

openGauss 数据库维护管理实验

姓名：____张洋____ 学号：____2111460____

实验步骤：

- openGauss 数据库安装
- 数据库性能检查实验
- 最大连接数设置实验
- 例行表、索引维护实验

实验报告

实验步骤截图：

截图 1：指导手册 25 页顺序扫描执行计划截图

```
postgres=# explain select * from student where std_id=30;
               QUERY PLAN
-----
Seq Scan on student (cost=0.00..1.62 rows=1 width=62)
  Filter: (std_id = 30)
(2 rows)
```

截图 2：指导手册 26 页索引扫描执行计划截图

```
postgres=# explain select /*+indexscan(student student_pkey)*/ * from student where std_id=30;
               QUERY PLAN
-----
[Bypass]
Index Scan using student_pkey on student (cost=0.00..8.27 rows=1 width=62)
  Index Cond: (std_id = 30)
(3 rows)
```

截图 3：将最大连接数设置为 8000 并验证设置是否成功截图（注意，指导手册中将最大连接数设置为 6000，怎样重新设置为 8000 呢？）

```
[omm@ecs-4966 ~]$ gsql -d postgres -p 26000 -r
gsql ((openGauss 2.0.0 build 78689da9) compiled at 2021-03-31 21:03:52 commit 0 last mr )
Non-SSL connection (SSL connection is recommended when requiring high-security)
Type "help" for help.

postgres=# SHOW max_connections;
 max_connections
-----
          8000
(1 row)
```

实验思考题：

1. 全表扫描和索引扫描的区别是什么？具体是如何实现的？比较两种扫描方式的 cost（提供查询结果截图），为什么全表扫描比索引扫描 cost 更小？在什么情况下通过主键进行查找会比全表扫描更节省时间？

全表扫描是指对于一张表中的每一行数据都进行遍历，以寻找符合条件的数据。全表扫描通常是在没有任何索引或者索引无法满足查询条件时，才会被使用。全表扫描的效率较低，因为需要遍历整张表，尤其是当表的数据量较大时，效率会更低。

索引扫描是指利用索引来定位符合条件的数据，从而避免了全表扫描的开销。具体实现方式是，先在索引中找到符合条件的记录，再通过主键或者聚簇索引找到数据行。由于索引通常只包含表中的部分列，因此在进行索引扫描时，只需要扫描索引列的数据，而不需要扫描整个表的数据。因此，索引扫描的效率通常比全表扫描高。

虽然索引扫描的效率通常比全表扫描高，但在某些情况下，全表扫描可能更加高效，

比如当查询条件无法使用索引时，或者需要查询表中大部分数据时。

总的来说，索引扫描是一种基于索引定位数据的查询方式，而全表扫描则是直接对表中的所有数据进行遍历的方式。

```
postgres=# explain select * from student where std_id=30;
               QUERY PLAN
-----
Seq Scan on student  (cost=0.00..1.62 rows=1 width=62)
  Filter: (std_id = 30)
(2 rows)

postgres=# explain select /*+indexscan(student student_pkey)*/ * from student where std_id=30;
               QUERY PLAN
-----
[Bypass]
Index Scan using student_pkey on student  (cost=0.00..8.27 rows=1 width=62)
  Index Cond: (std_id = 30)
(3 rows)
```

如上图所示，全表扫描比索引扫描的 cost 更小。

当表的数据量较小时，全表扫描的 cost 可能比索引扫描更小。因为当数据量较小时，全表扫描的 I/O 操作开销较小，而索引扫描需要多次 I/O 操作才能定位数据。

另外，在一些查询条件无法使用索引或者索引选择性不高的情况下，全表扫描的 cost 可能也比索引扫描要小。比如本题中的 student_ID 属性已经在查询条件的开头，此时全表扫描可能比索引扫描更快。

通过主键进行查找可以避免全表扫描，因为主键是唯一的，可以直接通过主键定位数据行，而不需要进行全表扫描。在以下情况下，通过主键进行查找会比全表扫描更节省时间：

当需要查询表中的某一行或少数几行数据并且全部数据量较大时，可以通过主键进行查找通常比全表扫描更快。因为通过主键查找只需要进行一次索引查找即可找到所需数据，而全表扫描需要遍历整张表，消耗更多时间。当查询条件中包含主键时，通过主

键进行查找通常也会比全表扫描更快。因为主键是唯一的，可以直接通过主键定位数据行，而不需要进行全表扫描。

2. 请列举一种需要重建索引的情况和原因，并说明 openGauss 中重建索引的方式有哪些。

如果索引数据分散在不同的数据块中，会导致 I/O 操作增多，查询效率降低，就需要重建索引来整理碎片。

在 openGauss 中，重建索引的方式有以下几种：

1. DROP 和 CREATE 方式：首先通过 DROP INDEX 命令删除原来的索引，然后通过 CREATE INDEX 命令重新创建索引。这种方式比较简单，但需要在重建过程中停止对表的访问，可能会影响业务的正常运行。
2. REINDEX 方式：使用 REINDEX 命令可以对表的所有索引进行重建，而不需要停止对表的访问。这种方式可以避免业务的中断，但重建速度较慢。
3. CLUSTER 方式：使用 CLUSTER 命令可以重组表和相关的索引，使得表的数据按照指定的顺序存储在磁盘上，可以提高查询性能。CLUSTER 操作实际上是将表按照指定的列排序，然后重建索引。
4. ALTER INDEX 方式：使用 ALTER INDEX 命令可以对索引进行修改和重建。
例如，可以对已经存在的索引增加新的列，或者修改存储结构。这种方式比较灵活，但需要谨慎操作，以免影响索引的正确性。