October, 2006

Advisor Answers

Identify Outlook Objects

VFP 9/8/7

Q: I'm automating Microsoft Outlook and I'd like to find a way to add a unique identifier to each item I'm working with, so I can track the items in my VFP database. Is there a way to do this?

A: In fact, it's already done for you. Every item in Outlook has an EntryID property that contains a GUID. So all you have to do is work with this existing property. For example, to store the EntryID of the first message in the Inbox, you can use code like this:

```
oOutlook = CREATEOBJECT("Outlook.Application")
oNameSpace = oOutlook.GetNameSpace("MAPI")

oInbox = oNameSpace.GetDefaultFolder(6) && Inbox
IF oInbox.Items.Count > 0
   oItem = oInbox.Items[1]
   cID = oItem.EntryID
   * Do something with it
ENDIF
```

Newly created items aren't assigned an EntryID until they're saved, so you need to populate the new item and call the Save method before grabbing the value. For example, this code creates a ContactItem, saves it and then retrieves the EntryID:

```
oContact = oOutlook.CreateItem(2) && ContactItem
WITH oContact
   .FullName = "Bill Gates"
   .Email1Address = "billg@microsoft.com"
   .Save()

   cID = .EntryID
   * Do something with it
ENDWITH
```

Be aware however, that in some circumstances, the EntryID for an item can change. In particular, for a personal store (PST), the EntryID changes when an item moves from the Drafts folder to another folder such as Sent Items.

Once you have an EntryID, finding the item in Outlook is easy. Just call the GetItemFromID method of the NameSpace object:

```
oItem = oNameSpace.GetEntryFromID(cID)
```

–Tamar

Check Whether a Word Document is Saved

VFP 9/8/7

Q: When automating Microsoft Word, I'd like to know whether a document has been saved or if I need to save it with code.

A: This is one of those items the designers of Word anticipated. The Word Document object has a Saved property, which contains a logical value. To check whether a document has been saved, just read the property. To save the document, call the Save method.

```
IF NOT oDocument.Saved
   oDocument.Save()
ENDIF
```

--Tamar

Open a Document Read-only

VFP 9/8/7

Q: In automating Word and Excel, I'd like to open documents and workbooks read-only, so that users can't change them. Is this possible?

A: The answer is "it depends." It depends what you mean by read-only. To me, opening something read-only means that it can't be changed. The Office applications take a different view of read-only—when you open a document or workbook read-only, you're unable to resave it to the same file, but you can modify it and save it with a new name.

If that approach is sufficient for your purposes, then you simply need to add a parameter to the Open method. In both Word and Excel, it's the third parameter and it's logical. In both cases, you can probably ignore the second parameter, passing .F.:

```
oDocument = oWord.Documents.Open("MyDocument.DOC", .f., .t.)
oWorkbook = oXL.Workbooks.Open("MyWorkbook.XLS", .f., .t.)
```

As always, you need to pass the fully-pathed filename to the Open method.

Once a document or workbook is open read-only, an interactive user can save changes by choosing File > Save As from the menu and specifying a new name. With Automation, you use the SaveAs method to accomplish the same thing.

If preventing the user from saving the changed document with a new name isn't good enough, you can disable the Save As item on the menu. To do so, you have to dig into the command hierarchy of Word and Excel. The CommandBars collection contains a list of all the menu bars and toolbars. You need to access the "Menu Bar" member of the collection. Then, you need to drill into the individual menu pads using the Controls collection. Finally, to access an individual item on a menu pad, you use the accChild collection. Each item has an Enabled property. For the English version of Word 2003, the following code works:

```
oMenu = oWord.CommandBars("Menu Bar")
oPad = oMenu.Controls("File")
oItem = oPad.accChild(5) && Save As is 5th on the menu
oItem.Enabled = .F.
```

You'll have to adjust the code based on the version and language you're using. Don't forget to re-enable the menu item when you're done.

–Tamar