

IS

```
CURSOR c_vendite IS
SELECT L1.Id Id1, L2.Id Id2, count(*) Conta
FROM ACQUISTI A1, ACQUISTI A2, LIBRI L1, LIBRI L2
WHERE A1.Cliente=A2.Cliente AND A1.Libro=L1.Id AND A2.Libro=L2.Id
      AND L1. Argomento=L2.Argomento AND A1.Libro <> L2.Id;
GROUP BY L1.Id, L2.Id;
/*coppie di libri diversi, ma dello stesso argomento, acquistati dallo stesso
cliente.*/
```

```
v_cur c_vendite%ROWTYPE;
v_acq NUMBER;
vlivello CORRELAZIONE.R_Livello%TYPE;
BEGIN
  OPEN c_vendite;
  LOOP
    FETCH c_vendite INTO v_cur;
    EXIT WHEN c_vendite %NOTFOUND;
    SELECT count(*) INTO v_acq
    FROM ACQUISTI A
    WHERE Libro=v_cur.Id1;

    /*Assumo che tutte le coppie di libri siano già presenti nel DB.*/
    vlivello := v_cur.Conta / v_acq;
    UPDATE CORRELAZIONE
    SET Livello=vlivello
    WHERE Libro = v_cur.Id1 AND LibroCorr = v_cur.Id2;
  END LOOP;
  CLOSE c_vendite;
END;
```

```

explain plan for select L_PARTKEY,AVG(L_RECEIPTDATE-O_ORDERDATE)
from ORDERS,LINEITEM,CUSTOMER
where L_ORDERKEY=O_ORDERKEY and O_CUSTKEY=C_CUSTKEY and
C_NAME='Customer#000000016'
group by L_PARTKEY;
@?/RDBMS/ADMIN/UTLXPLS;

```

Plan Table

Operation	Name	Rows	Bytes	Cost	Pstart	Pstop
SELECT STATEMENT		120	6K	128		
SORT GROUP BY		120	6K	128		
NESTED LOOPS		120	6K	124		
NESTED LOOPS		30	1K	34		
TABLE ACCESS BY INDEX	CUSTOMER	2	48	2		
INDEX RANGE SCAN	INDXNAME	2		1		
TABLE ACCESS BY INDEX	ORDERS	1M	18M	16		
INDEX RANGE SCAN	IX_CUST_O	1M		2		
TABLE ACCESS BY INDEX R	LINEITEM	6M	91M	3		
INDEX RANGE SCAN	IX_ORDER_	6M		2		

Il piano di esecuzione prevede inizialmente un nested loop con CUSTOMER esterna. La tabella esterna viene acceduta tramite indice unclustered sul campo C_NAME, anche l'accesso a quella esterna è tramite indice unclustered sul campo O_CUSTKEY.

$$NP_{Orders} = \lceil 1.500.000 \times 106 / (4096 \times 0,69) \rceil = 56.259$$

$$NP_{Customer} = \lceil 150.000 \times 158 / (4096 \times 0,69) \rceil = 8.386$$

$$NL_{C_NAME} = \lceil (150.000 \times 25 + 150.000 \times 4) / (4096 \times 0,69) \rceil = 1.540$$

$$NL_{O_CUSTKEY} = \lceil (150.000 \times 4 + 4 \times 1.500.000) / (4096 \times 0,69) \rceil = 2.336$$

$$\text{Costo di accesso a CUSTOMER: } h-1 + \lceil \text{sel(pred)} \times NL \rceil + EK \times \Phi(NT / NK, NP) = 2-1 + \lceil 1/150.000 \times 1.540 \rceil + 1 \times \Phi(150.000 / 150.000, 8.386) = 1 + 1 + 1 = \mathbf{3}$$

$$\text{Costo di accesso a ORDERS: } h-1 + \lceil \text{sel(pred)} \times NL \rceil + EK \cdot \Phi(NT / NK, NP) = 2-1 + \lceil 1/150.000 \times 2.336 \rceil + 1 \times \Phi(1.500.000 / 150.000, 56.259) = 1 + 1 + 10 = \mathbf{12}$$

$$\text{Costo del I}^\circ \text{ join} = \mathbf{3} + \mathbf{1} \times \mathbf{12} = \mathbf{15}$$

$$NP_{ORDERS+CUSTOMER} = \lceil 10 \times (106+158) / (4096 \times 0,69) \rceil = \mathbf{1}$$

$$NP_{LINEITEM} = \lceil 6.001.215 \times 113 / (4.096 \times 0,69) \rceil = 239.944$$

$$NL_{L_ORDERKEY} = \lceil (1.500.000 \times 4 + 4 \times 6.001.215) / (4096 \times 0,69) \rceil = 10.617$$

Accesso a LINEITEM

$$= 2-1 + \lceil 1/1.500.000 \times 10.617 \rceil + 1 \times \Phi(6.001.215 / 1.500.000, 239.944) = 1 + 1 + 4 = \mathbf{6}$$

$$\text{Costo del II}^\circ \text{ join (senza pipeline)} = \mathbf{1} + \mathbf{10} \times \mathbf{6} = \mathbf{61}$$

$$NP_{\text{ORDERS+CUSTOMER+LINEITEM}} = \lceil 6.001.215 / 1.500.000 \times 10 \times (106+158+113) / (4096 \times 0,69) \rceil = 6$$

Il sort necessario per il group by può essere svolto in memoria (6 < 100 buffer) e quindi, supponendo che il sistema non utilizzi la pipeline, richiede una lettura e una scrittura della relazione

$$\text{Costo totale} = 15 + 61 + 2 \times 6 = 88$$