

Re: Losing data in controls (and arrays) during unhandled exceptions.

Re: Losing data in controls (and arrays) during unhandled exceptions.

Source:

<http://www.tech-archive.net/Archive/Excel/microsoft.public.excel.programming/2006-12/msg01556.html>

- *From:* "NickHK" <TungCheWah@xxxxxxxxxxxxx>
 - *Date:* Wed, 6 Dec 2006 16:03:54 +0800
-

David,

Not sure what you have at the moment, although you seem to understand the concept.

Errors will "bubble up" through the routines until an active error handler is found. If one is not found, then you get your situation.

So if the first routine has a handler, you can deal with it there. Compare the output for i in routine below and when you comment out the "On Error" statement. This seems to be your situation.

e.g.

```
Dim i As Long
```

```
Private Sub Worksheet_SelectionChange(ByVal Target As Range)
```

```
On Error GoTo WS_SelectionChange_Handler
```

```
Debug.Print i
```

```
i = 10
```

```
Call No_Handler1
```

```
Exit Sub
```

```
WS_SelectionChange_Handler:
```

```
MsgBox Err.Description
```

```
End Sub
```

```
Private Function No_Handler1() As Long
```

```
Call No_Handler2
```

```
End Function
```

```
Private Function No_Handler2() As Long
```

```
'Cause an error
```

```
No_Handler2 = 1 / 0
```

```
End Function
```

NickHK

Re: Losing data in controls (and arrays) during unhandled exceptions.

Re: Losing data in controls (and arrays) during unhandled exceptions.

"David" <nodeo@xxxxxxxxxxxx> wrote in message
news:12nct27lq7pusa7@xxxxxxxxxxxxxxxxxxxxxxxxxxxx

Hi

Thanks for the quick reply!

"NickHK" <TungCheWah@xxxxxxxxxxxx> wrote in message
[news:uKvJ\\$5PGHHA.1216@xxxxxxxxxxxxxxxxxxxxxxxxxxxx](mailto:news:uKvJ$5PGHHA.1216@xxxxxxxxxxxxxxxxxxxxxxxxxxxx)

David,
Sounds like you need to include/improve your error handling. Do all
(necessary) routines have the statement;
On Error {one of the options }

Although this will not stop errors, it will allow you to test the
cause/current state and deal with the situation.

Most of the important functions have handlers.....the Worksheet
probably has more than a hundred subs and functions though....
I see your point though about improving the error handling
If the code was good enough unhandled exceptions shouldn't
actually happen.

As for populating with stored data, wouldn't the Workbook_Open event be
better ?

I can't actually remember why I chose Workbook_WindowActivate.
The first one I would have tried would have been Workbook_Open
but it was a while ago.....I'll look into it more, but that part seems to
work anyway (maybe I was hoping _Activate would be generated on
reset or something...??).

In VBA, the End statements causes all variables to be cleared. I would
assume this is effectively what happens in your error situation(s).
So basically, you need to deal with the error before it reaches this

stage.

You're probably right.....if there are no errors (or at least they're
handled), then there's no problem. I was just hoping that there was
some kind of event generated when the app is 'reset'. That would

Re: Losing data in controls (and arrays) during unhandled exceptions.

Re: Losing data in controls (and arrays) during unhandled exceptions.

cover any situations I haven't anticipated. Oh well.....I always feel better when I get a bit of confirmation I'm not missing something real obvious.

Thanks again!

David

NickHK

"David" <nodeo@xxxxxxxxxxxx> wrote in message
news:12ncmu2f100q367@xxxxxxxxxxxxxxxxxxxxxxxxxxxxx

Hi

I have a problem that's been plaguing me for a while now.

I have a fairly complicated Excel (97) Workbook made to create html pages (it's a template for generating html in effect). The root of the problem is that list boxes and arrays that contain data at runtime have to be persisted into a hidden worksheet when the app is shut down, otherwise it's lost. During startup, I found I can re-populate the controls and arrays from the hidden spreadsheet so the app is left in the same state as it was when shut down. This seems to work fine using the Workbook_WindowActivate event.

The real problem is when there's an un-handled exception. All data in the controls and arrays disappear, and once 'Reset' from the Visual Basic Editor, all the data is missing. In my program, if a person inadvertently saves the document (generates the html template), it's generated without the data... heh heh.....erasing my 'work in progress' in the html file. For the most part I can usually keep it together, but it's a hassle... and more than once I've generated a template with data missing.

(...by the way.....my code doesn't really generate all that many exceptions.....just a few.....sometimes.....but always at the worst possible moments.....)

I can see where a button could be set on one of the sheets to update the mess, but multiple working sheets mean multiple buttons ect. Shutting down the app at that point without saving also is not really an option, because all the other changes made since the last save are lost.....I guess that's normal when it comes to apps in general, but VBA allows you to continue

Re: Losing data in controls (and arrays) during unhandled exceptions.

from
where you left off, and that's a bit dangerous. Even if the app
was forced to completely terminate in such a case I think it
would be better. It would sure be better in the event someone
else needed to use it that wasn't so familiar with the
workings.

Is there some 'event' that is generated back to the Workbook
or app so that it knows it's in a 'reset' state??? That way, all
the
pages could be updated, such as at startup. I've tested all the
events generated in the Workbook_xxx but to no avail.....it
seems there is no indication whatsoever that the app has
been 'Reset' and the data is actually in quite a bit different
state
than it was before 'Reset'.

Maybe it's just a limitation I have to live with? I don't really
want to put error handlers in every function and
sub.....but..??
Could wrap the whole thing in a VB app, but that's a big step
and the convenience of Excel for such things would be
lost.....

Thanks for any help!

David Otte