

# B561 – Solutions for Assignment 3

Dirk Van Gucht & Mathias Niepert

October 5, 2007

## List of Queries

Given are the following queries in natural language:

1. Get all shipments where the quantity is between 400 and 850 inclusive.
2. Get all pairs of city names such that a supplier in the first city supplies a project in the second city with quantity at least 200.
3. Get project numbers for projects supplied by at least one supplier not in the same city.
4. Get project names for projects supplied by supplier S3.
5. Get part numbers for parts supplied to any project in London.
6. Get supplier numbers for suppliers supplying at least one part supplied by at least one supplier who supplies at least one red part.
7. Get supplier numbers for suppliers with a status lower than that of supplier S1.
8. Get part numbers for parts supplied to all projects in London.
9. Get project numbers and cities where the city has an “l” as the first or an “a” as the second letter of its name.
10. Get project numbers for projects supplied with part P1 in an average quantity greater than the greatest quantity in which any part is supplied to project J1.
11. Get project numbers for projects supplied with at least all parts available from supplier S1.

12. Get project numbers for projects that are supplied by every supplier who supplies some red part.

## Solutions

- (a) Formulate the queries (1) to (12) in SQL.

- (1) 

```
SELECT SID, PID, JID, QTY
FROM SPJ
WHERE QTY>=400 AND QTY<=850;
```
- (2) 

```
SELECT DISTINCT SUPPLIER.CITY, PROJECT.CITY
FROM S as SUPPLIER, SPJ as SHIPMENT, J as PROJECT
WHERE SUPPLIER.SID=SHIPMENT.SID
AND SHIPMENT.JID=PROJECT.JID
AND SHIPMENT.QTY>=200;
```
- (3) 

```
SELECT DISTINCT PROJECT.JID
FROM SPJ as SHIPMENT, S as SUPPLIER, J as PROJECT
WHERE SUPPLIER.SID=SHIPMENT.SID
AND SHIPMENT.JID=PROJECT.JID
AND SUPPLIER.CITY<>PROJECT.CITY;
```
- (4) 

```
SELECT PROJECT.JNAME
FROM J as PROJECT
WHERE PROJECT.JID IN
(SELECT PROJECT.JID
FROM SPJ as SHIPMENT
WHERE SHIPMENT.SID='S3');
```
- (5) 

```
SELECT DISTINCT SHIPMENT.PID
FROM SPJ as SHIPMENT
WHERE SHIPMENT.JID IN
(SELECT SHIPMENT.JID
FROM J as PROJECT
WHERE PROJECT.CITY='London');
```
- (6) 

```
SELECT DISTINCT SHIPMENT.SID
FROM SPJ as SHIPMENT
WHERE SHIPMENT.PID IN
(SELECT SHIPMENT2.SID
FROM SPJ as SHIPMENT2
WHERE SHIPMENT2.SID IN
(SELECT SHIPMENT3.SID
```

```

FROM SPJ as SHIPMENT3
WHERE SHIPMENT3.PID IN
    (SELECT PROJECT.PID
     FROM P as PROJECT
     WHERE PROJECT.COLOR='Red')));
(7) SELECT SUPPLIER.SID
FROM S as SUPPLIER
WHERE STATUS <
    (SELECT SUPPLIER2.STATUS
     FROM S as SUPPLIER2
     WHERE SUPPLIER2.SID='S1');
(8) SELECT DISTINCT SHIPMENT.PID
FROM SPJ as SHIPMENT, SPJ as SHIPMENT2
WHERE NOT EXISTS
    (SELECT *
     FROM J as PROJECT
     WHERE PROJECT.CITY='London'
     AND NOT EXISTS
        (SELECT *
         FROM SPJ as SHIPMENT3, SPJ as SHIPMENT4
         WHERE SHIPMENT4.PID = SHIPMENT2.PID
         AND SHIPMENT4.JID = PROJECT.JID));
(9) SELECT PROJECT.JID, PROJECT.CITY
FROM J as PROJECT
WHERE PROJECT.CITY LIKE 'l%' OR PROJECT.CITY LIKE '_a%';
(10) SELECT SHIPMENT.JID
FROM SPJ as SHIPMENT
WHERE SHIPMENT.PID='P1'
GROUP BY SHIPMENT.JID
HAVING AVG (QTY) >
    (SELECT MAX(QTY)
     FROM SPJ as SHIPMENT2
     WHERE SHIPMENT2.JID='J1');
(11) SELECT DISTINCT SHIPMENT.JID
FROM SPJ as SHIPMENT, SPJ as SHIPMENT2
WHERE NOT EXISTS
    (SELECT SHIPMENT3.PID
     FROM SPJ as SHIPMENT3, SPJ as SHIPMENT4
     WHERE SHIPMENT3.SID='S1'

```

```

AND NOT EXISTS
    (SELECT *
     FROM SPJ, SPJ as SHIPMENT5
     WHERE SHIPMENT5.PID=SHIPMENT4.PID
     AND SHIPMENT5.JID = SHIPMENT2.JID));
(12) SELECT DISTINCT SHIPMENT.JID
     FROM SPJ as SHIPMENT, SPJ as SHIPMENT2
     WHERE NOT EXISTS
         (SELECT *
          FROM SPJ, SPJ as SHIPMENT3
          WHERE EXISTS
              (SELECT *
               FROM SPJ, SPJ as SHIPMENT4
               WHERE SHIPMENT4.SID=SHIPMENT3.SID
               AND SHIPMENT4.PID IN
                   (SELECT PROJECT.PID
                    FROM P as PROJECT
                    WHERE PROJECT.COLOR='Red')
               AND NOT EXISTS
                   (SELECT *
                    FROM SPJ, SPJ as SHIPMENT5
                    WHERE SHIPMENT5.SID=SHIPMENT3.SID
                    AND SHIPMENT5.JID=SHIPMENT2.JID)));

```

(b) Formulate the queries (1), (5), (6), and (8) in the relational algebra.

- (1)  $\sigma_{[QTY \geq 400 \text{ AND } QTY \leq 850]}(SPJ)$
- (5)  $\pi_{[PID]}(SPJ \bowtie (\sigma_{[CITY='London']}(J)))$
- (6)  $\pi_{[SID]}(\pi_{[PID]}(\pi_{[SID]}(SPJ \bowtie \pi_{[PID]}(\sigma_{[Color='Red']}(P)))) \bowtie SPJ \bowtie SPJ)$
- (8)  $\pi_{[PID, JID]}(SPJ) \div \pi_{[JID]}(\sigma_{[CITY='London']}(J))$

(c) Formulate the queries (1), (2), (7), and (8) in the tuple relational calculus.

- (1)  $\{P \mid \exists X \in SPJ (X.QTY \geq 400 \wedge X.QTY \leq 850 \wedge X.SID = P.SID \wedge X.PID = P.SID \wedge X.JID = P.JID \wedge X.QTY = P.QTY)\}$
- (2)  $\{P \mid \exists X \in S \exists Y \in J \exists Z \in SPJ (Z.SID = X.SID \wedge Z.JID = Y.JID \wedge Z.QTY \geq 200 \wedge X.CITY = P.CITY1 \wedge Y.CITY = P.CITY2)\}$
- (7)  $\{P \mid \exists X \in S \exists Y \in S (Y.SID = 'S1' \wedge X.STATUS < Y.STATUS \wedge X.SID = P.SID)\}$
- (8)  $\{P \mid \exists X \in SPJ \forall Y \in J (X.CITY = 'London' \Rightarrow (\exists Z \in SPJ (Z.PID = X.PID \wedge Z.JID = Y.JID \wedge X.PID = P.PID)))\}$

(d) Formulate the queries (1), (3), (4), and (8) in the domain relational calculus.

- (1)  $\{\langle SX, PX, JX, QTYX \rangle \mid \langle SX, PX, JX, QTYX \rangle \in SPJ \wedge QTYX \geq 400 \wedge QTYX \leq 850\}$
- (3)  $\{\langle JX \rangle \mid \exists JNX, JCX (\langle JX, JNX, JCX \rangle \in J \wedge \exists SY, SNY, SSY, SCY (\langle SY, SNY, SSY, SCY \rangle \in S \wedge \exists PZ, QTYZ (\langle SY, PZ, JX, QTYZ \rangle \in SPJ \wedge JCX \neq SCY)))\}$
- (4)  $\{\langle JNX \rangle \mid \exists JX, JCX (\langle JX, JNX, JCX \rangle \in J \wedge \exists SY, PY, JY, QTYY (\langle SY, PY, JY, QTYY \rangle \in SPJ \wedge JX = JY \wedge SY = 'S1'))\}$
- (8)  $\{\langle PX \rangle \mid \forall JY, JNY, JCY ((\langle JY, JNY, JCY \rangle \in J \wedge JCY = 'London') \Rightarrow \exists SZ (\langle SZ, PX, JY \rangle \in SPJ))\}$