

Crosstabs In SQL

by

Joe Celko

copyright 2003

The Basic Idea

- The GROUP BY will partition the table into disjoint subsets and summarize them
- It is like a simple summary report, but really much weaker
- Crosstabs and “control break” or other reports have mixed levels of aggregation, so they are not tables (tables are sets; sets are one kind of thing)

Cross Tabulations -1

- Crosstabs are a report in the form of a grid with totals of the number of members in the combination represented by each cell of the grid
- Example: employees broken down by sex by race. (Actually, most of our employees are broken down by drugs and alcohol).

Cross Tabulations -2

- This is not a relational operation
- It converts values into attributes.
- For example "sex" is an attribute with the values "male" and "female"
- The crosstabs has columns for "male" and "female"
- Tables have a fixed, known number of columns; reports do not

Simple example

- Sex by Race

	W h i t e	B l a c k	A s i a n	L a t i n o	
M a l e	1 0 0	6 3	1 4	1 7	194
F e m a l e	8 7	4 8	1 2	1 9	166
	187	111	26	36	

Solution with Subquery

```
SELECT DISTINCT P0.race,  
  (SELECT COUNT(*)  
    FROM Personnel AS P1  
   WHERE P0.race = P1.race  
     AND sex = 'male') AS male_tally,  
  (SELECT COUNT(*)  
    FROM Personnel AS P1  
   WHERE P0.race = P1.race  
     AND sex = 'female') AS female_tally,  
  (SELECT COUNT(*)  
    FROM Personnel AS P1  
   WHERE P0.race = P1.race ) AS grand_tally  
FROM Personnel AS P0;
```

Solution with CASE expression

```
SELECT race,  
       (SUM WHEN sex = 'male' THEN 1 ELSE 0 END ) AS male _tally,  
       (SUM WHEN sex = 'female' THEN 1 ELSE 0 END ) AS female _tally  
FROM Personnel  
GROUP BY race;
```

- If you don't have CASE, you can fake it

```
SELECT race,  
       SUM (POSITION ('male' IN sex ) AS male _tally,  
       SUM (SIGN(ABS(POSITION('female' IN sex)-3))) AS female _tally  
FROM Personnel  
GROUP BY race;
```

Solution with Matrices -1

```
CREATE TABLE DeptMatrix  
(dept_id INTEGER NOT NULL,  
d100 INTEGER NOT NULL,  
d200 INTEGER NOT NULL,  
d300 INTEGER NOT NULL,  
d400 INTEGER NOT NULL,  
d500 INTEGER NOT NULL);
```

```
INSERT INTO DeptMatrix VALUES (100, 1,0,0,0,0);  
INSERT INTO DeptMatrix VALUES (200, 0,1,0,0,0);  
INSERT INTO DeptMatrix VALUES (300, 0,0,1,0,0);  
INSERT INTO DeptMatrix VALUES (400, 0,0,0,1,0);  
INSERT INTO DeptMatrix VALUES (500, 0,0,0,0,1);
```

Solution with Matrices -2

```
CREATE TABLE StateMatrix  
(state CHAR(2) NOT NULL,  
ma INTEGER NOT NULL,  
ga INTEGER NOT NULL,  
ca INTEGER NOT NULL,  
tx INTEGER NOT NULL);
```

```
INSERT INTO StateMatrix VALUES ('MA', 1,0,0,0);  
INSERT INTO StateMatrix VALUES ('GA', 0,1,0,0);  
INSERT INTO StateMatrix VALUES ('CA', 0,0,1,0);  
INSERT INTO StateMatrix VALUES ('TX', 0,0,0,1);
```

Solution with Matrices -3

```
SELECT
SUM(ma * d100), SUM(ca * d100),SUM(tx * d100),SUM(ga *
    d100),
SUM(ma * d200), SUM(ca * d200),SUM(tx * d100),SUM(ga *
    d200),
SUM(ma * d300), SUM(ca * d300),SUM(tx * d300),SUM(ga *
    d300),
SUM(ma * d400), SUM(ca * d400),SUM(tx * d400),SUM(ga *
    d400),
SUM(ma * d500), SUM(ca * d500),SUM(tx * d500),SUM(ga *
    d500)
FROM DeptMarix AS D1, StateMatrix AS S1, Employees AS E1
WHERE E1.state = S1.state
    AND E1.dept_id = D1.dept_id ;
```

Questions & Answers

?