

数据库课程实验报告-数据库调优

| 基本信息 | | |
|---------|------------------|--|
| 姓名: | 汪畅 | |
| 专业: | 地图学与地理信息系统 | |
| 学号: | 201028007010019 | |
| 年级: | 2010 | |
| 院系: | 资源与环境学院 | |
| E-mail: | 675015515@qq.com | |

1. 实验目的

学会使用查询代价分析工具，并能够针对某个复杂的 SQL 语句分析各运算的代价。

2. 实验内容和步骤

本实验选择两个表如下

| ID | customer_name | account_number |
|----|---------------|----------------|
| 1 | 汪畅 | 002 |
| 2 | 汪畅 | 005 |
| 3 | 张三 | 001 |
| 4 | 张三 | 007 |
| 5 | 李四 | 003 |
| 6 | 王五 | 008 |
| 7 | 王五 | 009 |
| 8 | 刘三 | 006 |
| 9 | 汪畅 | 004 |
| 10 | 刘三 | 010 |

depositor 表

| account_number | branch_name | balance |
|----------------|-------------|---------|
| 001 | 北京建行 | 10000 |
| 002 | 上海工行 | 10000 |
| 003 | 北京建行 | 10000 |
| 004 | 天津农行 | 10000 |
| 005 | 北京建行 | 1000 |
| 006 | 北京建行 | 1000 |
| 007 | 上海工行 | 1000 |
| 008 | 北京建行 | 10000 |
| 009 | 天津农行 | 2000 |
| 010 | 上海工行 | 1000 |

account 表

查询汪畅在北京建行里的存款。

方法一：

1. 查询 depositor 表中含有汪畅的存款信息，查询 account 表中含有北京建行的存款信息；
2. 利用两个表的 account_name 属性连接，得到汪畅在北京建行里的存款。

查询语句如下：

```
use bank
go
select b.customer_name,a.branch_name,a.balance
from (select * from dbo.account where dbo.account.branch_name='北京
建行') as a,
(select dbo.depositor.account_number,dbo.depositor.customer_name
from dbo.depositor where dbo.depositor.customer_name='汪畅') as b
where a.account_number=b.account_number
```

方法二：

1. 利用 account_name 属性连接两个表到一个表中；
2. 通过属性值“汪畅”和“北京建行”查询 1 中所得到的表，得到汪畅在北京建行里的存款。

查询语句如下：

```
use bank
go
select a.customer_name,a.branch_name,a.balance
from (select
dbo.depositor.customer_name,dbo.account.branch_name,dbo.account.b
alance from dbo.account,dbo.depositor where
dbo.account.account_number=dbo.depositor.account_number) as a
where a.customer_name='汪畅' and a.branch_name='北京建行'
```

3. 实验结果及结论

(1) 实验结果

得到的结果如下：

查询 2：(与该批有关的) 查询开销：100%

```
select b.customer_name,a.branch_name,a.balance from (select * fro
```



方法一

查询 2：(与该批有关的) 查询开销：100%

```
select a.customer_name,a.branch_name,a.balance from (select dbo.dep
```



方法二

其中，select 的一些参数值如下：

| SELECT | |
|---------|-----------|
| 缓存的计划大小 | 11 字节 |
| 估计运算符开销 | 0 (0%) |
| 估计子树大小 | 0.0067388 |
| 估计行数 | 2 |

方法一

| SELECT | |
|---------|-----------|
| 缓存的计划大小 | 11 字节 |
| 估计运算符开销 | 0 (0%) |
| 估计子树大小 | 0.0072366 |
| 估计行数 | 5 |

方法二

嵌套循环的一些参数值如下：

| 嵌套循环 | |
|---|----------------|
| 对于顶部 (外部) 输入的每一行，扫描底部 (内部) 输入，然后输出匹配的行。 | |
| 物理运算 | 嵌套循环 |
| Logical Operation | Inner Join |
| 估计 I/O 开销 | 0 |
| 估计 CPU 开销 | 0.0000084 |
| 估计运算符开销 | 0.0000112 (0%) |
| 估计子树大小 | 0.0067388 |
| 估计行数 | 2 |
| 估计行大小 | 33 字节 |
| 节点 ID | 0 |

方法一

| 嵌套循环 | |
|---|----------------|
| 对于顶部 (外部) 输入的每一行，扫描底部 (内部) 输入，然后输出匹配的行。 | |
| 物理运算 | 嵌套循环 |
| Logical Operation | Inner Join |
| 估计 I/O 开销 | 0 |
| 估计 CPU 开销 | 0.0000209 |
| 估计运算符开销 | 0.0000281 (0%) |
| 估计子树大小 | 0.0072366 |
| 估计行数 | 5 |
| 估计行大小 | 33 字节 |
| 节点 ID | 0 |

方法二

聚集索引扫描的一些参数值如下：

| 聚集索引扫描 | |
|-------------------|----------------------|
| 整体扫描聚集索引或只扫描一定范围。 | |
| 物理运算 | 聚集索引扫描 |
| Logical Operation | Clustered Index Scan |
| 估计 I/O 开销 | 0.003125 |
| 估计 CPU 开销 | 0.0001614 |
| 估计运算符开销 | 0.0032864 (49%) |
| 估计子树大小 | 0.0032864 |
| 估计行数 | 2 |
| 估计行大小 | 23 字节 |
| 已排序 | False |
| 节点 ID | 1 |

方法一

| 聚集索引扫描 | |
|-------------------|----------------------|
| 整体扫描聚集索引或只扫描一定范围。 | |
| 物理运算 | 聚集索引扫描 |
| Logical Operation | Clustered Index Scan |
| 估计 I/O 开销 | 0.003125 |
| 估计 CPU 开销 | 0.000168 |
| 估计运算符开销 | 0.003293 (46%) |
| 估计子树大小 | 0.003293 |
| 估计行数 | 5 |
| 估计行大小 | 23 字节 |
| 已排序 | False |
| 节点 ID | 1 |

方法二

聚集索引查找的一些参数值如下：

| 聚集索引查找 | |
|-------------------|----------------------|
| 扫描聚集索引中特定范围的行。 | |
| 物理运算 | 聚集索引查找 |
| Logical Operation | Clustered Index Seek |
| 估计 I/O 开销 | 0.003125 |
| 估计 CPU 开销 | 0.0001581 |
| 估计运算符开销 | 0.0034412 (51%) |
| 估计子树大小 | 0.0034412 |
| 估计行数 | 1 |
| 估计行大小 | 27 字节 |
| 已排序 | True |
| 节点 ID | 2 |

方法一

| 聚集索引查找 | |
|-------------------|----------------------|
| 扫描聚集索引中特定范围的行。 | |
| 物理运算 | 聚集索引查找 |
| Logical Operation | Clustered Index Seek |
| 估计 I/O 开销 | 0.003125 |
| 估计 CPU 开销 | 0.0001581 |
| 估计运算符开销 | 0.0039155 (54%) |
| 估计子树大小 | 0.0039155 |
| 估计行数 | 1 |
| 估计行大小 | 27 字节 |
| 已排序 | True |
| 节点 ID | 2 |

方法二

(2) 实验结论

查询汪畅在北京建行里的存款有两种查询思路：先投影，后连接（方法一）；先连接，后投影（方法二）。

方法一：两个表分别投影得到的数据去除了那些不相关的数据，大大减小了元组的数量，从而是磁盘块能容纳更多的元组，减少了 I/O 操作代价。连接后，就得到结果了。但是投影消耗的计算量较大，所以 cpu 开销较大。

方法二：连接操作需要把其中一个表的所有元组与另外一个表中的元组对应，连接后元组数量可能增加，这样需要更多的磁盘空间。但进行投影时候，计算量没有方法一的大，所以 cpu 开销较小。

（3）问题分析

根据上述分析，该实验不能够明显的比较出两种查询方法效率的大小，这是因为数据量较小，数据的类型没有明显的特点（比如说一个表相应的数据较多，一个表对应的数据较少）。但是，本次试验可以看出两种查询方法的优缺点，还是达到了实验的目的。