

数据库课程实验报告-查询分析优化

基本信息

姓名：蔡衡
专业：地图学与地理信息系统
学号：201028007410002
年级：研一
院系：资环学院
E-mail：18162335@qq.com

一. 实验目的

- (1) 学会使用 SQL Server 中的查询分析代价分析工具
- (2) 针对某个复杂的 SQL 语句分析各运算的代价。

二. 实验准备

- (1) 下列组件是完成本实验所必须的
Microsoft SQL Server 2005 Database Engine;
Microsoft SQL Server 2005 Management studio

- (2) 实验用表,

1. dbo.branch, 属性为 branch_name, branch_city, assets

| | branch_name | branch_city | assets |
|---|-------------|-------------|---------|
| ▶ | brighton | brooklyn | 7100000 |
| | downtownw | brooklyn | 9000000 |
| | mianus | horseneck | 400000 |
| | north town | rye | 3700000 |
| | perryridge | horseneck | 1700000 |
| | pownal | bennington | 300000 |
| | redwood | palo alto | 2100000 |
| | round hill | horseneck | 8000000 |
| * | NULL | NULL | NULL |

2. dbo.loan, 包括属性为 loan_number, branch_name, amount

| | loan_number | branch_name | amount |
|---|-------------|-------------|--------|
| ▶ | L-11 | Round hill | 900 |
| | L-14 | downtown | 1500 |
| | L-15 | perryridge | 1500 |
| | L-16 | perryridge | 1300 |
| | L-17 | downtown | 1000 |
| | L-23 | redwood | 2000 |
| | L-93 | mianus | 500 |
| * | NULL | NULL | NULL |

3. dbo.borrower, 属性为 customer_name, loan_number

| | customer_name | loan_number |
|---|---------------|-------------|
| ▶ | smith | L-11 |
| | jackson | L-14 |
| | hayes | L-15 |
| | adams | L-16 |
| | jones | L-17 |
| | williams | L-18 |
| | smith | L-23 |
| | curry | L-93 |
| * | NULL | NULL |

4. dbo.depositor, 属性为 customer_name, account_number

| | customer_name | account_number |
|---|---------------|----------------|
| ▶ | johnson | A-101 |
| | hayes | A-102 |
| | johnson | A-201 |
| | smith | A-215 |
| | jones | A-217 |
| | lindsay | A-222 |
| | turner | A-305 |
| * | NULL | NULL |

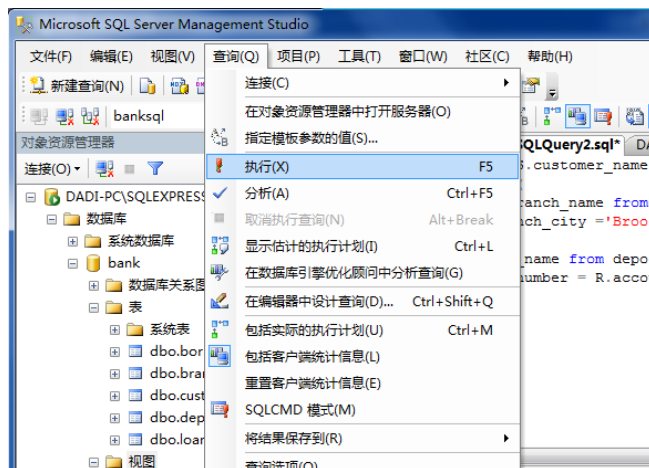
5. dbo.customer, 属性为 customer_name, customer_street, customer_city

| | customer_name | customer_street | customer_city |
|---|---------------|-----------------|---------------|
| ▶ | adams | spring | pittsfield |
| | brooks | senator | brooklyn |
| | curry | north | rye |
| | glenn | sand hill | woodside |
| | green | walnut | stamford |
| | hayes | main | harrison |
| | johnson | alma | palo alto |
| | jones | main | harrison |
| | lindsay | park | pittsfield |
| | smith | north | rye |
| | turner | putnam | stamford |
| | williams | nassau | princeton |
| * | NULL | NULL | NULL |

三. 实验内容和步骤

(1) 新建查询

SQL 2005中查询分析器和事件探查器都被整合到企业管理器中，打开企业管理器直接点击“新建查询”就可以打开查询分析器，其中打开查询分析器需要对数据库进行连接。本实验利用查询管理器中的“分析”“显示估计的执行计划”和“包括客户端统计信息”三个功能对查询代价进行分析。



(2) 利用的复杂SQL查询语句进行查询

/*找出所有位于Brooklyn的所有支行都有帐户的客户*/

```
select distinct S.customer_name from depositor as S
where not exists(
```

```
  (select branch.branch_name from branch
   where branch.branch_city = 'Brooklyn' and branch.branch_name
   not in
```

```
  (select R.branch_name from depositor as T, account as R
```

```
   Where T.account_number = R.account_number and S.customer_name = T.customer_name )))
```

(3) 利用查询管理器中的“分析”“显示估计的执行计划”和“包括客户端统计信息”三个功能对查询代价进行分析。

四. 实验结果及结论

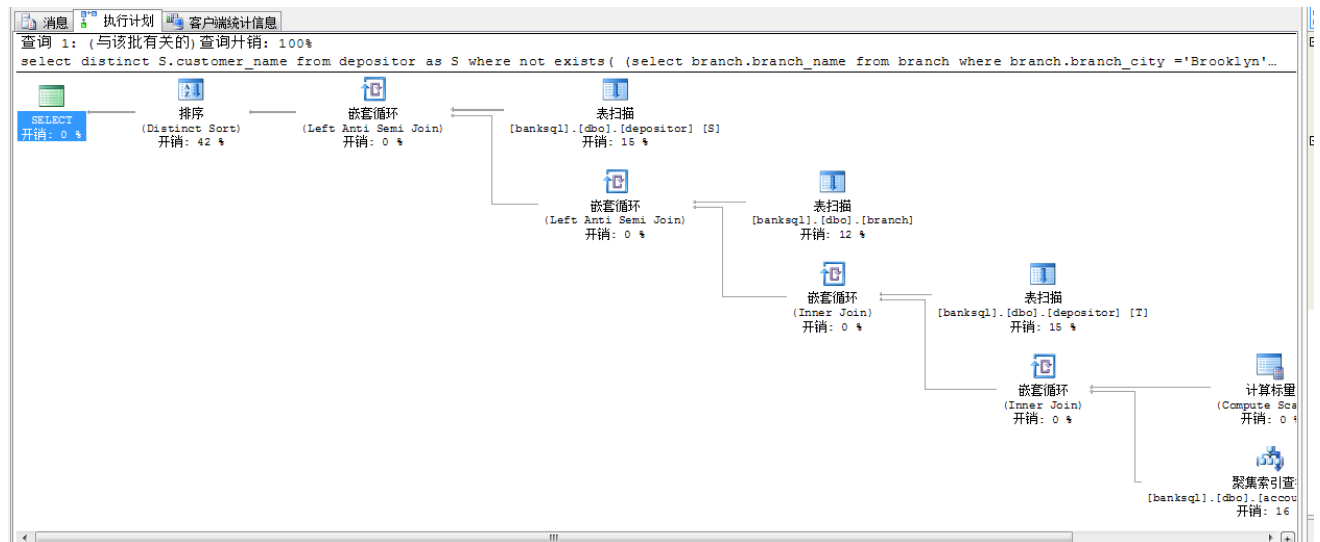
(1) 实验结果

| 结果 | |
|----|---------------|
| | customer_name |
| 1 | hayes |
| 2 | johnson |
| 3 | jones |
| 4 | lindsay |
| 5 | smith |
| 6 | tuner |

查询结果：

显示估计的执行计划：

找出所有位于 Brooklyn 的所有支行都有帐户的客户进行的查询，其代价估计见下图：



查询 1: (与该批有关的) 查询开销: 100%

```
select distinct S.customer_name fr
```

客户端统计信息

有关的) 查询开销: 100%

```
: S.customer_name from depositor as S where no
```

排序

SELECT

| | |
|---------|-----------|
| 缓存的计划大小 | 29 字节 |
| 估计运算符开销 | 0 (0%) |
| 估计子树大小 | 0.0273318 |
| 估计行数 | 5.5 |

语句

```
select distinct
S.customer_name from
depositor as S
where not exists(
(select branch.branch_name
from branch
where branch.branch_city
= 'Brooklyn' and
branch.branch_name
not in
(select R.branch_name from
depositor as T, account as R
Where T.account_number =
R.account_number and
S.customer_name =
T.customer_name )))
```

排序

对输入进行排序。

| | |
|-------------------|-----------------|
| 物理运算 | 排序 |
| Logical Operation | Distinct Sort |
| 估计 I/O 开销 | 0.0112613 |
| 估计 CPU 开销 | 0.0001261 |
| 估计运算符开销 | 0.0113874 (42%) |
| 估计子树大小 | 0.0273318 |
| 估计行数 | 5.5 |
| 估计行大小 | 27 字节 |
| 节点 ID | 0 |

输出列表

```
[banksql].[dbo].
[depositor].customer_name
```

排序依据

```
[banksql].[dbo].
[depositor].customer_name 升序
```

SQL Server Enterprise Manager - SQLQuery1.sql* 对象资源管理器详细信...

```

from depositor as S

m depositor as S where not exists ( (select

```

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 | Left Anti Semi Join |
| 估计 I/O 开销 | 0 |
| 估计 CPU 开销 | 0.0000293 |
| 估计运算符开销 | 0.0000293 (0%) |
| 估计子树大小 | 0.0159444 |
| 估计行数 | 6.3 |
| 估计行大小 | 27 字节 |
| 节点 ID | 1 |

输出列表
[banksql].[dbo].
[depositor].customer_name

外部引用
[banksql].[dbo].
[depositor].customer_name

DADI-PC\SQLEXPRESS (9.0)

```

exists ( (select branch.branch_name from

```

表扫描
扫描表中的行。

| 物理运算 | 表扫描 |
|--------------------------|-----------------|
| Logical Operation | Table Scan |
| 估计 I/O 开销 | 0.0038657 |
| 估计 CPU 开销 | 0.0001647 |
| 估计运算符开销 | 0.0040304 (15%) |
| 估计子树大小 | 0.0040304 |
| 估计行数 | 7 |
| 估计行大小 | 27 字节 |
| 已排序 | False |
| 节点 ID | 2 |

对象
[banksql].[dbo].[depositor] [S]

输出列表
[banksql].[dbo].
[depositor].customer_name

DADI-PC\SQLEXPRESS (9.0)

```

ot exists ( (select branch.branch_name from

```

表扫描
[banksql].[dbo].[depositor] [S]
开销: 15 %

嵌套循环
(Left Anti Semi Join)
开销: 0 %

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 | Left Anti Semi Join |
| 估计 I/O 开销 | 0 |
| 估计 CPU 开销 | 0.0000042 |
| 估计运算符开销 | 0.0000333 (0%) |
| 估计子树大小 | 0.0118847 |
| 估计行数 | 1 |
| 估计行大小 | 9 字节 |
| 节点 ID | 3 |

外部引用
[banksql].[dbo].[branch].branch_name

DADI-PC\SQLEXPRESS (9.0)

```

:branch.branch_name from branch where branch.branch_ci

```

表扫描
[banksql].[dbo].[branch] [S]
开销: 15 %

表扫描
扫描表中的行。

| 物理运算 | 表扫描 |
|--------------------------|-----------------|
| Logical Operation | Table Scan |
| 估计 I/O 开销 | 0.0032035 |
| 估计 CPU 开销 | 0.0000796 |
| 估计运算符开销 | 0.0032897 (12%) |
| 估计子树大小 | 0.0032897 |
| 估计行数 | 1 |
| 估计行大小 | 52 字节 |
| 已排序 | False |
| 节点 ID | 4 |

谓词
[banksql].[dbo].[branch].[branch_city]
='Brooklyn'

对象
[banksql].[dbo].[branch]

输出列表
[banksql].[dbo].[branch].branch_name

DADI-PC\SQLEXPRESS (9.0)

h_name from branch where branch.branch_city =

表扫描

扫描表中的行。

物理运算

| Logical Operation | Table Scan |
|-------------------|-----------------|
| 估计 I/O 开销 | 0.0032035 |
| 估计 CPU 开销 | 0.0000796 |
| 估计运算符开销 | 0.0032897 (12%) |
| 估计子树大小 | 0.0032897 |
| 估计行数 | 1 |
| 估计行大小 | 52 字节 |
| 已排序 | False |
| 节点 ID | 4 |

谓词

[banksql].[dbo].[branch].[branch_city] = 'Brooklyn'

对象

[banksql].[dbo].[branch]

输出列表

[banksql].[dbo].[branch].branch_name

聚集索引查找

扫描聚集索引中特定范围的行。

物理运算

| 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 | 0.0044162 (16%) |
| 估计子树大小 | 0.0044162 |
| 估计行数 | 1 |
| 估计行大小 | 52 字节 |
| 已排序 | True |
| 节点 ID | 14 |

谓词

[banksql].[dbo].[depositor].
[account_number] as [T].[account_number]
= CONVERT_IMPLICIT(nchar(15),[banksql].
[dbo].[account].[account_number] as [R].
[account_number],0) AND ([banksql].[dbo].
[account].[branch_name] as [R].
[branch_name] IS NULL OR [banksql].[dbo].
[branch].[branch_name]=[banksql].[dbo].
[account].[branch_name] as [R].
[branch_name])

对象

[banksql].[dbo].[account].
[PK_account_023D5A04] [R]

输出列表

[banksql].[dbo].[account].account_number,
[banksql].[dbo].[account].branch_name

Seek 谓词

范围起始值: [banksql].[dbo].
[account].account_number > 标量运算符
((Expr1014)), 范围结束值: [banksql].[dbo].
[account].account_number < 标量运算符
((Expr1015))

从上面图中看出，大部分开销用于了排序和表扫描。

客户端统计信息：

```
select distinct S.customer_name from depositor as S
where not exists(
(select branch.branch_name from branch
where branch.branch_city = 'Brooklyn' and branch.branch_name
not in
(select R.branch_name from depositor as T, account as R
Where T.account_number = R.account_number and S.customer_name = T.customer_name )))
```

| | 查询试验 5 | 查询试验 4 | 查询试验 3 | 查询试验 2 | 查询试验 1 | 平均 |
|--------------------------------|----------|----------|----------|----------|----------|-------------|
| 客户端执行时间 | 15:20:39 | 15:20:34 | 15:11:36 | 15:11:18 | 15:10:49 | |
| 查询配置文件统计信息 | | | | | | |
| INSERT、DELETE 和 UPDATE 语句的数目 | 0 | → 0 | → 0 | → 0 | → 0 | → 0.0000 |
| INSERT、DELETE 或 UPDATE 语句影响的行数 | 0 | → 0 | → 0 | → 0 | → 0 | → 0.0000 |
| SELECT 语句的数目 | 1 | ↑ 0 | ↓ 1 | → 1 | → 1 | → 0.8000 |
| SELECT 语句返回的行数 | 1 | ↑ 0 | ↓ 6 | → 6 | ↑ 1 | → 2.8000 |
| 事务数 | 0 | → 0 | → 0 | → 0 | → 0 | → 0.0000 |
| 网络统计信息 | | | | | | |
| 服务器往返的次数 | 3 | → 3 | ↑ 1 | → 1 | ↓ 3 | → 2.2000 |
| 从客户端发送的 TDS 数据包 | 3 | → 3 | ↑ 1 | → 1 | ↓ 3 | → 2.2000 |
| 从服务器接收的 TDS 数据包 | 8 | ↑ 3 | ↑ 1 | → 1 | ↓ 8 | → 4.2000 |
| 从客户端发送的字节数 | 818 | ↑ 806 | ↑ 680 | → 680 | ↓ 818 | → 760.4000 |
| 从服务器接收的字节数 | 23832 | ↑ 63 | ↓ 203 | → 203 | ↓ 23832 | → 9626.6000 |
| 时间统计信息 | | | | | | |
| 客户端处理时间 | 1 | ↑ 0 | ↓ 1 | ↓ 3 | ↑ 0 | → 1.0000 |
| 总执行时间 | 3 | ↑ 0 | ↓ 1 | ↓ 3 | → 3 | → 2.0000 |
| 服务器应答等待时间 | 2 | ↑ 0 | → 0 | → 0 | ↓ 3 | → 1.0000 |

(2) 实验结论

利用 SQL2005 的查询代价分析工具，可看到查询的各种运算的开销以及不同的查询实验的各统计信息，为查询优化提供了依据。