands of ever-increasing massive test data. Aiming at the characteristics of large magnitude, low write throughput and poor access efficiency of existing test data, a test data storage management method based on time series database is proposed. By comparing and analyzing the reading and writing performance of aerospace project test data in traditional relational database and time series database, it is verified that the performance of data writing and data query based on time series database is obviously better than that of relational database, and it has the capabilities of fast saving and fast accessing. This method is applied to test data management system and data processing display and analysis system, which can meet the requirements of real-time writing and fast reading of test data processing.

Key words Time series database; Test data; Aerospace

收稿日期:2023-07-27

作者简介:曹雪君(1988-),女,硕士,主要从事系统软件设计工作,E-mail:lfexjexj@126.com;姚 娜(1982-),女,硕士,主要从事仿真模型建模、设计与实现工作;王天秀(1983-),女,硕士,主要从事系统测试软件开发设计工作;童 心(1985-),男,硕士,主要从事系统软件设计工作;范景怡(1995-),女,硕士,主要从事数据处理软件设计工作。

0 引 言

近几年,航天事业取得了突飞猛进的发展,航天装备型号经过数次迭代,功能更加完善,性能更加可靠。随着型号种类的增加、硬件设备的迭代更新、型号对系统功能要求的与日俱增及对性能要求的愈加严格,控制系统在设计方面的复杂度也随之不断提升,从而导致试验数据量级的爆发式增长。对数据库的存储与访问带来了极大挑战。小型航天装备型号进行一次 2 h 左右的模拟综合试验,仅 1553B 一种数据类型,就能产生 600MB ~ 800MB 的原始二进制文件数据,将二进制数据解析成具有可读性的规格化数据后,可达到 3GB 甚至更大,在使用数据库对数据进行存储时,数据条数的存储规模可达到千万级,达到了传统关系型数据库的瓶颈。近些年随着新一代高速传输协议 Glink 的快速发展和应用,一次综合试验产生的二进制原始数据可达到几十 GB 的量级,解析后甚至能够达到 TB 量级,传统关系型数据库已无法满足型号试验数据存储的需求,需要寻找一种新的数据存储方案来满足现阶段海量试验数据的存储需求。

航天领域试验数据是在试验过程中对各种设备中的数据进行实时采集得到的,是随着时间序列产生的。在对试验数据进行处理时,侧重于对数据进行插入和查询操作,几乎不涉及对数据的修改。近年来时序数据库快速发展^[1],其独有的特点恰好满足具有上述特征的海量试验数据的存储需求。

1 相关研究

随着各行各业对海量数据存储需求的日益激增,时序数据库^[2]成为近几年的热点,并在电力^[3-5]、海上监测^[6]、铁路^[7]及地震监测^[8]等多个领域得到广泛应用。此外,文献[9]设计了基于 InfluxDB 时序数据库的工业大数据存储管理系统,提高了对工业数据可视化的效率。文献[10]将时序数据库用于自动化运维技术,克服了传统自动化运维系统管理效率低、运算过程复杂等问题,提高了信息查询效率。

2 时序数据库概述

2.1 时序数据

时序数据是指按照时间序列产生的数据,主要

特点是每个数据有一个具体的时间标签,即时间戳。按照时间顺序排列数据,即可对数据进行趋势分析、异常点分析等。

航天领域试验中产生的数据即是标准的时序数据。这些数据在试验过程中随着时间不断产生并被采集,每个数据有唯一的时间戳。单独的数据点一般没有较多意义,只有将数据按照时间序列排序,对数据进行整体的趋势研究与分析,探究其规律,才能挖掘数据间的价值,分析出数据的正常与异常。

此外,时序数据在产生时即具有确定的值,几乎不涉及修改操作。如某一时刻产生的温度值,即代表产生时刻的温度值,下一时刻变化后的温度值会有新的时间戳。图 1 表示主机板温度随采样时间的变化规律示例,主机板温度即为时序数据。

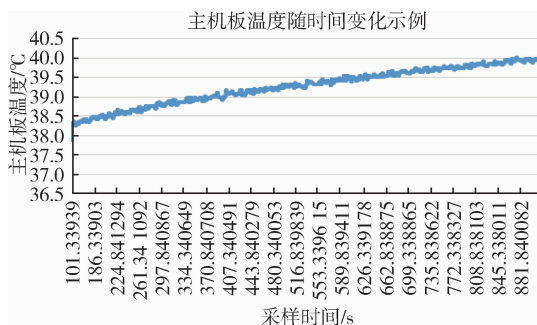


图 1 主机板温度时序数据示例

2.2 时序数据库与关系型数据库区别

时序数据库是对时序数据进行存储的数据库,它与传统的关系型数据库有较大区别。关系型数据库主要用于描述数据之间的关系,利用行式结构进行存储;时序数据库主要用于对单个参数按照时间序列进行存储,采用列式存储结构,描述单个数据参数与时间戳之间的关系,不同数据参数之间无明显的关联关系。

关系型数据库侧重对数据的增、删、改和查操作,主要维护数据之间的关联关系;时序数据库由于时序数据的特点,主要涉及数据的插入与查询操作,几乎不涉及数据的修改操作。关系型数据库在进行数据插入和查询时,经常涉及多个表之间的关联关系,因此插入和查询效率有限。当数据量级较高时,随着数据量的增大,数据插入和查询效率会呈指数倍的降低。时序数据库的列式存储解决了关系型数据库的上述瓶颈,在数据插入和查询时有较强的性能优势。

3 基于时序数据库的存储设计方案

3.1 试验数据层次结构分析

为了建立时序数据库表结构,首先分析试验数据结构。试验数据有如下结构特点:

每个型号包含若干个试验项(如等效器测试、单项检查测试等)。每个试验项包含多个测试项(如等效器分系统测试、MF1 和 MF2 等)。每个测试项下有不同的帧类型(如平台数据及 xx 制导遥测数据等)。每个帧类型中有不同的参数(如电流、电压、马赫数和飞行高度等)。试验数据具体层次结构关系可参考图 2,其中试验项和测试项表示试验的流程信息。

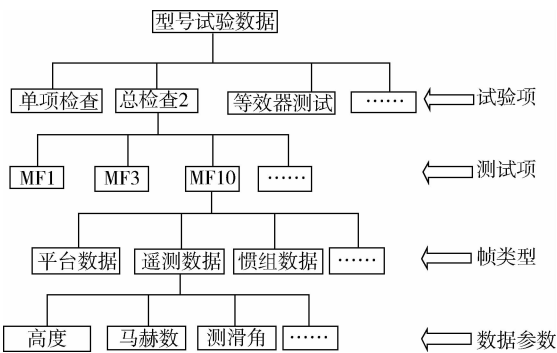


图 2 试验数据层次结构图

3.2 时序数据库概念模型

通过对试验数据层次结构特点的分析,建立时序数据库概念结构模型,并以某个具体参数(如 25A 电流)为例,介绍时序数据库的相关概念。

时间戳	25A 电流	试验项	测试项	帧类型
x1	24.89	试验1	测试1	消息1
x2	25.02	试验1	测试1	消息1
x3	25.06	试验1	测试2	消息2
x4	24.98	试验2	测试3	消息1
x5	24.94	试验2	测试4	消息2
x6	25.05	试验2	测试4	消息2

数据点

时间戳 字段值 标签

图 3 时序数据库结构图

图 3 中,每行表示 1 个数据点信息,每个数据点由时间戳、字段值和标签列字段组成。第 1 列为时间戳,即数据点产生的时刻。第 2 列为字段值,即数据点在某一时刻(即时间戳)下的具体数值,该数据随着时间戳变化。对于每个数据点而言,其字段值对应的时间戳是唯一的,即图中每一行的时间戳字

段是唯一的。后 3 列“试验项、测试项、帧类型”为标签列,标签通常不随时间戳变化,用于表征数据点的属性。每种标签列中的属性数量一般不会无限增长,通常在试验前即可确定。例如,对于某个具体型号而言,该型号共有若干个试验项,每个试验项下面有数个测试项,每个测试项中又包含多个帧类型,每个帧类型中有若干种参数。因此,每个数据点在某个具体时间戳下产生的字段值,只会属于某个试验项下面的某个测试项中的某个帧类型,即用“试验项、测试项、帧类型”作为标签表示某一时刻数据点的属性信息。

3.3 时序数据库物理建模

根据上述试验数据的特点及概念模型,试验数据分为单参数采集量时序数据(如电流值)以及流程指令数据(如试验项开始、测试项开始),在进行数据建模时,对上述两类数据分别建立不同的物理模型。对于流程指令数据,单独建立一张 events 表,该表中的数据量较少,用于存储每次试验过程中的流程信息。

对于数据量随着时间递增的数据参数(采集量),每个数据参数建立一张子表,子表中的数据量随着试验过程不断增加。由于采集量与试验项、测试项和帧类型相关,因此子表命名规则为试验项 + 测试项 + 帧类型 + 采集量,以保证每个子表名称的唯一性。子表是在试验过程中当需要时动态建立的,建立后再写入业务数据。

由于每类数据参数维护一张子表,导致子表的数量庞大,难以管理,而且经常会用到表之间的关系做聚合操作,使数据库操作变得复杂。为了解决此问题,时序数据库引入了超级表(Super Table,简称为 STable)的概念。

超级表是指某一特定类型的数据参数的集合。同一类型的数据参数,其表结构一致,但每个表的静态属性(标签)是不一样的。因此描述一个超级表,除需要定义数据参数的表结构之外,还需要定义其标签的模式。对于试验数据而言,创建一个超级表 metrics,用来管理所有与时间相关的试验参数数据。

此外,将二进制试验数据解析成具有可读性的结构化数据时,需要有一套解析规则,即数据解析协议。数据解析协议与时间无关,它描述了如何将二进制字节流转换成基础数据类型。因此还需建立一张超级表 protocols,用于存储数据解析协议。

根据上述分析,共有 3 个超级表,具体含义见表 1。

表 1 超级表组成	
表名	表含义说明
events	流程指令表,存放与试验流程相关的数据(试验项、测试项)
metrics	物理量表,存放试验中所有物理量的数据值
protocols	协议表,存放数据解析协议

表 1 中,protocols 协议表用于将原始二进制数据解析成规格化的数据;events 流程指令表用于存储与流程相关的数据;metrics 物理量表存储了试验过程中所有的采集参数数据。由于 metrics 物理量表中的数据较多,本文对数据库性能进行对比分析时即采用该表中的数据。metrics 物理量表的表结构见表 2。

表 2 metrics 超级表——物理量表				
序号	列名	类型	含义	备注说明
1	ts	timestamp	时间戳:采集时间	ms 精度
2	ts_origin	timestamp	原始时间	ms 精度
3	bit_value	bool	布尔类型数据	
4	ut_value	tinyint unsigned	无符号字符类型数据	每个物理量在所有 value 字段中仅填一个
5	
6	exp_id	varchar(14)	试验项 id	标签列
7	id	varchar(36)	参数 id	标签列
8	frame	int	帧类型 id	标签列
9	metric_id	int	物理量 id	标签列
10	frame_name	varchar(100)	帧名称	标签列
11	value_type	tinyint	参数值类型	标签列

4 实验对比分析

4.1 试验数据来源

型号在试验过程中,会产生多种类型的试验数据,如 1553B 的 MT 和 RT 数据、网络数据及 KGIO 数据等。有的数据为二进制类型,有的数据为可读的文本类型。虽然试验数据类型不同,但每种试验数据均依据一定的解析规则进行解析,解析过程并无本质区别。

在对传统关系型数据库和时序数据库进行对比分析时,选择了一种类型经典且内容较多的数据进行对比实验,即 1553B_MT 类型数据,原始二进制数据文件大小为 342MB。解析后的数据共有 3000 多万条数据点,分布在 100 多个不同的帧类型中。

4.2 关系型数据库 MySQL 入库性能分析

在引入时序数据库前,通常使用关系型数据库 MySQL 存储与管理试验数据。针对上节 342MB 的原始二进制 1553B 数据,解析后的数据条数为 3000 多万条,本节统计分析了该数据的解析与入库耗时。在数据入库时,为了优化入库性能,采用了批

量入库操作,每 10 万条数据批量入库一次。

由于原始二进制数据在解析后才能入库,因此首先统计数据解析过程的耗时,以便对后续数据入库过程的耗时进行分析。此外,本节还对数据解析与解析后的文本数据本地磁盘存储(简称存文本)进行了对比实验。纯解析耗时、解析 + MySQL 入库耗时和解析 + 存文本耗时见表 3。

表 3 数据解析、入库与存文本耗时对比			
数据量级/万	解析耗时/s	解析 + 入库耗时/s	解析 + 存文本耗时/s
3041.9707	422.438	894.206	573.370

通过表 3 中的数据可知,解析 + 入库耗时减去纯解析耗时,为 471.768 s,约为数据入库耗时;解析 + 存文本耗时减去纯解析耗时,为 150.932 s,约为存文本耗时。可以看出,MySQL 数据入库耗时约为存文本耗时的 3 倍。

对数据入库与数据存文本耗时进一步对比分析,分别统计不同数据条数下纯解析耗时、解析 + MySQL 入库耗时及解析 + 存文本耗时,其耗时对比数据见图 4。

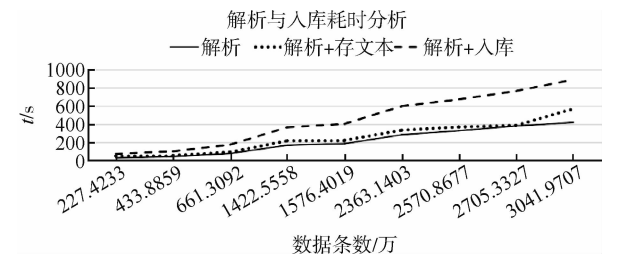


图 4 解析与入库耗时对比分析

通过观察图 4,解析 + 存文本耗时与纯数据解析耗时相差不大,其差值即为数据存文本所耗用的时间,属于数据存盘的正常耗用时间。而数据解析 + 入库的耗时时间明显比纯数据解析时间长得多,并且与数据解析 + 存文本所耗用的时间差也较大,说明数据入库过程本身占用了较长的时间,从而表明关系型数据库 MySQL 的数据入库性能较差。

4.3 数据库性能对比

以下对关系型数据库和时序数据库的入库性能进行对比分析。关系型数据库采用 MySQL 数据库,时序数据库采用 TDEngine 数据库。TDEngine 数据库是一款开源的时序数据库,支持开源版应用与商业版应用。

4.3.1 数据入库性能对比

由于原始数据是二进制格式,需要先将非结构化的二进制形式的数据解析成结构化的标准数据,才能进行入库操作。因此,对数据库入库性能进行对比分析时所采用的耗时时间,不是单纯的数据库入库时间,而是包含了数据解析的时间。数据解析采用了相同的解析算法,且由上节可知数据解析对数据入库性能影响较小,因此总时间差异可以反映数据入库时间的差异。

图 5 为不同数据量下入库操作的时间对比数据。当数据量较小时,数据入库所耗用的时间差别不大。当数据量达到千万级别以上时,MySQL 数据库的入库时间明显增大,是时序数据库入库时间的两倍以上。随着数据量级的不断增大,MySQL 数据库与时序数据库入库所耗用的时间差也在不断增大。

4.3.2 数量总量查询性能对比

在进行数据查询分析时,由于数据量较大,若显示所有数据查询结果,屏幕输出查询结果会占用较长的时间,且该时间远大于查询语句本身占用的时间,导致该时间无法作为数据查询本身占用的时间进行对比分析。因此,在对比分析数据查询性能时,采用查询不同数据量下的数据总条数所消耗的

时间,作为数据库查询的性能指标进行对比分析。由于不同试验项中包含的数据总数不同,因此可通过查询不同试验项中所有数据点总个数所耗用的时间,对数据库的查询性能进行对比分析。

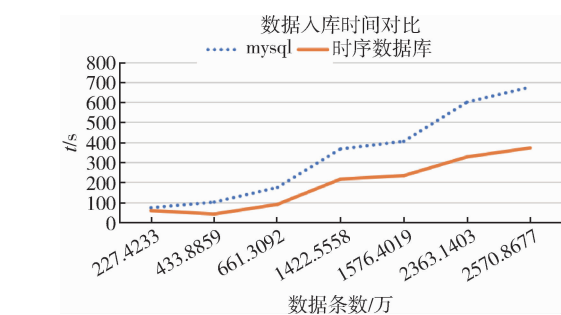


图 5 不同数据量下入库时间对比

以数据库查询语句 `select count(*) from table` 为例分析数据查询耗时性能,其中 `table` 表示表名。在 `sql` 语句中加入 `where` 子句,通过 `where` 子句筛选试验项,从而对不同试验项中的数据总量进行查询。

表 4 为不同数据量下查询数据总量的时间对比,可以看出,MySQL 数据库的数据查询时间随着数据量的增长呈明显递增趋势,而时序数据库的数据查询时间无明显变化。当数据量级为千万级别时,时序数据库在查询数据总量时用不到 0.1 s。因此,与传统关系型数据库 MySQL 相比,时序数据库在数据查询方面有着明显的性能优势。

表 4 不同数据量下查询数据总量耗时		
数据量级/万	MySQL 数据库查询耗时/s	时序数据库查询耗时/s
227.4233	1.36	0.05387
433.8859	2.76	0.055851
661.3092	3.70	0.055893
1422.5558	7.24	0.059740
1576.4019	7.92	0.062832
2363.1403	12.32	0.063568
2570.8677	14.36	0.077691

4.3.3 不同帧类型中数据总量查询性能对比

上节的数据查询性能只统计了不同试验下(即不同数据量级下)所有数据参数总量的查询时间对比,并未区分数据类型。而在实际应用中,经常会根据数据类型(即帧类型)对数据进行统计分析。此节选择某个试验项中的数据,对该试验项中不同帧类型下的数据参数总量进行查询耗时分析。

在关系型数据库 MySQL 中,查询不同帧类型的

数据需要对两张表(帧消息表、参数数据表)进行关联查询,耗用时间较长。在某次试验项中数据总条数为 1222914 的数据库中,查询某种帧类型下的数据总量(总量为 1207360),总共耗时 2406.89 s,而时序数据库查询用时不到 1 s。可见,传统关系型数据库并不适用于该情形下的查询操作,因此不再统计其它数据量级下关系型数据库的数据查询耗时,只分析时序数据库在不同帧类型下的数据总量查询耗时,具体见表 5。

表 5 时序数据库在不同帧类型下的数据查询耗时		
序号	查询后数据条数	查询耗时/s
1	17	0.020945
2	42384	0.02194
3	54900	0.021942
4	89023	0.021942
5	129520	0.022939
6	4283524	0.021938
7	7734540	0.026928

表 5 的试验中,选择的试验项中共有 25700481 条数据,分别统计了 7 组不同帧类型下的数据总量查询耗时。第 2 列“查询后数据条数”表示该帧类型下的数据总条数,第 3 列“查询耗时/s”表示查询该帧类型下数据总量所耗用的时间。通过表 5 可以看出,尽管不同帧类型中包含的数据条数差别很大(小到 17 条,大到 700 多万条),但查询时间却相差不大,且几乎是瞬间得到查询结果。这是由于在时序数据库中,试验名称和帧类型均作为数据的标签,所以数据查询时间与数据量级并无明显关系。

5 基于时序数据库的应用实践

随着航天领域型号试验数据量级的爆发式增长,时序数据库在航天领域有着广阔的应用前景,并在型号应用中逐步取代传统关系型数据库。目前,时序数据库已经在下列应用实践中进行了初步探索,取得了较好的效果。

5.1 试验数据管理系统

型号中产生的试验数据类型繁多、数量庞大,为了统一管理型号中的试验数据,某些项目应用试验数据管理系统对所有型号产生的试验数据进行统一管理。试验数据管理系统通过区分型号与用户角色,对用户与数据进行权限管理。通过数据匹

配模板,规格化型号中的非结构化数据,在底层存储时采用时序数据库对规格化后的海量试验数据进行存储管理。

试验数据管理系统能够查询分析试验数据,支持文本导出、曲线绘图、曲线对比分析和数据过滤等功能。由于单次试验产生的试验数据量级可达到亿级,在查询数据时,传统关系型数据库已无法支撑该数据量级的存储与查询,时序数据库应运而生,很好地解决了试验数据存储与管理中数据存取方面遇到的性能瓶颈问题。

5.2 数据处理显示与分析系统

型号在试验中产生的试验数据,部分较重要数据需要在试验过程中通过显控软件实时显示,以便试验人员实时观察试验状况;其余数据在试验后通过关联分析、曲线分析和数据回放等方式进行判读即可。因此,在处理与分析试验数据时,需满足实时与事后两方面的处理能力。

数据处理显示与分析系统采用时序数据库存储并管理数据可满足上述要求。试验数据产生时,首先被规格化并解析,然后存入时序数据库。需要实时展示的数据,通过发布订阅机制,在界面服务对数据进行订阅,每当有符合条件的数据入库后,后台的数据存储服务及时将数据推送至界面服务,由界面服务显示处理。试验结束后,用户可回放认为存在异常的数据,回放时数据存储服务从数据库中读取数据并推送至界面服务。此外,试验数据还支持导出结构化文本、导出分析报告等功能。时序数据库的单参数列式存储,使数据存储与查询性能得到了显著提升,可满足海量试验数据显示与分析的处理需求。

6 结 论

通过在传统关系型数据库和时序数据库上分别对航天型号试验数据进行数据入库及数据查询的性能对比分析,验证了时序数据库在数据入库和数据查询方面的性能均明显优于关系型数据库。在试验数据管理系统和数据处理显示与分析系统中采用时序数据库作为存储层,使时序数据库在航天领域得到了工程实践应用。

参 考 文 献

[1] 杨婕. 时序数据库发展研究[J]. 广东通信技术, 2020,40(3):46-48 + 57. (YANG Jie. Research on the

- development of time series database [J]. Guangdong Communication Technology, 2020, 40(3):46-48+57.)
- [2] 郑博, 王煜彤, 燕钰, 等. 时间序列数据库管理: 技术、系统与展望[J]. 工业技术创新, 2022, 9(4):12-21. (ZHENG Bo, WANG Yutong, YAN Yu, et al. Management of time series database: technology, system and prospect [J]. Industrial Technology Innovation, 2022, 9(4):12-21.)
- [3] 冯仟, 龚鸣, 邓宏涛. 基于电力系统的时序数据库分析与研究[J]. 长江信息通信, 2021(9):140-143. (FENG Qian, GONG Ming, DENG Hongtao. Analysis and research of time series database based on power system [J]. Changjiang Information & Communications, 2021(9):140-143.)
- [4] 李立雪, 赵君力, 魏清新. 基于时序数据库的数字孪生模型动态数据建模[C]// 中国航空工业技术装备工程协会年会, 西安, 2022年11月25日. (LI Lixue, ZHAO Junli, WEI Qingxin. Dynamic data modeling of digital twin model based on time series database [C]// China Aviation Industries Technology Equipment Engineering Association, Xi'an, November 25, 2022.)
- [5] 王风硕, 曾丽. 基于时序数据库的智慧用电应用研究[J]. 电脑知识与技术, 2021, 17(30):48-50. (WANG Fengshuo, ZENG Li. Research on smart electricity application based on time series data [J]. Computer Knowledge and Technology, 2021, 17(30):48-50.)
- [6] 郑宝锋, 刘修国, 崔芳姿, 等. 基于时序数据库的CGSsafety系统设计与实现[J]. 地理空间信息, 2021, 19(11):585-589. (ZHENG Baofeng, LIU Xiguoguo, CUI Fangzi, et al. Design and implementation of CGSsafety system based on time series database [J]. Geospatial Information, 2021, 19(11):585-589.)
- [7] 李畅. 基于时序数据库的信号集中监测数据存储和分析[J]. 铁路通信信号工程技术. 2020, 17(12):95-99. (LI Chang. Data storage and analysis of railway signal centralized monitoring system based on time-series database [J]. Railway Signalling & Communication Engineering, 2020, 17(12):95-99.)
- [8] 陈通, 韩雪君, 马延路. 时序数据库在海量地震波形数据分布式存储与处理中的应用初探[J]. 中国地震, 2022, 38(4):799-809. (CHEN Tong, HAN Xuejun, MA Yanlu. Preliminary application of time series database in distributed storage and processing of massive seismic waveform data [J]. Earthquake Research in China, 2022, 38(4):799-809.)
- [9] 王红涛, 王志超, 陈峰, 等. 基于时序数据库的工业大数据应用研究[J]. 重型机械, 2020, (4):17-21. (WANG Hongtao, WANG Zhichao, CHEN Feng, et al. Research on industrial big data application based on time series database [J]. Heavy Machinery, 2020, (4):17-21.)
- [10] 李明, 张靖, 吴尚, 等. 基于时序数据库的自动化运维技术研究[J]. 信息技术, 2021(6):102-107. (LI Ming, ZHANG Jing, WU Shang, et al. Research on automatic operation and maintenance technology based on time-series database [J]. Information Technology, 2021(6):102-107.)