

分布式数据库系统部分课后题答案 (by 谢龙)

第五章

EMP			ASG			
ENO	ENAME	TITLE	ENO	PNO	RESP	DUR
E1	J. Doe	Elect. Eng	E1	P1	Manager	12
E2	M. Smith	Syst. Anal.	E2	P1	Analyst	24
E3	A. Lee	Mech. Eng.	E2	P2	Analyst	6
E4	J. Miller	Programmer	E3	P3	Consultant	10
E5	B. Casey	Syst. Anal.	E3	P4	Engineer	48
E6	L. Chu	Elect. Eng.	E4	P2	Programmer	18
E7	R. Davis	Mech. Eng	E5	P2	Manager	24
E8	J. Jones	Syst. Anal.	E6	P4	Manager	48
			E7	P3	Engineer	36
			E8	P3	Manager	40

PROJ				PAY	
PNO	PNAME	BUDGET	LOC	TITLE	SAL
P1	Instrumentation	150000	Montreal	Elect. Eng.	40000
P2	Database Develop.	135000	New York	Syst. Anal.	34000
P3	CAD/CAM	250000	New York	Mech. Eng.	27000
P4	Maintenance	310000	Paris	Programmer	24000

Figure 5.3. Modified Example Database

5.1 p_1 : TITLE < "Programmer" and p_2 : TITLE > "Programmer".

(a) 根据 $\{p_1, p_2\}$ 对关系 EMP 进行水平分片: $EMP_1 = \sigma_{TITLE < "Programmer"}(EMP)$;

$EMP_2 = \sigma_{TITLE > "Programmer"}(EMP)$; 分片结果为:

EMP ₁			EMP ₂		
ENO	ENAME	TITLE	ENO	ENAME	TITLE
E1	J. Doe	Elect. Eng	E2	M. Smith	Syst. Anal.
E3	A. Lee	Mech. Eng.	E5	B. Casey	Syst. Anal.
E6	L. Chu	Elect. Eng.	E8	J. Jones	Syst. Anal.
E7	R. Davis	Mech. Eng			

图 5.1.1.

(b) 分片结果(EMP1, EMP2)不满足分片的正确性规则, 项"E4, J. Miller, Programmer"不在任何一个分片中, 其原因是: 谓词 $\{p_1, p_2\}$ 对关系 EMP 的划分并不完全。

(c) 可以这样修改 p_1 和 p_2 使其对 EMP 的划分符合分片的正确性规则:

p_1 : TITLE < "Programmer" and p_2 : TITLE \geq "Programmer"

根据新的谓词得到如图 5.1.2 的分片结果。

从图 5.1.2 可以看出关系 EMP 中的每一项都属于且仅属于 EMP1 或 EMP2 中的一个, 因此这个分片满足完整性 (Completeness) 和互斥性 (Disjointness); 又关系 $EMP = EMP1 \cup$

EMP2, 因此这个分片满足重构性 (Reconstruction)。因此这个分片满足分片的正确性规则。

EMP1			EMP2		
ENO	ENAME	TITLE	ENO	ENAME	TITLE
E1	J. Doe	Elect. Eng	E2	M. Smith	Syst. Anal.
E3	A. Lee	Mech. Eng.	E4	J. Miller	Programmer
E6	L. Chu	Elect. Eng.	E5	B. Casey	Syst. Anal.
E7	R. Davis	Mech. Eng	E8	J. Jones	Syst. Anal.

图 5.1.2

5.2

根据题目可以得到如下信息:

1. 根据第一个应用, 可以得到 5 个谓词:

p_1 : RESP = "Manager"; p_2 : RESP = "Consultant"; p_3 : RESP = "Engineer";
 p_4 : RESP = "Programmer"; p_5 : RESP = "Analyst"

2. 根据第二个应用, 可以得到 2 个谓词: p_6 : DUR < 20; p_7 : DUR ≥ 20;

根据得到的谓词, 我们对其取小项。共可得到 $5 \times 2 = 10$ 个不同的小项:

小项表

小项号	谓词组合	谓词组合详细信息
m_1	$p_1 \wedge p_6$	RESP = "Manager" \wedge DUR < 20
m_2	$p_1 \wedge p_7$	RESP = "Manager" \wedge DUR ≥ 20
m_3	$p_2 \wedge p_6$	RESP = "Consultant" \wedge DUR < 20
m_4	$p_2 \wedge p_7$	RESP = "Consultant" \wedge DUR ≥ 20
m_5	$p_3 \wedge p_6$	RESP = "Engineer" \wedge DUR < 20
m_6	$p_3 \wedge p_7$	RESP = "Engineer" \wedge DUR ≥ 20
m_7	$p_4 \wedge p_6$	RESP = "Programmer" \wedge DUR < 20
m_8	$p_4 \wedge p_7$	RESP = "Programmer" \wedge DUR ≥ 20
m_9	$p_5 \wedge p_6$	RESP = "Analyst" \wedge DUR < 20
m_{10}	$p_5 \wedge p_7$	RESP = "Analyst" \wedge DUR ≥ 20

我们根据以上这 10 个小项对 ASG 表进行水平分片, 共得到 10 个分片, 但其中根据小项 m_4 , m_5 和 m_8 得到的分片为空, 故没有写出。

ASG1

ENO	PNO	RESP	DUR
E1	P1	Manager	12

ASG2

ENO	PNO	RESP	DUR
E5	P2	Manager	24
E6	P4	Manager	48
E8	P3	Manager	40

ASG3

ENO	PNO	RESP	DUR
E3	P3	Consultant	10

ASG6

ENO	PNO	RESP	DUR
E3	P4	Engineer	48
E7	P3	Engineer	36

ASG7

ENO	PNO	RESP	DUR
E4	P2	Programmer	18

ASG9

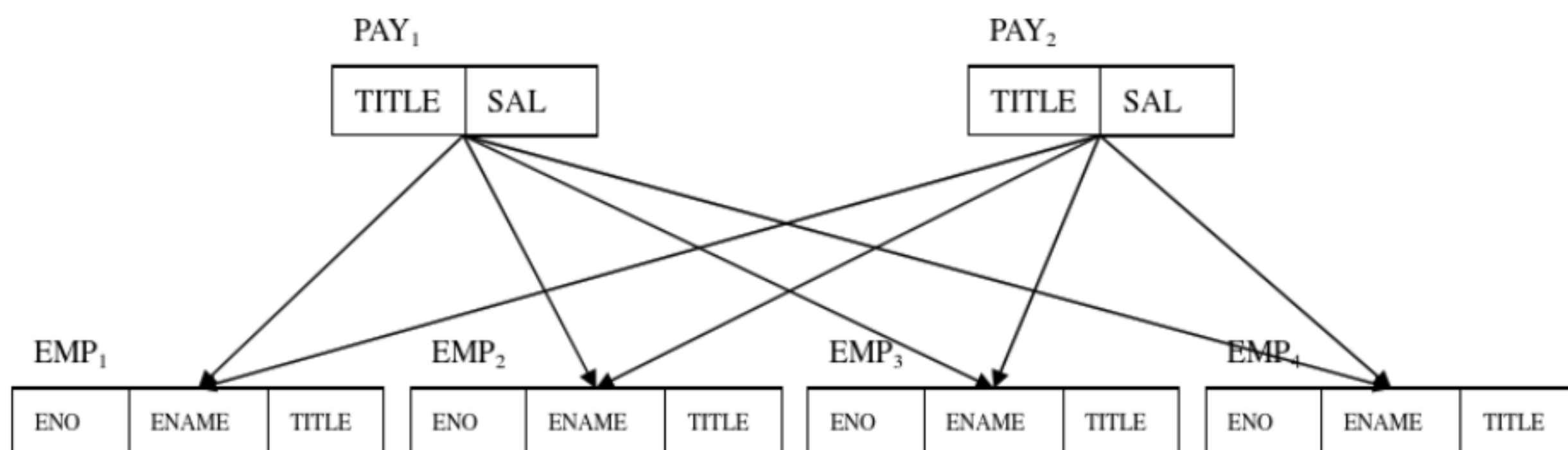
ENO	PNO	RESP	DUR
E2	P2	Analyst	6

ASG10

ENO	PNO	RESP	DUR
E2	P1	Analyst	24

5.3

$EMP \bowtie_{TITLE} PAY$ 的连接图如下：



这个图显然不是一个简单图。

我们可以通过将 PAY 关系根据 EMP 关系的分片进行诱导分片，即

$$PAY_1 = PAY \bowtie_{TITLE} EMP_1; PAY_2 = PAY \bowtie_{TITLE} EMP_2; PAY_3 = PAY \bowtie_{TITLE} EMP_3; PAY_4 = PAY \bowtie_{TITLE} EMP_4;$$

或 将 EMP 关系根据 PAY 关系的分片进行诱导分片（推荐此方案），即

$$EMP_1 = EMP \bowtie_{TITLE} PAY_1; EMP_2 = EMP \bowtie_{TITLE} PAY_2;$$

两种新的分片方案的连接图如下（推荐第二个方案）：

根据如下关系代数表达式进行分片后的 $EMP \bowtie_{TITLE} PAY$ 连接图：

$$EMP_1 = \sigma_{TITLE = "Elect. Eng."}(EMP); \quad EMP_2 = \sigma_{TITLE = "Syst. Anal."}(EMP);$$

$$EMP_3 = \sigma_{TITLE = "Mech. Eng."}(EMP); \quad EMP_4 = \sigma_{TITLE = "Programmer"}(EMP);$$

$$PAY_1 = PAY \bowtie_{TITLE} EMP_1; PAY_2 = PAY \bowtie_{TITLE} EMP_2; PAY_3 = PAY \bowtie_{TITLE} EMP_3; PAY_4 = PAY \bowtie_{TITLE} EMP_4;$$

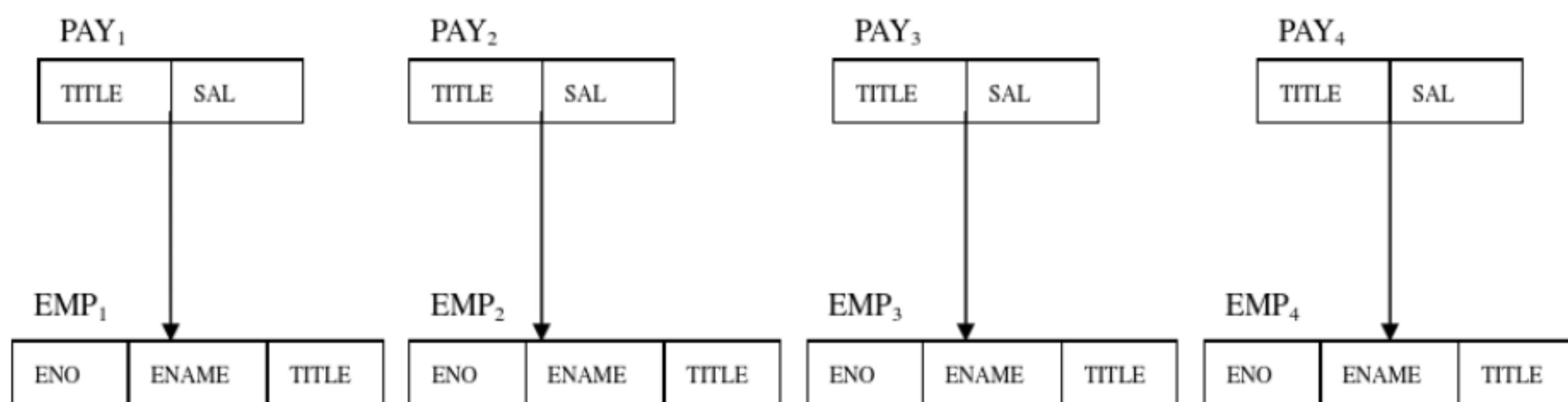


图 5.3.1

根据如下关系代数表达式分片后得到的 $EMP \bowtie_{TITLE} PAY$ 连接图：

$PAY_1 = \sigma_{SAL \geq 30000} (PAY)$; $PAY_2 = \sigma_{SAL < 30000} (PAY)$;

$EMP_1 = EMP \bowtie PAY_1$; $EMP_2 = EMP \bowtie PAY_2$

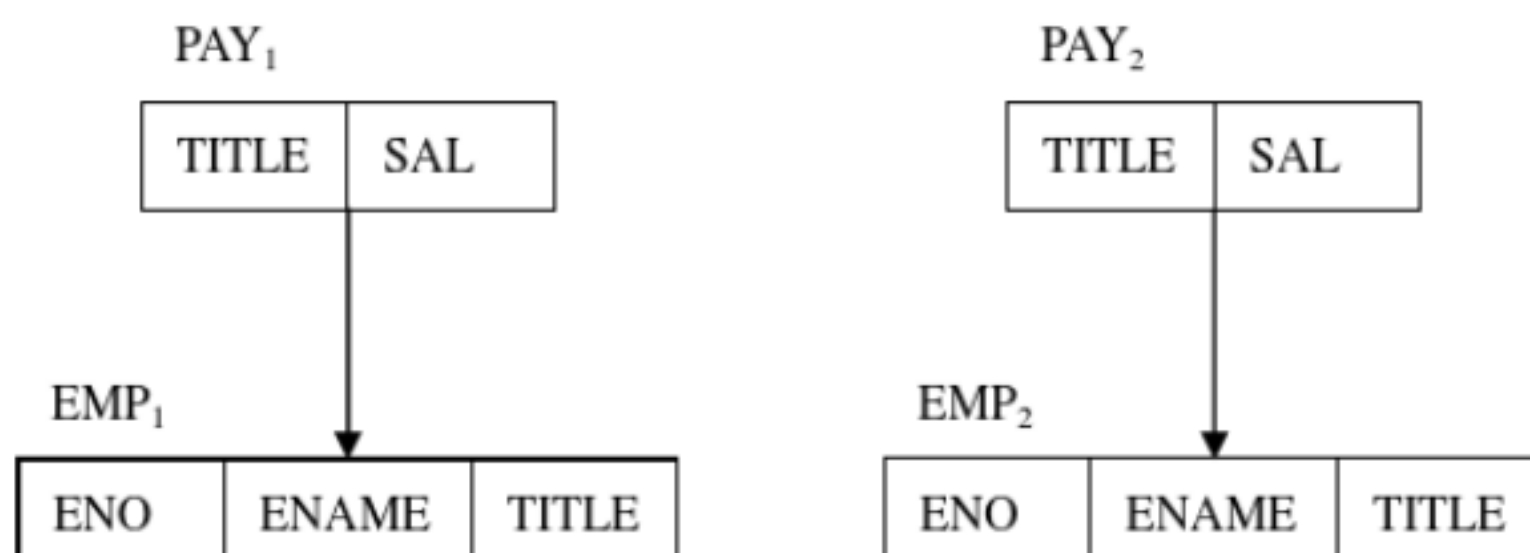


图 5.3.2

我们可以看出根据新的分片方案，我们得到的 $EMP \bowtie_{TITLE} PAY$ 的连接图 5.4.1 和图 5.4.2 都是简单图。

5.5 $p_1: SAL < 30000$ and $p_2: SAL \geq 30000$;

根据谓词 p_1 和 p_2 对 PAY 进行水平分片的关系代数表达式为：

$PAY_1 = \sigma_{SAL < 30000} (PAY)$; $PAY_2 = \sigma_{SAL \geq 30000} (PAY)$;

得到的分片结果为：

PAY ₁		PAY ₂	
TITLE	SAL	TITLE	SAL
Mech. Eng.	27000	Elect. Eng.	40000
Programmer	24000	Syst. Anal.	34000

图 5.5.1

再根据对 PAY 的分片结果 PAY₁ 和 PAY₂，对 EMP 进行诱导水平分片，其关系代数表达式为：

$EMP_1 = EMP \bowtie PAY_1$; $EMP_2 = EMP \bowtie PAY_2$

得到的分片结果为：

EMP ₁			EMP ₂		
ENO	ENAME	TITLE	ENO	ENAME	TITLE
E3	A. Lee	Mech. Eng.	E1	J. Doe	Elect. Eng.
E4	J. Miller	Programmer	E2	M. Smith	Syst. Anal.
E7	R. Davis	Mech. Eng.	E5	B. Casey	Syst. Anal.
			E6	L. Chu	Elect. Eng.
			E8	J. Jones	Syst. Anal.

图 5.5.2

完整性说明：对于 EMP 的诱导水平分片，根据引用完整性，member 关系（EMP）的任一个元组中的外键(TITLE)一定在 owner（PAY）关系中存在。按外键(TITLE)相等的条件连接时，一定不会丢失 member（EMP）关系中的元组。

互斥性说明：对于诱导水平分片，如果连接图是简单的，则诱导水平分片满足互斥性。

1. PAY₁, PAY₂ 的交集为空。

2.每个雇员仅有一个 TITLE。

3.每个 TITLE 仅有一个 SAL。

此时 EMP 的分片是互斥的。

假设一个雇员可以有多个 TITLE，一个 TITLE 有多个 SAL，则可能出现不满足 Disjointness 的情况。

重构性说明：关系 $EMP = EMP_1 \cup EMP_2$

5.6 查询集 $Q = \{q_1, q_2, q_3, q_4, q_5\}$ 属性集 $A = \{A_1, A_2, A_3, A_4, A_5\}$ 站点集 $S = \{S_1, S_2, S_3\}$ $ref_i(q_k) = 1$

$$\begin{array}{c}
 \begin{array}{c} A_1 \quad A_2 \quad A_3 \quad A_4 \quad A_5 \\
 \begin{array}{c} q_1 \begin{bmatrix} 0 & 1 & 1 & 0 & 1 \end{bmatrix} \\
 q_2 \begin{bmatrix} 1 & 1 & 1 & 0 & 1 \end{bmatrix} \\
 q_3 \begin{bmatrix} 1 & 0 & 0 & 1 & 1 \end{bmatrix} \\
 q_4 \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \end{bmatrix} \\
 q_5 \begin{bmatrix} 1 & 1 & 1 & 0 & 0 \end{bmatrix} \end{array} \end{array} \quad (a)
 \end{array}$$

$$\begin{array}{c}
 \begin{array}{c} S_1 \quad S_2 \quad S_3 \\
 \begin{array}{c} q_1 \begin{bmatrix} 10 & 20 & 0 \end{bmatrix} \\
 q_2 \begin{bmatrix} 5 & 0 & 10 \end{bmatrix} \\
 q_3 \begin{bmatrix} 0 & 35 & 5 \end{bmatrix} \\
 q_4 \begin{bmatrix} 0 & 10 & 0 \end{bmatrix} \\
 q_5 \begin{bmatrix} 0 & 15 & 0 \end{bmatrix} \end{array} \end{array} \quad (b)
 \end{array}$$

根据公式 $aff(A_i, A_j) = \sum_{kluse(qk, Ai)=1 \wedge kluse(qk, Aj)=1} \sum_{\forall site_l} ref_l(q_k) acc_l(q_k)$

$$aff(A_1, A_1) = \sum_{k=2,3,5} \sum_{l=1}^3 acc_l(q_k) = acc_1(q_2) + acc_3(q_2) + acc_2(q_3) + acc_3(q_3) + acc_2(q_5) = 70$$

$$aff(A_1, A_2) = \sum_{k=2,5} \sum_{l=1}^3 acc_l(q_k) = acc_1(q_2) + acc_3(q_2) + acc_2(q_5) = 30$$

$$aff(A_1, A_3) = \sum_{k=2,5} \sum_{l=1}^3 acc_l(q_k) = acc_1(q_2) + acc_3(q_2) + acc_2(q_5) = 30$$

\vdots

$$aff(A_5, A_4) = \sum_{k=3} \sum_{l=1}^3 acc_l(q_k) = acc_2(q_3) + acc_3(q_3) = 40$$

$$aff(A_5, A_5) = \sum_{k=1}^3 \sum_{l=1}^3 acc_l(q_k) = acc_1(q_1) + acc_2(q_1) + acc_3(q_1) + acc_1(q_2) + acc_3(q_2) + acc_2(q_3) + acc_3(q_3) = 85$$

从而得到了本题的 AA 矩阵：

$$\begin{array}{c}
 \begin{array}{c} A_1 \quad A_2 \quad A_3 \quad A_4 \quad A_5 \\
 \begin{array}{c} A_1 \begin{bmatrix} 70 & 30 & 30 & 40 & 55 \end{bmatrix} \\
 A_2 \begin{bmatrix} 30 & 60 & 60 & 0 & 45 \end{bmatrix} \\
 A_3 \begin{bmatrix} 30 & 60 & 70 & 0 & 45 \end{bmatrix} \\
 A_4 \begin{bmatrix} 40 & 0 & 0 & 40 & 0 \end{bmatrix} \\
 A_5 \begin{bmatrix} 55 & 45 & 45 & 0 & 85 \end{bmatrix} \end{array} \end{array}
 \end{array}$$

应用 BEA 算法由现有的 AA 矩阵，计算 CA 矩阵，过程如下：

1. 首先，从 AA 矩阵中任选两列放入 CA 矩阵中，这里我们选第一列和第二列，得到初始 CA 矩阵如图 5.6.1 (a) 所示。
2. 将第 A_k 列，根据 BEQ 算法插入到 CA 矩阵中。插入方法为：
对每个可插入的位置根据公式：

$$cont(A_i, A_k, A_j) = 2bond(A_i, A_k) + 2bond(A_k, A_j) - 2bond(A_i, A_j) \quad (1.1)$$

$$bond(A_x, A_y) = \sum_{z=1}^n aff(A_z, A_x)aff(A_z, A_y) \quad (1.2)$$

计算其 $cont()$ 函数值，选择其中最大的为插入点。

3. 插入所有列后，对得到的 CA 矩阵的行序参照其列序进行调整，最终得到 CA 矩阵。
现以将 A_3 列插入到 CA 中来说明第二步的计算过程。

顺序 (0-3-1)

$$cont(A_0, A_3, A_1) = 2bond(A_0, A_3) + 2bond(A_3, A_1) - 2bond(A_0, A_1)$$

其中

$$bond(A_0, A_1) = bond(A_0, A_3) = 0$$

$$bond(A_3, A_1) = 30 \times 70 + 60 \times 30 + 70 \times 30 + 0 \times 40 + 45 \times 55 = 8475$$

$$\text{因此 } cont(A_0, A_3, A_1) = 16950$$

顺序 (1-3-2)

$$cont(A_1, A_3, A_2) = 2bond(A_1, A_3) + 2bond(A_3, A_2) - 2bond(A_1, A_2)$$

$$bond(A_1, A_3) = bond(A_3, A_1) = 8475$$

$$bond(A_3, A_2) = 10725$$

$$bond(A_1, A_2) = 6375$$

$$\text{因此 } cont(A_1, A_3, A_2) = 25650$$

顺序 (2-3-4)

$$cont(A_2, A_3, A_4) = 2bond(A_2, A_3) + 2bond(A_3, A_4) - 2bond(A_2, A_4)$$

其中

$$bond(A_2, A_3) = bond(A_3, A_2) = 10725$$

$$bond(A_3, A_4) = 1200$$

$$bond(A_2, A_4) = 1200$$

$$\text{因此 } cont(A_2, A_3, A_4) = 21450$$

其中最大的是顺序 (1-3-2)，因此将 A_3 插到 A_1 和 A_2 之间。我们得到如图 5.6.1(b) 的 CA 矩阵。
对其他列做相似操作，我们就得到了如图 5.6.1(c) 的 CA 矩阵。在对这个的得到的 CA 矩阵进行第 3 步的调整，我们就得到了最终的 CA 矩阵，如图 5.6.1(d)。

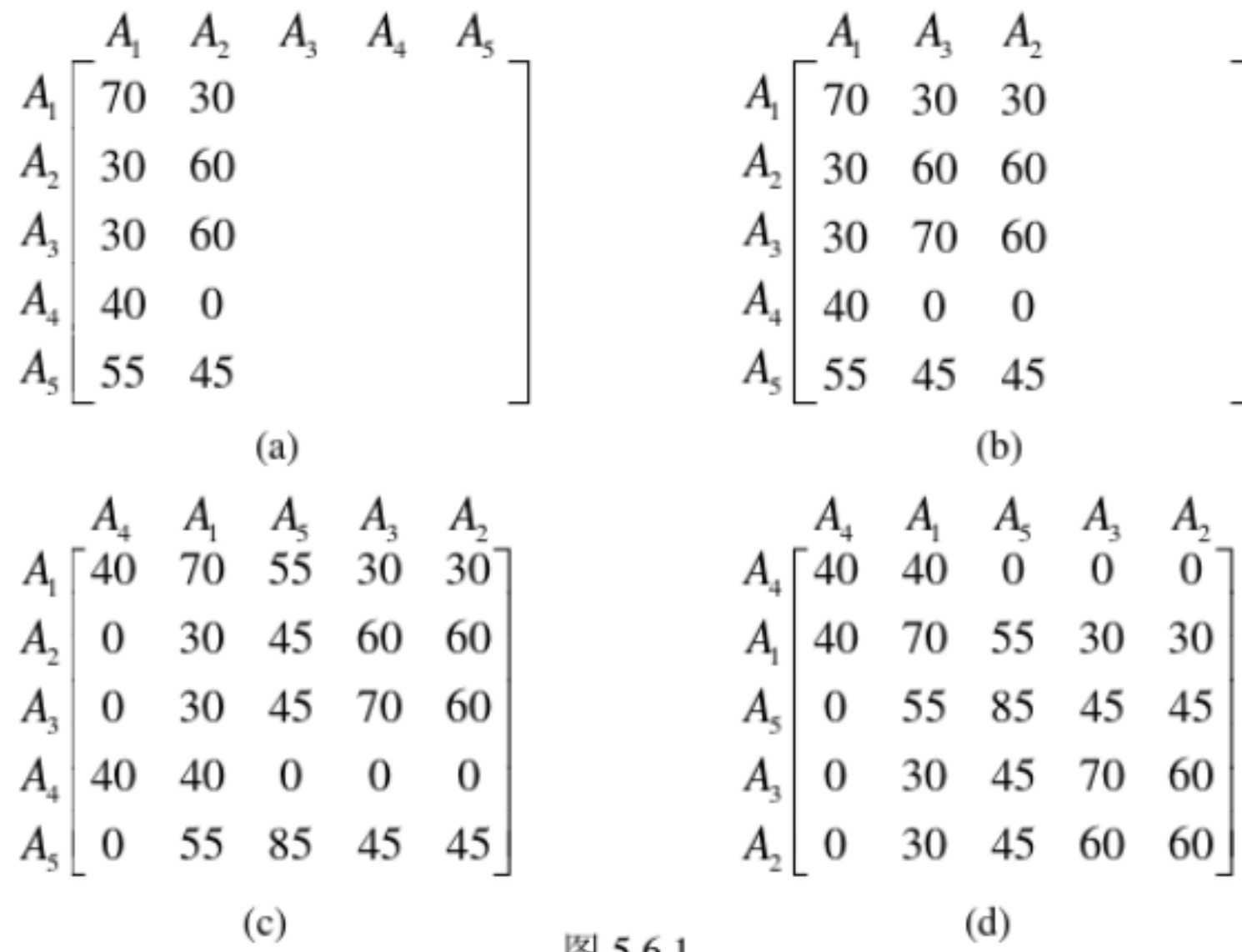


图 5.6.1

然后，我们再根据 Partitioning 算法对 CA 矩阵进行计算，找出最优划分点。最优划分点就是使方程： $z = CTQ * CBQ - COQ^2$ 最大的点，其中

$$CTQ = \sum_{q_i \in TQ} \sum_{\forall S_j} ref_j(q_i) acc_j(q_i)$$

$$CBQ = \sum_{q_i \in BQ} \sum_{\forall S_j} ref_j(q_i) acc_j(q_i)$$

$$COQ = \sum_{q_i \in OQ} \sum_{\forall S_j} ref_j(q_i) acc_j(q_i)$$

在本题中共有 4 个待计算点，其计算过程如下：

$n = 1$ 时

$$TQ_1 = \emptyset; \quad BQ_1 = \{q_1, q_2, q_4, q_5\}; \quad OQ_1 = \{q_3\}.$$

$$z_1 = CTQ_1 * CBQ_1 - COQ_1^2 = 0 * (30 + 15 + 10 + 15) - 40^2 = -1600$$

$n = 2$ 时

$$TQ_2 = \emptyset; \quad BQ_2 = \{q_1, q_4\}; \quad OQ_2 = \{q_2, q_3, q_5\}.$$

$$z_2 = CTQ_2 * CBQ_2 - COQ_2^2 = 0 * (30 + 10) - (15 + 40 + 15)^2 = -4900$$

$n = 3$ 时

$$TQ_3 = \{q_3\}; \quad BQ_3 = \{q_4\}; \quad OQ_3 = \{q_1, q_2, q_5\}.$$

$$z_3 = CTQ_3 * CBQ_3 - COQ_3^2 = 40 * 10 - (30 + 15 + 15)^2 = -3200$$

$n = 4$ 时

$$TQ_4 = \{q_3, q_4\}; \quad BQ_4 = \emptyset; \quad OQ_4 = \{q_1, q_2, q_5\}.$$

$$z_4 = CTQ_4 * CBQ_4 - COQ_4^2 = (40 + 10) * 0 - (30 + 15 + 15)^2 = -3600$$

因此最佳划分点在 $n=1$ 处，形成的分片集合为

$F = \{R_1, R_2\}$ 其中 $R_1 = \{A_1, A_4\}, R_2 = \{A_1, A_2, A_3, A_5\}$.

5.9

完整性：根据引用完整性，member 关系的任一个元组中的外键一定在 owner 关系中存在。
按外键相等的条件连接时，一定不会丢失 member 关系中的元组。

重构性：对于关系 R， $FR = \{R_1, R_2, \dots, R_w\}$ $R = \bigcup R_i$ 对于所有的 $R_i \in FR$

互斥性：对于诱导水平分片，分片时使用了半连接，增加了问题的复杂性。如果连接图是简单的（如果只有一个连接进入一个分片或从一个分片发出），则诱导水平分片满足这个规则。

第八章

8.1

根据题中查询得到如下谓词表达式：

p_1 : $RESP = \text{"Analyst"}$; p_2 : $PNO = \text{"P2"}$; p_3 : $DUR = 12$;

查询条件可以表示为如下合取范式：

$$p_1 \wedge \neg(p_2 \vee p_3) \wedge \neg p_2 \wedge p_3$$

通过幂等率对其简化，过程如下：

$$p_1 \wedge \neg(p_2 \vee p_3) \wedge \neg p_2 \wedge p_3$$

$$= p_1 \wedge \neg p_2 \wedge \neg p_3 \wedge \neg p_2 \wedge p_3$$

$$= p_1 \wedge (\neg p_2 \wedge \neg p_2) \wedge (\neg p_3 \wedge p_3)$$

$$= p_1 \wedge \neg p_2 \wedge false$$

$$= false$$

因此，原查询条件永假。

SELECT	ENO
FROM	ASG
WHERE	RESP = "Analyst"
AND	NOT(PNO="P2" OR DUR=12)
AND	PNO ≠ "P2"
AND	DUR=12

8.2

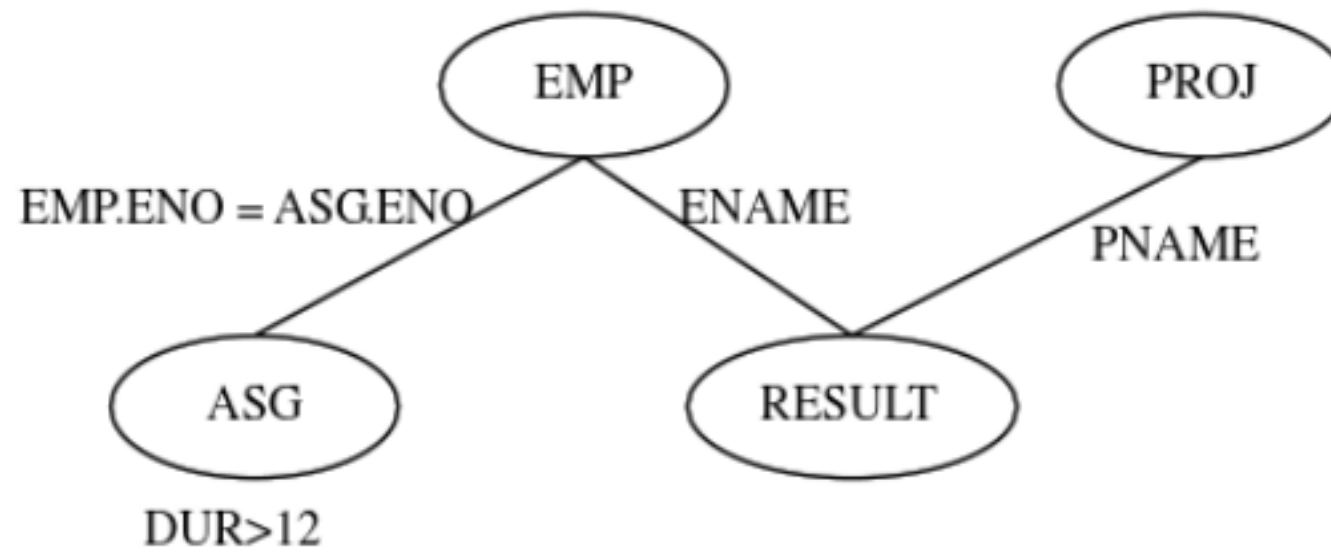


图 8.2.1 Query Graph

$\Pi_{NAME, PNAME} (PROJ \bowtie_{PNO} ((\sigma_{DUR > 12} ASG) \bowtie_{ENO} EMP))$

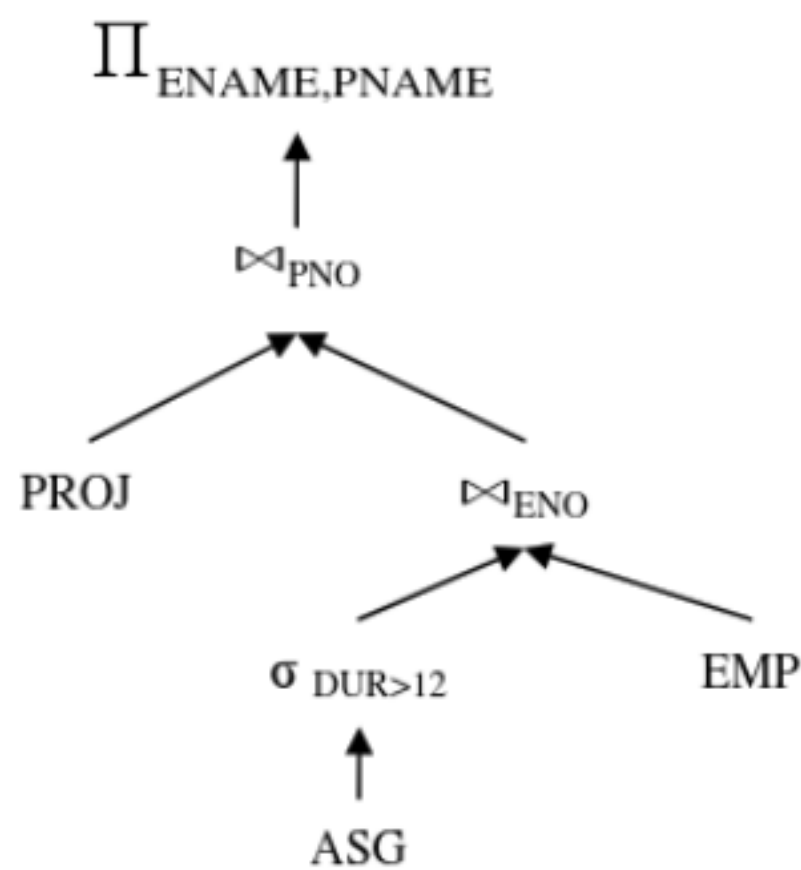


图 8.2.2 Operator Tree

8.6

根据题中的 SQL 查询语句，得到图 8.6(a)的操作树；根据公式 $(R_1 \cup R_2) \bowtie S = (R_1 \bowtie S) \cup (R_2 \bowtie S)$

将连接操作下移，再根据分片信息将空关系 $PROJ_1 \bowtie (\sigma_{PNO=4} ASG)$ 去掉，得到图 8.6(b) 的操作树；再将投影操作下移以减少产生的中间结果的规模，得到最终的优化操作树如图 8.6(c)。

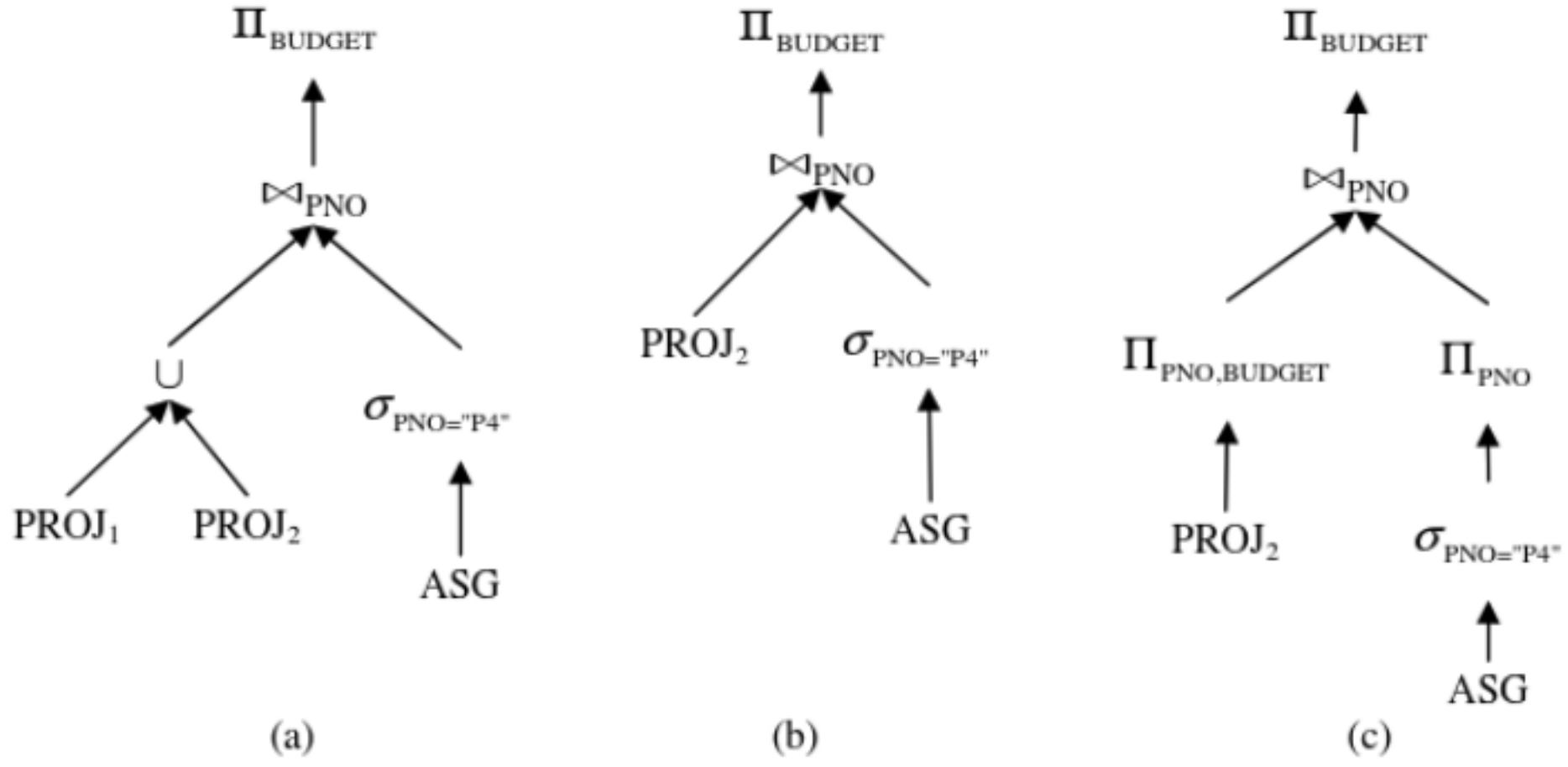


图 8.6

8.8

$PROJ_1 = \sigma_{PNO \leq "P2"}(ASG)$
 $PROJ_2 = \sigma_{PNO > "P2"}(ASG)$
 $ASG_1 = ASG \bowtie PROJ_1$
 $ASG_2 = ASG \bowtie PROJ_2$
 $EMP_1 = \Pi_{ENO, ENAME}(EMP)$
 $EMP_2 = \Pi_{ENO, TITLE}(EMP)$

SELECT ENAME
 FROM EMP, ASG, PROJ
 WHERE PROJ.PNO = ASG.PNO
 AND PNAME = "Instrumentation"
 AND EMP.ENO = ASG.ENO

$\Pi_{ENAME}((EMP_1 \bowtie_{ENO} EMP_2) \bowtie_{ENO} ((ASG_1 \cup ASG_2) \bowtie_{PNO} \sigma_{PNAME="Instrumentation"}(PROJ_1 \cup PROJ_2)))$

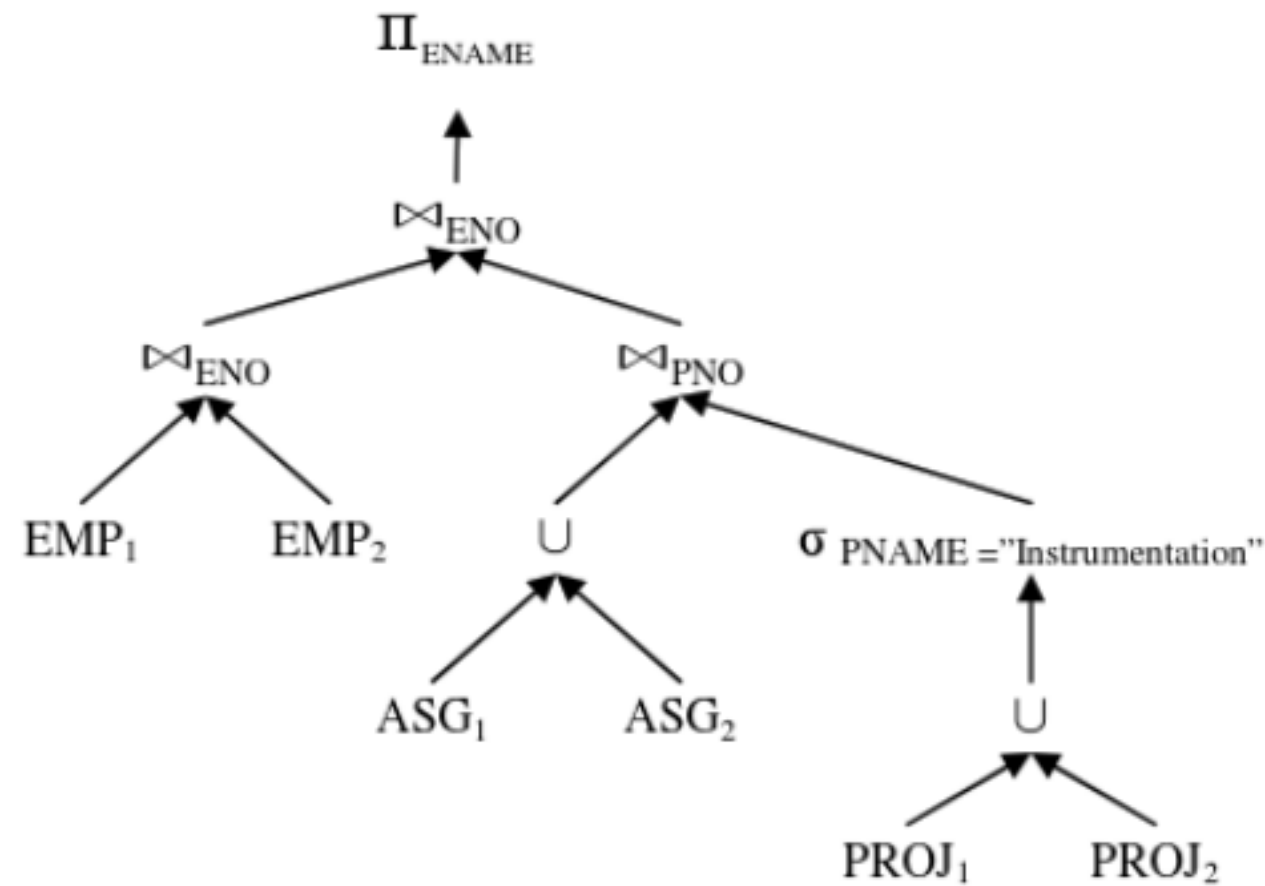


图 8.8.1 Generic query

$$\Pi_{ENAME}((EMP_1 \bowtie_{ENO} EMP_2) \bowtie_{ENO} (((\sigma_{PNAME="Instrumentation"} PROJ_1) \bowtie_{PNO} ASG_1) \cup ((\sigma_{PNAME="Instrumentation"} PROJ_2) \bowtie_{PNO} ASG_2)))$$

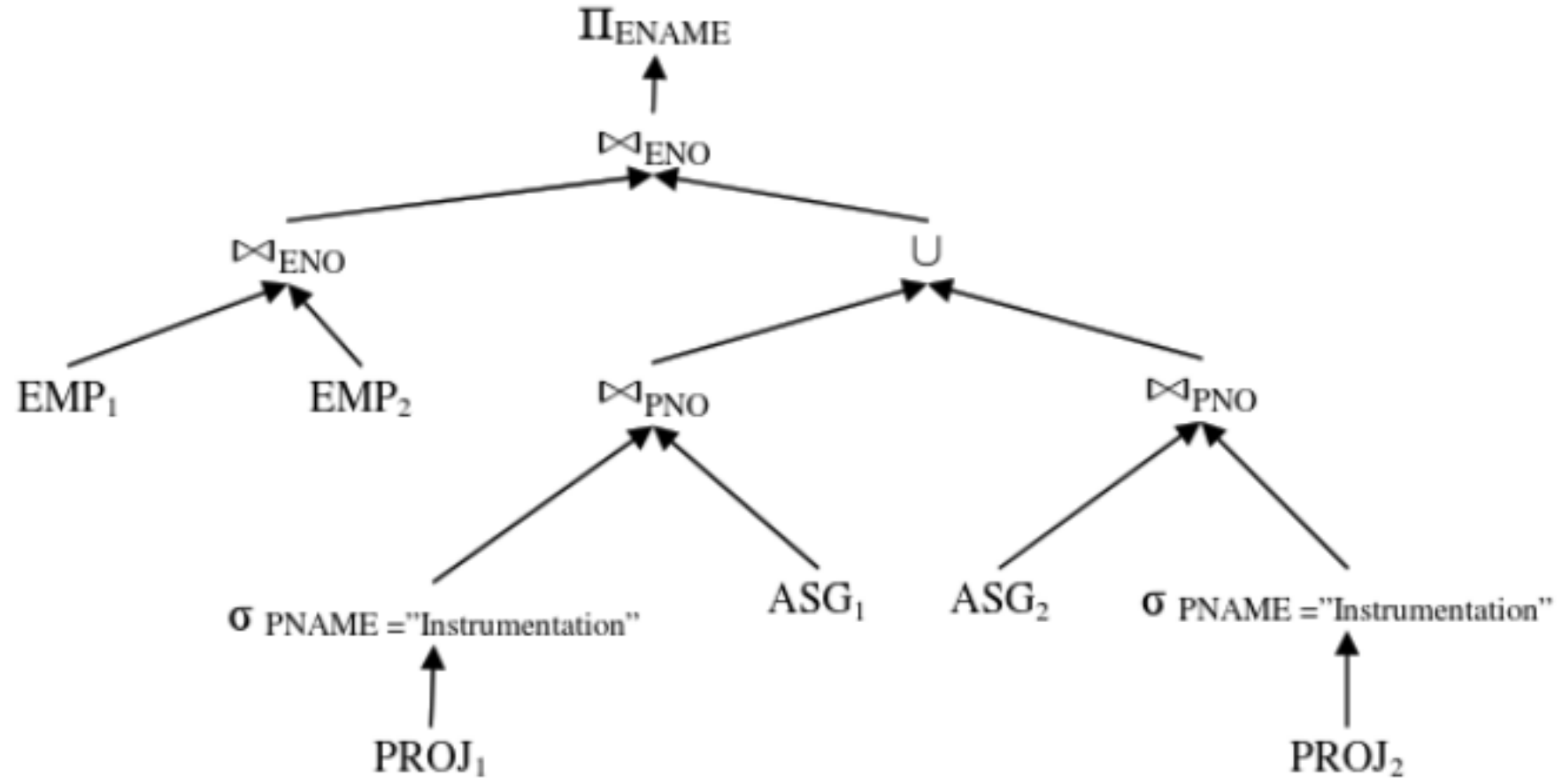


图 8.8.2 Reduced query after unions up

$$\Pi_{ENAME} (EMP_1 \bowtie_{ENO} (\Pi_{ENO} (\Pi_{PNO} (\sigma_{PNAME="Instrumentation"} PROJ_1) \bowtie_{PNO} \Pi_{PNO, ENO} ASG_1) \cup \Pi_{ENO} (\Pi_{PNO} (\sigma_{PNAME="Instrumentation"} PROJ_2) \bowtie_{PNO} \Pi_{PNO, ENO} ASG_2)))$$

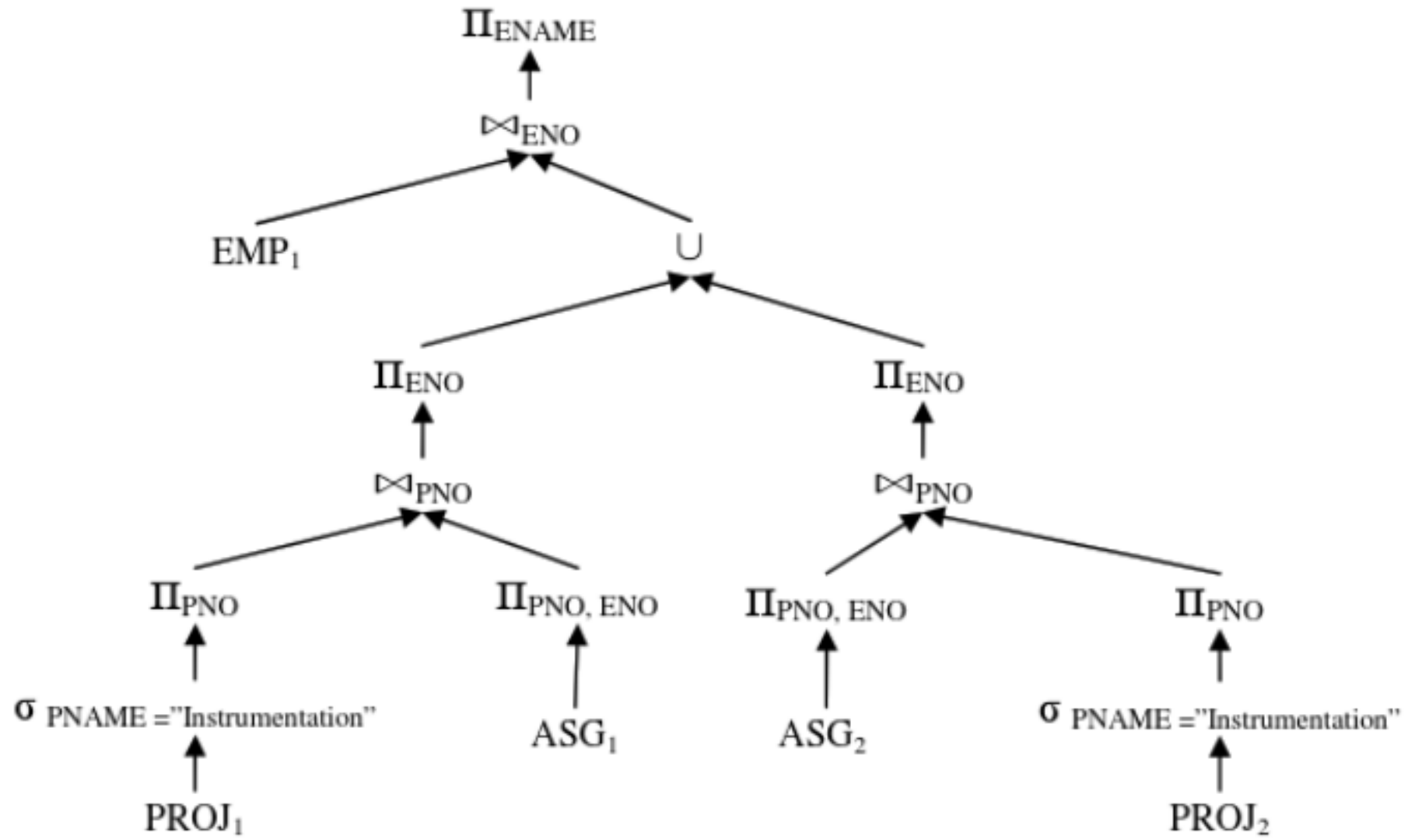


图 8.8.3 Reduced query after pushing projection

根据题中 SQL 查询语句，首先得到如图 8.8.1 所示的普通查询操作树；首先将并操作提前，根据 ASG 的分片信息可知得到的关系 $ASG_1 \bowtie_{ENO} (\sigma_{PNAME="Instrumentation"} PROJ_2)$ 和关系 $ASG_2 \bowtie_{ENO} (\sigma_{PNAME="Instrumentation"} PROJ_1)$ 是空的，因此将这两个空关系删除，从而得到如图 8.8.2 的查询操作树；然后在将投影操作下移，根据 EMP 的分片信息可知关系 $\Pi_{ENAME, ENO}(EMP_1 \bowtie_{ENO} EMP_2) = EMP_1$ ，因此查询操作树得到进一步简化，最终我们得到了如图 8.8.3 所示的经过简化和优化的查询操作树。

8.3

将题中所给出的 SQL 查询语句转化为如下关系代数表达式：

$$\Pi_{ENAME, PNAME} (\sigma_{(TITLE = "Elect.Eng.") \vee (PNO < "P3") \wedge (DUR > 12)} (EMP \bowtie ASG \bowtie PROJ))$$

$$\Pi_{ENAME, PNAME} ((\sigma_{(TITLE = "Elect.Eng.") \wedge (DUR > 12)} (EMP \bowtie ASG \bowtie PROJ)) \cup (\sigma_{(PNO < "P3") \wedge (DUR > 12)} (EMP \bowtie ASG \bowtie PROJ)))$$

$$\Pi_{ENAME, PNAME} (((\sigma_{TITLE = "Elect.Eng."} (EMP)) \bowtie (\sigma_{DUR > 12} ASG)) \bowtie PROJ) \cup ((\sigma_{(PNO < "P3") \wedge (DUR > 12)} (ASG)) \bowtie EMP \bowtie PROJ))$$

$$\Pi_{ENAME, PNAME} ((\Pi_{ENAME, PNO} (\Pi_{ENAME, ENO} (\sigma_{TITLE = "Elect.Eng."} (EMP)) \bowtie_{ENO} \Pi_{ENO, PNO} (\sigma_{DUR > 12} ASG))) \cup \Pi_{ENAME, PNO} (\Pi_{ENO, PNO} (\sigma_{(PNO < "P3") \wedge (DUR > 12)} (ASG)) \bowtie_{ENO} \Pi_{ENAME, ENO} EMP)) \bowtie_{PNO} \Pi_{PNAME, PNO} PROJ)$$

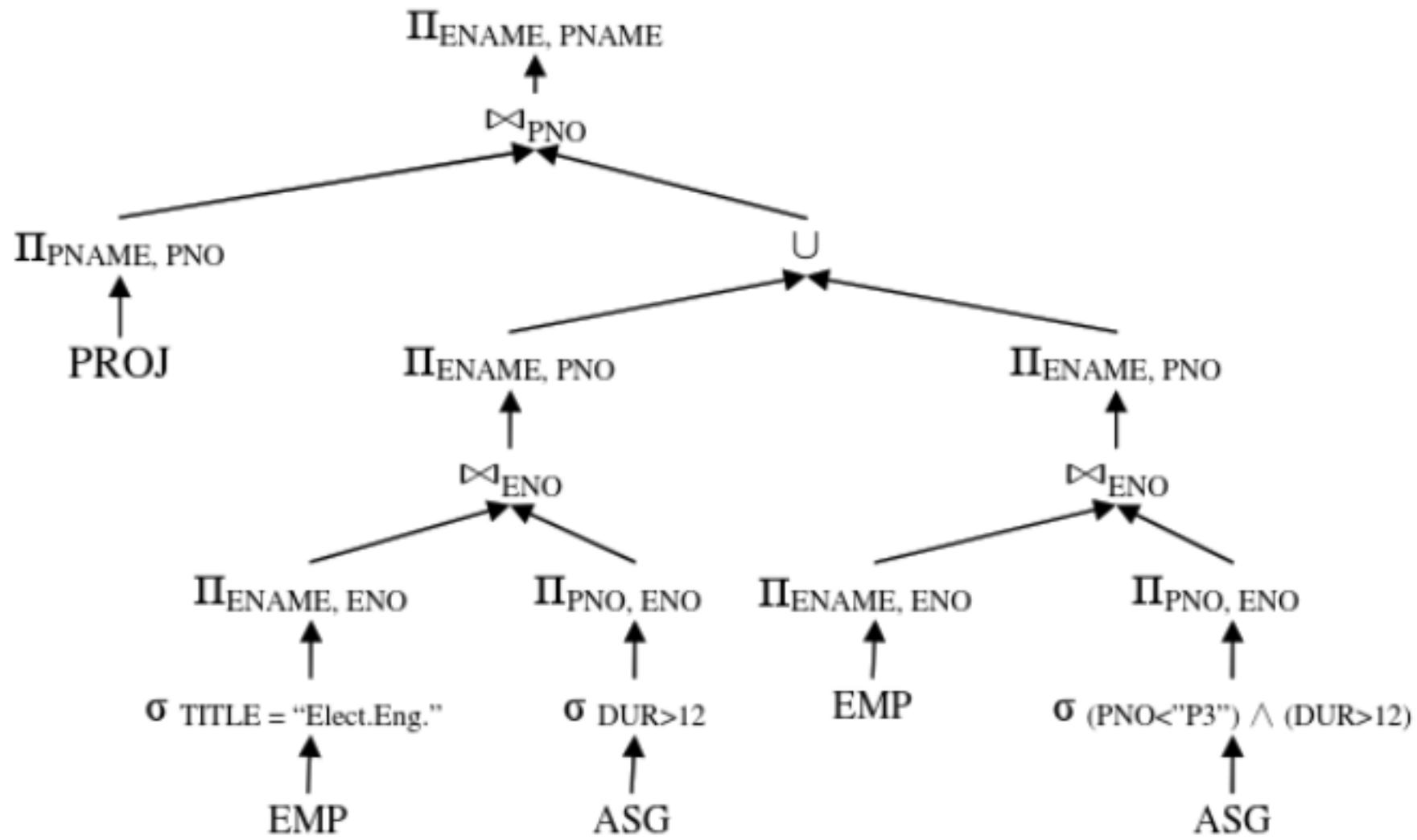


图 8.3.1 优化后的查询操作树

8.5

题中所给的 SQL 查询语句可以映射成如下关系代数表达式：

$$\Pi_{ENAME, SAL} (\sigma_{(BUDGET > 200000) \vee (DUR > 24)} (PAY \bowtie EMP \bowtie ASG \bowtie PROJ))$$

$$\Pi_{ENAME, SAL} (\sigma_{(BUDGET > 200000)} (PAY \bowtie EMP \bowtie ASG \bowtie PROJ) \cup \sigma_{(DUR > 24)} (PAY \bowtie EMP \bowtie ASG \bowtie PROJ))$$

$$\Pi_{ENAME, SAL} (\Pi_{ENAME, TITLE} (\Pi_{ENO} (\Pi_{PNO} (\sigma_{BUDGET > 200000} (PROJ)) \bowtie_{PNO} (\Pi_{PNO, ENO} ASG)) \bowtie_{ENO} EMP) \bowtie_{TITLE} PAY) \cup \Pi_{ENAME, SAL} (\Pi_{ENAME, TITLE} (\Pi_{ENO} (\sigma_{DUR > 24} (ASG)) \bowtie_{ENO} EMP) \bowtie_{TITLE} PAY))$$

$$\Pi_{ENAME, SAL} (\Pi_{ENAME, TITLE} ((\Pi_{ENO} (\Pi_{PNO} (\sigma_{BUDGET > 200000} (PROJ)) \bowtie_{PNO} (\Pi_{PNO, ENO} ASG)) \cup \Pi_{ENO} (\sigma_{DUR > 24} (ASG))) \bowtie_{ENO} EMP) \bowtie_{TITLE} PAY)$$

第九章

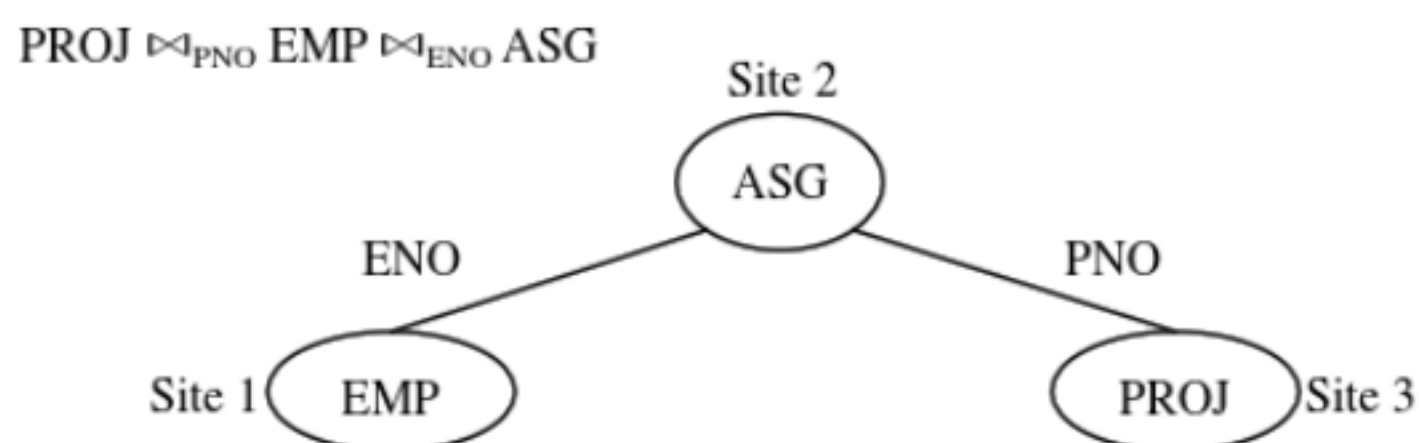


Figure 9.10. Join Graph of Distributed Query

9.2 $size(EMP) = 100$, $size(ASG) = 200$, $size(PROJ) = 300$, $size(EMP \bowtie ASG) = 300$ and $size(ASG \bowtie PROJ) = 200$

应用动态规划(Dynamic programming)进行求解:

1. 首先选择 $size$ 最小的关系,从题中给的数据我们可以看出是 EMP 关系 $size(EMP) = 100$,它在 Site1 上,可以将它送到 Site2 或 Site3 上,这是总的传输时间 $Total_time = T_{MSG} + T_{TR} * 100$

2. 如果将 EMP 传送到 Site2,并在 Site2 上与 ASG 进行连接操作得到的结果的 $size$ 为 $size(EMP \bowtie ASG) = 300$ 。此时可以做的传输为: a)将 $(EMP \bowtie ASG)$ 传输到 Site3; b)将 Site3 上的 $PROJ$ 传输到 Site2。而由于 $size(EMP \bowtie ASG) = size(PROJ) = 300$,因此无论此时如何传输最终的总体传输时间都相同 $Total_time = 2 * T_{MSG} + T_{TR} * (100 + 300) = 2 * T_{MSG} + T_{TR} * 400$ 。

3. 如果将 EMP 传输到 Site3,由于 EMP 无法同 $PROJ$ 进行连接操作,因此这时的可用传输为 a)将 Site2 上的 ASG 传输到 Site3,因为 $size(ASG) = 200$,因此这种情况下的总体传输时间 $Total_time = 2 * T_{MSG} + T_{TR} * (100 + 200) = 2 * T_{MSG} + T_{TR} * 300$; b)将 Site3 上的 $PROJ$ 和 EMP 全部传输到 Site2,这时的总体传输时间 $Total_time = 2 * T_{MSG} + T_{TR} * (100 + 100 + 300) = 2 * T_{MSG} + T_{TR} * 500$ 。

综上,我们找到了最优传输方案:

$EMP \rightarrow Site3, ASG \rightarrow Site3$, 这时的总体传输时间 $Total_time = 2 * T_{MSG} + T_{TR} * 300$;



图 9.2

9.3

由于题目要求重响应时间最短,因此应该提高数据传输的并行度。因此应采用 $(ASG, EMP) \rightarrow Site3; (ASG, PROJ) \rightarrow Site1; (EMP, PROJ) \rightarrow Site2$ 中的一种。又由于

$\max(size(ASG), size(EMP)) = size(ASG) = 200$; $\max(size(ASG), size(PROJ)) = size(PROJ) = 300$;
 $\max(size(EMP), size(PROJ)) = size(PROJ) = 300$

因此选用方案 $(ASG, EMP) \rightarrow Site3$, 这时的总反应时间最小:

$$Response_time = T_{MSG} + T_{TR} * \max(size(ASG), size(EMP)) = T_{MSG} + T_{TR} * 200$$

第十一章

$$\begin{aligned} S_1 &= W_2(x), W_1(x), R_3(x), R_1(x), C_1, W_2(y), R_3(y), R_3(z), C_3, R_2(z), C_2 \\ S_2 &= R_3(z), R_3(y), W_2(y), R_2(z), W_1(x), R_3(x), W_2(x), R_1(x), C_1, C_2, C_3 \\ S_3 &= R_3(z), W_2(x), W_2(y), R_1(x), R_3(x), R_2(z), R_3(y), C_3, W_1(x), C_2, C_1 \\ S_4 &= R_2(z), W_2(x), W_2(y), C_2, W_1(x), R_1(x), A_1, R_3(x), R_3(z), R_3(y), C_3 \end{aligned}$$

11.1

其中 S_3, S_4 冲突等价(*conflict equivalent*), 拥有共同的冲突操作偏序关系:

$$W_2(x) \prec R_3(x), W_2(y) \prec R_3(y)$$

S_1, S_4 也冲突等价, 拥有共同的冲突等价操作偏序关系:

$$W_2(x) \prec R_3(x), W_2(y) \prec R_3(y)$$

但 S_1, S_3 并不冲突等价。

11.2

其中 S_1, S_4 可串行化:

S_1 等价与串行化操作: $T_2 \rightarrow T_1 \rightarrow T_3$; S_4 等价与串行化操作: $T_2 \rightarrow T_3$

11.3

对于执行计划:

$$S = R_1(x), R_1(y), W_1(x), R_2(x), W_1(y), C_1, W_2(x), C_2$$

这个执行计划可以被基本 2PL 接受, 但不能被 S2PL 接受。其被基本 2PL 接受后的加/解锁为:

$$S = wl_1(x), R_1(x), wl_1(y), R_1(y), W_1(x), lr_1(x), wl_2(x), R_2(x), W_1(y), lr_1(y), C_1, W_2(x), lr_2(x), C_2$$

修订信息:

1/13/2007 12:11:32 PM

11.1 $W_2(x) \prec R_3(x), W_2(y) \prec R_3(y)$ 符号更正

11.1 $rl_1(x)$ 更正为 $wl_2(x)$

$S = wl_1(x), R_1(x), wl_1(y), R_1(y), W_1(x), lr_1(x), wl_2(x), R_2(x), W_1(y), lr_1(y), C_1, W_2(x), lr_2(x), C_2$

8.3 修正无 $DUR > 12$ 选择条件的错误

1/14/2007 11:13:09 PM

5.3 推荐使用第二个方案

5.5 更正原来的完整性说明, 互斥性说明和重构性说明

5.6 BEQ 更正为 BEA

5.6 更正公式 1.1 最后一项, 添加系数 2

8.8 对第二步和第三步的优化逻辑进行了修改, 将并操作下调。

8.3 删除化简第三步, 更正最后的执行操作树

8.5 增加一步优化步骤

致谢

感谢徐皓老师的悉心指导。

下面是同徐皓老师通信记录及徐皓老师对原答案指出的错误：

谢龙，你好！

我看到8.3题，还没看完。先给你发一个邮件，将已看过的题的想法告诉你。全部看完后，再给你发一个邮件。

一、5.3题的第二个方案可能更合适（图5.3.2）。因为第一个方案中PAY关系的KEY是TITLE，PAY关系做诱导分片后，其每个分片只剩下一个元组。

二、5.5题的完整性说明：对于EMP的诱导水平分片，根据引用完整性，member关系(EMP)的任一个元组中的外键(TITLE)一定在owner(PAY)关系中存在。按外键(TITLE)相等的条件连接时，一定不会丢失 member(EMP)关系中的元组。

5.5题的互斥性说明：对于诱导水平分片，如果连接图是简单的
则诱导水平分片满足互斥性。

1. PAY1, PAY2的交集为空。

2. 每个雇员仅有一个TITLE。

3. 每个TITLE仅有一个SAL。

此时EMP的分片是互斥的。

假设一个雇员可以有多个TITLE，一个TITLE有多个SAL，则可能出现不满足Disjointness的情况。

三、5.6题BEA，误写为BEQ。公式1.1的第三项少了一个系数2。

四、8.8题的图8.8.2，我觉得不用合并上移，即不用把合并移到根节点。如果这样做，就把“EMP1连接EMP2”分配到左、右两个子树中，但是并没有在两个子树中得到缩减。而在每个子树中都得与EMP1连接一次（图8.8.3）。

徐皓 2007年1月14日