**Batch Template Changes**

Templates are used to quickly define a standard look for a document or a group of documents. The wonderful thing about templates is that you can define one template to give your document one look, and another template to give it an entirely different look. All you need to do, of course, is change which template is associated with a document.

[Changing the template associated with one or two documents is quite easy](http://wordribbon.tips.net/T005998_Changing_the_Attached_Template.html). What if you have a directory full of documents whose templates need to be changed? This can get quite tedious very quickly. This is where a macro can come to the rescue—to relieve that old tedium and do the mundane very quickly. The following macro, ChangeTemplates, modifies all the documents in a particular directory to make sure they use the template you want.

Sub ChangeTemplates()

 Dim strDocPath As String

 Dim strTemplateB As String

 Dim strCurDoc As String

 Dim docCurDoc As Document

 ' set document folder path and template strings

 strDocPath = "C:\path to document folder\"

 strTemplateB = "C:\path to template\templateB.dotx"

 ' get first doc - only time need to provide file spec

 strCurDoc = Dir(strDocPath & "\*.doc?")

 ' ready to loop (for as long as file found)

 Do While strCurDoc <> ""

 ' open file

 Set docCurDoc = Documents.Open(FileName:=strDocPath & strCurDoc)

 ' change the template

 docCurDoc.AttachedTemplate = strTemplateB

 ' save and close

 docCurDoc.Close wdSaveChanges

 ' get next file name

 strCurDoc = Dir

 Loop

 MsgBox "Finished"

End Sub

In order to use the macro, simply make sure that the strDocPath and strTemplateB variables are set properly. The macro changes the template associations for all documents in a particular directory. If you want something a little more discriminating, then a different macro is in order. For instance, you may want to have the macro examine each document and only change those that use TemplateA so that they now use TemplateB. In this instance you will find the following macro very handy:

Sub TemplateBatchChange()

 Dim objPropertyReader

 Dim strFolder As String

 Dim strFileName As String

 Dim objThisDoc As Word.Document

 Dim strFindTemplate As String

 Dim strReplaceTemplate As String

 Dim strAffectedDocs As String

 On Error Resume Next

 'Create the PropertyReader object

 Set objPropertyReader = CreateObject("DSOleFile.PropertyReader")

 If Err.Number <> 0 Then

 MsgBox "You must install the DSOleFile component. See " & \_

 "http://support.microsoft.com/kb/224351"

 GoTo FinishUp

 End If

 'Get the template names

 strFindTemplate = UCase(InputBox("Name of template to find"))

7

 strReplaceTemplate = InputBox("Name of replacement template")

 'Make sure it's a valid template. Try to create a new document based on it.

 Set objThisDoc = Word.Documents.Add(strReplaceTemplate, Visible:=False)

 If Err.Number <> 0 Then

 'No such template

 MsgBox "There is no accessible template named " & strReplaceTemplate

 GoTo FinishUp

 End If

 'Close the test document

 objThisDoc.Close wdDoNotSaveChanges

 On Error GoTo ErrorHandler

 'Get the current documents path

 strFolder = Word.Application.Options.DefaultFilePath(wdDocumentsPath) \_

 & Word.Application.PathSeparator

 'Examine all Word documents in the directory

 'Get the first document name

 strFileName = Dir(strFolder & "\*.doc?")

 While strFileName <> ""

 'Look at the template name

 If UCase(objPropertyReader.GetDocumentProperties \_

 (strFolder & strFileName).Template) = strFindTemplate Then

 'It matches. Open the document

 Set objThisDoc = Word.Documents.Open \_

 (FileName:=strFileName, Visible:=False)

 'Change the attached template

 objThisDoc.AttachedTemplate = strReplaceTemplate

 'Save the change

 objThisDoc.Close wdSaveChanges

 'Note the document

 strAffectedDocs = strAffectedDocs & strFileName & ", "

 End If

 'Get the next document

 strFileName = Dir

 Wend

 'Report the results

 If strAffectedDocs = "" Then

 MsgBox "No documents were changed.", , "Template Batch Change"

 Else

 'Remove the trailing comma and space

 strAffectedDocs = Left(strAffectedDocs, Len(strAffectedDocs) - 2)

 MsgBox "These documents were changed: " & \_

 strAffectedDocs, , "Template Batch Change"

 End If

 GoTo FinishUp

ErrorHandler:

 Set objThisDoc = Nothing

 Set objPropertyReader = Nothing

 Err.Raise vbError + 1001, "TemplateBatchChange", \_

 "TemplateBatchChange encountered an error: " & Err.Description

FinishUp:

 'Release object references

 Set objThisDoc = Nothing

 Set objPropertyReader = Nothing

End Sub

In order to use this macro, you must first make sure that the DSOLEFILE component (available free from Microsoft) is installed on your system. You can find out more about this component by referring to Microsoft's 224351 Knowledge Base article. The macro provides you a chance to specify a template name to be replaced and the name of the template to replace it with. It even checks to make sure that the replacement template exists.

This macro searches for documents in the default Word document folder. If you want to search in a different directly, you should make sure that the strFolder variable gets set to the full path of the folder you want used.