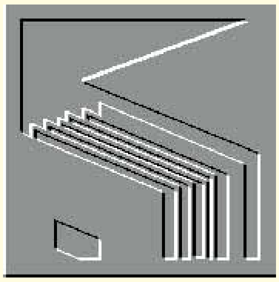


The **Perfect** Gift

A chastened Mark Whitehorn eats more humble pie than mince pie, and asks: "What do database developers *really* want for Christmas?" Plus, using the Count function, and more lottery logic.

OK Santa, I am a humbled and chastened individual. Last year I was greedy: you know it and now I know it. I got carried away. I asked for a powerful workstation, a big file server, and an even bigger database server. Then I added a new network to the list, and finally a new house in which to keep it all. I was too greedy and, quite rightly, nothing appeared on Christmas Morn. Not so much as a mouse mat with a picture of a penguin.

I realise now that I was foolish and self-centred. There are far more deserving causes than my own self gratification, so this year I'll ask for things from which we can all benefit.

For a while now I have been covering those features which make a DBMS actually Relational (in other words, make it worthy of the term RDBMS). As I have been doing this, it has come home to me more forcefully than ever that the majority of DBMSs sold for the PC simply don't make the grade. This isn't a question of simple pedantry, or an insistence that definitions are met; it's about wanting DBMSs that actually deliver data security, speed and ease of use. I would be grateful if, over the next year, you could make all the manufacturers look seriously at their products, and either remove the word "Relational" from the box or deliver the goods. If you need a few names of offending manufacturers, try IBM (Approach), Borland (Paradox, dBASE) and Microsoft (FoxPro).

Fig 1 John S.Graham's sample code

```
Sub Form_Current ()
    'The ComboBox is called "Pick List"
    'The form is called "Employees"

    Dim ctlPickList As Control
    Dim rstEmployees As Recordset
    Dim rstEmployeesClone As Recordset
    Dim strName As String
    Dim intRow As Integer

    Set rstEmployees = Me.RecordsetClone
    rstEmployees.Bookmark = Me.Bookmark

    rstEmployees.MovePrevious
    If rstEmployees.BOF Then
        GoTo AtFirstRow
    End If

    Set rstEmployeesClone = rstEmployees.Clone()
    rstEmployeesClone.Bookmark = rstEmployees.Bookmark

    intRow = 0
    Do Until rsttEmployeesClone.BOF
        rsttEmployeesClone.MovePrevious
        intRow = intRow + 1
    Loop

    strName = [Pick List].ItemData(intRow)
    [Pick List].Value = strName

    rsttEmployees.Close
    rsttEmployeesClone.Close

Exit Sub

AtFirstRow:
    strName = [Pick List].ItemData(0)
    [Pick List].Value = strName
    Exit Sub

End Sub

Sub_Form_Load ()
    'Sets the default value for the combo when the form loads

    Dim strName As String

    strName = [Pick List].ItemData(0)
    [Pick List].DefaultValue = strName

End Sub
```

Secondly, we (database developers) need more tools. Santa, you have no idea what it is like nowadays. Gone are the days when we could ask for, and get, a whole year in which to fail to deliver a complete database. Two-year extensions on projects which were originally scheduled for a mere six months are no longer the norm. Nowadays we are sometimes expected to deliver working systems within a few paltry months. I have even heard of projects with detailed specifications and, almost unbelievably, penalty clauses. What we really need are more tools to take the slog out of the design and implementation work. After all, while much of our work is, of course, an art form and could never be replaced by a mere computer program, much of it is essentially a slog. With those bits we could really do with some help.

Access 95's new database designer is a step in the right direction, and of course Asymetrix's InfoModeler, SDP's S-Designor Professional are doing a great job, but we need more. We also need these tools to be more tightly integrated into the RDBMSs that we use on a daily basis. Anything you could do in this direction would be greatly appreciated.

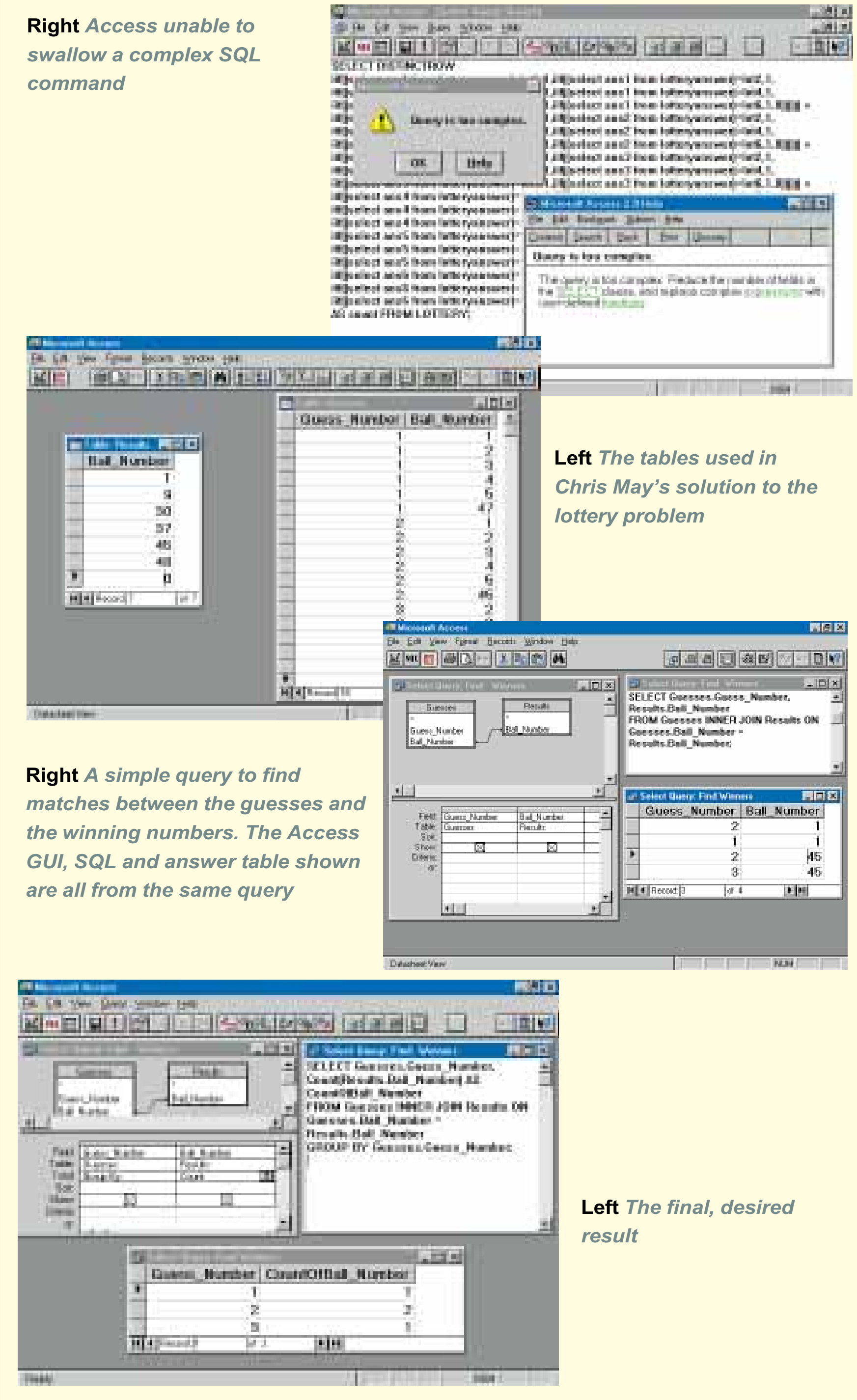
Well, that about wraps it up for the present (sorry, about the pun) I hope that Ms Claus, and all of the little sub-Clauses, are fine.

Regards,
Mark.

Tips and tricks

Well, I asked for tips and tricks and here is some code sample from John S. Graham: "How do you feel about covering the elements of DAO in future

Fig 2 Lottery solutions



```
SELECT DISTINCTROW
...
AS count FROM LOTTERY;
```

Query is too complex.

OK Help

Query is too complex

The query is too complex. Reduce the number of fields in the SELECT clause, and replace complex expressions with user-defined functions.

Right *Access unable to swallow a complex SQL command*

Ball_Number
1
9
30
37
45
48
0

Guess_Number	Ball_Number
1	1
1	2
1	3
1	4
1	6
1	47
2	1
2	2
2	3
2	4
2	6
2	45
3	2

Left *The tables used in Chris May's solution to the lottery problem*

```
SELECT Guesses.Guess_Number,
Results.Ball_Number
FROM Guesses INNER JOIN Results ON
Guesses.Ball_Number =
Results.Ball_Number;
```

Select Query: Find Winners

Guess_Number	Ball_Number
2	1
1	1
2	45
3	45

Record: 3 of 4

Right *A simple query to find matches between the guesses and the winning numbers. The Access GUI, SQL and answer table shown are all from the same query*

Select Query: Find Winners

```
SELECT Guesses.Guess_Number,
Count(Results.Ball_Number) AS
CountOfBall_Number
FROM Guesses INNER JOIN Results ON
Guesses.Ball_Number =
Results.Ball_Number
GROUP BY Guesses.Guess_Number;
```

Guess_Number	CountOfBall_Number
1	1
2	2
3	1

Left *The final, desired result*

articles?" *(Sounds like a good idea to me. MW).*

"I have some sample code that illustrates how to use complex things like BookMarks and Recordset Clones. This code is attached to the 'Current' event of a bound form which is the container object for a ComboBox. The ComboBox draws its data from the same query as the form and its main purpose is to move the form to a record corresponding to the value selected in the combo. This is trivial stuff, as the code to do it is in the Access Help file. However, if the combo displays a value and the user moves to a different record (using the form navigation buttons), the value in the combo remains static and does not follow the form. My code corrects this so that the combo 'follows the form' in addition to the form 'following the combo'." (See John's sample code in *Fig 1*)

● And from Bob McQuattie: "Before identifying my problem, could I also add to the article in October's *PCW*" (*in which I*

Fig 3 Roger Hipperson's SQL query

```
SELECT DISTINCTROW
iif((select ans1 from lotteryanswer)=lot1,1,iif((select ans1 from lotteryanswer)=lot2,1,
iif((select ans1 from lotteryanswer)=lot3,1,iif((select ans1 from lotteryanswer)=lot4,1,
iif((select ans1 from lotteryanswer)=lot5,1,iif((select ans1 from lotteryanswer)=lot6,1,0)))))) +
iif((select ans2 from lotteryanswer)=lot1,1,iif((select ans2 from lotteryanswer)=lot2,1,
iif((select ans2 from lotteryanswer)=lot3,1,iif((select ans2 from lotteryanswer)=lot4,1,
iif((select ans2 from lotteryanswer)=lot5,1,iif((select ans2 from lotteryanswer)=lot6,1,0)))))) +
iif((select ans3 from lotteryanswer)=lot1,1,iif((select ans3 from lotteryanswer)=lot2,1,
iif((select ans3 from lotteryanswer)=lot3,1,iif((select ans3 from lotteryanswer)=lot4,1,
iif((select ans3 from lotteryanswer)=lot5,1,iif((select ans3 from lotteryanswer)=lot6,1,0)))))) +
iif((select ans4 from lotteryanswer)=lot1,1,iif((select ans4 from lotteryanswer)=lot2,1,
iif((select ans4 from lotteryanswer)=lot3,1,iif((select ans4 from lotteryanswer)=lot4,1,
iif((select ans4 from lotteryanswer)=lot5,1,iif((select ans4 from lotteryanswer)=lot6,1,0)))))) +
iif((select ans5 from lotteryanswer)=lot1,1,iif((select ans5 from lotteryanswer)=lot2,1,
iif((select ans5 from lotteryanswer)=lot3,1,iif((select ans5 from lotteryanswer)=lot4,1,
iif((select ans5 from lotteryanswer)=lot5,1,iif((select ans5 from lotteryanswer)=lot6,1,0)))))) +
iif((select ans6 from lotteryanswer)=lot1,1,iif((select ans6 from lotteryanswer)=lot2,1,
iif((select ans6 from lotteryanswer)=lot3,1,iif((select ans6 from lotteryanswer)=lot4,1,
iif((select ans6 from lotteryanswer)=lot5,1,iif((select ans6 from lotteryanswer)=lot6,1,0))))))
AS count FROM LOTTERY;
```

talked about the problems of installing Access as a network application. MW).

"1. We have installed Access 2.0 on individual networked PCs, with the database stored on the server. During development, only one machine was accessing this, so no problems. However, once we tried to go live, we found that messages were coming back advising that the database was already in use. We had omitted to amend the default access to "shared" rather than exclusive. A simple solution, but one which can drive you mad until you find the answer.

2. The article on modifying forms, etc made reference to "cut and paste" to a modified form. I would have thought that the simpler solution would be to open the original form, and then "save As", giving a modified form name; this can then be used for further changes without disturbing the original. Or have I missed something?

No, Windows often provides several ways of doing thing, these are just two alternatives. I find the Cut and Paste solution (especially with Ctrl C and Ctrl V), to be slightly faster, but it's just a matter of taste.

3. My problem. My database contains records which are extracted for the purpose of producing labels, for product certification. Within each batch of line items, there can be a variable number of labels per line item. I have now managed to construct the necessary form and print routines to produce the required number of labels. These are sent, via the server, to an Epson FX1170 Dot matrix printer, and printed two-across.

However, different customers take different labels, and while there is a marginal difference in width, the problem stems from the length of the label. My labels form was originally designed around a label length of 1020mm although only approx. 800 mm is used (there is a pre-printed company logo). This label is, to all intents, satisfactory.

However, when I try to alter the settings to the other label length of 860mm, these changes being made in both Control Panel and Print setup, the label form becomes misaligned due to spacing up more than required, and seems to have a distance of 1020 mm between the first line of print on consecutive labels. Try as I might, I cannot find any reason for this spacing, as all the settings seem to be correct. Can you make any suggestion?"

The simple answer is no, apart from questioning the overall efficiency of the printer driver. Has anyone else come across this problem?

● In the Nov. issue of *PCW* I published a lottery problem from Mark Broadbent. I supplied a solution in code, but wondered if anyone could come up with a solution which consisted of a single query.

Roger Hipperson replied: "Here is the (unpleasant) SQL query to check the rows of numbers in lottery against the ONE row of results in a table called lotteryanswer," shown in *Fig 3*. It is enough to make you want to do the lottery.

Nothing would induce me to do the lottery (except of course the thought of all the money). I tried to insert this SQL statement into Access and it replied succinctly (see *Fig 2*) but I am more than prepared to take Roger's word for its accuracy. He is, however, quite correct that it is unpleasant.

● A different solution came from Chris May: "I think your lottery SQL problem in the current PCW is more of a database design problem than an SQL one. If you simplify (normalise) the tables a bit more by making each ball prediction and result a row in its own right, you can just join the guesses and results tables on ball number and use "count(*)" and "group by" to find how many guesses match the results.

Here's the example, using your data:

```
select guess_number,
count(*) as matching_balls
from guesses, results
where guesses.ball_number
= results.ball_number
group by guess_number."
```

And Chris supplied the appropriate tables. This solution clearly comes from an RDBMS other than Access (Ingress, in fact), since the SQL syntax is different. It can be converted to the standard Access dialect as:

```
SELECT DISTINCTROW Guesses.Guess_Number, Count(Results.Ball_Number) AS
CountOfBall_Number
FROM Guesses INNER JOIN Results ON
Guesses.Ball_Number =
Results.Ball_Number
GROUP BY Guesses.Guess_Number;
```

The burning question is "How does it work?".We start with two tables, "Results", which holds this weeks results from the Lottery, and "Guesses" which contains three guesses that we have made. These tables hold the same data as used in the November issue, it has simply been redistributed (see *Fig 2*).

A simply query can be used to find which numbers from the Results table are also found in the Guesses table (see *Fig 2*). Each time a match is found for a particular guess, a new record appears in the answer table. Thus we see two matches for guess 2, and one each for the other two.

Finally, all we have to do is to count the number of matches for each guess and we have accomplished the task (see *Fig 2*). I like this solution because it seems cleaner and more elegant that the one I originally suggested.

PCW Contacts

Mark Whitehorn welcomes readers' correspondence and ideas for the Databases column. He's on **m.whitehorn@dundee.ac.uk**