DBTech EXT

BI&KDD VLW Meeting

Dimitris A. Dervos
dad@it.teithe.gr
http://aetos.it.teithe.gr/~dad

Adobe Connect Pro® Virtual Classroom,
University of Malaga,
December 08, 2010, 7.00-8.00 pm CET

Meeting Objectives

- Relaxed, informal web meeting…
- …of a preparatory nature in view of the 13 December meeting:
  - familiarize ourselves with the IBM command editor window, the command line window, and the control center user interface
  - SQL views: classical vs. runtime types…
- … all by means of a simple (numbers) example
- Most important of all, today's meeting will NOT be an instructor's monologue; the participants are encouraged to participate and do some of the talking!

Can submit/execute SQL commands:

- Directly, using the command editor window
- Directly, using the command line processor window
- As an SQL script, via the command line processor window

Note: may copy-paste across your OS and the PCV VM environment!

A Simple Task

Will generate all the natural numbers (i.e. 0,1,2,3...), up to a pre-specified length, say up to 4 digits (9999)

'Testbed' Creation

```
CONNECT TO db01 USER dbtech001;
DROP TABLE mydbs.r;
CREATE TABLE mydbs.r (a
VARCHAR(100));
DROP TABLE mydbs.f;
CREATE TABLE mydbs.f (a
VARCHAR(100));

INSERT INTO mydbs.f(a) VALUES ('1');
INSERT INTO mydbs.f(a) VALUES ('2');
INSERT INTO mydbs.f(a) VALUES ('3');
INSERT INTO mydbs.f(a) VALUES ('4');
INSERT INTO mydbs.f(a) VALUES ('5');
INSERT INTO mydbs.f(a) VALUES ('6');
INSERT INTO mydbs.f(a) VALUES ('7');
INSERT INTO mydbs.f(a) VALUES ('8');
INSERT INTO mydbs.f(a) VALUES ('9');

INSERT INTO mydbs.r(a) SELECT * FROM
mydbs.f.a;
INSERT INTO mydbs.r(a) VALUES('0');
CONNECT RESET;
```

'Testbed' Relations

F

A
1
2
3
4
5
6
7
8
9

R

A
1
2
3
4
5
6
7
8
9
0

SQL Views Talk

```
CREATE VIEW mydbs.A(a)
AS SELECT mydbs.R.a FROM mydbs.R;

CREATE VIEW mydbs.B(b)
AS SELECT mydbs.F.a CONCAT mydbs.R.a
FROM mydbs.F, mydbs.R;

CREATE VIEW mydbs.C(c)
AS SELECT mydbs.F.a CONCAT R1.a CONCAT R2.a
FROM mydbs.F, mydbs.R R1, mydbs.R R2;

CREATE VIEW mydbs.D(d)
AS SELECT mydbs.F.a CONCAT R1.a CONCAT R2.a CONCAT R3.a
FROM mydbs.F, mydbs.R R1, mydbs.R R2, mydbs.R R3;
```

SQL Views Talk

```
CREATE VIEW mydbs.A(a)
AS SELECT mydbs.R.a FROM mydbs.R;

CREATE VIEW mydbs.B(b)
AS SELECT mydbs.F.a CONCAT mydbs.R.a
FROM mydbs.F, mydbs.R;

CREATE VIEW mydbs.C(c)
AS SELECT mydbs.F.a CONCAT R1.a CONCAT R2.a
FROM mydbs.F, mydbs.R R1, mydbs.R R2;

CREATE VIEW mydbs.D(d)
AS SELECT mydbs.F.a CONCAT R1.a CONCAT R2.a CONCAT R3.a
FROM mydbs.F, mydbs.R R1, mydbs.R R2, mydbs.R R3;
```

```
CREATE VIEW mydbs.U1(u1)
AS (SELECT * FROM mydbs.A UNION SELECT * FROM
mydbs.B);

CREATE VIEW mydbs.U2(u2)
AS (SELECT * FROM mydbs.U1 UNION SELECT * FROM
mydbs.C);

CREATE VIEW mydbs.U3(u3)
AS (SELECT * FROM mydbs.U2 UNION SELECT * FROM
mydbs.D);

SELECT * FROM mydbs.U3
ORDER BY int(u3);
```

Runtime Version (SQL Standard)

```
SELECT *
FROM ( SELECT *
        FROM ( SELECT *
                FROM ( SELECT *
                        FROM (SELECT mydbs.R.a
                                FROM mydbs.R)
                        UNION
                        SELECT *
                        FROM (SELECT mydbs.F.a CONCAT mydbs.R.a
                                FROM mydbs.F, mydbs.R)
                        )
                UNION
                SELECT *
                FROM ( SELECT mydbs.F.a CONCAT R1.a CONCAT R2.a
                        FROM mydbs.F, mydbs.R R1, mydbs.R R2)
                )
        UNION
        SELECT *
        FROM ( SELECT mydbs.F.a CONCAT R1.a CONCAT R2.a CONCAT R3.a
                FROM mydbs.F, mydbs.R R1, mydbs.R. R2, mydbs.R R3)
         )  AS U3(u3)
ORDER BY int(u3)
```

Runtime Version (IBM DB2)

```
WITH
A(a)    as (select mydbs.R.a from  mydbs.R),
B(b)    as (select  mydbs.F.a CONCAT  mydbs.R.a
            from  mydbs.F,  mydbs.R),
...
```

Runtime Version (IBM DB2)

```
WITH
A(a)    as (select mydbs.R.a from  mydbs.R),
B(b)    as (select  mydbs.F.a CONCAT  mydbs.R.a
            from  mydbs.F,  mydbs.R),
C(c)    as (select  mydbs.F.a CONCAT R1.a CONCAT R2.a
            from  mydbs.F, mydbs.R R1, mydbs.R R2),
D(d)    as (select  mydbs.F.a CONCAT R1.a CONCAT R2.a CONCAT R3.a
            from  mydbs.F, mydbs.R R1, mydbs.R R2, mydbs.R R3),
...
```

Runtime Version (IBM DB2)

```
WITH
A(a)    as (select mydbs.R.a from  mydbs.R),
B(b)    as (select  mydbs.F.a CONCAT  mydbs.R.a
            from  mydbs.F,  mydbs.R),
C(c)    as (select  mydbs.F.a CONCAT R1.a CONCAT R2.a
            from  mydbs.F, mydbs.R R1, mydbs.R R2),
D(d)    as (select  mydbs.F.a CONCAT R1.a CONCAT R2.a CONCAT R3.a
            from  mydbs.F, mydbs.R R1, mydbs.R R2, mydbs.R R3),
U1(u1) as (select * from A UNION select * from B),
U2(u2) as (select * from U1 UNION select * from C),
U3(u3) as (select * from U2 UNION select * from D)

select * from U3 order by int(u3)
```

...Incidentally: a Recursive Version (IBM DB2)

```
WITH U(counter_col,u1) AS
        (SELECT 1, mydbs.R.a
         FROM mydbs.R
         UNION ALL
         SELECT counter_col+1, mydbs.F.a CONCAT U.u1
         FROM mydbs.F, U
         WHERE U.counter_col<4)
SELECT u1
FROM U
ORDER BY int(u1)
```

Thank You!