2022 年 OceanBase 数据库大赛

决赛赛题

OceanBase 数据库大赛组委会

2022 年 11 月 9 日

一、 赛题背景说明

OceanBase 是蚂蚁集团完全自研的一款分布式数据库，在 TPC-C 和 TPC-H 上分别刷新世界纪录，支撑阿里和蚂蚁 9 年双 11 业务。随着 2021 年 6 月 1 号正式对外开源，越来越多的金融客户、电信运营商、政府合作单位、互联网企业开始使用 OceanBase。OceanBase 也开始介入越来越多的场景，比如随着业务不断增长，数据量也在不断增长，原来的集中式 Oracle、MySQL、DB2 开始支撑不了业务的快速增长，寻找能支撑业务稳定发展并能弹性扩容的系统，因此 OceanBase 开始不断升级换代老的 Oracle、MySQL 与 DB2。

在升级换代 Oracle、MySQL 与 DB2 的过程中，需要解决的第一件事情，就是将数据无缝从历史系统中，导入到 OceanBase 当中。很多时候，导入操作会在一个有限的运维窗口中才能进行，如果一个 10TB 的数据，消耗 8 个小时，则可能会横跨一个运维窗口(通常凌晨 2 点到早上 9 点)，会冲击正常的业务压力，因此，导入速度会至关重要，站在用户的角度，期望 10TB 这样 OceanBase 社区版的基础上，能在 1 个小时内完成。

二、 比赛题目：旁路导入

当前 OceanBase 导入数据的方案是将文本文件转换成 batch insert 语句执行插入，执行路径长，需要经过 SQL、事务、转储等。在很多传统的数据库当中，提供 “direct path load” 方式，这种方式是一种可以“走捷径”的方案，跳过 SQL 与事务，直接将数据存储在 SSTable 中，因此称这种方案为“旁路导入”，这种方式相对过去 batch insert 方式，更底层、更直接，性能上也会有量级提升，是我们解决导入问题最佳方式之一。

目前 OceanBase 还未包含旁路导入的功能，选手们可以参考我们提供的 demo 工程实现指定场景的旁路导入功能，同时优化导入的性能。优化链路包含解析 CSV 文件、转换为 OceanBase 内部数据结构、写入 SSTable 存储等。

三、 参赛收获

旁路导入是 OceanBase 业务中，一个真实场景，也是一个极富挑战的场景，上手简单(从单一核心模块即可入手)，但发挥空间巨大(也可以涉及数据库的各个模块)，参赛选手可以学习到许多关于数据库以及工程方面的知识。具体但不限于包括如下收获：

- 可以体会成绩在一点一点持续提高，获得解决真实场景的成就感
- 接触真实工业界数据库的存储引擎，并学习和接触数据库最核心的模块之一 ——存储引擎
    - 学习 LSM Tree 的原理。OceanBase 存储底层采用 LSM Tree 结构，同学们在做此题目时需要将数据直接写为 SSTable（LSM Tree 中的一个核心数据结构）文件，会对 LSM Tree 有更加深刻的认识；
    - 平衡查找树原理。OceanBase 存储底层结构使用 LSM Tree，而其中的每个 SSTable 使用多阶层的平衡多叉树作为排序结构，而没有采用 skip-list 结构。因此同学们会对平衡查找的原理也会有更深入的理解，同时也可以帮助理解磁盘数据访问的优化原理；

- 接触真实高并发高性能系统框架，系统性学习底层编程技巧和规范

    - 多线程与任务调度。为了能够充分的利用机器的资源，让性能成绩提高，必然需要充分发挥计算机多核的优势，采用多线程的方式，提高任务的并发处理能力。另外，多线程处理中，任务调度的效率，也会直接影响到整体的执行效率；
    - 文件读取优化。文件读写是许多工程项目中需要用到的技能，而如何高效的读取数据充分利用磁盘带宽，也会影响到程序的执行效率，这里也涉及到了零拷贝技术的应用，同时让同学们对文件系统读写模块底层理解的更加透彻；
    - 文件排序。在数据库系统中，由于磁盘比内存大很多，在外存中对数据进行排序，也是常见的场景。这里在导入数据时，也需要考虑数据放在外存中排序，如何做的更加的高效，是选手们努力的目标之一。
    - 性能调优手段，学习如何使用 CPU 调优、内存调优。

OceanBase 希望大家以学习数据库技术为目的参与比赛，所以决赛设置为帮助选手成长为主要目标，竞技为第二目标，也欢迎大家成为 OceanBase 开源大家庭中的一员，OceanBase 社区组织架构可以参考 社区组织成为 Committer 的第一步是成为 Contributor，可以在 issues 列表 中找到一个容易上手的 issue，然后提交关联的 pull request，操作过程请参考 Contribute to OceanBase。成为 Contributor 之后，对 OceanBase 的整个开发流程已经有了基本了解，也可以帮助自己做决赛试题。通过决赛中学习到的信息,进一步找一些更复杂的 BUG 或功能点,提交 pull request,成为Contributor。

四、 赛题后台测试流程说明

测试分为评分测试和正确性验证测试。通过正确性测试的程序才会有有效成绩。数据要求落盘，并重启进行验证。

1、评分测试

在单机 OceanBase 集群中创建一个空表，执行 load data 命令将 CSV 文件导入到此表中。导入数据速度越快，分数越高。导入数据的量，会比内存大很多。另外，CSV 文件中的数据是无序的，会被打乱，在旁路导入功能中需要对数据进行排序。

关于表结构和索引的测试说明，请查看如下详情：

```
create table lineitem_bulk (
    l_orderkey BIGINT NOT NULL,
    l_partkey BIGINT NOT NULL,
    l_suppkey INTEGER NOT NULL,
    l_linenumber INTEGER NOT NULL,
    l_quantity DECIMAL(15,2) NOT NULL,
    l_extendedprice DECIMAL(15,2) NOT NULL,
    l_discount DECIMAL(15,2) NOT NULL,
    l_tax DECIMAL(15,2) NOT NULL,
    l_returnflag char(1) DEFAULT NULL,
    l_linestatus char(1) DEFAULT NULL,
```

```
    l_shipdate date NOT NULL,
    l_commitdate date DEFAULT NULL,
    l_receiptdate date DEFAULT NULL,
    l_shipinstruct char(25) DEFAULT NULL,
    l_shipmode char(10) DEFAULT NULL,
    l_comment varchar(44) DEFAULT NULL,
    primary key(l_orderkey, l_linenumber))
```

以下是一个 CSV 测试示例文件，供大家参考。注意：此数据仅为示例文件，真实的测试数据，并没有按照 primary key (l_orderkey, l_linenumber)排序。

```
5999972|133109|8136|2|44|50252.40|0.08|0.00|N|O|1996-05-24|1996-07-22|1996-05-
27|COLLECT COD|RAIL| the furiously express pearls. furi|
5999972|152761|2762|3|3|5441.28|0.04|0.01|N|O|1996-08-31|1996-06-02|1996-09-
22|DELIVER IN PERSON|MAIL|sual accounts al|
5999973|176345|1380|1|50|71067.00|0.04|0.01|N|O|1997-07-27|1997-09-07|1997-08-
10|TAKE BACK RETURN|FOB|gular excuses. |
5999974|25360|5361|1|24|30848.64|0.02|0.03|R|F|1993-08-15|1993-10-07|1993-09-
01|COLLECT COD|MAIL|express dependencies. express, pendi|
5999974|10463|5466|2|46|63179.16|0.08|0.06|R|F|1993-09-16|1993-09-21|1993-10-
02|COLLECT COD|RAIL|dolites wake|
5999975|7272|2273|1|32|37736.64|0.07|0.01|R|F|1993-10-07|1993-09-30|1993-10-
21|COLLECT COD|REG AIR|tructions. excu|
5999975|6452|1453|2|7|9509.15|0.04|0.00|A|F|1993-11-02|1993-09-23|1993-11-
19|DELIVER IN PERSON|SHIP|lar pinto beans aft|
```

2、正确性测试

在数据导入完成后，执行合并动作，完成后重启节点，再使用表对比方式进行验证数据正确性，确保数据一致。选手可以通过简单进行 select count(primary_key) from lineitem_bulk, 做一个快速简单的验证。

3、测试环境说明

单机环境部署单节点 OceanBase 集群，机器规格 8C 16G。测试脚本与 OceanBase 集群在同一台机器上。

4、后台测试流程如下所示：

1) 选手提交测试，包含代码分支、commit id 等信息
2) 后台程序拉取代码
3) 编译
4) 创建测试表
5) 加载数据。超时时间 30 分钟。
6) 计算导入速度
7) major merge: 将数据合并到基线。超时时间 30 秒。
8) 重启
9) 正确性验证

5、如何提交测试

与初赛相似，选手填写 git url、branch 和 commit id 信息。

6、约束条件

为了大家的精力更加集中在业务场景的代码优化中，同时为了提高测试效率，再次做一些约束与约定。

- 比赛基于分支 2022_competition (https://github.com/oceanbase/oceanbase/tree/2022_competition)；
- 不允许修改影响程序运行效率的编译选项或指令集；
- 不允许调整 build.sh、cmake/Env.cmake 文件以及 deps 目录下所有内容；
- 存储引擎底层存储结构仍使用原有 LSM Tree，可以优化，但不能另写存储引擎；
- 不允许改变 SQL 执行路径。比如不通过存储引擎，直接返回数据；
- 比赛结束后会拉取代码查重，代码重复度较高将会取消成绩；
- 比赛后需要保持代码对测试后台人员可见，如果大赛工作人员在评审截止时间之前无法拉取代码，将会取消成绩；
- 参赛代码仓库需要设置为私有（private），因为开放代码导致与其它参赛队伍代码雷同将会取消成绩。

五、 决赛成绩排名规则

测试分为评分测试和正确性验证测试。通过正确性测试的程序才会有有效成绩。数据要求落盘，并重启进行验证。

1、决赛评分规则

基于决赛成绩的计算公式，所得数据即为比赛分数。分数越高，导入速度越快，则排名越高，最终筛选出前 12 支队伍进入最终的夺冠之夜。特别说明，如出现分数相同的情况，选择提交成绩时间最早的团队获胜。

决赛成绩计算公式：比赛分数 = 数据量 ÷ 导入时间

公式说明：导入时间具体是指导入之前与之后记录时间戳，统计此时间差，作为导入

时间。

2、公平竞赛规则

（1） 抄袭行为：凡未能保证原创性的竞赛行为均视为抄袭行为，将取消成绩。例如：

- 直接引用他人代码；
- 私自与其他队伍或非本队伍成员进行互相抄袭的；

（2） 不正当竞争行为： 凡未遵从竞赛宗旨，恶意获取高分的行为均被视为不正当竞争行为，将取消成绩。例如：

- 可以通过除竞赛规定途径之外的其他途径接触到竞赛相关数据的人员参加竞赛的；
- 邀约参赛团队名单之外的人员参与解题与方案设计，或以外包、求助等形式在参赛团队之外完成赛题的；
- 参赛者以任何形式使用竞赛提供数据之外的任何数据参赛的（大赛主页明确规定可以使用的除外）；
- 人工标注部分或所有测试集、标注或刻意跳过测试用例、正确性测试与性能测试代码逻辑明显不同，并作为结果进行提交的；
- 使用竞赛规定外的计算资源的（大赛主页未做任何规定的除外）；
- 在同一个比赛中，使用多个账号参赛的；
- 利用平台或规则漏洞进行参赛的。

（3）蓄意破坏行为：凡通过恶意手段对比赛平台、评估系统和环境进行破坏的均视为蓄意破坏行为，将取消成绩。例如：

- 蓄意上传携带病毒文件的；
- 蓄意发起对比赛平台、评估系统的攻击，扰乱比赛秩序的。

六、 附录及参考资料

1、 关于编译加速

OceanBase 代码量大，编译一次花费很久时间，这里使用 ccache 缓存加速编译，因此对 OceanBase 的 cmake/Env.cmake 做了改造。另外，deps/3rd 目录在编译初始化时会下载依赖的 rpm 包，并且解压，这里将初始化完成的环境做备份，后续每次编译时，将备份的 3rd 目录移动到需要编译的目录中以加速编译。

2、数据导入

导入数据量大约 40G，数据使用 TPC-H 工具生成，然后执行打乱，如果参考旁路导入 demo 程序的话，数据在导入之前需要进行排序。

数据导入操作步骤如下：

```
SET GLOBAL secure_file_priv = "";
-- 执行换成后，退出客户端，再重新连接

-- 设置导入超时时间。时间单位是“微秒”
set global ob_query_timeout=36000000000;

-- 导入数据。需要调整文件路径
load data infile "load_data_10.csv" into table lineitem_bulk fields terminated by
"|";
```

如果不做任何优化，测试数据需要约 1.1 小时导入完成，是超出当前给定的导入超时时间的。

3、如何查找 issue

如果现有的 github 上找不到合适的上手 issue，可以自行创建新的 issue 来改善 OceanBase 现状。比如优化 OceanBase 的编译、安装部署、代码质量、typo 错误、中文注释修改为英文注释、给代码加注释以帮助更多的人理解代码工作原理。如果没有思路，还可以联系后台工作人员，一起找到一些比较简单的 issue。

4、源码部署

可以使用 OceanBase 源码中自带的 obd.sh 工具部署，目录在源码中的 tools/deploy/obd.sh ，使用方法可以执行 obd.sh -h。

这里会简单介绍如何使用另一种源码来部署一个单节点集群，同时也是测试后台使用的部署方法。

前提:提前安装 obd,参考 obd github 首页(https://github.com/oceanbase/obdeploy)。

1) 下载源码
2) 编译。执行 sh build.sh release --init --make
3) 编译完成可以找到二进制文件 build_release/src/observer/observer
4) 制作 obd 镜像。在 build_release 目录下执行：

```
# 将编译好的东西安装到当前目录某个位置
# 这里会报错，不过可以忽略错误
make install DESTDIR=.

# 制作 obd 镜像
```

```
# -n 镜像名字，-V 版本号，-p 路径，-f 强制更新镜像，-t 标签，与部署配置文件对应
obd mirror create -n oceanbase-ce -V 4.0.0.0 -p ./usr/local/ -f -t final_2022

```

5. 使用 obd 部署

```
# 使用 obd 部署单节点集群。obd 会自动启动
# -f 会删除之前部署数据，如果有的话
obd cluster autodeploy final_2022 -c final_2022.yaml -f

```

obd 其它常用命令：

```
# 重启集群
obd cluster restart final_2022

# 销毁之前部署的集群
obd cluster destroy final_2022

```

6、如何使用 ccache 加速编译

ccache 是一种编译缓存工具，可以将编译出的中间文件放置在某个目录下，下次编译时可以直接使用，进而加速编译。

可以把提供的 demo 代码中的 cmake/Env.cmake 文件覆盖掉自己的代码，就可以使用这个功能。默认情况下，缓存目录在 $HOME/.ccache，大小是 5G。

7、如何做 major freeze

执行 SQL：

```
ALTER SYSTEM MAJOR FREEZE;
```

major freeze 是后台执行的，可以使用下面的语句判断是否执行完成：

```
select sum(last_scn) as last_scn_sum, sum(frozen_scn) as frozen_scn_sum from oceanbase.CDB_OB_MAJOR_COMPACTION;
```

这个 SQL 会返回一行两列数据，如果两列的值相等，就说明执行完成。

8、测试环境 OceanBase 部署配置文件

```
oceanbase-ce:
tag: final_2022
```

```
servers:
# Please don't use hostname, only IP can be supported
- 127.0.0.1
global:
#  The working directory for OceanBase Database. OceanBase Database is started under this directory. This is a required field.
home_path: /data/final/final_2022
# The directory for data storage. The default value is $home_path/store.
# data_dir: /data
# The directory for clog, ilog, and slog. The default value is the same as the data_dir value.
# redo_dir: /redo
# Please set devname as the network adaptor's name whose ip is  in the setting of severs.
# if set severs as "127.0.0.1", please set devname as "lo"
# if current ip is 192.168.1.10, and the ip's network adaptor's name is "eth0", please use "eth0"
devname: lo
mysql_port: 2881 # External port for OceanBase Database. The default value is 2881. DO NOT change this value after the cluster is started.
rpc_port: 2882 # Internal port for OceanBase Database. The default value is 2882. DO NOT change this value after the cluster is started.
zone: zone1
# if current hardware's memory capacity is smaller than 50G, please use the setting of "mini-single-example.yaml" and do a small adjustment.
memory_limit: 14G # The maximum running memory for an observer
# The reserved system memory. system_memory is reserved for general tenants. The default value is 30G.
system_memory: 12G
datafile_size: 150G # The percentage of the data_dir space to the total disk space. This value takes effect only when datafile_size is 0. The default value is 90.
log_disk_size: 20G
syslog_level: INFO # System log level. The default value is INFO.
enable_syslog_wf: false # Print system logs whose levels are higher than WARNING to a separate log file. The default value is true.
enable_syslog_recycle: true # Enable auto system log recycling or not. The default value is false.
max_syslog_file_count: 4 # The maximum number of reserved log files before enabling auto recycling. The default value is 0.
```

9、推荐参考资料和相关链接

1) 旁路导入 demo 程序参考： 旁路导入 github pull request
https://github.com/oceanbase/oceanbase/pull/1107

2) 如何编译 OceanBase 源码：https://www.oceanbase.com/docs/community-developer-advance-10000000000627375
3) LSM-Tree 介绍：https://www.oceanbase.com/docs/community-developer-advance-0000000000634013
4) 如何为 OceanBase 贡献源代码：https://www.oceanbase.com/docs/community-developer-advance-10000000000627380
5) 存储格式介绍：https://www.oceanbase.com/docs/community-developer-advance-0000000000660124
6) 社区组织： https://open.oceanbase.com/community/organization
7) Issue 列表：https://github.com/oceanbase/oceanbase/issues
8) Contribute to OceanBase（如何贡献代码）：https://github.com/oceanbase/oceanbase/wiki/Contribute-to-OceanBase