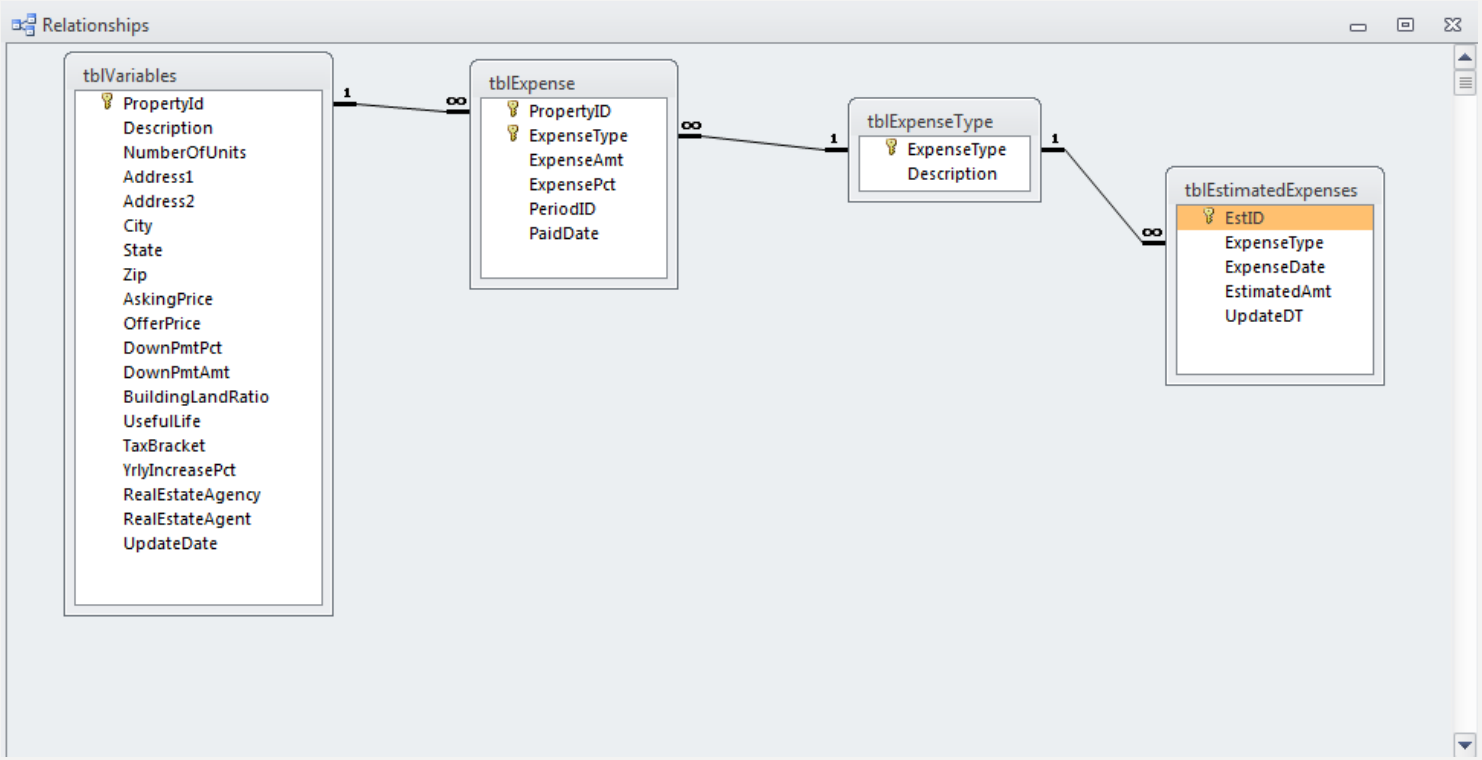
BOUND DENORMALIZED FORMS

WHY?

- What do you do when your client has Excel on the brain and insists on retaining the look and feel of his unnormalized spreadsheet?
- This example takes a properly normalized set of tables and shows how to select specific items from the many-side table and make them look like they are part of the one-side record. No code is required to perform this pivot; just a query for each field to pivot and another to bring them all together. Although, you will need code to properly set the foreign key for new records.
- Adding an amount to one of the empty "cells" inserts a row in the many-side table. Deleting an amount should delete a row but I can't figure out how to get rid of the "deleted" message so I just left the null value.
- This example could be used to pivot a small number of many-side records to produce a view similar to a spreadsheet while still being updatable. Examples might be days in a week for a timekeeping app or months in a year for a forecasting app.

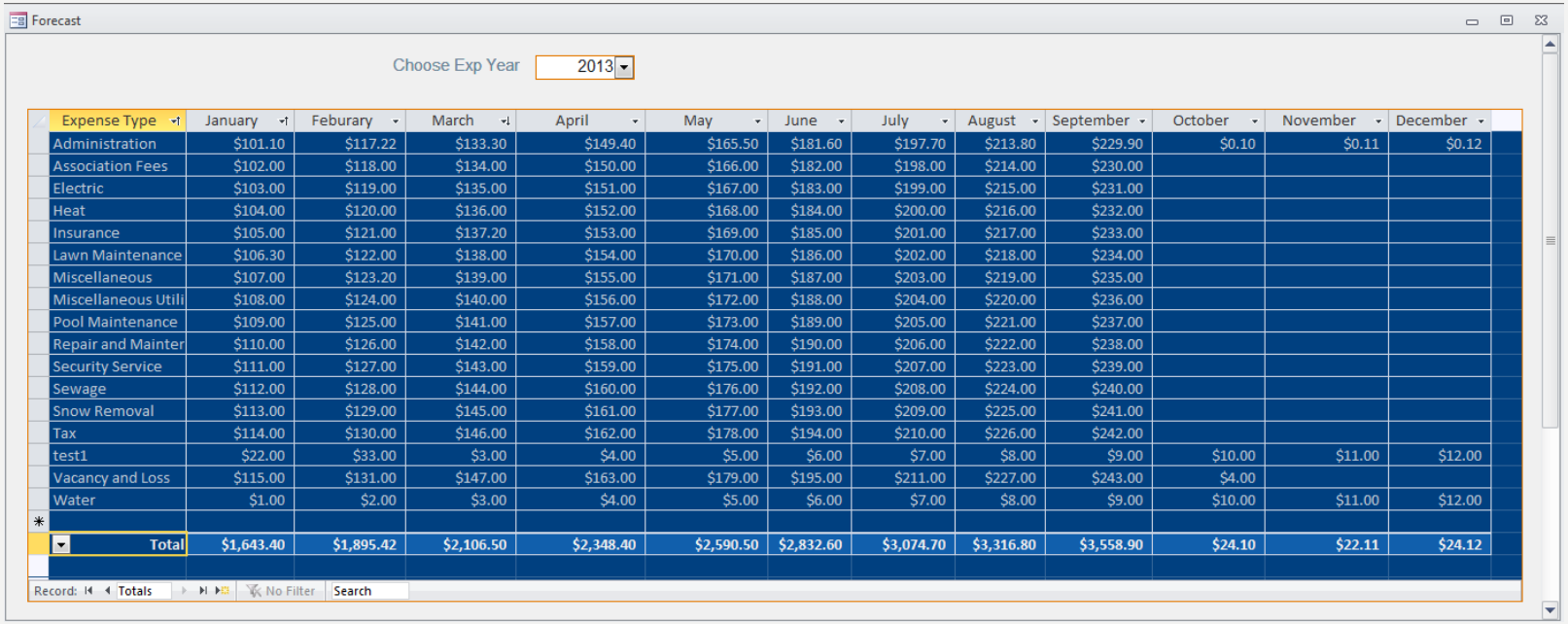
- The second set of example data I chose is not really a good fit since the columns are neither limited nor predictable and so the form would require maintenance each time a new expense type was added. But it does show the flexibility in adding rows. Plus I included a report based on this data to show how you can have variable columns in a report.

THE SCHEMA


Relationships
tblVariables
PropertyId
Description
NumberOfUnits
Address1
Address2
City
State
Zip
AskingPrice
OfferPrice
DownPmtPct
DownPmtAmt
BuildingLandRatio
UsefulLife
TaxBracket
YrlyIncreasePct
RealEstateAgency
RealEstateAgent
UpdateDate
1
∞
tblExpense
PropertyID
ExpenseType
ExpenseAmt
ExpensePct
PeriodID
PaidDate
∞
1
tblExpenseType
ExpenseType
Description
1
∞
tblEstimatedExpenses
EstID
ExpenseType
ExpenseDate
EstimatedAmt
UpdateDT

THE FORM – DATASHEET VIEW

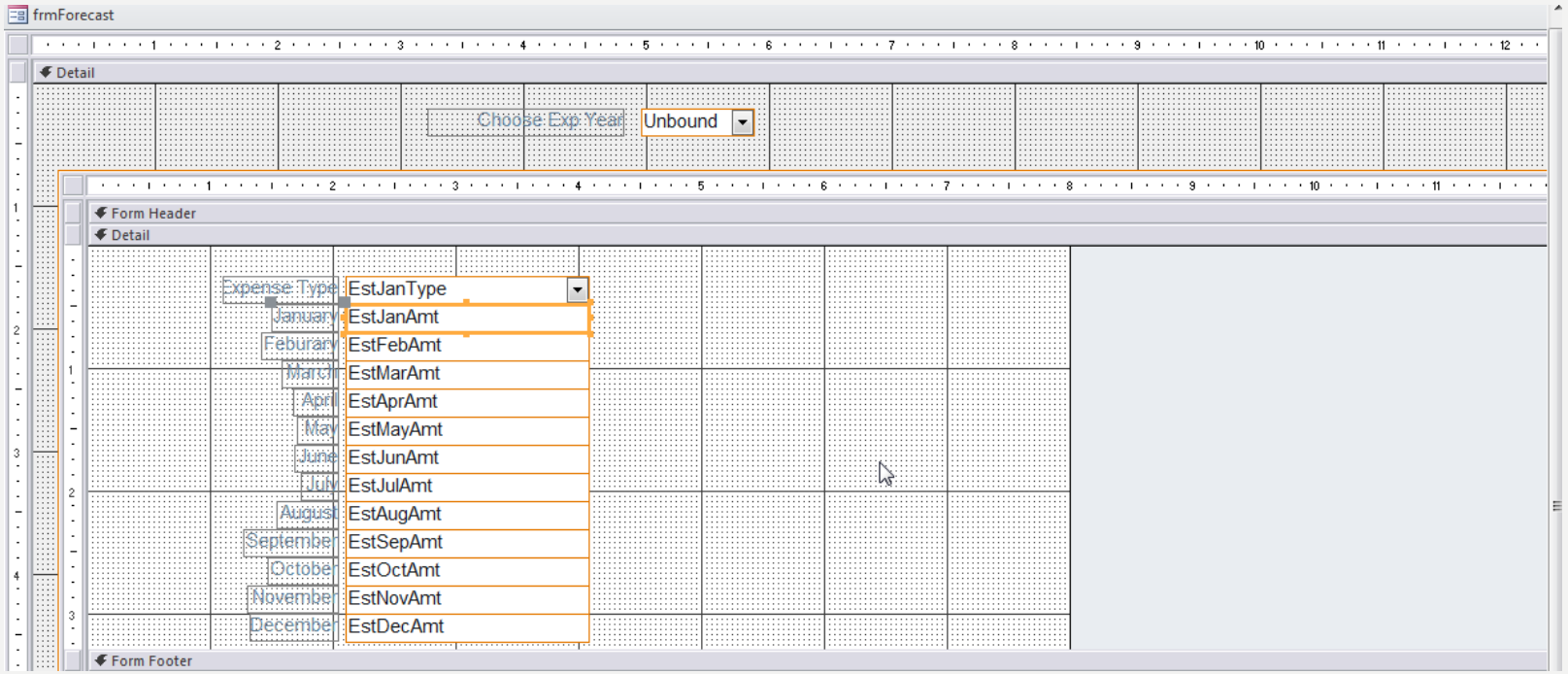
The Expense type and Jan data come from the "main" query but each other month comes from a separate query that is joined to the main query.



Expense Type	January	Feburary	March	April	May	June	July	August	September	October	November	December
Administration	$101.10	$117.22	$133.30	$149.40	$165.50	$181.60	$197.70	$213.80	$229.90	$0.10	$0.11	$0.12
Association Fees	$102.00	$118.00	$134.00	$150.00	$166.00	$182.00	$198.00	$214.00	$230.00			
Electric	$103.00	$119.00	$135.00	$151.00	$167.00	$183.00	$199.00	$215.00	$231.00			
Heat	$104.00	$120.00	$136.00	$152.00	$168.00	$184.00	$200.00	$216.00	$232.00			
Insurance	$105.00	$121.00	$137.20	$153.00	$169.00	$185.00	$201.00	$217.00	$233.00			
Lawn Maintenance	$106.30	$122.00	$138.00	$154.00	$170.00	$186.00	$202.00	$218.00	$234.00			
Miscellaneous	$107.00	$123.20	$139.00	$155.00	$171.00	$187.00	$203.00	$219.00	$235.00			
Miscellaneous Utili	$108.00	$124.00	$140.00	$156.00	$172.00	$188.00	$204.00	$220.00	$236.00			
Pool Maintenance	$109.00	$125.00	$141.00	$157.00	$173.00	$189.00	$205.00	$221.00	$237.00			
Repair and Mainter	$110.00	$126.00	$142.00	$158.00	$174.00	$190.00	$206.00	$222.00	$238.00			
Security Service	$111.00	$127.00	$143.00	$159.00	$175.00	$191.00	$207.00	$223.00	$239.00			
Sewage	$112.00	$128.00	$144.00	$160.00	$176.00	$192.00	$208.00	$224.00	$240.00			
Snow Removal	$113.00	$129.00	$145.00	$161.00	$177.00	$193.00	$209.00	$225.00	$241.00			
Tax	$114.00	$130.00	$146.00	$162.00	$178.00	$194.00	$210.00	$226.00	$242.00			
test1	$22.00	$33.00	$3.00	$4.00	$5.00	$6.00	$7.00	$8.00	$9.00	$10.00	$11.00	$12.00
Vacancy and Loss	$115.00	$131.00	$147.00	$163.00	$179.00	$195.00	$211.00	$227.00	$243.00	$4.00		
Water	$1.00	$2.00	$3.00	$4.00	$5.00	$6.00	$7.00	$8.00	$9.00	$10.00	$11.00	$12.00
Total	$1,643.40	$1,895.42	$2,106.50	$2,348.40	$2,590.50	$2,832.60	$3,074.70	$3,316.80	$3,558.90	$24.10	$22.11	$24.12

THE FORM – DESIGN VIEW

Design view of Form. The filter is on the main form. The subform is shown in datasheet view. If you compare this to the final query, you will see that each control comes from a separate query. That is what makes the form add new rows to the properly normalized table. I have only tested this sample with Jet/ACE. I'm not sure it will work with SQL Server but do try it.



THE MONTH QUERIES

Each column you want to pivot requires a separate query. This example has one for each month.

qEstJan

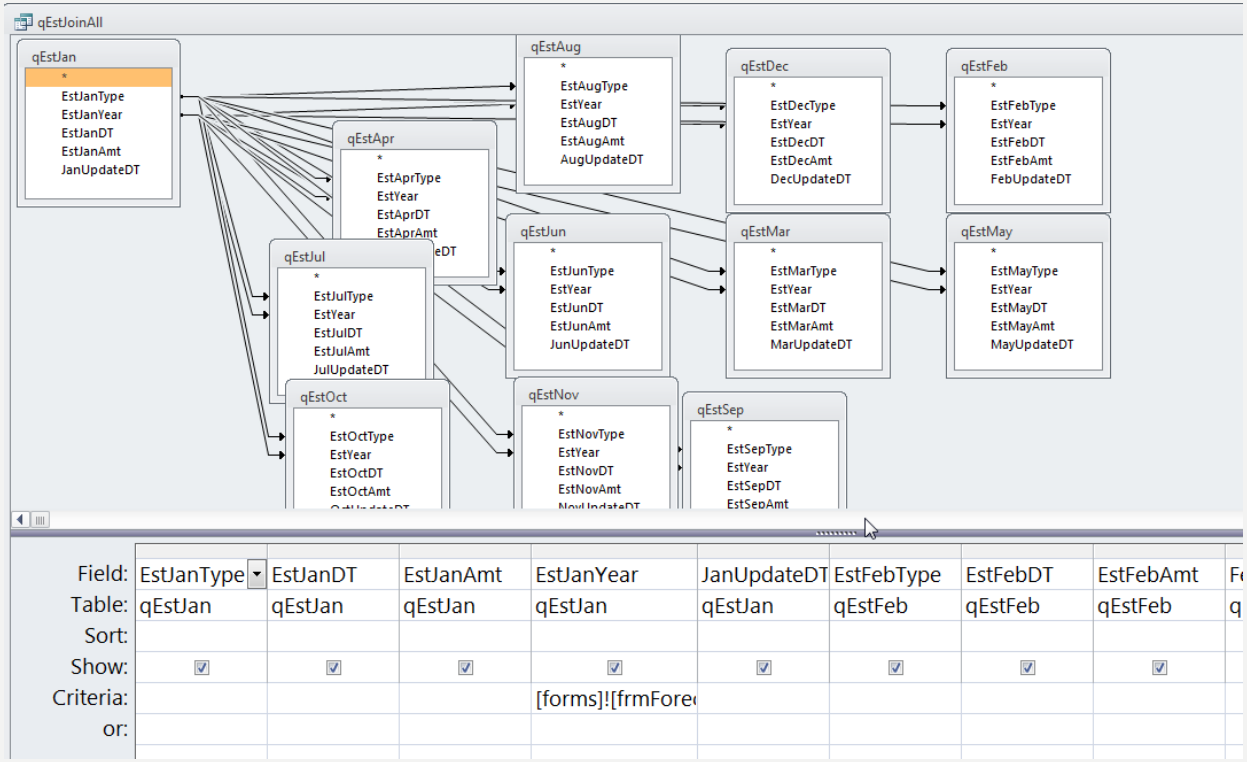
```
SELECT tblEstimatedExpenses.ExpenseType AS EstJanType
   , Year([ExpenseDate]) AS EstYear
   , tblEstimatedExpenses.ExpenseDate AS EstJanDT
   , tblEstimatedExpenses.EstimatedAmt AS EstJanAmt
   , tblEstimatedExpenses.UpdateDT AS JanUpdateDT
FROM tblEstimatedExpenses
WHERE Month([ExpenseDate])=1;
```

qEstSep

```
SELECT tblEstimatedExpenses.ExpenseType AS EstSepType
   , Year([ExpenseDate]) AS EstYear
   , tblEstimatedExpenses.ExpenseDate AS EstSepDT
   , tblEstimatedExpenses.EstimatedAmt AS EstSepAmt
   , tblEstimatedExpenses.UpdateDT AS SepUpdateDT
FROM tblEstimatedExpenses
WHERE Month([ExpenseDate])=9;
```

FINAL QUERY

The final query uses a left join to connect the main query with all the others. Data must exist always exist for the main query for this to work. This example uses Jan and connects the other 11 months.



CODE FOR EACH CONTROL.

This example hard-codes the month but gets the year from the form's filter

```
Private Sub EstJulAmt_BeforeUpdate(Cancel As Integer)
    If IsNull(Me.EstJulAmt.OldValue) Then
        Me.EstJulDT = "7/1/" & Me.Parent!cboYearFilter
        Me.EstJulType = Me.cboEstJanType
    End If
    Me.JulUpdateDT = Now()
End Sub
```

OTHER INTERESTING FEATURES

RESIZE SUBFORM

This code goes into the main form

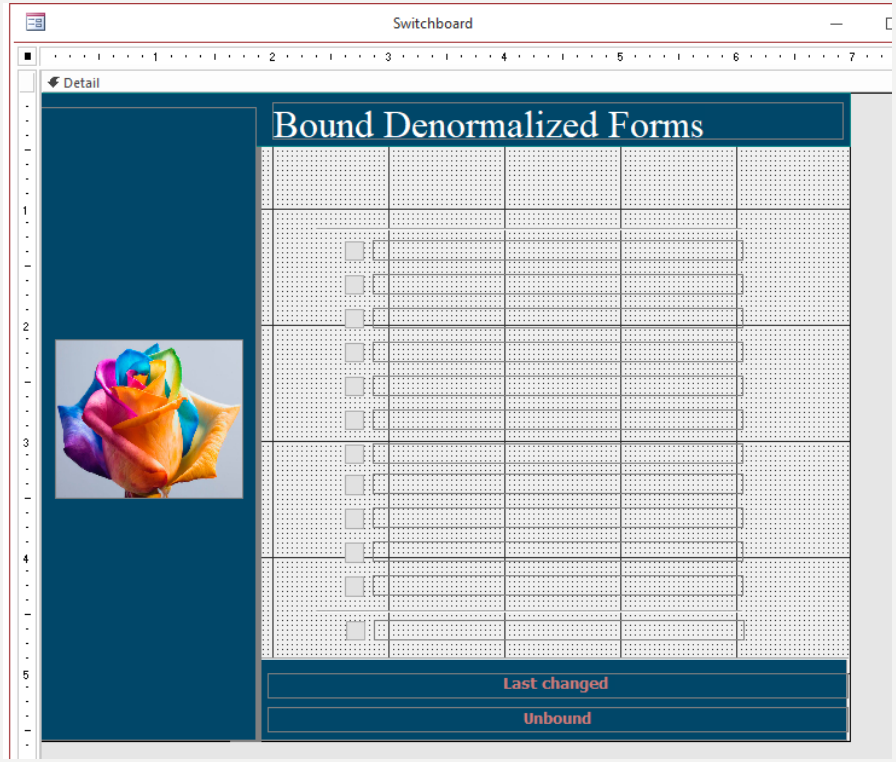
```
Private Sub Form_Resize()
    Const LeftRightPadding = (0.125 + 0.25) * 1440
    Const TopBottomPadding = (0.5 + 0.375) * 1440

    Me.sfrmEstJoinAll.Height = Me.InsideHeight - TopBottomPadding
    Me.sfrmEstJoinAll.Width = Me.InsideWidth - LeftRightPadding
End Sub
```

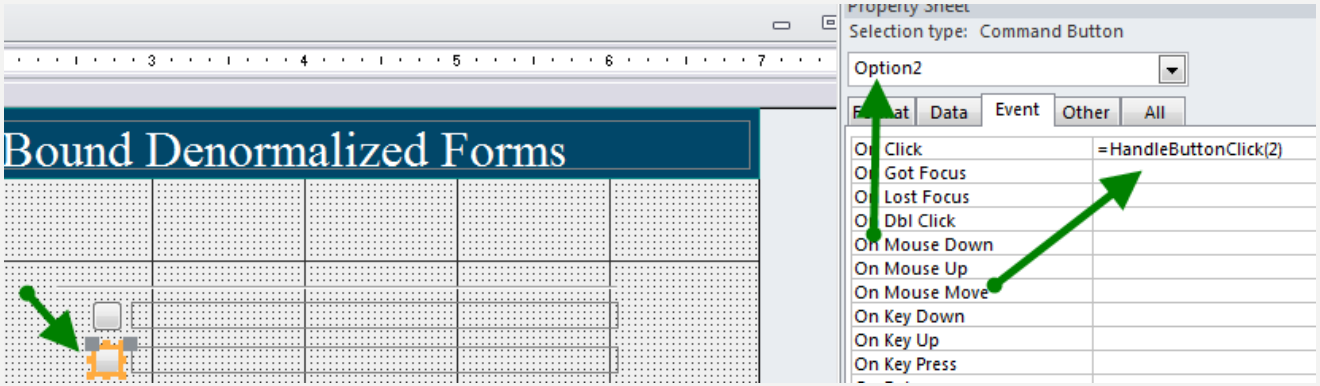
COLOR THE DS BACKGROUND

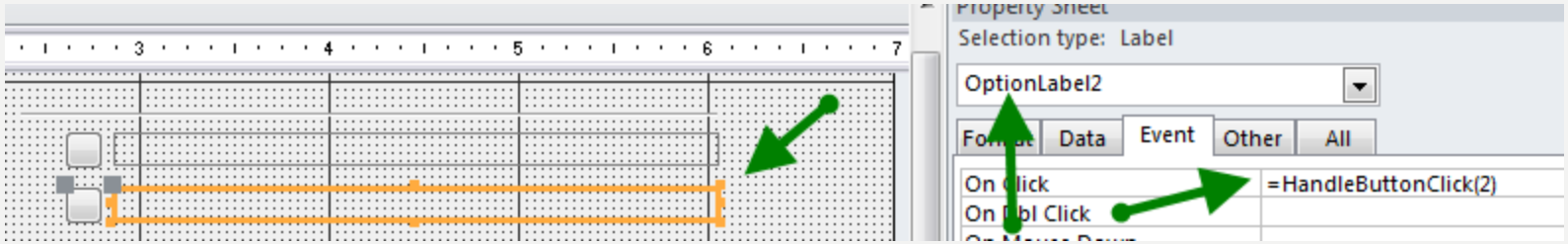
- Private Sub Form_Open(Cancel As Integer)
- Me.DatasheetBackColor = 8404992
- Me.DatasheetForeColor = -2147483633
- Me.DatasheetAlternateBackColor = 8404992
- End Sub

LONG SWITCHBOARD



1. Add the additional controls – 4 sets in this example
2. Name the controls using the existing pattern





3. Change the constant that defines the number of buttons

```
Private Sub FillOptions()
' Fill in the options for this switchboard page.

    ' The number of buttons on the form.
    Const conNumButtons = 12

    Dim dbs As DAO.Database
    Dim rst As DAO.Recordset
    Dim strSql As String
    Dim intOption As Integer

    ' Set the focus to the first button on the form,
    ' and then hide all of the buttons on the form
    ' but the first.  You can't hide the field with the f
    Me![Option1].SetFocus
```