

Access import Excel – multiple Worksheets from multiple Workbooks

Source:

<http://www.tech-archive.net/Archive/Access/microsoft.public.access.externaldata/2007-02/msg00175.html>

- *From:* InOverHead <InOverHead@xxxxxxxxxxxxxxxxxxxxxxxxxxxx>
 - *Date:* Wed, 21 Feb 2007 07:43:13 -0800
-

Summary: I am attempting to write an access 2003 (11.5614.5606) routine to automate importing multiple excel workbooks (25+) each containing multiple worksheets (280+) into a single table in access. Each worksheet has the same columns and headings. I can create a link to each table or import each table into a single access file, but if I try to get access to process through the entire directory appending records I get an error. I used the windows API file browse dialog in some of the other examples to select the directory containing the workbooks. Currently this code processes part of the workbooks then crashes with an Error 3620 – The connection for viewing your linked Microsoft Excel worksheet was lost.

*****Code*****

```
Option Compare Database
Option Explicit
Dim varRet As Variant 'System Message

Sub Link_To_Excel(Optional LinkImportUnion As String)
'Macro Loops through the specified directory (strPath)
'and passes filenames for ALL Excel files to module for
'linking tables in the Access Database.

Dim strPath As String 'Directory Path
Dim strFile As String 'Filename
Dim strFileList() As String 'File Array
Dim intFile As Integer 'File Number

strPath = BrowseFolder("Select Excel Files directory to process!")
If strPath = "" Then
MsgBox ("Cancelled. No Directory Selected.")
Exit Sub
Else
strPath = TrailingSlash(strPath)
'Loop through the folder & build file list
strFile = Dir(strPath & "*.xls")
While strFile <> ""
'add files to the list
intFile = intFile + 1
ReDim Preserve strFileList(1 To intFile)
```

Access import Excel – multiple Worksheets from multiple Workbooks

```
strFileList(intFile) = strFile
strFile = Dir()
Wend
'see if any files were found
If intFile = 0 Then
MsgBox "No files found"
Exit Sub
End If
'cycle through the list of files & link to Access
'Check out the TransferSpreadsheet options in the Access Visual
Basic Help
'file for a full description & list of optional settings
For intFile = 1 To UBound(strFileList)
Call GetXLsht(strPath & strFileList(intFile), LinkImportUnion)
Next
MsgBox UBound(strFileList) & " Files were Linked"
End If
End Sub

*****

Private Sub cmdAbout_Click()
DoCmd.OpenForm "USysAboutScreen"
End Sub

Private Sub cmdDeleteLink_Click()
LinksDelete
End Sub

Private Sub cmdQuit_Click()
DoCmd.Quit
End Sub

Private Sub cmdUnion_Click()
Dim strSQL As String
If fExistTable("XLAssembled") Then ' check if import table exists
strSQL = "DROP TABLE [XLAssembled];"
DoCmd.RunSQL strSQL
End If
Call Link_To_Excel("Union")

End Sub

*****

Private Sub GetXLsht(xlInFile As String, strAction As String)

Dim XL As Excel.Application
Set XL = CreateObject("Excel.Application")
Dim xlWrkBk As Excel.Workbook
Dim xlsht As Excel.Worksheet
Dim xlRow As Long
Dim xlCol As Long
```

Access import Excel – multiple Worksheets from multiple Workbooks

```
Dim xlshtcnt As Integer
Dim actioncase As Integer
Dim varRet As Variant
Dim strSQL As String
Set xlWrkBk = GetObject(xlInFile)
DoCmd.SetWarnings False
On Error GoTo ok_error
If strAction = "Import" Then actioncase = 1
If strAction = "Link" Then actioncase = 2
If strAction = "Union" Then actioncase = 3
For xlshtcnt = 1 To xlWrkBk.Worksheets.Count
Set xlsht = xlWrkBk.Worksheets(xlshtcnt)
Debug.Print xlWrkBk.Worksheets(xlshtcnt).Name
'delete table if it exists
' On Error Resume Next
Select Case actioncase
Case Is = 1 ' Import
varRet = SysCmd(acSysCmdSetStatus, "Now Importing '" &
xlWrkBk.Worksheets(xlshtcnt).Name & "'....")
DoCmd.TransferSpreadsheet acImport, acSpreadsheetTypeExcel9,
xlWrkBk.Worksheets(xlshtcnt).Name, xlInFile, True
Case Is = 2 ' Link
varRet = SysCmd(acSysCmdSetStatus, "Now linking '" &
xlWrkBk.Worksheets(xlshtcnt).Name & "'....")
DoCmd.TransferSpreadsheet acLink, acSpreadsheetTypeExcel9,
xlWrkBk.Worksheets(xlshtcnt).Name, xlInFile, True
Case Is = 3 ' Union
varRet = SysCmd(acSysCmdSetStatus, "Creating Union '" &
xlWrkBk.Worksheets(xlshtcnt).Name & "'....")
If Not fExistTable("XLAssembled") Then ' check if import table
exists
'code to create table from header on first worksheet
DoCmd.TransferSpreadsheet acImport, acSpreadsheetTypeExcel9,
xlWrkBk.Worksheets(xlshtcnt).Name, xlInFile, True
DoCmd.Rename "XLAssembled", acTable,
xlWrkBk.Worksheets(xlshtcnt).Name
Else
'code to append data
' create VBA link to XI data worksheet
DoCmd.TransferSpreadsheet acLink, acSpreadsheetTypeExcel9,
xlWrkBk.Worksheets(xlshtcnt).Name, xlInFile, True
'insert excel records into table
strSQL = "INSERT INTO XLAssembled SELECT [" &
xlWrkBk.Worksheets(xlshtcnt).Name & "].* FROM [" &
xlWrkBk.Worksheets(xlshtcnt).Name & "];"
DoCmd.RunSQL strSQL
' Delete VBA link to XI data worksheet
DoCmd.DeleteObject acTable, xlWrkBk.Worksheets(xlshtcnt).Name
End If

Case Else
```

Access import Excel – multiple Worksheets from multiple Workbooks

```
varRet = SysCmd(acSysCmdSetStatus, "Error Occurred Incorrect
Parameter passed to GetXLsht...")
MsgBox ("Incorrect Parameter passed to GetXLsht")
End Select
varRet = SysCmd(acSysCmdSetStatus, "Successfully linked "" & xlshtcnt &
"" worksheets from workbook....")
Next xlshtcnt
varRet = SysCmd(acSysCmdSetStatus, " ")
DoCmd.SetWarnings True
XL.Quit
Set XL = Nothing
Set xlWrkBk = Nothing
Set xlsht = Nothing
Set varRet = Nothing

Exit Sub
ok_error:
MsgBox Err.Description, , "ERROR " & Err.Number & " OK"
'press F8 to step through code and fix problem
Stop
XL.Quit
Set XL = Nothing
Set xlWrkBk = Nothing
Set xlsht = Nothing
Set varRet = Nothing
Resume
End Sub
*****

Sub LinksDelete(Optional strConnectionString As String = "")
'This function removes links to tables with specified connections
'If strConnectionString is omitted all links will be removed
Dim tdf As TableDef

For Each tdf In CurrentDb.TableDefs
If tdf.Connect <> "" Then
If InStr(1, tdf.Connect, strConnectionString, vbTextCompare) > 0 Then
varRet = SysCmd(acSysCmdSetStatus, "Link "" & tdf.Name & ""
now being removed....")
DoCmd.DeleteObject acTable, tdf.Name
End If
End If
Next tdf
varRet = SysCmd(acSysCmdSetStatus, " ")
End Sub

*****End Code*****
```