



MySQL8.0统计信息总结

原创 闫建(Rock Yan) 云和恩墨技术服务团队 2025-03-17

755



概念描述

在MySQL8.0中，统计信息（Statistics）是优化器（Optimizer）用来生成执行计划的重要依据，它直接影响SQL性能。

统计信息管理

MySQL提供了两种统计信息的管理方式：非持久化统计信息（Non-Persistent Statistics）和持久化统计信息（Persistent Statistics）。这两种方式在存储、更新机制以及对执行计划的影响上有所不同。

1. 非持久性优化器统计信息

非持久优化器化统计信息（non-persistent optimizer statistics）是指MySQL InnoDB 存储引擎的统计信息仅存储在内存中，而不会持久化到磁盘。当 MySQL 服务重启时，这些统计信息会丢失，并在下次访问表时重新计算。非持久化统计信息的行为由参数 innodb_stats_persistent 控制，当该参数设置为 OFF 时，统计信息即为非持久化的，缺省情况下MySQL的统计信息是持久化的（innodb_stats_persistent=ON）。

非持久优化器统计信息通常会在以下几种情况下触发更新：

1) 手动执行 **analyze table** 命令。

2) 在innodb_stats_on_metadata=ON 的情况下，执行**show table status,show index status**或者查询**information_schem**库下的 **tables**表和**statistics**表。

说明：默认情况下innodb_stats_on_metadata是关闭的，开启innodb_stats_on_metadata会降低具有大量表或者索引的库的访问速度，并减少查询语句执行计划的稳定性。

3) **MySQL**客户端连接时启用自动补全功能 **--auto-rehash** （默认启用）

说明：禁用它（**--no-auto-rehash**）可以加快连接速度，减少内存占用，但需要手动输入完整的 **S**QL 语句

4) 首次打开表时。

5) 自上次统计信息更新后，**innodb**检测到表有**1/16**的数据被修改时。

innodb_stats_transient_sample_pages参数

以上几种情况下会触发非持久优化器统计信息的自动更新，对于非持久优化器统计信息还有一个参数来控制 **innodb_stats_transient_sample_pages**，统计信息数据更新机制是基于innodb表的索引页的数据量来估算的，默认情况下这个参数就是用于控制 innodb表的统计信息采样页面数量，默认为8个



Rock Yan

关注

45

文章

127

粉丝

107K+

浏览量



获得了 220 次点赞



内容获得 61 次评论



获得了 368 次收藏

TA的专栏



关于MySQL的那些事儿

收录 39 篇内容

热门文章

升级OpenSSL：CVE-2016-2183漏洞处理解决方案

2024-04-15

10469浏览

MySQL参数优化系列之- join_buffer_size

2023-05-16

8428浏览

MySQL8.0的 UNDO 表空间管理

2020-03-08

5319浏览

采用keepalived（VIP）作为MySQL主从高可用架构时的一些建议

2023-04-30

5202浏览

MySQL DBA 日常运维常用命令总结

2024-04-30

4753浏览

在线实训环境入口



MySQL在线实训环境

查看详情 >>

最新文章

MySQL8.0直方图功能简介

2025-03-21

411浏览

MySQL8.0分区表之范围分区

2025-01-24

270浏览

MySQL8.0新特性-通用表达式WITH

2024-12-13

354浏览

MySQL未提交事务导致的TRUNCATE表阻塞挂起问题处理

2024-12-13

732浏览

联合主键表导致MySQL Shell逻辑备份异常（备份时间超长影响正常业务！）

2024-05-07

711浏览

页。增大 innodb_stats_transient_sample_pages 的值会提高统计信息的准确性，但会增加计算开销，减小该值会降低统计信息的准确性，但也会减少计算开销。

在大多数情况下，默认值 8 是一个合理的平衡点，既能提供足够的统计信息准确性，又不会带来过多的性能开销。如果发现查询优化器选择了不理想的执行计划，可以尝试逐步增大 innodb_stats_transient_sample_pages 的值，观察查询性能是否改善。如果查询性能要求极高，且表数据量较大，可以适当减小该值以减少开销。

2. 持久性优化器统计信息

在MySQL5.6版本之前，InnoDB表的统计信息是动态计算的（即“非持久化统计信息”），这些统计信息不会持久化到磁盘，而是在每次需要时进行计算。这种方式虽然灵活，但是有一定的缺点：

- 统计信息可能会频繁变化，导致查询优化器选择的执行计划不太稳定。
- 动态计算统计信息会增加查询的开销，尤其是在表数据量较大时。

为了解决以上问题，从MySQL5.6版本开始引入了持久化统计信息功能，将统计信息持久化存储到磁盘，并在表数据发生重大变化时自动更新。

持久化统计信息主要由参数innodb_stats_persistent决定，该值默认为ON（启用持久化统计信息）。

3. 持久化统计信息存储在哪里？

- mysql.innodb_table_stats表
该表存储了innodb表的统计信息，包括表的行数，数据页数量等：

```
mysql >desc mysql.innodb_table_stats;
+-----+-----+-----+-----+-----+-----+
| Field          | Type          | Null | Key | Default          | Extra |
+-----+-----+-----+-----+-----+-----+
| database_name  | varchar(64)   | NO   | PRI | NULL             |       |
| table_name     | varchar(199)  | NO   | PRI | NULL             |       |
| last_update    | timestamp     | NO   |     | CURRENT_TIMESTAMP | DEFAULT_GENERATED |
| n_rows         | bigint unsigned | NO   |     | NULL             |       |
| clustered_index_size | bigint unsigned | NO   |     | NULL             |       |
| sum_of_other_index_sizes | bigint unsigned | NO   |     | NULL             |       |
+-----+-----+-----+-----+-----+-----+
6 rows in set (0.00 sec)

-- 主要字段说明：
database_name  ：数据库名
table_name     ：表名
last_update    ：最近更新时间
n_rows         ：表的行数
clustered_index_size ：聚集索引的大小，单位为页pages
sum_of_other_index_sizes ：其他索引的总大小，单位为页pages
```

- mysql.innodb_index_stats表
该表存储了InnoDB表索引的统计信息，包括索引的基数、页数等

目录

- 概念描述
- 统计信息管理
 - 1. 非持久性优化器统计信息
 - 2. 持久性优化器统计信息
 - 3. 持久化统计信息存储在哪里？
 - 4. 持久化统计信息的准确性由谁来决...
 - 5. 统计信息的准确性如何受影响？
 - 6. 如何提高统计信息的准确性？
- 相关参数
- 总结
- 参考文档

- 2. 持久性优化器统计信息
- 3. 持久化统计信息存储在哪里？


```
mysql > desc mysql.innodb_index_stats;

+-----+-----+-----+-----+-----+-----+
| Field          | Type          | Null | Key | Default          | Extra          |
+-----+-----+-----+-----+-----+-----+
| database_name  | varchar(64)   | NO   | PRI | NULL             |               |
| table_name     | varchar(199)  | NO   | PRI | NULL             |               |
| index_name     | varchar(64)   | NO   | PRI | NULL             |               |
| last_update    | timestamp     | NO   |     | CURRENT_TIMESTAMP | DEFAULT_GENERATED c
| stat_name      | varchar(64)   | NO   | PRI | NULL             |               |
| stat_value     | bigint unsigned | NO   |     | NULL             |               |
| sample_size    | bigint unsigned | YES  |     | NULL             |               |
| stat_description | varchar(1024) | NO   |     | NULL             |               |
+-----+-----+-----+-----+-----+-----+

8 rows in set (0.00 sec)

-- 主要字段说明：
database_name  ：数据库名
table_name     ：表名
index_name     ：索引名
last_update    ：最近更新时间
stat_name      ：统计信息的名称（如 n_diff_pfx01，n_leaf_pages，size 等）。
stat_value     ：统计信息值
sample_size    ：样本大小
stat_description ：统计信息描述
```

```
示例：查询large_table表索引的统计信息
mysql >select * from mysql.innodb_index_stats where table_name='large_table';

+-----+-----+-----+-----+-----+-----+
| database_name | table_name  | index_name  | last_update    | stat_name  | stat_v
+-----+-----+-----+-----+-----+-----+
| testdb       | large_table | PRIMARY    | 2025-03-14 14:53:46 | n_diff_pfx01 | 49
| testdb       | large_table | PRIMARY    | 2025-03-14 14:53:46 | n_leaf_pages | 
| testdb       | large_table | PRIMARY    | 2025-03-14 14:53:46 | size        | 
| testdb       | large_table | idx_lt_field1 | 2025-03-14 14:53:46 | n_diff_pfx01 | 9
| testdb       | large_table | idx_lt_field1 | 2025-03-14 14:53:46 | n_diff_pfx02 | 50
| testdb       | large_table | idx_lt_field1 | 2025-03-14 14:53:46 | n_leaf_pages | 
| testdb       | large_table | idx_lt_field1 | 2025-03-14 14:53:46 | size        | 
+-----+-----+-----+-----+-----+-----+

7 rows in set (0.00 sec)

--说明 stat_name 列中常见的统计信息名称及其含义：
1. n_diff_pfx01 表示索引第一列的不同值数量
2. n_diff_pfx02 表示索引前两列的不同值数量:如果索引是多列索引（例如（col1, col2）），n_diff_pfx02 表示
3. n_leaf_pages 表示索引的叶子页数量。叶子页是实际存储索引数据的页。
4. size 表示索引的总页数，包括叶子页和非叶子页。
```

4. 持久化统计信息的准确性由谁来决定？

在MySQL8.0中，持久化统计信息的准确性由采样数据和统计信息计算方式决定。MySQL 通过分析表的索引和数据分布来生成统计信息，这些统计信息直接影响查询优化器的决策。

MySQL 通过以下方式决定统计信息的准确性：

（1）采样数据：MySQL 使用 **innodb_stats_persistent_sample_pages** 参数控制采样页数。默认值为20，表示从表中随机采样20个数据页来计算统计信息，采样页数越多，统计信息越准确，但计算开销也越大。

（2）统计信息计算方式：基数估算（通过分析索引中的不同值Cardinality来估算查询的选择性）和直方图（MySQL 8.0 引入了直方图统计信息，用于更精确地估算数据分布）。

（3）自动重新计算：**innodb_stats_auto_recalc**参数默认启用控制此行为 -> 如果表中超过 10% 的数据发生了变化，那么MySQL将会自动重新计算统计信息。

（4）手动更新：可以使用analyze table命令手动更新统计信息

```
analyze table table_name;
```

5. 统计信息的准确性如何受影响？

持久性优化器统计信息的准确性可能受到以下因素的影响：

(1) 采样页数不足

如果 innodb_stats_persistent_sample_pages 设置过小，采样数据可能不足以准确反映表的实际数据分布，导致统计信息不准确。

- 4. 持久化统计信息的准确性由谁来决...
- 5. 统计信息的准确性如何受影响？
- 6. 如何提高统计信息的准确性？
- 7. 相关参数
- 总结
- 参考文档

(2) 数据分布不均匀

如果表中的数据分布不均匀（例如某些值出现频率极高），统计信息可能无法准确反映查询的选择性。

(3) 索引结构变化

如果表的索引结构发生变化（例如添加或删除索引），统计信息可能过时，导致查询优化器选择次优的执行计划。

(4) 表数据变化

如果表的数据发生大量变化（例如插入、更新或删除大量数据），统计信息可能过时，需要重新计算。

(5) 直方图统计信息未启用

如果未启用直方图统计信息，MySQL 可能无法准确估算复杂查询的选择性。

6. 如何提高统计信息的准确性？

(1) 增加采样页数

```
-- 增加 innodb_stats_persistent_sample_pages 的值，例如：  
SET GLOBAL innodb_stats_persistent_sample_pages = 50;
```

(2) 启用直方图统计信息

```
-- 使用 ANALYZE TABLE 命令生成直方图统计信息，例如：  
ANALYZE TABLE table_name UPDATE HISTOGRAM ON column_name;
```

(3) 定期更新统计信息：定期执行 ANALYZE TABLE 命令，确保统计信息是最新的。

(4) 优化表的碎片整理

```
--表碎片可能导致统计信息不准确，定期对表进行优化，以减少碎片  
OPTIMIZE TABLE example_table;  
  
或者：  
ALTER TABLE my_table ENGINE=InnoDB;
```

说明：两者都会导致表被锁定，因此在生产环境中使用时需要谨慎，尤其是在大表上执行这些操作时。

(5) 优化索引：确保表的索引设计合理，避免冗余或无效的索引。

7. 相关参数

以下是与持久性优化器统计信息相关的重要参数：

- innodb_stats_persistent: 是否启用持久性统计信息（默认 ON）。
- innodb_stats_auto_recalc: 是否自动重新计算统计信息（默认 ON）。
- innodb_stats_persistent_sample_pages: 采样页数（默认 20）。
- innodb_stats_method: 统计信息计算方法（如 nulls_equal、nulls_unequal 等）。

总结

持久性优化器统计信息的准确性由很多因素都有关系包括采样数据、统计信息计算方式，表数据分布等。为了提高准确性，可以增加采样页数、启用直方图统计信息、定期更新统计信息，并优化索引设计。通过合理配置和监控，可以确保查询优化器选择最优的执行计划。

参考文档

https://dev.mysql.com/doc/refman/8.0/en/innodb-persistent-stats.html
https://dev.mysql.com/doc/refman/8.0/en/innodb-statistics-estimation.html
https://dev.mysql.com/doc/refman/8.0/en/innodb-performance-optimizer-statistics.html

🔗 墨力计划 mysql 统计信息

「喜欢这篇文章，您的关注和赞赏是给作者最好的鼓励」

关注作者

赞赏

【版权声明】本文为墨天轮用户原创内容，转载时必须标注文章的来源（墨天轮），文章链接，文章作者等基本信息，否则作者和墨天轮有权追究责任。如果您发现墨天轮中有涉嫌抄袭或者侵权的内容，欢迎发送邮件至：contact@modb.pro进行举报，并提供相关证据，一经查实，墨天轮将立刻删除相关内容。

文章被以下合辑收录



关于MySQL的那些事儿（共39篇）

记录MySQL数据库的点点滴滴

收藏合辑

评论

分享你的看法，一起交流吧~



淡定

LV4

在MySQL8.0中，统计信息（Statistics）是优化器（Optimizer）用来生成执行计划的重要依据，它直接影响SQL性能。

3月前

点赞

评论

相关阅读

📖 新书首发！《MySQL 8.0 实用手册》限量赠送

墨天轮福利君

654次阅读

2025-07-09 16:41:05

MySQL锁定位实践指南

Digital Observer

503次阅读

2025-06-29 19:27:55

MySQL 8.0 性能优化实战：性能提升的全方位调优方案

shunwah

M

466次阅读

2025-06-27 15:15:45

MySQL 升级到8.0这个参数变化一定要清楚

蔡璐

307次阅读

2025-07-01 13:14:17

mysql 1z0-909每日一题8

山丘smith

285次阅读

2025-06-27 11:07:37

MySQL 排序优化指南

Cui Hulong

279次阅读

2025-07-07 14:14:14

6月“墨力原创作者计划”获奖名单公布！

墨天轮编辑部

269次阅读

2025-07-09 15:22:35

ACDU周度精选 | 本周数据库圈热点 + 技术干货分享（2025/7/11期）

墨天轮小助手

266次阅读

2025-07-11 15:14:25

mysql 1z0-909每日一题4

山丘smith

258次阅读

2025-06-23 14:52:36

MySQL数据库主从同步中断问题分析:Column x of table cjc.t1 cannot be converted from type xx to type xx

陈举超

238次阅读

2025-07-05 10:08:52