

Re: Problem with VBA Code behind form

Source: <http://www.tech-archive.net/Archive/Access/microsoft.public.access.forms/2005-02/2214.html>

From: Tom (tmillington_at_aavf.co.uk)

Date: 02/18/05

Date: Fri, 18 Feb 2005 23:23:48 -0000

"Dirk Goldgar" <dg@NOdataSPAMgnostics.com> wrote in message
news:eZNU\$qqFFHA.228@TK2MSFTNGP15.phx.gbl...

> "Tom" <tmillington@aavf.co.uk> wrote in message

> news:5aedna5ocNQj-IvfRVnyhw@eclipse.net.uk

>>

>> *In answer to your questions*

>>

>>> *1. When you talk about the form's display being "corrupt", what do
>>> you mean? What are you actually seeing?*

>>

>> *A bit like parralel vertical bars or slats of wood, in that it looks
>> like some lines have moved up half a line leaving a gap underneath
>> through which you can see the report. I have never seen anything like
>> it before.*

>

> *Huh. I've never seen anything like that before. I'd be more inclined
> to suspect a problem with your video driver than anything else. What
> version and service-level of Access are you using, and what operating
> system?*

Access 2000 with all updates installed. Tried on two PCs: (a) Pentium 330
with 584Mb RAM running Windows 2000 and onboard graphics, and (b) Sony Vaio
laptop with Centrino 1.7M CPU and 512Mb Ram, running XP Pro with Radeon 64Mb
graphics card. Same on both.

>

>>> *2. What is the recordsource of the report? If you're getting no
>>> data on the report when you update the table first, it sounds as
>>> though there must be criteria in the recordsource that exclude the
>>> records you've just updated.*

>>>

>>

>> *The recordsource for both the form and the report are a query, the
>> basis of which is the same table that is updated by the other part of
>> the code. Basically, the query calls from the table details of all
>> existing operator cards with a status of 'N' for new. Only new cards
>> can be printed. Each record is displayed as part of the continuous
>> form and has a checkbox under a non-visible command button. Clicking*

>> on the button sets the check box, thereby selecting that record for
>> further processing.
>
> I'm curious. Why use a transparent button to check a check box? Why
> not just let the user check the check box itself? Is the check box not
> bound to a field in the form's recordsource?

I tried this originally but because it is a bound continuous form if you check one box it checks it for every record. Found this fix in this newsgroup and it seems to work.

>
>> When the user clicks on the print button, the
>> report is run using the same query as used to load the form, but the
>> selection is limited further by only choosing the records checked on
>> the form. I guess I can try using the table as the source for the
>> report to see what that does as the form is feeding through the
>> selection criteria. The update element just selects all records from
>> the underlying table where the record number (field REC_ID, an
>> autonumber field) is checked on the form.
>
> If your code behind the "Print" button first updates the table and sets
> the selected records' Status to 'P', then those records won't show up on
> the report, will they?

Yes, because the report uses the field REC_ID on the form where checkboxes are set to get the record numbers to update. The status is only checked when the form is loaded or opened to confirm which records are to be displayed for selection.

>
>> I can see how conflicts are occurring because everything sources
>> from, or is written back to, the same table. I suppose I could use
>> temporary tables but I am reluctant to do that as this is going to be
>> rolled out over a dozen PCs, after converting the back end to
>> server-based MSDE tables, rather than Access.
>
> I don't think see that there's any problem having everything bound to
> the same table, so long as you get the sequence of events right. In a
> multiuser, server-based environment, you could have editing conflicts if
> two users are working on the same set of records, but there are ways to
> deal with those, and I don't think you've even gotten to that point yet.
>
>>> 3. What is "MySelected"? I *guess* it's a function that returns the
>>> comma-delimited list of record IDs, but it would be nice to see
>>> what's in there.
>>>
>>
>> Good deduction. The code is below.
>>
>> =====
>>
>> Private Function MySelected() As String

```
>>
>> Dim i As Integer
>>
>> For i = 1 To colCheckBox.Count
>> If MySelected <> "" Then
>> MySelected = MySelected & ", "
>> End If
>> MySelected = MySelected & colCheckBox(i)
>>
>> Next i
>>
>> End Function
>>
>> =====
>
> Okay, so now I have to ask what "colCheckBox" is. Have you done
> something very clever to avoid binding the check boxes on your
> continuous form to a field in the form's recordsource?
```

I have reproduced somebody else's 'very clever' (courtesy of Albert D Kallal – <http://www.members.shaw.ca/AlbertKallal/msaccess/msaccess.html>). I had to tweak it a bit, but it works.

```
>
>>> 4. Is there code in the report that also updates the OPERATOR_CARDS
>>> table?
>>
>> No. I didn't think of trying that one.
>>
>> I'm sorry about the length of the reply. I just this helps you help
>> me.
>
> It does indeed, but there are still some hazy areas.
>
> Here is a how I think I would do something like this. If only one user
> (at a time) will be authorized to print these records, I'd probably
> define an additional value for the Status field, meaning "to be
> printed". The act of selecting a record on the form -- whether by a
> button on each record or by some other means -- would set the Status to
> that value.
>
> I'd change the report's recordsource to a query that returns only the
> records with a Status of "to be printed". That way, the report would
> need no special criteria at run time, and if it doesn't get printed the
> first time for some reason, the records will still be marked for a
> reprint. I'd have code in the Report's Close event that displays a
> MsgBox dialog to ask the user if all the records printed successfully.
> If the user answers yes, I'd execute an update query to change Status to
> some other value for all those records where Status was "to be printed",
> and then I'd requery the form (if it's open).
>
> On the other hand, if multiple users are going to be marking records and
```

> *printing, and each user must print only her own marked set of records,*
> *then I'd take a different approach. I'd probably use a local table in*
> *the front-end to store the IDs of the records to be printed -- selecting*
> *a record for printing would result in its ID being stored in the local*
> *table. Then the report would just be based on a query that inner joins*
> *the main table with the "IDsToPrint" table on the ID field, so only*
> *those records would get printed. If the report prints successfully, I'd*
> *first execute an update query to change the status of the records in the*
> *main table, and then execute a delete query to empty the IDsToPrint*
> *table.*

At this time, we believe there will only ever be one card administrator, and only about 100 or so active cards. With this low number of records in mind, I thought it would be simplest to work the following status sequence, thereby allowing a query just to look for a specific status when an action is required.

- 1). An operator can only have one active card at a time.
- 2). When a user is first allocated a card, it has status (N)ew.
- 3). The sequence under discussion here takes the (N)ew card, prints it, changes it status to (P)rinted and stamps the record's PRINT_DATE field.
- 4). The new card now exists (barcode format) but has to be (A)ctivated as it may not be convenient to hand over the card at the time it is printed. When they want to (A)ctivate the card, it (S)uspends any active card. The newly activated card goes to status (A)ctivated and has current date loaded into the ACTIVATED_DATE field. The old card has its status set to (S)uspended and has the current date loaded into its SUSPENDED_DATE field. In this way, we keep the transactions both separated, and provide our own audit trail. Each action also logs the administrator's ID and the PC network name.

For your interest, this is the administration part of a barcoded data collection system for reporting scrap during the manufacturing process: ie, operators scan in a job number, an operation number, the amount of items they received to work on, the number of goods ones produced, the number of scrap items, and a reason code for any scrap produced (all known as Work in Progress). The WiP side is the easy bit and I have already done that.

Regards

Tom

>
> --
> *Dirk Goldgar, MS Access MVP*
> *www.datagnostics.com*
>
> *(please reply to the newsgroup)*
>
>