Nested Table Expressions, Common Table Expressions, and Recursive SQL

Peter G Backlund
Peter Backlund DB2-Konsult AB
BacklundDB2@telia.com

GROUP BY Expression

- Select Lastname, Hiredate
  From Emp
  Order by Year(Hiredate)

OK in V7

- Select Year(Hiredate), Avg(Salary)
  From Emp
  Group by Year(Hiredate)

NOK in V7
OK in V8

NTE - Nested Table Expression

- Select Hireyear, Avg(Salary)
  From
  (Select Year(Hiredate) as Hireyear
  , Salary
  From Emp) as NTE
  Group by Hireyear

- Not needed in DB2 for z/OS V8!

Who Earn More Than...

- Who earn more than **the company** average?

```
SELECT LASTNAME
FROM    EMP
WHERE  SALARY >
    (SELECT   AVG(SALARY)
     FROM     EMP)
```

- Who earn more than **their own department** average?

```
SELECT LASTNAME
FROM    EMP [AS] X
WHERE  SALARY >
  (SELECT    AVG(SALARY)
   FROM      EMP
   WHERE     WORKDEPT = X.WORKDEPT)
```

- Performance can be bad!

Using an NTE

In a COBOL program we could first compute salary averages for the departments and then compare each employee's salary against this

- With an NTE we can do as follows

```
SELECT   LASTNAME
FROM  EMP
  JOIN
   (SELECT  WORKDEPT
            ,  AVG(SALARY) AS DEPSAL
    FROM        EMP
    GROUP BY    WORKDEPT
                ) AS DEP
   ON   EMP.WORKDEPT = DEP.WORKDEPT
   AND SALARY              > DEPSAL
```


Using an NTE

- An alternative way to name the columns

```
SELECT   LASTNAME
FROM
   EMP
  JOIN
   (SELECT  WORKDEPT
            ,  AVG(SALARY)
    FROM        EMP
    GROUP BY    WORKDEPT
                ) AS DEP(wd, sal)
   ON   WORKDEPT = WD
   AND SALARY       > SAL
```

Correlated NTE

- We want a list of
  Department Number, Department Name, Number of Employees

```
Select Deptno, Deptname, Number
From   Dept
   Join
        Table (Select Count(*) as Number
                 From   Emp
                 Where Workdept = Dept.Deptno) as NTE
On 1=1;
```

- The NTE must be written **after** the Dept

- The **Table** is necessary

Alternatives

```
-- join with NTE
Select deptno, deptname, antal
 from dept join
 (Select workdept,count(*) as antal from emp group by workdept) x
 on  deptno=workdept;
-- join tables
 Select workdept, max(deptname), count(*)
 from emp join dept
 on workdept=deptno
 group by workdept;
-- scalar fullselect
 Select deptno, deptname
 ,(Select count(*) from emp where workdept=dept.deptno)
  from dept ;
```

New in DB2 for z/OS V8!

Alternatives

```
-- join with NTE
Select deptno, deptname, coalesce(antal,0)
 from dept left join
 (Select workdept,count(*) as antal from emp group by workdept) x
 on  deptno=workdept;
```


Performance

- **Emp:** 10,000,000 rows; 7,500 pages - **Dept:** 33 rows
- Create Index Iwork on Emp(Workdept)
- Create table MQT as
  (Select workdept, count(*) as antal from emp group by workdept)

		Iwork	MQT	Iwork+MQT
Correlated NTE	129,0	2,17	0,00	0,00
Join with NTE	13,6	2,60	0,00	0,00
Join tables	25,0	3,60	20,5	3,52
Scalar Fullselect	130,0	2,17	0,00	0,00

Another Problem

- We want to know

  - what is the largest/smallest average department salary

  - which department has this (or which departments!)

- Pseudo-SQL

```
Select Max(Avg(Salary))
From  Emp
```

EMPNO	LAST NAME	SEX	WORK DEPT	SALARY	HIRE DATE
10	Alf	M	A1	24,000	1973-01-01
70	Gus	M	A1	30,000	1968-07-07
20	Bea	F	C1	29,000	1973-02-02
40	Dave	M	C1	31,000	1973-04-04
30	Carl	M	C1	33,000	1963-03-03
60	Fred	M	E1	30,000	1968-06-06
50	Eva	F	D1	27,000	1963-05-05

Using a View

- Create View DeptSal as
  Select Workdept, Avg(Salary) as AvgSal
  From Emp
  Group by Workdept

- Select * From DeptSal

- Select Max(AvgSal)
  From Deptsal

- Select Workdept, AvgSal
  From DeptSal
  Where Avgsal =
  (Select Min(AvgSal)
  From DeptSal)

EMPNO	LAST NAME	SEX	WORK DEPT	SALARY	HIRE DATE
10	Alf	M	A1	24,000	1973-01-01
70	Gus	M	A1	30,000	1968-07-07
20	Bea	F	C1	29,000	1973-02-02
40	Dave	M	C1	31,000	1973-04-04
30	Carl	M	C1	33,000	1963-03-03
60	Fred	M	E1	30,000	1968-06-06
50	Eva	F	D1	27,000	1963-05-05

WORK DEPT	AVGSAL
A1	27,000
E1	30,000
C1	31,000
D1	27,000

31,000

WORK DEPT	AVGSAL
A1	27,000
D1	27,000

A View Litters

- When you create a view, information is stored in the catalog in the following tables (at least)

  SYSIBM.SYSTABLES
  SYSIBM.SYSCOLUMNS
  SYSIBM.SYSVIEWS
  SYSIBM.SYSVIEWDEP
  SYSIBM.SYSTABAUTH

- Who does the cleaning?

NTE - Example

- What is the largest average department salary?

```
Select Max(AvgSal)
From
(Select Workdept, Avg(Salary) as AvgSal
 From   Emp
 Group By Workdept) as DeptSal
```

- We are using the nested table as an "**inline view**"

- Nested tables may be needed to get
  the correct result ==> **outer join**

NTE - Example...

- Which department has the smallest average salary?

```
Select Workdept, AvgSal
 From  (Select Workdept, Avg(Salary) as AvgSal
          From    Emp
         Group By Workdept) as DeptSal
 Where AvgSal =
 (Select Min(AvgSal)
  From    (Select Workdept, Avg(Salary) as AvgSal
           From    Emp
          Group By Workdept) as DeptSal)
```


CTE - Common Table Expression

```
With DeptSal as
 ( Select Workdept, Avg(Salary) as AvgSal
  From    Emp
  Group By Workdept )

Select  Workdept, AvgSal
From   DeptSal
Where AvgSal =
        (Select Min(AvgSal)
         From   DeptSal)
```

CTE - Common Table Expression

```
With DeptSal(Workdept, AvgSal) as
 ( Select Workdept, Avg(Salary)
  From    Emp
  Group By Workdept )

Select  Workdept, AvgSal
From   DeptSal
Where AvgSal =
          (Select Min(AvgSal)
           From   DeptSal)
```


Is the NTE or CTE Really Necessary?

```
Select Workdept, Avg(Salary)
From  Emp
Group By Workdept
Having Avg(Salary) <=
    All ( Select  Avg(Salary)
          From    Emp
          Group By Workdept)
```

Performance

- **Emp:** 10,000,000 rows; 7,500 pages - **Dept:** 33 rows
- Create Index Iwork on Emp(Workdept)
- Create table MQT as
  (Select workdept, count(*) as antal from emp group by workdept)

		Iwork	MQT
View	11,9	11,5	11,9
NTE	23,8	23,8	23,7
CTE	11,5	12,2	11,7
"All"	25,8	28,7	27,1

CTE - Another Example

```
With
  Memp as
(Select Empno, Lastname, Salary
 From   Emp
 Where Sex = 'M')

 , HiSalMen as
 (Select Lastname, Salary
  From Memp
  Where Salary > 30000)

Select *
From HiSalMen
```

EMPNO	LAST NAME	SEX	WORK DEPT	SALARY	HIRE DATE
10	Alf	M	A1	24,000	1973-01-01
70	Gus	M	A1	30,000	1968-07-07
20	Bea	F	C1	29,000	1973-02-02
40	Dave	M	C1	31,000	1973-04-04
30	Carl	M	C1	33,000	1963-03-03
60	Fred	M	E1	30,000	1968-06-06
50	Eva	F	D1	27,000	1963-05-05

LAST NAME	SALARY
Carl	33,000
Dave	31,000

The Honda Problem

- How many screws are needed for a Honda?

	Carparts	
Part	**Subpart**	**Number**
Honda	Steering Wheel	1
Honda	Wheel	4
Honda	Screw	7
Steering Wheel	Screw	4
Wheel	Tire	1
Wheel	Screw	3

Recursive SQL - How Many Screws?

With t(p,s,n) as

```
(Select Part, Subpart, Number
 From   Carparts
 Where Subpart = 'Screw'
   Union all
 Select Part, Subpart, Number*n
 From   Carparts, t
 Where Subpart = p)

Select sum(n)
 from t
 where p = 'Honda'
```

Honda	SteeringW	1
Honda	Wheel	4
Honda	Screw	7
SteeringW	Screw	4
Wheel	Tire	1
Wheel	Screw	3

Honda	Screw	7
SteeringW	Screw	4
Wheel	Screw	3
Honda	SteeringW	1*4 = 4
Honda	Wheel	4*3 = 12

Final result: 23

Recursive SQL - How Many Screws?

```
With t(p,n) as

(Select Part, Number
 From   Carparts
 Where Subpart = 'Screw'
   Union all
 Select Part, Number*n
 From   Carparts, t
 Where Subpart = p)

Select sum(n)
 from t
 where p = 'Honda'
```

Honda	SteeringW	1
Honda	Wheel	4
Honda	Screw	7
SteeringW	Screw	4
Wheel	Tire	1
Wheel	Screw	3

Honda	7
SteeringW	4
Wheel	3
Honda	1*4 = 4
Honda	4*3 = 12

Final result: 23

Recursive SQL - What Constitutes a Honda?

```
With t(p,s,n) as

(Select Part, Subpart, Number
 From   Carparts
 Where Part = 'Honda'
   Union all
 Select Part, Subpart, Number*n
 From   Carparts, t
 Where Part = s)

Select s, sum(n)
 from t
 group by s
```

Honda	SteeringW	1
Honda	Wheel	4
Honda	Screw	7
SteeringW	Screw	4
Wheel	Tire	1
Wheel	Screw	3

Honda	SteeringW	1
Honda	Wheel	4
Honda	Screw	7
SteeringW	Screw	1*4 = 4
Wheel	Tire	4*1 = 4
Wheel	Screw	4*3 = 12

Result:

Screw	23
SteeringW	1
Tire	4
Wheel	4

Recursive SQL - What Constitutes a Honda?

With t(s,n) as

(Select Subpart, Number
 From Carparts
 Where Part = 'Honda'
  Union all
 Select Subpart, Number*n
 From Carparts, t
 Where Part = s)

Select s, sum(n)
 from t
 group by s

Honda	SteeringW	1
Honda	Wheel	4
Honda	Screw	7
SteeringW	Screw	4
Wheel	Tire	1
Wheel	Screw	3

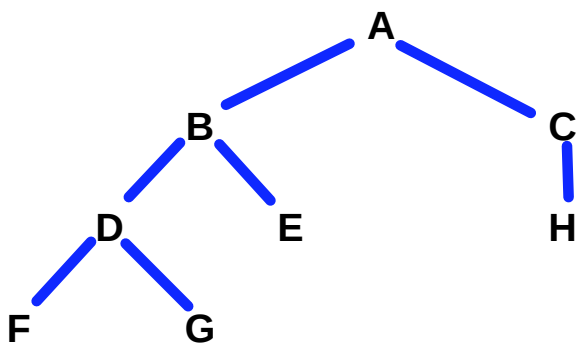
SteeringW	1
Wheel	4
Screw	7
Screw	1*4 = 4
Tire	4*1 = 4
Screw	4*3 = 12

Result:

Screw	23
SteeringW	1
Tire	4
Wheel	4

The Department Hierarchy

- Which departments report (direct or indirect) to 'B'?



Mydept

Deptno	Admrdept
A	?
B	A
D	B
F	D
G	D
E	B
C	A
H	C

Which Departments Report to 'B'?

```
With x(d) as
(Select deptno
 from mydept
 where admrdept = 'B'
   Union All
 Select deptno
 from mydept, x
 where admrdept = d
 )

 select d from x
```

Deptno	Admrdept
A	A
B	A
D	B
F	D
G	D
E	B
C	A
H	C

D
E
F
G

Which Departments Report to 'A'?

```
With x(d) as
(Select deptno
 from mydept
 where admrdept = 'A'
   Union All
 Select deptno
 from mydept, x
 where admrdept = d
 )

 select d from x
```

Count Number of Iterations

```
With x(d,a ,level) as
(Select deptno, admrdept ,0
 from mydept
 where admrdept = 'A'
   Union All
 Select deptno, admrdept ,level+1
 from mydept, x
 where admrdept = d
 and level < 2
 )

 select * from x
```

A	A	0
B	A	0
C	A	0
A	A	1
B	A	1
C	A	1
D	B	1
E	B	1
H	C	1
A	A	2
B	A	2
C	A	2
D	B	2
E	B	2
H	C	2
F	D	2
G	D	2

Eliminate Offending Row

```
With x(d) as
(Select deptno
 from mydept
 where admrdept = 'A'
 and deptno <> workdept
   Union All
 Select deptno
 from mydept, x
 where admrdept = d
 )

 select d from x
```

Deptno	Admrdept
A	A
B	A
D	B
F	D
G	D
E	B
C	A
H	C

B
C
D
E
H
F
G

Eliminate Offending Row

Update Mydept
Set Admrdept = NULL
Where Deptno = 'A';

Deptno	Admrdept
A	*null*
B	A
D	B
F	D
G	D
E	B
C	A
H	C

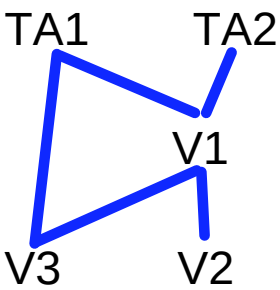
```
With x(d) as
(Select deptno
 from mydept
 where admrdept = 'A'
   Union All
 Select deptno
 from mydept, x
 where admrdept = d
 )

 select d from x
```

B
C
D
E
H
F
G

View Dependency

create table ta1(t1 int);
create table ta2(t2 int);
create view v1 as select t1 from ta1,ta2;
create view v2 as select t1 from v1;
create view v3 as select ta1.t1 from ta1,v1;




```
with t(name) as
(select dname
 from sysibm.sysviewdep
 where bname ='TA1'
 union all
 select dname
 from sysibm.sysviewdep, t
 where  name=bname)
 select distinct * from t
 order by 1;
```

NAME
V1
V2
V3

Generate Testdata

```
with t(n,x) as
(select 1,rand(2)
 from sysibm.sysdummy1
 union all
 select n+1,rand()
 from t
 where n<10)

select n, dec(x,5,3) as r
from t
```

n	r
1	0.001
2	0.891
3	0.738
4	0.543
5	0.899
6	0.599
7	0.445
8	0.806
9	0.326
10	0.558

Generate EMP Table

```
Insert into emp select
with t(n) as (select 1 from sysibm.sysdummy1
                union all
                select n+1 from t
                 where n<100)
 select char(100000+n)
 , substr('ACBCDEFGHIJKLMNOPQRSTUVWXYZÅÄÖ',int(rand()*29)+1,1)
 ||substr('abcdefghijklmnopqrstuvwxyzåäö',int(rand()*29)+1,1)
 ||substr('abcdefghijklmnopqrstuvwxyzåäö',int(rand()*29)+1,1)
 ||substr('abcdefghijklmnopqrstuvwxyzåäö',int(rand()*29)+1,1)
 ,  case when rand()>0.9 then 'M' else 'F' end
 ,  'A' || substr(digits(smallint(rand()*99)),4,2)
 ,  round(rand()*30000,-2)+10000
 , date('1980-01-01') + int(rand()*3650) days
  from t
```

Generated EMP Table

EMPNO	LASTNAME	SEX	WORKDEPT	SALARY	HIREDATE
100001	Aåkm	F	A89	30800	1984-09-12
100002	Sgxq	F	A18	30800	1980-11-05
100003	Dcåw	F	A62	17800	1983-09-23
100004	Cqsö	M	A03	39000	1988-08-21
100005	Eouj	F	A19	22900	1986-02-18
100006	Zhgp	F	A62	12100	1989-07-10
100007	Rwåa	F	A03	38500	1988-09-13
100008	Soår	F	A27	33500	1980-03-03
100009	Itkz	F	A12	11100	1983-04-25
100010	Mpjb	F	A45	15100	1982-12-19
100011	Pwsu	F	A20	10100	1988-03-07
100012	Udhx	F	A89	15900	1987-04-14

Thanks to **GRAEME BIRCHALL**