

## Re: Rolling back 'folders delete' using VBA

---

*Source:*

<http://www.tech-archive.net/Archive/Excel/microsoft.public.excel.programming/2007-08/msg04617.html>

---

- *From:* Nayan <[Nayan@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx](mailto:Nayan@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx)>
  - *Date:* Wed, 29 Aug 2007 12:50:03 -0700
- 

Hi Bob,

Sorry for delayed reply.

The fact is that I know only 4 conditions where you can not delete files/folders at run time using File system object.

There could be other conditions too (that I have not faced so far).

I was thinking of a process in which we actually start cutting the files in a folder (for eg C:\Temp) one by one and paste in a different location (say C:\Temp1 created at run time ) with same sub folder structure.

if we come across an error while deleting a file from original location. ( on path C:\Temp) then we restore back all the files that we already cut from new location (C:\Temp1) to original location.

Thus we may store new location of each file in an array while we cut them. On error go through this array and copy back all the files and delete the folder (C:\Temp1) created for backup at runtime.

Please let me know if there is any better way.

Thanks

Nayan

"Bob Phillips" wrote:

I can do you the first two, but I don't know how to do the second two.

--

---

HTH

Bob

Re: Rolling back 'folders delete' using VBA

(there's no email, no snail mail, but somewhere should be gmail in my addy)

"Nayan" <Nayan@xxxxxxxxxxxxxxxxxxxxxxxxxxxx> wrote in message  
news:A2BF4BB5-E66E-4D30-AD08-1C4FF6D184CC@xxxxxxxxxxxxxxxxxxxx

Hi Bob,

Thanks for the code you sent.

I have gone through the code and found that if you keep a read only file in C:\test folder, while deleting that file it will produce "Path/file access" error.

We need to know ( before kill commnad " Kill arfiles(i)" in the code) if the file can be deleted or not

The few scenarios in which file can not be deleted are...

- 1) If its a read only file
- 2) If its already opened in an application.
- 3) if its currrupted
- 4) if its a source safe file

Thus just opening a file successfully and adding it to an array does not confirm that we can delete that file.

Any thoughts or suggestions?

Thanks in advance for your help.

Nayan.

"Bob Phillips" wrote:

In the code that you have, I presume that there is a test that traps the error. AT this point, if and when an error is encountered, then bale out.

If an error is not encountered, add the file path and name to the array, and continue.

If you get to the end without baling out then the array is full

## Re: Rolling back 'folders delete' using VBA

of  
deletable  
items.

Something like this

Option Explicit

```
Private cnt As Long
Private arfiles
Private FSO As Object
```

```
Sub Folders()
Const START_FOLDER = "C:\Test"
Dim i As Long
```

```
Set FSO = CreateObject("Scripting.FileSystemObject")
cnt = 0
ReDim arfiles(1 To 1)
If SelectFiles(START_FOLDER) Then
For i = LBound(arfiles) To UBound(arfiles)
Kill arfiles(i)
Next i
End If
End Sub
```

---

```
Public Function SelectFiles(Optional sPath As String) As
Boolean
```

---

```
Dim oSubFolder As Object
Dim oFolder As Object
Dim oFile As Object
Dim oFiles As Object
```

```
SelectFiles = True
Set oFolder = FSO.GetFolder(sPath)
Set oFiles = oFolder.Files
For Each oFile In oFiles
On Error Resume Next
Workbooks.Open Filename:=oFile.Path
On Error GoTo 0
If Err.Number <> 0 Then
SelectFiles = False
Exit Function
Else
cnt = cnt + 1
ReDim Preserve arfiles(1 To cnt)
arfiles(cnt) = oFile.Path
ActiveWorkbook.Close savechanges:=False
End If
```

Re: Rolling back 'folders delete' using VBA

Next oFile

```
For Each oSubFolder In oFolder.Subfolders
If Not SelectFiles(oSubFolder.Path) Then
SelectFiles = False
Exit Function
End If
Next
```

End Function

--  
---  
HTH

Bob

(there's no email, no snail mail, but somewhere should be  
gmail in my  
addy)

"Nayan" <Nayan@xxxxxxxxxxxxxxxxxxxxxxxxxxxx> wrote in  
message  
[news:E36CE6D4-0CB7-4EA5-ACEC-D351BFBF7138@xxxxxxxxxxxxxxxxxxxx](mailto:news:E36CE6D4-0CB7-4EA5-ACEC-D351BFBF7138@xxxxxxxxxxxxxxxxxxxx)

Bob,

Thanks for the idea.

Can you please let me know how to check if  
a file is deletable or not  
before  
deleting it.

Thanks again

Nayan

"Bob Phillips" wrote:

The best way would be to  
not delete anything, but to  
build an array of  
all  
deletable files. When you  
get the error, just quit, if you  
don't,

Re: Rolling back 'folders delete' using VBA

iterate  
the array and delete them  
all.

--  
----  
HTH

Bob

(there's no email, no snail  
mail, but somewhere should  
be gmail in my  
addy)

"Nayan"  
<Nayan@xxxxxxxxxxxxxxxxxxxxxxxxxxxx>  
wrote in message  
[news:5FFBE6EF-D398-4593-A9DA-CF6EBDFA1655@xxxxxxxxxxxxxxxxxxxx](mailto:news:5FFBE6EF-D398-4593-A9DA-CF6EBDFA1655@xxxxxxxxxxxxxxxxxxxx)

Hello,

(Environment  
Excel 2003)  
(operating  
system  
Windows  
XP  
professional)

I am  
deleting a  
root folder  
in a vba  
macro using  
filesystem  
object.  
This  
root folder  
has many  
subfolders  
in it.

If any of the  
files (any  
type of the  
file) in any  
subfolder  
can not

Re: Rolling back 'folders delete' using VBA

be  
deleted for  
any reason  
(such as if  
file is read  
only or  
already  
opened  
in  
an  
application)  
then I get  
'permission  
denied'  
error. At  
this point I  
want  
to  
rollback the  
entire  
deleting  
process.

Basically on  
any type of  
error while  
deleting the  
root folder I  
want  
to  
rollback the  
deleting  
process.

Any ideas  
or  
suggestion  
as how this  
can be  
achieved  
programmatically  
in  
Excel  
macro.

Thanks in  
advance

Nayan

Re: Rolling back 'folders delete' using VBA