Video 15-2：代数优化（一）

（对应教科书9.3小节）

代 数 优 化

观察1：一个代数表达式（查询），可以有多个结果相同但是形式不同的表达式

观察2：这些表达式的执行效率是不一样的，有时候还差别巨大！

代数优化

理想：

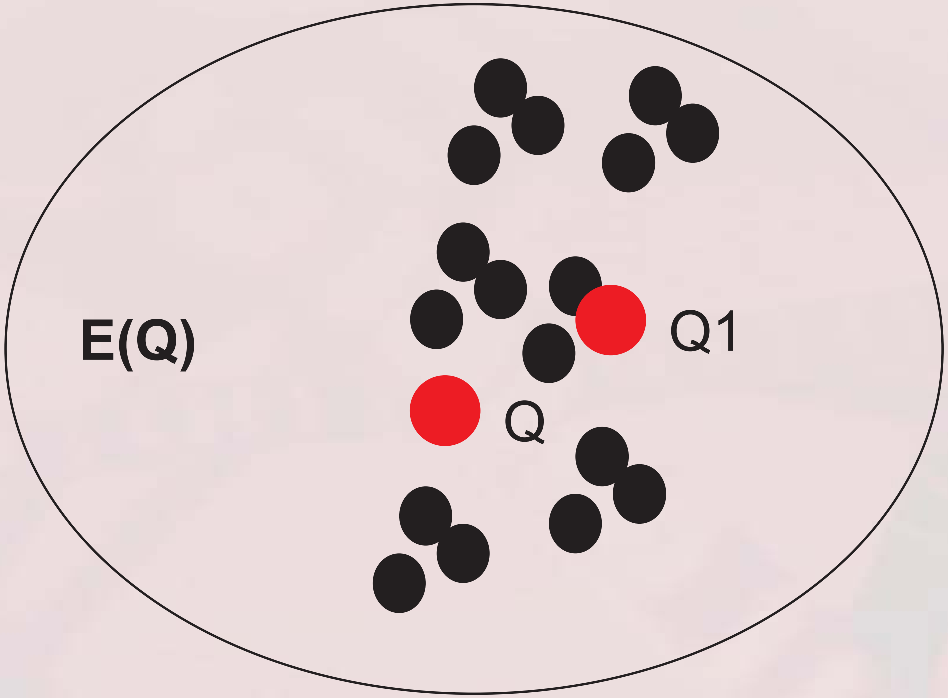
1 找出查询Q的全部等价表达式（结果相同）E(Q)

2 对E(Q)中的每一个表达式计算其执行代价：

$f(q)$, $q \in E(Q)$ q是Q的一个等价表达式

3 找出执行代价最小的表达式

问题：成本太高！


E(Q)
Q1
Q

代数优化

现实：

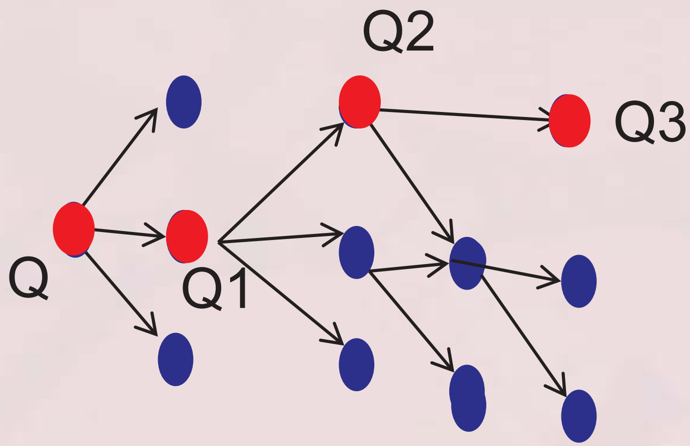
1 从查询Q出发，按照事先确定的规则对Q进行变换，获得Q1，Q1与Q必须等价

2 确保Q1的执行代价比Q的执行代价低。

3 直到找不到执行代价更小的等价表达式，结束

问题：不一定是最优的。

代数优化



代数优化的两个基本问题

- ❖ 关系代数表达式的等价变换规则
  - ■ 等价的含义：指用相同的关系代替两个表达式中相应的关系所得到的结果是相同的
  - ■ 什么样的变换一定是等价的。（避免去证明等价）
  - ■ 两个关系表达式E_1和E_2是等价的，可记为$E_1 \equiv E_2$
- ❖ 启发式规则
  - ■ 什么样的变换一定是“好”的，即执行代价更小。

关系代数表达式等价变换规则（1）

❖ 连接、笛卡尔积交换律

设E_1和E_2是关系代数表达式，F是连接运算的条件，则有

$E_1 \times E_2 \equiv E_2 \times E_1$

$E_1 \bowtie E_2 \equiv E_2 \bowtie E_1$

$E_1 \underset{F}{\bowtie} E_2 \equiv E_2 \underset{F}{\bowtie} E_1$

关系代数表达式等价变换规则（2）

❖ 连接、笛卡尔积的结合律

设E_1，E_2，E_3是关系代数表达式，F_1和F_2是连接运算的条件

$$(E_1 \times E_2) \times E_3 \equiv E_1 \times (E_2 \times E_3)$$

$$(E_1 \bowtie E_2) \bowtie E_3 \equiv E_1 \bowtie (E_2 \bowtie E_3)$$

$$(E_1 \underset{F_1}{\bowtie} E_2) \underset{F_2}{\bowtie} E_3 \equiv E_1 \underset{F}{\bowtie} (E_2 \underset{F_2}{\bowtie} E_3)$$

关系代数表达式等价变换规则（3）

3.投影的串接定律

$$\pi_{A_1,A_2,\cdots A_n}(\pi_{B_1,B_2,\cdots B_m}(E)) \equiv \pi_{A_1,A_2,\cdots A_n}(E)$$

- *E*是关系代数表达式
- A_i(i=1，2，…，n)，B_j(j=1，2，…，m)是属性名
- {A_1，A_2，…，A_n}构成{B_1，B_2，…，B_m}的子集

关系代数表达式等价变换规则（4）

4.选择的串接定律

$$\sigma_{F_1}(\sigma_{F_2}(E)) \equiv \sigma_{F_1 \wedge F_2}(E)$$

- E是关系代数表达式，F_1、F_2是选择条件
- 选择的串接律说明选择条件可以合并,这样一次就可检查全部条件

关系代数表达式等价变换规则（5）

5.选择与投影操作的交换律

$$\sigma_F(\pi_{A_1, A_2, \cdots A_n}(E)) \equiv \pi_{A_1, A_2, \cdots A_n}(\sigma_F(E))$$

- 选择条件F只涉及属性A_1，…，A_n。
- 若F中有不属于A_1，…，A_n的属性B_1，…，B_m有更一般规则：

$$\pi_{A_1, A_2, \cdots A_n}(\sigma_F(E)) \equiv \pi_{A_1, A_2, \cdots A_n}(\sigma_F(\pi_{A_1, A_2, \cdots A_n, B_1, B_2, \cdots B_m}(E)))$$

关系代数表达式等价变换规则（6）

6. 选择与笛卡尔积的交换律

- 如果F中涉及的属性都是E_1中的属性，则

$\sigma_F(E_1 \times E_2) \equiv \sigma_F(E_1) \times E_2$

- 如果$F=F_1 \wedge F_2$，并且F_1只涉及E_1中的属性，F_2只涉及E_2中的属性，则由上面的等价变换规则可推出：

$\sigma_F(E_1 \times E_2) \equiv \sigma_{F_1}(E_1) \times \sigma_{F_2}(E_2)$

- 若F_1只涉及E_1中的属性，F_2涉及E_1和E_2两者的属性，则仍有

$\sigma_F(E_1 \times E_2) \equiv \sigma_{F_2}(\sigma_{F_1}(E_1) \times E_2)$

它使部分选择在笛卡尔积前先做。

关系代数表达式等价变换规则（7）

7. 选择与并的分配律

设$E=E_1 \cup E_2$，E_1，E_2有相同的属性名，则

$$\sigma_F(E_1 \cup E_2) \equiv \sigma_F(E_1) \cup \sigma_F(E_2)$$

关系代数表达式等价变换规则（8）

8. 选择与差运算的分配律

若E_1与E_2有相同的属性名，则

$\sigma_F(E_1-E_2)\equiv\sigma_F(E_1)-\sigma_F(E_2)$

关系代数表达式等价变换规则（9）

9. 选择对自然连接的分配律

$\sigma_F(E_1 \bowtie E_2) \equiv \sigma_F(E_1) \bowtie \sigma_F(E_2)$

F只涉及E_1与E_2的公共属性

关系代数表达式等价变换规则（10）

10. 投影与笛卡尔积的分配律

设E_1和E_2是两个关系表达式，A_1，…，A_n是E_1的属性，B_1，…，B_m是E_2的属性，则

$$\pi_{A_1,A_2,\cdots A_n,B_1,B_2,\cdots B_m}(E_1 \times E_2) \equiv \pi_{A_1,A_2,\cdots A_n}(E_1) \times \pi_{B_1,B_2,\cdots B_m}(E_2)$$

关系代数表达式等价变换规则（11）

11. 投影与并的分配律

设E_1和E_2有相同的属性名，则

$$\pi_{A_1,A_2,\cdots A_n}(E_1 \cup E_2) \equiv \pi_{A_1,A_2,\cdots A_n}(E_1) \cup \pi_{A_1,A_2,\cdots A_n}(E_2)$$

Video 15-2：代数优化（二）

（对应教科书9.3小节）

代数优化的两个基本问题

- 关系代数表达式的等价变换规则
  - 等价的含义：指用相同的关系代替两个表达式中相应的关系所得到的结果是相同的
  - 两个关系表达式E_1和E_2是等价的，可记为$E_1 \equiv E_2$
- 启发式规则
  - 什么样的变换一定是“好”的，即执行代价更小。

查询树的启发式优化

❖ 典型的启发式规则

（1）选择运算应尽可能先做

在优化策略中这是最重要、最基本的一条。

（2）把投影运算和选择运算同时进行

如有若干投影和选择运算，并且它们都对同一个关系操作，则可以在扫描此关系的同时完成所有的这些运算以避免重复扫描关系。

查询树的启发式优化（续）

（3） 把投影同其前或其后的双目运算结合起来，没有必要为了去掉某些字段而扫描一遍关系。

（4） 把某些选择同在它前面要执行的笛卡尔积结合起来成为一个连接运算，连接特别是等值连接运算要比同样关系上的笛卡尔积省很多时间。

查询树的启发式优化（续）

（5） 找出公共子表达式

- 如果这种重复出现的子表达式的结果不是很大的关系
- 并且从外存中读入这个关系比计算该子表达式的时间少得多
- 则先计算一次公共子表达式并把结果写入中间文件是合算的。
- 当查询的是视图时，定义视图的表达式就是公共子表达式的情况

查询树的启发式优化（续）

算法：关系表达式的优化

输入：一个关系表达式的查询树

输出：优化的查询树

方法：

（1）利用等价变换规则4把形如$\sigma_{F1 \wedge F2 \wedge \ldots \wedge Fn}(E)$变换为$\sigma_{F1}(\sigma_{F2}(\ldots(\sigma_{Fn}(E))\ldots))$。

（2）对每一个选择，利用等价变换规则4～9尽可能把它移到树的叶端。

规则4：选择的串接定律

$$\sigma_{F_1}(\sigma_{F_2}(E)) \equiv \sigma_{F_1 \wedge F_2}(E)$$

规则4：　合并或分解选择运算

规则5-9：选择运算与其他运算交换

查询树的启发式优化（续）

（3）对每一个投影利用等价变换规则3，5，10，11中的一般形式尽可能把它移向树的叶端。

■ 注意：

- 等价变换规则3使一些投影消失或使一些投影出现
- 规则5把一个投影分裂为两个，其中一个有可能被移向树的叶端

（4）利用等价变换规则3～5，把选择和投影的串接合并成单个选择、单个投影或一个选择后跟一个投影，使多个选择或投影能同时执行，或在一次扫描中全部完成

查询树的启发式优化（续）

（3）对每一个投影利用等价变换规则3，5，10，11中的一般形式尽可能把它移向树的叶端。

- 注意：
  - 等价变换规则
  - 规则5把一个
    的叶端

规则3：合并或分解投影运算
规则4：合并或分解选择运算
规则5：投影运算与选择运算交换

（4）利用等价变换规则3～5，把选择和投影的串接合并

个投影，使多个

全部完成

规则3：合并或分解投影运算
规则5,10,11：投影运算与其他运算交换

查询树的启发式优化（续）

（5）把上述得到的语法树的内节点分组。

- 每一双目运算(×，⋈，∪，-)和它所有的直接祖先为一组(这些直接祖先是(σ，π运算)。
- 如果其后代直到叶子全是单目运算，则也将它们并入该组
- 但当双目运算是笛卡尔积(×)，而且后面不是与它组成等值连接的选择时，则不能把选择与这个双目运算组成同一组

查询树的启发式优化（续）

❖ [例9.4]下面给出[例9.3]中 SQL语句的代数优化示例

（1）把SQL语句转换成查询树，如下图所示

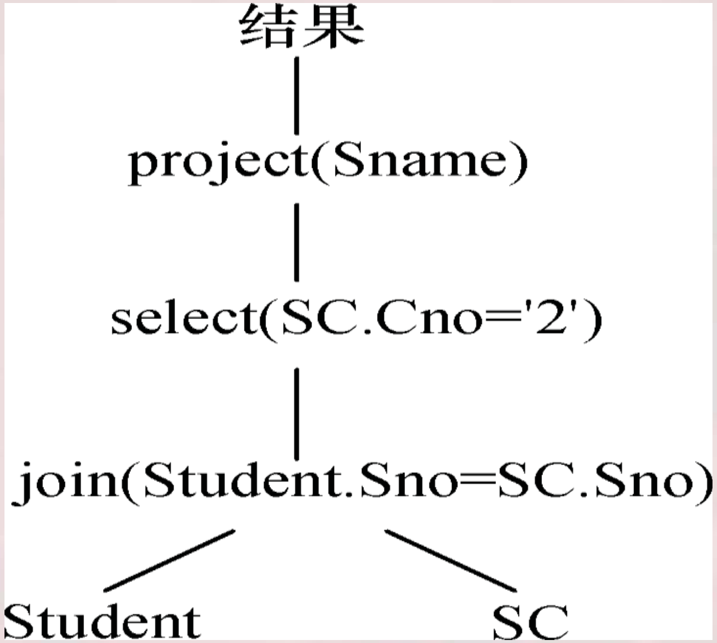


图9.3 查询树图

查询树的启发式优化（续）

为了使用关系代数表达式的优化法，假设内部表示是关系代数语法树，则上面的查询树如图9.4所示。

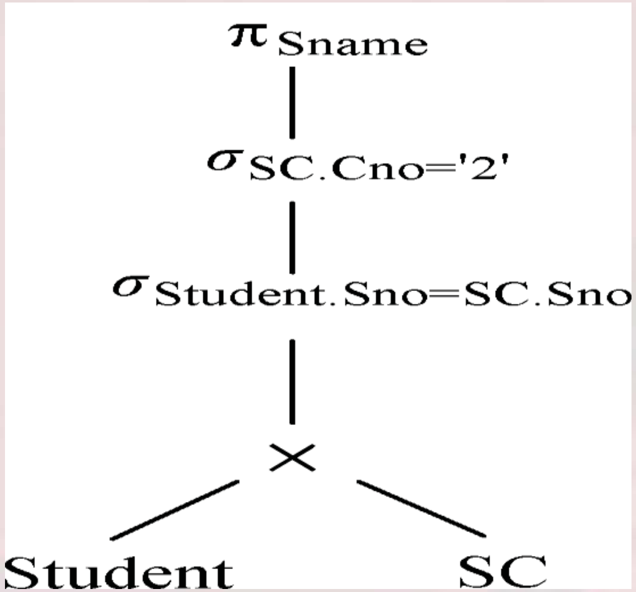


图9.4 关系代数语法树图

查询树的启发式优化（续）

（2 ）对查询树进行优化

■ 利用规则4、6把选择$\sigma_{SC.Cno='2'}$移到叶端，图9.4查询树便转换成下图优化的查询树。这就是9.2.2节中Q3的查询树表示。