

How to update existing items in an Outlook folder to use a new custom form

This article was previously published under Q201087

Article ID	: 201087
Last Review	: August 13, 2004
Revision	: 1.0

SUMMARY

This article describes how to update items in an Outlook folder so that those items open using a different form than the form that the items are currently associated with.

MORE INFORMATION

For example, you might want to change the form that is used for all of the items in a folder in the following situations:

- You use the default Outlook form to enter 10 contacts into your Contacts folder. You then create a custom form for contacts and enter 10 additional contacts. You want the first 10 contacts to use the new custom form when they are opened.
- You create a custom form for contacts and enter 10 contacts using the custom form. You then import 100 contacts from a text file. The 100 imported contacts use the default form instead of the custom form.
- You have a public folder that contains 1,000 posted items based on the default post form. You then create a custom form that shows the items in a way that is important to your work. You want to apply the new form to the 1000 existing items.
- You installed the Small Business Customer Manager or the Small Business Forms Manager (both which change your default contact form), and you want to change back to another form.

A property of the item called "message class" determines the form the item uses. You cannot change the message class of an item manually. However, you can write Microsoft Visual Basic Scripting Edition (VBScript) or Visual Basic Automation code to change the message class for all of the items in a folder.

When you create and publish a custom form, the form is assigned a message class. This message class determines which form is associated with an item. The format of the name is "IPM.*Form_Type*.*Form_Name*", where *Form_Type* is the type of form (Contact, Task, and so on) and *Form_Name* is the name of the custom form. For example, if you create a new contact form, name it Revised, and then publish it to your Contacts folder, the message class is IPM.Contact.Revised.

NOTE: In each of the following operations, it is imperative that you enter the new message class name exactly as the name that was used when the form was published.

The following table lists the various names that are used for message classes:

Item	Default folder	Default message class
Contact	Contacts	IPM.Contact
Task	Tasks	IPM.Task
Appointment	Calendar	IPM.Appointment
Note	Notes	IPM.StickyNote
Journal Entry	Journal	IPM.Activity
Mail	Inbox	IPM.Note

NOTE: You cannot customize and publish the Note form.

To see the message class for an existing item, add the Message Class field as one of the columns in the current view. The message class in this view is read-only; you cannot type a different message class to change the form manually.

To add the message class to your view, follow these steps:

1. Change the view to a table view, such as the **Phone List** view in the Contacts folder.
2. Right-click the column-header in the view, and then click **Field Chooser** on the shortcut menu.
3. In the **Field Chooser** list, click to select **All Contact Fields**.
4. Drag the **Message Class** field to the view column-header to add the field as a column.

To change the Message Class field of existing items, you need to use Visual Basic Scripting Edition (VBScript) code in an Outlook form, or Visual Basic code from another program, to automate Outlook and change the Message Class fields.

You can use the following two methods to change Message Class fields. Use the second method only if you do not have Microsoft Word 97 or Microsoft Word 2000, or if you cannot obtain the Omsgclas.exe file.

Download the Omsgclas.exe Utility

To change Message Class fields, download Omsgclas.exe, which contains a Word 97 or Word 2000 document with a macro that

changes Outlook message classes. This is the same utility that is available for Microsoft Outlook 97 and Word 97. It also works with Outlook 2000 and Word 2000. The macro runs automatically when you open the document.

When you open Omsgclas in Word 2000, you may find that the Word document will come up but the macro will not run. To correct this problem, you must reset the macro security in Word 2000. While in Word, point to Macro on the Tools menu and click Security. Change the security to low or medium. This will enable the macro to run.

For more information on obtaining the Omsgclas.exe file, please see the following article in the Microsoft Knowledge Base:

[201089](http://support.microsoft.com/kb/201089/EN-US/) (<http://support.microsoft.com/kb/201089/EN-US/>) Word document to change message class of Outlook items

Create a VBScript Routine

Microsoft provides programming examples for illustration only, without warranty either expressed or implied, including, but not limited to, the implied warranties of merchantability and/or fitness for a particular purpose. This article assumes that you are familiar with the programming language being demonstrated and the tools used to create and debug procedures. Microsoft support professionals can help explain the functionality of a particular procedure, but they will not modify these examples to provide added functionality or construct procedures to meet your specific needs. If you have limited programming experience, you may want to contact a Microsoft Certified Partner or the Microsoft fee-based consulting line at (800) 936-5200. For more information about Microsoft Certified Partners, please visit the following Microsoft Web site:

<http://www.microsoft.com/partner/referral/> (<http://www.microsoft.com/partner/referral/>)

For more information about the support options that are available and about how to contact Microsoft, visit the following Microsoft Web site:

<http://support.microsoft.com/default.aspx?scid=fh;EN-US;CNTACTMS> (<http://support.microsoft.com/default.aspx?scid=fh;en-us;contactms>)

Follow the steps below to create and run a VBScript routine that will change all the items in a folder to a specified form. This example assumes that you have published a new form called MyNewForm in the current folder. If you use a different title for your form, modify the form title used in the third line of code in the section "Enter the VBScript Code."

There are three tasks to this solution.

- Create a new item to store the VBScript code.
- Enter the VBScript code and save the form.
- Run the VBScript code.

Create a New Item to Store the VBScript Code

1. On the **File** menu, point to **New**, and then click **Mail Message**.
2. On the **Tools** menu, point to **Forms**, and then click **Design This Form** to enter form design mode.

Enter the VBScript Code and Save the Form

1. On the **Form** menu, click **View Code**.
2. In the Script Editor, type the following code. You do not need to enter the lines that begin with an apostrophe, since these lines are comments that are ignored when executed.

```
Sub Item_Open

    ' Change the following line to your new Message Class
    NewMC = "IPM.Contact.MyNewForm"

    Set CurFolder = Application.ActiveExplorer.CurrentFolder
    Set AllItems = CurFolder.Items
    NumItems = CurFolder.Items.Count

    ' Loop through all of the items in the folder
    For I = 1 to NumItems

        Set CurItem = AllItems.Item(I)

        ' Test to see if the Message Class needs to be changed
        If CurItem.MessageClass <> NewMC Then

            ' Change the Message Class
            CurItem.MessageClass = NewMC
        End If
    Next I
End Sub
```

```
        ' Save the changed item  
        CurItem.Save  
  
    End If  
  
Next  
  
MsgBox "Done."  
  
End Sub
```



3. On the **File** menu, click **Close**.
4. On the **File** menu, click **Save As**. Make sure the default setting for file type is **Outlook Template (.oft)**, and then select a location to save the file. Enter a file name for the form and then click **OK**.
5. Close the item by clicking the **X** in the upper-right corner of the item window and then click **No** when prompted to save changes.

Run the VBScript Code

1. Open the folder that contains the items you wish to update.
2. To run the VBScript code, open the item again by using Windows Explorer to locate the file and then double-click the .oft file. The code will run automatically because it was entered into an **Item_Open** event procedure. If you receive a macro warning, click **Enable Macros**.
3. Wait while the code changes the message class for all of the items in the currently selected folder. Depending on the number of items, this may take several minutes. When the code finishes, you should receive a message that says **Done**.

NOTE: If you wish to edit the VBScript code later to change the name of the message class, hold down the SHIFT key when you open the item. This prevents the VBScript code from executing and you can go into design mode, make changes to the VBScript code, and save the form.

REFERENCES

For additional information about available resources and answers to commonly asked questions about Microsoft Outlook 2000 solutions, please see the following articles in the Microsoft Knowledge Base:

[146636](http://support.microsoft.com/kb/146636/EN-US/) (http://support.microsoft.com/kb/146636/EN-US/) Questions about custom forms and Outlook solutions

[241220](http://support.microsoft.com/kb/241220/EN-US/) (http://support.microsoft.com/kb/241220/EN-US/) General information about the Existing Items converter

APPLIES TO

- Microsoft Outlook 2000 Standard Edition

Keywords: kbhowto kbprogramming kbautomation kbdisplay kbusage kbupdate kbopenfile KB201087