

Entwicklerdokumentation

FlowFact Events

Inhalt

1	Vorbemerkung	5
2	Callback_NewAppointment	6
2.1	Beispiel	6
2.2	Beschreibung	6
3	Callback_DeleteAppointment	7
3.1	Beispiel	7
3.2	Beschreibung	7
4	Callback_AppointmentMove	8
4.1	Beispiel	8
4.2	Beschreibung	8
5	Callback_BeforePhoneCall	9
5.1	Beispiel	9
5.2	Beschreibung	9
6	Callback_BeforeCharacteristicChanged	10
6.1	Beispiel	10
6.2	Beschreibung	10
7	Callback_AfterCharacteristicChanged	11
7.1	Beispiel	11
7.2	Beschreibung	11
8	DetailPopupBefore	12
8.1	Beispiel	12
8.2	Beschreibung	12
9	DetailPopupClick	13
9.1	Beispiel	13
9.2	Beschreibung	13
10	PaneRSChanged	14
10.1	Beispiel	14
10.2	Beschreibung	14
11	BeforePaneRSChanged	15
11.1	Beispiel	15
11.2	Beschreibung	15
12	DetailChanged	16
12.1	Beispiel	16
12.2	Beschreibung	16
13	AfterDetailChanged	17
13.1	Beispiel	17

13.2	Beschreibung	17
14	AfterExecuteCommand	18
14.1	Beispiel	18
14.2	Beschreibung	18
15	BeforeExecuteCommand	19
15.1	Beispiel	19
15.2	Beschreibung	19
16	Fill	20
16.1	Beispiel	20
16.2	Beschreibung	20
17	Callback_AfterFlowFactStart	21
17.1	Beispiel	21
17.2	Beschreibung	21
18	PaneActivate	22
18.1	Beispiel	22
18.2	Beschreibung	22
19	PaneDeActivate	23
19.1	Beispiel	23
19.2	Beschreibung	23
20	PaneInitialize	24
20.1	Beispiel	24
20.2	Beschreibung	24
21	X_MergeFieldGet	25
21.1	Beispiel	25
21.2	Beschreibung	25
22	X_MergeFieldFillList	26
22.1	Beispiel	26
22.2	Beschreibung	26
23	Callback_BeforTransfer	27
23.1	Beispiel	27
23.2	Beschreibung	27
24	Callback_AfterTransfer	28
24.1	Beispiel	28
24.2	Beschreibung	28
25	EventBeforeStartDoc	29
25.1	Beispiel	29
25.2	Beschreibung	29
26	BeforeDetailDeleted	30
26.1	Beispiel	30
26.2	Beschreibung	30

27	EventAfterDetailsFilled	31
27.1	Beispiel	31
27.2	Beschreibung	31
28	Callback_StartDetailEdit	32
28.1	Beispiel	32
28.2	Beschreibung	32
29	Callback_BeforeFillCell	33
29.1	Beispiel	33
29.2	Beschreibung	33
30	EventGetStandardPermission	34
30.1	Beispiel	34
30.2	Beschreibung	34
31	Callback_BeforeXmlSave	35
31.1	Beispiel	35
31.2	Beschreibung	35
32	Kontakt	36

1 Vorbemerkung

Dieses Dokument beinhaltet eine lose Auflistung der FlowFact Events incl. einiger Beispiele.

Von den hier aufgelisteten Events werden die Fehler geloggt. Das mitloggen der Events ist zu Debugzwecken für die Entwickler gedacht. Sollten Fehler auftreten, können Sie diese unter dem FlowFact Pfad „Dat\ScriptError.LOG“ nachlesen.

Es besteht keine Gewähr auf Vollständigkeit.

2 **Callback_NewAppointment**

2.1 **Beispiel**

```
Sub Callback_NewAppointment(ByVal rsAKT, ByRef blnCancel)
```

```
    msgbox "Termin erstellt – Datum: " & rsAKT.fields("Datum").value
```

```
    blnCancel = false
```

```
End Sub
```

2.2 **Beschreibung**

Dieses Event wird ausgelöst sobald ein Termin über den Kalender angelegt wird. Mit Hilfe des 2. Parameters kann der Vorgang abgebrochen werden.

Ab Build 356 verfügbar

3 **Callback_DeleteAppointment**

3.1 **Beispiel**

Sub Callback_DeleteAppointment(ByVal strAktDSN, ByRef blnCancel)

 MsgBox "Termin gelöscht – Aktivitäten DSN: " & strAktDSN

 blnCancel = false

End Sub

3.2 **Beschreibung**

Dieses Event wird ausgelöst sobald ein Termin über den Kalender(Rechtsklick→ Löschen) gelöscht wird. Mit Hilfe des 2. Parameters kann der Vorgang abgebrochen werden.

Ab Build 356 verfügbar

4 Callback_AppointmentMove

4.1 Beispiel

```
Sub Callback_AppointmentMove(ByVal strAktDSN, ByRef strNewDateStart,  
ByRef strNewDateEnd, ByRef blnConfirmChanges, ByRef blnCancel)
```

```
    MsgBox "Termin bewegt " & strNewDateStart
```

```
    blnConfirmChanges = false
```

```
    blnCancel = false
```

```
End Sub
```

4.2 Beschreibung

Dieses Event wird ausgelöst sobald ein Termin über den Kalender verschoben oder die Startzeit/Endzeit verändert wurde, aber noch nicht in der Datenbank gespeichert worden ist. Der 2. Parameter entspricht der „neuen Start Uhrzeit“ – der 3. Parameter entsprechend der „neuen Endzeit“. Parameter 4 gibt an ob der „Verschieben bestätigen“ Dialog angezeigt werden soll. Mithilfe des 5. Parameters kann der Verschiebungsvorgang abgebrochen werden.

Ab Build 356 verfügbar

5 **Callback_BeforePhoneCall**

5.1 **Beispiel**

Sub Callback_BeforePhoneCall(ByVal strAdDSN, ByVal lngRow, ByRef blnCancel)

 MsgBox "telefonieren " & strAdDSN & "|" & lngRow & "|"

 blnCancel = false

End Sub

5.2 **Beschreibung**

Dieses Event wird ausgelöst sobald der FlowFact Benutzer ein Telefonat mit einer Adresse führen will. Der 2. Parameter definiert den Index des Eintrages in der Telefon Spalte. Mithilfe des 3. Parameters kann der Vorgang abgebrochen werden.

Ab Build 356 verfügbar

6 Callback_BeforeCharacteristicChanged

6.1 Beispiel

```
Sub Callback_BeforeCharacteristicChanged(ByVal strPane, ByVal strMKMDSN, ByVal  
blnMKMAdded, ByRef blnCancel)
```

```
    If (blnMKMAdded) then
```

```
        MsgBox "Bevor Merkmal hinzugefügt wurde – DSN: " & strMKMDSN
```

```
    Else
```

```
        MsgBox "Bevor Merkmal entfernt wurde – DSN: " & strMKMDSN
```

```
    End If
```

```
    blnCancel = false
```

```
End Sub
```

6.2 Beschreibung

Dieses Event wird ausgelöst sobald im Merkmalsdialog ein Merkmal hinzugefügt oder entfernt wurde. Wenn der 3. Parameter „True“ ist wurde ein Merkmal hinzugefügt – wenn er „False“ ist wurde ein Merkmal entfernt. Über den 4. Parameter kann der Vorgang abgebrochen werden.

Ab Build 356 verfügbar

7 Callback_AfterCharacteristicChanged

7.1 Beispiel

```
Sub Callback_AfterCharacteristicChanged(ByVal strPane, ByVal strMKMDSN, ByVal  
bInMKMAdded)
```

```
    If (bInMKMAdded) Then
```

```
        MsgBox "Nachdem ein Merkmal hinzugefügt wurde – DSN: " & strMKMDSN
```

```
    Else
```

```
        MsgBox "Nachdem ein Merkmal entfernt wurde: - DSN: " & strMKMDSN
```

```
    End If
```

```
End Sub
```

7.2 Beschreibung

Dieses Event wird ausgelöst sobald im Merkmalsdialog ein Merkmal hinzugefügt oder entfernt wurde und diese Änderung bereits im Merkmals Fenster angezeigt wurde. Wenn der 3. Parameter „True“ ist wurde ein Merkmal hinzugefügt – wenn er „False“ ist wurde ein Merkmal entfernt.

Ab Build 356 verfügbar

8 DetailPopupBefore

8.1 Beispiel

Sub DetailPopupBefore(ByVal strName, ByVal strZeile, ByVal strFeldart, Byref strBeschriftung)

 If strZeile Mod 2 = 1 Then

 strBeschriftung="Notarttermin vereinbaren"

 End If

End Sub

8.2 Beschreibung

Dieses Ereignis wird ausgelöst wenn die rechte Maustaste in der Detailliste gedrückt wird und bevor das PopUp erscheint. Dadurch lässt sich ein individueller Menüeintrag erzeugen Momentan allerdings nicht für alle Eventklassen.

Ab Build 283 verfügbar.

9 DetailPopupClick

9.1 Beispiel

Sub DetailPopupClick(ByVal strBefehl)

MsgBox "Befehl " & strBefehl & " wurde via Popup ausgewählt!"

End Sub

9.2 Beschreibung

Wenn im PopUpMenü ein Menüeintrag (den man selbst mit „DetailPopupBefore“ hinzugefügt hat) angeklickt wurde, wird dieses Ereignis ausgelöst.

Ab Build 283 verfügbar.

10 PaneRSChanged

10.1 Beispiel

```
Sub PaneRSChanged(ByVal strTabellenName, ByVal rsDatensatz, ByVal lngStatus)
```

```
    MsgBox "Afterchanged:" & rsDatensatz ("Kennung") & ""
```

```
End Sub
```

```
Public Enum ffPaneRSState          'lngStatus
```

```
    ffRsNew = 1
```

```
    ffRsChanged = 2
```

```
    ffRsDeleted = 3
```

```
End Enum
```

10.2 Beschreibung

Dieses Ereignis wird ausgelöst nachdem ein Datensatz geändert wurde und dies bereits in der Datenbank gesichert wurde.

Ab Build < 283 verfügbar.

Achtung: Seit Build 356 feuern auch Änderungen im Kalender dieses Event.

11 BeforePaneRSChanged

11.1 Beispiel

```
Sub BeforePaneRSChanged(ByVal strTabellenName, ByVal rs, ByVal lngStatus)
```

```
    MsgBox "Beforechanged:" & rs("Kennung") & ""
```

```
End Sub
```

```
Public Enum ffPaneRSState          'lngStatus
```

```
    ffRsNew = 1
```

```
    ffRsChanged = 2
```

```
    ffRsDeleted = 3
```

```
End Enum
```

11.2 Beschreibung

Dieses Ereignis wird ausgelöst bevor ein Datensatz in der Datenbank geändert wurde.

Ab Build < 283 verfügbar.

Achtung: Seit Build 356 feuern auch Änderungen im Kalender dieses Event.

12 DetailChanged

12.1 Beispiel

```
Sub DetailChanged(ByVal strDetTabellenName, ByRef rsDetails)
```

```
    If IsNumeric(rsDetails("Eingabe").Value) Then
```

```
        'Zahlen Feldarten
```

```
        MsgBox "Es ist eine Nummer"
```

```
        rsDetails("von_num") .Value = 1000
```

```
        rsDetails("bis_num") .Value = 1000
```

```
    End If
```

```
    'Ja/Nein Feldarten
```

```
    'Ja = 1/ Nein = 0 nur im von_num und bis_num eintragen
```

```
    MsgBox "Es ist ein Ja/Nein Feld"
```

```
    If rsDetails("Eingabe").Value = "Ja" Then
```

```
        rsDetails("von_num") .Value = 1
```

```
        rsDetails ("bis_num") .Value = 1
```

```
    Else
```

```
        rsDetails("von_num") .Value = 0
```

```
        rsDetails ("bis_num") .Value = 0
```

```
    End If
```

```
End Sub
```

12.2 Beschreibung

Nachdem der Benutzer einen Detaildatensatz geändert und die Enter Taste gedrückt hat, wird dieses Event noch vor dem Datenbankupdate gefeuert. In „rsDetails“ befinden sich schon die neuen Detaildatensätze. Das Recordset kann ByRef manipuliert werden. „Von_num“, „Bis_num“ etc. müssen dann jedoch manuell befüllt werden. Im Anschluss erfolgt ein Datenbankupdate auf Dal Schicht. Es erfolgen keine Plausibilitätsprüfungen mehr.

Ab Build 283 verfügbar.

13 AfterDetailChanged

13.1 Beispiel

```
Sub AfterDetailChanged(ByVal strTabellenName, ByVal rsDatensatz)
```

```
    MsgBox "Änderungen an Detaildatensatz in der " & strTabellenName & " Tabelle  
    wurden vorgenommen"
```

```
End Sub
```

13.2 Beschreibung

Hier ist der Datensatz jetzt schon gespeichert. Das bietet sich an wenn man selbst noch andere Felder upgedatet hat und hier jetzt z.B. die Anzeige refreshen möchte.

Ab Build < 283 verfügbar.

14 AfterExecuteCommand

14.1 Beispiel

```
Sub EventAfterExecuteCommand(ByVal strPane, ByVal strCommand)
```

```
    MsgBox "Befehl erfolgreich ausgeführt: " & strCommand
```

```
End Sub
```

14.2 Beschreibung

Dieses Event wird ausgelöst nachdem ein Command ausgeführt wird.

Ab Build < 283 verfügbar.

15 BeforeExecuteCommand

15.1 Beispiel

```
Sub BeforeExecuteCommand(ByVal strPane, ByVal strCommand, Byref blnCancel)
```

```
    If strcommand="DUPLICATE" Then
```

```
        MsgBox "Du darfst nicht duplizieren!"
```

```
        blnCancel=True
```

```
    End If
```

```
End Sub
```

15.2 Beschreibung

Dieses Event wird ausgelöst bevor ein Befehl ausgeführt wird. Mithilfe des Parameters blnCancel lässt sich die Ausführung des Befehls abbrechen.

Ab Build < 283 verfügbar.

16 **Fill**

16.1 **Beispiel**

Sub Fill(ByVal strTabelle, ByVal ePut, ByRef rsDatensatz)

MsgBox "Fill: " & strTabelle & "/" & cstr(ePut) & "/" & rsDatensatz (0)

End Sub

'ePut:

Global Const CLR_AND_GET_BUFFER = 3

Global Const PUT_BUFFER = 1

Global Const GET_BUFFER = 0

16.2 **Beschreibung**

Dieses Event wird aufgerufen wenn sich der Hauptdatensatz ändert z.B. wenn eine andere Adresse, Objekt, Anfrage ect. ausgewählt wird.

Ab Build < 283 verfügbar.

17 **Callback_AfterFlowFactStart**

17.1 **Beispiel**

```
Sub Callback_AfterFlowFactStart ()
```

```
    MsgBox "FlowFact wurde erfolgreich gestartet "
```

```
End Sub
```

17.2 **Beschreibung**

Dieses Event wird gefeuert nachdem FlowFact vollständig geladen ist.

Ab Build 360 verfügbar.

18 PaneActivate

18.1 Beispiel

```
Sub PaneActivate(ByVal strPane)
```

```
    If strPane="F_OBJ" Then
```

```
        ' Einen Karteireiter verstecken (emakeln 1)
```

```
        Application.Panes("F_OBJ").Controls("ssTab").Tabs(6).Visible = False
```

```
        ' Schaltet den Befehl DEMOOBJ frei
```

```
        commands("DEMOOBJ").enabled = True
```

```
    End If
```

```
End Sub
```

18.2 Beschreibung

Änderungen an der Oberfläche sollten in der Regel besser im Activate Ereignis erfolgen da im Initialize-Ereignis die Oberfläche u.U. noch nicht vollständig initialisiert ist.

Ab Build 283 verfügbar.

19 PaneDeActivate

19.1 Beispiel

```
Sub PaneDeActivate(ByVal strPane)
```

```
    If strPane="F_OBJ" Then
```

```
        commands("DEMOOBJ").enabled = false
```

```
    End If
```

```
End Sub
```

19.2 Beschreibung

Dieses Event wird ausgelöst sobald ein Fenster deaktiviert wurde.

Ab Build < 283 verfügbar.

20 PanelInitialize

20.1 Beispiel

```
Sub PanelInitialize(ByVal strPane)
```

```
    If strPane="F_OBJ" Then
```

```
        ' Init-Code
```

```
    End If
```

```
End Sub
```

20.2 Beschreibung

Dieses Event wird ausgelöst sobald ein Fenster neu geladen wird. Jedoch ist dieses Fenster zu dem Zeitpunkt noch nicht vollständig aufgebaut.

Ab Build 283 verfügbar.

21 X_MergeFieldGet

21.1 Beispiel

```
Function X_MergeFieldGet(ByVal strMergefield, ByVal rsDatensatz, Byref blnSuccess)
```

```
    If instr(strMergefield, "X_TEST") > 0 Then
```

```
        x_mergefieldget = "X_Platzhaltername"
```

```
    End If
```

```
End Function
```

21.2 Beschreibung

Ein X_Platzhalter wird in Vorlagen benutzt und (kann) eine Zusammenstellung von verschiedenen Datenbankfeldern aus FlowFact sein. Dieser Platzhalter wird im Script definiert und muss zwingend mit „X_“ im Namen anfangen. In Parameter „strMergefield“ ist der Platzhaltername selbst enthalten. Dieses Event wird ausgelöst sobald ein X-Platzhalter benutzt wird. Im „rsDatensatz“ ist der Datensatz der ausgewählt wurde, um die Vorlage zu starten. (Adress-, Objekt-, Anfragedatensatz usw.)

Ab Build < 283 verfügbar.

22 X_MergeFieldFillList

22.1 Beispiel

```
Sub X_MergeFieldFillList(ByRef list)
```

```
    list.additem "x_test"
```

```
End Sub
```

22.2 Beschreibung

Dieses Event wird ausgelöst sobald der Platzhalter Dialog aufgerufen wird. Hier sollten alle selbst definierten X-Platzhalter hinzugefügt werden. „list“ ist in diesem Fall eine „ListBox“ dem wir einzelne X-Platzhalter zuweisen können.

Ab Build < 283 verfügbar.

23 Callback_BeforTransfer

23.1 Beispiel

```
Function Callback_BeforeTransfer(ByVal rsboersen, ByVal blnIsFtp, ByVal strPath,  
ByVal strFileName, ByVal strFileFilter)
```

```
    MsgBox "Die Übertragung wird gestartet "
```

```
End Function
```

23.2 Beschreibung

Dieses Event wird ausgelöst bevor eine Übertragung zu den Börsen gestartet wird. Es kann nur in einem Script ausgelöst werden, das im Ordner „FlowFact\system\Services\OpenImmo“ liegt. Soll dieses Event jedoch ausgelöst werden wenn eine Webserverübertragung stattfindet, muss das Script im Ordner „FlowFact\system\Services\Webserver“ vorhanden sein. Somit wird dieses Event ausgelöst, wenn eine Webserverübertragung gestartet wird (FlowFact Webserver nicht einbezogen). Obwohl dieses Event eine Funktion ist, spielt der Rückgabewert für FlowFact für die Weiterverarbeitung keine Rolle.

24 Callback_AfterTransfer

24.1 Beispiel

```
Function Callback_AfterTransfer(ByVal rsboersen, ByVal blnIsFtp, ByVal blnTransferOk)
```

```
    MsgBox = "Die Übertragung ist abgeschlossen"
```

```
End Function
```

24.2 Beschreibung

Dieses Event wird ausgelöst nachdem die Übertragung zu den Börsen beendet wurde. Der Parameter „blnTransferOK“ ist „True“ wenn die Übertragung erfolgreich war. Dieses Event kann nur in einem Script ausgelöst werden, dass im Ordner „FlowFact\system\Services\OpenImmo“ liegt. In „rsBoersen“ befinden sich die Börsen zu denen gerade übertragen wurde. Soll dieses Event jedoch ausgelöst werden wenn eine Webserverübertragung stattgefunden hat, muss das Script im Ordner „FlowFact\system\Services\Webserver“ vorhanden sein. Somit wird dieses Event ausgelöst, wenn eine Webserverübertragung beendet wurde (FlowFact Webserver nicht einbezogen). Obwohl dieses Event eine Funktion ist, spielt der Rückgabewert für FlowFact für die Weiterverarbeitung keine Rolle.

25 EventBeforeStartDoc

25.1 Beispiel

```
Function EventBeforeStartDoc(ByVal strFilename, ByVal strExt)
```

```
    MsgBox "Die Datei wird gleich geöffnet"
```

```
    EventBeforeStartDoc = false
```

```
Function End
```

25.2 Beschreibung

Dieses Event wird ausgelöst bevor eine Datei aus dem Anhang einer Aktivität geöffnet wird. Der Parameter „strExt“ beinhaltet den Dateitypen der geöffnet werden soll. Wenn der Rückgabewert „True“ ist wird die Datei von FlowFact verarbeitet. Ist der Rückgabewert „False“ z.B. bei Dateitypen wie „JPG“ wird die Datei geöffnet und somit für den Benutzer sichtbar.

26 BeforeDetailDeleted

26.1 Beispiel

```
Sub BeforeDetailDeleted(ByVal strDetailsTable, ByVal strDetailsDsn)
```

```
    MsgBox "Der Detaildatensatz wird gleich gelöscht"
```

```
End Sub
```

26.2 Beschreibung

Dieses Event wird ausgelöst bevor ein Detaildatensatz gelöscht wird. In „strDetailsDSN“ ist die DSN des Detaildatensatzes enthalten. Diese Detail Datensatznummer kann von verschiedenen Tabellen sein (ADDET, OBJDET, ANFDET).

27 EventAfterDetailsFilled

27.1 Beispiel

```
Sub EventAfterDetailsFilled(ByVal strTablename)
```

```
    MsgBox "Der Datensatz wurde neu geladen"
```

```
End Sub
```

27.2 Beschreibung

Dieses Event wird ausgelöst nachdem ein Datensatz ausgewählt und dazu der Detaildatensatz geladen wurde. (Im Objekte-, Adress-, Anfrage-, Aktivitätendetails)
Dieses Event verwendet man um zu überprüfen, ob bestimmte Datensätze aufgerufen werden.

28 Callback_StartDetailEdit

28.1 Beispiel

```
Function Callback_StartDetailEdit(ByVal strFldartDsn, ByVal strPaneName, ByRef blnCancel)
```

```
    If strPaneName = "F_OBJ" Then
```

```
        MsgBox "Die Verlinkung zu Adressen und Projekte werden gleich ausgeschaltet."  
        blnCancel = True
```

```
    End If
```

```
End Function
```

28.2 Beschreibung

Dieses Event wird ausgelöst nachdem ein Detaildatensatz vom Benutzer geändert, jedoch noch nicht in die Datenbank gespeichert wurde. Mit diesem Event kann man die Verlinkung der Feldarten vom Typ Adresse und Projekte bei der Eingabe ausschalten. Wenn der Parameter „blnCancel = True“ ist, ist die Verlinkung ausgeschaltet und somit in den neu eingegebenen Feldern nur als Text enthalten. Obwohl dieses Event eine Funktion ist, spielt der Rückgabewert für FlowFact für die Weiterverarbeitung keine Rolle. Es ist ratsam dieses Event mit Vorsicht zu genießen.

29 Callback_BeforeFillCell

29.1 Beispiel

```
Sub Callback_BeforeFillCell(ByVal strFldartDsn, ByVal strPaneName, ByRef blnCancel)
```

```
    MsgBox "Ein Optionsfeld wird gleich geändert"
```

```
    blnCancel = True
```

```
    MsgBox "Das Optionsfeld wird nicht aufgeklappt und somit auch nicht geändert"
```

```
End Sub
```

29.2 Beschreibung

Dieses Event wird ausgelöst nachdem ein Optionsfeld (Detaildatensatz) zum ändern ausgewählt wurde. Mit diesem Event kann man bestimmte Optionsfelder in einem Fenster abfragen. Mit `blnCancel = True` wird verhindert, dass dieses Optionsfeld geöffnet und somit geändert werden kann.

30 EventGetStandardPermission

30.1 Beispiel

```
Sub EventGetStandardPermission(ByVal strTableName, ByRef lngIndex)
```

```
    MsgBox "Hier kann man eine Rechteschablone auswählen"  
    lngIndex = 2          'lngIndex ist die neu ausgewählte Rechteschablone
```

```
End Sub
```

30.2 Beschreibung

Dieses Event wird ausgelöst nachdem eine Aktivität, Anfrage, Adresse, Objekt usw. erstellt wird. Im Anschluss kann man eine andere Rechteschablone nehmen. Standardmäßig ist die Schablone 4 ausgewählt. Die Schablonen sind wie folgt aufgeteilt.

5	3
1	2
6	4

31 Callback_BeforeXmlSave

31.1 Beispiel

```
Function Callback_BeforeXmlSave(ByVal rsBoerse, ByRef oXmlDoc, ByRef strPath,  
ByRef strFilename)
```

```
    strFilename = "C:\Meine Uebertragungen"
```

```
End Function
```

31.2 Beschreibung

Dieses Event wird ausgelöst bevor die XML Datei, die zur Übertragung zu den Portalen benötigt wird, gespeichert wird. Es tritt noch vor dem „Callback_BeforeTransfer“ Event auf. Dieses Event kann nur in einem Script ausgelöst werden, dass im Ordner „FlowFact\system\Services\OpenImmo“ liegt.

Hinweis: „oXmlDoc“ ist vom Typ „MSXML2.DOMDocument40“. Obwohl dieses Event eine Funktion ist, spielt der Rückgabewert für FlowFact für die Weiterverarbeitung keine Rolle.

32 **Kontakt**

Maik Godinho (Research & Development)

FlowFact AG
Industriestraße 161
D - 50999 Köln

Telefon +49 (0)2236 - 880 - 0
Telefax +49 (0)2236 - 880 - 111

info@flowfact.de
www.flowfact.de