

OUTLOOK E-Mails separat in einem ausgewählten Zielverzeichnis sichern

Dr. Volker Thormählen, 2. März 2019

Inhalt

Abbildungen	I
Listings	I
Tabellen.....	I
1 Problemstellung	1
2 Benutzerdialoge	2
2.1 Order in OUTLOOK auswählen	2
2.2 Auswahl des Zielordners.....	2
3 Ergebnis einer OUTLOOK-Sicherung.....	3
4 Benötigte VBA-Prozeduren.....	3
4.1 Hauptprogramm	3
4.2 Aufrufe im Hauptprogramm	5
4.3 Listing des benutzerdefinierten Unterprogramms	5
4.4 Listings der benutzerdefinierten Funktionen.....	6
5 Praktische Relevanz.....	6
Literaturverzeichnis.....	II

Abbildungen

Abb. 1: Quellordner in OUTLOOK auswählen	2
Abb. 2: Auswahl eines Zielordners	2
Abb. 3: Ausschnitt aus dem Zielverzeichnis.....	3

Listings

Listing 1: Hauptprogramm.....	4
Listing 2: OUTLOOK-Verzeichnisse holen	5
Listing 3: Pfad des Ziel-Ordners bestimmen	6
Listing 4: Ungültige Zeichen aus einem Verzeichnispfad entfernen	6
Listing 5: Bestimmte Sonderzeichen aus dem Betreff einer OUTLOOK-E-Mail-Nachricht entfernen	6

Tabellen

Tab 1: Aufrufe im Hauptprogramm	5
---------------------------------------	---

1 Problemstellung

In diesem Beitrag wird ein kommentiertes, interaktives VBA¹-Makro präsentiert mit dem OUTLOOK-E-Mails² in einem ausgewählten Zielverzeichnis auf einem verfügbaren Datenträger eines Rechners unter Windows 7 gesichert werden können.

Der Name des VBA-Makros lautet: *SaveAllEmails_ProcessAllSubFolders*. Die Suche danach im Internet führt zu 99 Treffern. Aber keine der stichprobenweise geöffneten Fundstellen enthält eine ausführliche Kommentierung des Quellcodes und/oder eine aussagekräftige Funktionsbeschreibung der Anwendung. Das Ziel dieses Beitrags ist, die genannten Defizite zu beheben.

Das VBA-Makro erfordert zwei Eingaben des Benutzers im Dialog:

- Auswahl eines Quellordner in OUTLOOK
- Auswahl eines Zielordners auf einem verfügbaren Datenträger

Alle E-Mails im Quellordner (nebst dazugehörigen Unterordnern) werden in einem ausgewählten Ordner auf einem verfügbaren Datenträger eines Rechners unter Windows 7 gespeichert. Die Ordnerhierarchie wird beibehalten (d. h. die Dateien werden in benannten Ordnern gespeichert, die dieselbe Struktur wie die OUTLOOK-Ordner haben.) Jede OUTLOOK-E-Mail-Nachricht wird separat gespeichert. Ein Doppelklick darauf genügt, um sie in OUTLOOK zu öffnen und anzuzeigen.

Die folgende Darstellung des VBA-Makros ist angelehnt an [1]. Nicht beschrieben wird allerdings, wie ein Makro in OUTLOOK erstellt wird. Im Internet sind genug einschlägige Anleitungen zu finden.

Am Ende dieses Beitrags wird auf die Notwendigkeit der ordnungsgemäßen Sicherung betrieblicher E-Mails nach den gesetzlichen Vorgaben in Deutschland hingewiesen.

¹ Dieses Akronym steht für **V**isual **B**asic for **A**pplications. Das ist eine proprietäre Programmiersprache der Firma *Microsoft* für deren *Office-Suite* unter dem Betriebssystem *Windows*.

² *Outlook* ist in erster Linie ein proprietäres E-Mail-Dienstprogramm der Firma *Microsoft* unter dem Betriebssystem *Windows*.

2 Benutzerdialoge

2.1 Order in OUTLOOK auswählen

Der erste Benutzerdialog beinhaltet die Auswahl eines Quellordners in OUTLOOK:

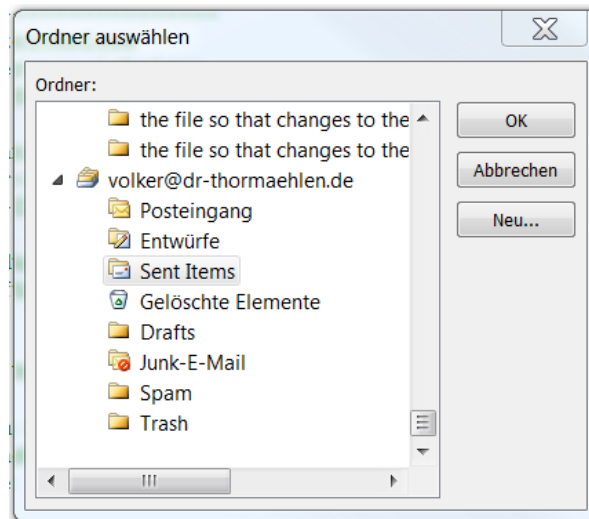


Abb. 1: Quellordner in OUTLOOK auswählen

In Abb. 1 wird dargestellt, dass der Ordner *Sent Items* (dt.: *Gesendete Objekte*) markiert ist. Nach Betätigung der Schaltfläche *OK* ist dieser Ordner als Datenquelle ausgewählt.

2.2 Auswahl des Zielordners

Der zweite Benutzerdialog beinhaltet die Auswahl eines Zielordners auf einem verfügbaren Datenträger des aktuellen Rechners unter Windows 7.

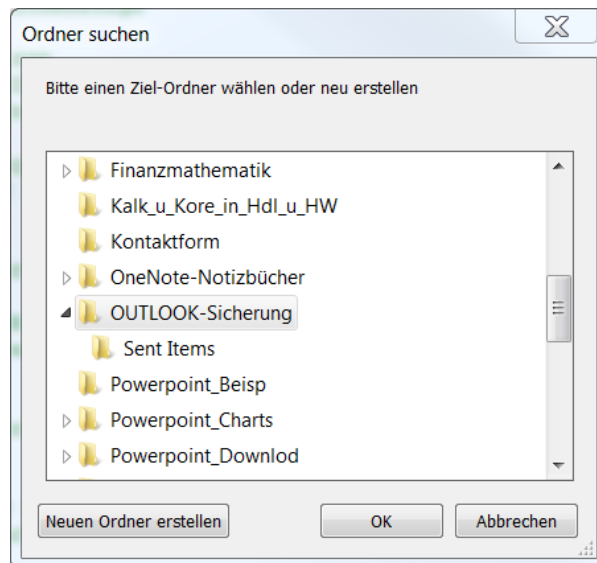


Abb. 2: Auswahl eines Zielordners

In Abb. 2 ist *OUTLOOK-Sicherung* als Ziel-Ordner markiert. Nach Betätigung der Schaltfläche *OK* ist dieser Ordner als Zielverzeichnis ausgewählt.

Der vollständige Pfad zum Ziel-Verzeichnis der *OUTLOOK-Sicherung* lautet in diesem Fall:

C:\Users\...\Documents\OUTLOOK-Sicherung\Sent Items³

3 Ergebnis einer OUTLOOK-Sicherung

Im Zielverzeichnis (s. oben) befinden sich u. a. folgende OUTLOOK-Elemente (hier nur ein typischer Ausschnitt):

Name	Änderungsdatum	Typ	Größe
✉ 20181215-1933_WG Foto von Annette Thormaehlen	16.02.2019 16:31	Outlook-Element	147 KB
✉ 20181217-0948_AW Bitte Korrekturlesen	16.02.2019 16:31	Outlook-Element	824 KB
✉ 20181224-1642_AW Weihnachts- und Neujahrsgr	16.02.2019 16:31	Outlook-Element	70 KB
✉ 20190106-1443_AW Walkingausflug Montag, 07.01.2019	16.02.2019 16:31	Outlook-Element	18 KB
✉ 20190106-1658_AW Familie und Verwandtschaft	16.02.2019 16:31	Outlook-Element	25 KB
✉ 20190107-1542_AW Grabpflege MoorhusenNeuendorf	16.02.2019 16:31	Outlook-Element	22 KB
✉ 20190109-1212_WG Besuch in Ratingen	16.02.2019 16:31	Outlook-Element	18 KB

Abb. 3: Ausschnitt aus dem Zielverzeichnis

Ein Doppelklick auf ein bestimmtes OUTLOOK-Element (s. Abb. 3) bewirkt dessen sofortige Anzeige in OUTLOOK mit entsprechendem Nachrichtenformat. Der Dateityp dieser OUTLOOK-Elemente lautet (*.msg)⁴, also *nicht* (*.pst)⁵.

4 Benötigte VBA-Prozeduren

4.1 Hauptprogramm

Der Quellcode des Hauptprogramms wird in Listing 1 wiedergegeben:

```
Option Explicit
Dim strSavePath As String ' Verzeichnispfad

Sub SaveAllEmails_ProcessAllSubFolders()
' VBA-Makro zum Sichern von ausgewählten OUTLOOK-Nachrichten
' in einem vom Benutzer bestimmten Zielverzeichnis.
'-----
Dim lngFolder As Long ' Schleifenzähler f. Verzeichnisse
Dim lngSubFolder As Long ' Schleifenzähler f. Unterverzeichnisse
Dim lngStartPos As Long ' Startposition in einer Zeichenfolge

Dim colFolders As New Collection ' Auflistung der Verzeichnisse
Dim colStoreID As New Collection ' Auflistung der Speicher-IDs
Dim colEntryID As New Collection ' Auflistung der Eintrag-IDs

Dim objApp As OUTLOOK.Application ' Microsoft Office-Anwendung
Dim objNS As OUTLOOK.NameSpace ' Abstraktes Stammobjekt für Datenquellen
Dim objFolder As OUTLOOK.MAPIFolder ' Verzeichnis
Dim objMsg As OUTLOOK.MailItem ' Nachricht (E-Mail)
Dim objPicked As Object ' Vom Benutzer gewähltes Verzeichnis
```

³ Dt.: *Gesendete Objekte*

⁴ Der Dateityp (*.msg) kennzeichnet das sog. *Outlook Message Format*.

⁵ Der Dateityp einer OUTLOOK-*Datendatei* lautet (*.pst): **personal storage table**. Dieser Dateityp kann in OUTLOOK mit den standardmäßig eingebauten Werkzeugen exportiert und importiert werden.

```

Dim FSO                As Object                ' Akronym f. "File System Object"
                                                ' (f. den Zugriff auf das Windows-Dateisystem)

Dim strFullFileNm     As String                ' vollständiger Dateiname
Dim strFolder         As String                ' Verzeichnis
Dim strFullPath       As String                ' vollständiger Pfad zum Verzeichnis
Dim strTargetFolder   As String                ' Zielverzeichnis
Dim strSubject        As String                ' Betreff einer E-Mail-Nachricht
Dim strSubjectClean   As String                ' bereinigter Betreff einer E-Mail-Nachricht
Dim strReceived       As String                ' Zugangszeitpunkt einer E-Mail-Nachricht

' FSO-Objekt erstellen.
Set FSO = CreateObject("Scripting.FileSystemObject")
' OUTLOOK starten.
Set objApp = OUTLOOK.Application
' Datenquellen bestimmen.
Set objNS = objApp.GetNamespace("MAPI")
Set objPicked = objNS.PickFolder
If objPicked Is Nothing Then
    MsgBox "Nichts ausgewählt!", vbCritical, "Fehler"
    GoTo ExitSub:
End If

' Ziel-Ordner bestimmen.
Call BrowseForFolder(strSavePath)
Call GetFolder(colFolders, colEntryID, colStoreID, objPicked)

' Alle Verzeichnisse durchlaufen
For lngFolder = 1 To colFolders.Count
    ' Ungültige Zeichen aus dem Verzeichnispfad entfernen.
    strFolder = RemoveIllegalCharacters(colFolders(lngFolder))
    lngStartPos = InStr(3, strFolder, "\") + 1
    strFolder = Mid(strFolder, lngStartPos, 256)
    strFullPath = strSavePath & "\" & strFolder & "\"
    strTargetFolder = Left(strFullPath, Len(strFullPath) - 1) & "\"
    If Not FSO.FolderExists(strFullPath) Then
        FSO.CreateFolder (strFullPath)
    End If
    Set objFolder = objApp.Session.GetFolderFromID(colEntryID(lngFolder), colStoreID(lngFolder))
    On Error Resume Next
    For lngSubFolder = 1 To objFolder.Items.Count
        Set objMsg = objFolder.Items(lngSubFolder)
        With objMsg
            ' Empfangszeitpunkt einer OUTLOOK-Meldung formatieren
            strReceived = Format(.ReceivedTime, "YYYYMMDD-hhmm")
            ' Sonderzeichen aus der Betreff-Zeile einer OUTLOOK-Meldung entfernen
            strSubject = RemoveSpecialCharacters(.Subject)
        End With
        ' (.msg) kennzeichnet den Dateityp einer OUTLOOK E-Mail-Nachricht
        strFullFileNm = Left(strTargetFolder & strReceived & "_" & strSubject & ".msg", 256)
        objMsg.SaveAs strFullFileNm, 3
    Next lngSubFolder
    On Error GoTo 0
Next lngFolder
If lngFolder >= colFolders.Count Then MsgBox "Normales Ende der OUTLOOK-Sicherung.", vbInformation,
"Erfolg!"
ExitSub:
Set objApp = Nothing
Set objNS = Nothing
Set objFolder = Nothing
Set objMsg = Nothing
Set objPicked = Nothing
End Sub

```

Listing 1: Hauptprogramm

4.2 Aufrufe im Hauptprogramm

Im Hauptprogramm (s. Listing 1) werden 3 benutzerdefinierte Funktionen und 1 benutzerdefiniertes Unterprogramm eingesetzt (s. Tab 1).

Name	Typ	Aufgabe
GetFolder	Unterprogramm	OUTLOOK-Verzeichnisse holen
BrowseForFolder	Funktion	Ziel-Verzeichnis bestimmen.
RemoveIllegalCharacters	Funktion	Ungültige Zeichen aus einem Pfadnamen entfernen
RemoveSpecialCharacters	Funktion	Bestimmte Sonderzeichen aus dem Betreff einer Nachricht entfernen

Tab 1: Aufrufe im Hauptprogramm

4.3 Listing des benutzerdefinierten Unterprogramms

Das in Listing 2 gezeigte Unterprogramm dient zur Ermittlung der relevanten OUTLOOK-Ordner.

```
Sub GetFolder(colFolders As Collection, colEntryID As Collection, colStoreID As Collection, fldr As MAPIFolder)
    ' OUTLOOK-Verzeichnisse holen.
    Dim objFolder As MAPIFolder
    With fldr
        ' Auflistungen belegen.
        colFolders.Add .FolderPath ' Gibt den Pfad des aktuellen Ordners an.
        colEntryID.Add .EntryID ' Wird gesetzt, nachdem das jeweilige Element gespeichert o. gesendet wurde.
        colStoreID.Add .StoreID ' StoreID ist eine Eigenschaft des aktuellen Ordners.
        For Each objFolder In .Folders
            GetFolder colFolders, colEntryID, colStoreID, objFolder
        Next objFolder
    End With
    Set objFolder = Nothing
End Sub
```

Listing 2: OUTLOOK-Verzeichnisse holen

4.4 Listings der benutzerdefinierten Funktionen

```
Function BrowseForFolder(strSavePath As String, Optional OpenAt As String) As String
    ' Pfad des Ziel-Ordners bestimmen.
    Dim objShell As Object
    Dim objFolder As Object
    Dim strEnviro As String
    On Error Resume Next
    ' Windows-Umgebungsvariablen abrufen
    strEnviro = CStr(Environ("USERPROFILE"))
    Set objShell = CreateObject("Shell.Application")
    Set objFolder = objShell.BrowseForFolder(0, "Bitte einen Ziel-Ordner wählen oder neu erstellen", 0,
(strEnviro & "\Documents\"))
    strSavePath = objFolder.Path
    Set objFolder = Nothing
    Set objShell = Nothing
    On Error GoTo 0
End Function
```

Listing 3: Pfad des Ziel-Ordners bestimmen

```
Function RemoveIllegalCharacters(strName As String) As String
    ' Ungültige Zeichen aus dem Verzeichnispfad entfernen.
    Dim objRegex As Object ' Def. f. regulären Ausdruck
    Set objRegex = CreateObject("VBScript.RegExp")
    With objRegex
        ' Suchmuster f. ungültige Zeichen definieren.
        .Pattern = "[\\" & Chr(10) & Chr(13) & Chr(34) & "!@#\$%&^*\(\)\=\+\|[\]\{\}\|\`'\';\:\<>?\/\,]"
        .Global = True
        .IgnoreCase = True
    End With
    ' Ungültige Zeichen durch Leerzeichen ersetzen.
    RemoveIllegalCharacters = objRegex.Replace(strName, " ")
    Set objRegex = Nothing
End Function
```

Listing 4: Ungültige Zeichen aus einem Verzeichnispfad entfernen

```
Function RemoveSpecialCharacters(strSubject as String) as String
    ' Bestimmte Sonderzeichen aus der Betreff-Zeile einer OUTLOOK-E-Mail-Nachricht entfernen,
    ' aber NICHT Unterstrich, Leerzeichen, Bindestrich und/oder Punkt.
    Dim RegExp As Object
    Set RegExp = CreateObject("vbscript.regexp")
    With RegExp
        .Pattern = "[^\w -.]'"
        .IgnoreCase = True
        .Global = True
        RemoveSpecialCharacters = .Replace(strSubject, "")
    End With
    Set RegExp = Nothing
End Function
```

Listing 5: Bestimmte Sonderzeichen aus dem Betreff einer OUTLOOK-E-Mail-Nachricht entfernen

5 Praktische Relevanz

Zur ordnungsgemäßen Aufbewahrung und Archivierung betrieblicher E-Mails nach den gesetzlichen Vorschriften in Deutschland siehe [2].

Literaturverzeichnis

- [1] D. Poremsky, „Saving All Messages to the Hard Drive Using VBA,“ 29 08 2018. [Online]. Available: <https://www.slipstick.com/developer/saving-messages-to-the-hard-drive-using-vba/>. [Zugriff am 11 02 2019].
- [2] o. V., „E-Mail-Archivierung,“ 05 04 2017. [Online]. Available: <https://de.wikipedia.org/wiki/E-Mail-Archivierung>. [Zugriff am 2019 02 17].