

Entwicklerdokumentation

# FLOWFACT

# Events



**FLOWFACT GmbH**

Holweider Straße 2a · D-51065 Köln

T +49 221 995 90-0 · F +49 221 995 90-111

[www.flowfact.de](http://www.flowfact.de) · [info@flowfact.de](mailto:info@flowfact.de)

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



<b>13</b>	<b>CanDelete</b>	<b>18</b>
<b>14</b>	<b>DetailChanged</b>	<b>19</b>
14.1	Beispiel	19
14.2	Beschreibung	20
<b>15</b>	<b>AfterDetailChanged</b>	<b>21</b>
15.1	Beispiel	21
15.2	Beschreibung	21
<b>16</b>	<b>AfterExecuteCommand</b>	<b>22</b>
16.1	Beispiel	22
16.2	Beschreibung	22
<b>17</b>	<b>BeforeExecuteCommand</b>	<b>23</b>
17.1	Beispiel	23
17.2	Beschreibung	23
<b>18</b>	<b>Fill</b>	<b>24</b>
18.1	Beispiel	24
18.2	Beschreibung	24
<b>19</b>	<b>Callback_AfterFlowFactStart</b>	<b>25</b>
19.1	Beispiel	25
19.2	Beschreibung	25
<b>20</b>	<b>PaneActivate</b>	<b>26</b>
20.1	Beispiel	26
20.2	Beschreibung	26
<b>21</b>	<b>PaneDeActivate</b>	<b>27</b>
21.1	Beispiel	27
21.2	Beschreibung	27
<b>22</b>	<b>PaneInitialize</b>	<b>28</b>
22.1	Beispiel	28
22.2	Beschreibung	28
<b>23</b>	<b>X_MergeFieldGet</b>	<b>29</b>
23.1	Beispiel	29
23.2	Beschreibung	29
<b>24</b>	<b>X_MergeFieldFillList</b>	<b>30</b>
24.1	Beispiel	30
24.2	Beschreibung	30
<b>25</b>	<b>EventBeforeStartDoc</b>	<b>31</b>
25.1	Beispiel	31
25.2	Beschreibung	31

<b>26</b>	<b>BeforeDetailDeleted</b>	<b>32</b>
26.1	Beispiel	32
26.2	Beschreibung	32
<b>27</b>	<b>EventAfterDetailsFilled</b>	<b>33</b>
27.1	Beispiel	33
27.2	Beschreibung	33
<b>28</b>	<b>Callback_StartDetailEdit</b>	<b>34</b>
28.1	Beispiel	34
28.2	Beschreibung	34
<b>29</b>	<b>Callback_BeforeFillCell</b>	<b>35</b>
29.1	Beispiel	35
29.2	Beschreibung	35
<b>30</b>	<b>EventGetStandardPermission</b>	<b>36</b>
30.1	Beispiel	36
30.2	Beschreibung	36
<b>31</b>	<b>Callback_BeforeXmlSave</b>	<b>37</b>
31.1	Beispiel	37
31.2	Beschreibung	37
<b>32</b>	<b>Mail</b>	<b>38</b>
32.1	MessageReceived	38
32.2	MessageProcessed	38
32.3	MessageSend	38
32.4	MessageSendError	40
<b>33</b>	<b>Webserver – Events</b>	<b>41</b>
33.1	Callback_BeforeTransfer	41
33.2	Callback_AfterTransfer	41
<b>34</b>	<b>OpenImmo – Events</b>	<b>43</b>
34.1	Callback_OpenImmoPicInit	43
34.2	Callback_OpenImmoPicGet	43
34.3	Callback_OpenImmoPicCopy	44
34.4	Callback_BeforeXmlSave	44
34.5	Callback_BeforeTransfer	44
34.6	Callback_AfterTransfer	45
34.7	Callback_GetGeolocation	45
<b>35</b>	<b>IS24Rest Events</b>	<b>47</b>
35.1	Callback_BeforeTransfer	47
35.2	Callback_AfterTransfer	47



# 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

(siehe auch „Hinweis FlowFact 2014“)

### 2.1 Beispiel

```
Sub Callback_NewAppointment( 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\_AfterNewAppointment

### 3.1 Beispiel

```
Sub Callback_AfterNewAppointment(ByVal rsAKTI)
```

```
    MsgBox "Kennung erstellter Termin : " &  
rsAKT.fields("Kennung").value
```

```
End Sub
```

### 3.2 Beschreibung

Dieser Event wird ausgelöst, sobald ein Termin über den Kalender angelegt und gespeichert wurde. Die übergebenen Werte sind nicht editierbar. Der Vorgang ist nicht mehr abzubrechen.

Ab Build 660 verfügbar



## 4 Callback\_DeleteAppointment

### 4.1 Beispiel

```
Sub Callback_DeleteAppointment(ByVal strAktDSN, ByRef blnCancel)
```

```
    MsgBox "Termin gelöscht – Aktivitäten DSN: " & strAktDSN
```

```
    blnCancel = false
```

```
End Sub
```

### 4.2 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

## 5 Callback\_AppointmentMove

### 5.1 Beispiel

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

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

```
        blnConfirmChanges = false
```

```
    blnCancel = false
```

```
End Sub
```

### 5.2 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

## 6 Callback\_BeforePhoneCall

### 6.1 Beispiel

```
Sub Callback_BeforePhoneCall(ByVal strAdDSN, ByVal lngRow, ByRef  
bInCancel)
```

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

```
    bInCancel = false
```

```
End Sub
```

### 6.2 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

## 7 Callback\_BeforeCharacteristicChanged

### 7.1 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
```

### 7.2 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

## 8 Callback\_AfterCharacteristicChanged

### 8.1 Beispiel

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

```
    If (blnMKMAdded) Then
```

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

```
    Else
```

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

```
    End If
```

```
End Sub
```

### 8.2 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

## 9 DetailPopupBefore

### 9.1 Beispiel

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

    If strZeile Mod 2 = 1 Then

        strBeschriftung="Notartermin vereinbaren"

    End If

End Sub

### 9.2 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.

## 10 DetailPopupClick

### 10.1 Beispiel

Sub DetailPopupClick(ByVal strBefehl)

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

End Sub

### 10.2 Beschreibung

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

Ab Build 283 verfügbar.

# 11 PaneRSChanged

(siehe auch „Hinweis FlowFact 2014“)

## 11.1 Beispiel

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

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

```
End Sub
```

```
Public Enum ffPaneRSState    lngStatus
```

```
    ffRsNew = 1
```

```
    ffRsChanged = 2
```

```
    ffRsDeleted = 3
```

```
End Enum
```

## 11.2 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.



## 12 BeforePaneRSChanged

(siehe auch „Hinweis FlowFact 2014“)

### 12.1 Beispiel

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

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

```
End Sub
```

```
Public Enum ffPaneRSState `lngStatus
```

```
    ffRsNew = 1
```

```
    ffRsChanged = 2
```

```
    ffRsDeleted = 3
```

```
End Enum
```

### 12.2 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.

## 13 CanDelete

(siehe auch „Hinweis FlowFact 2014“)

## 14 DetailChanged

(siehe auch „Hinweis FlowFact 2014“)

### 14.1 Beispiel

```
Sub DetailChanged(ByVal strDetTabellenName, 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
```

## 14.2 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.

## 15 AfterDetailChanged

(siehe auch „Hinweis FlowFact 2014“)

### 15.1 Beispiel

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

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

```
End Sub
```

### 15.2 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.

## 16 AfterExecuteCommand

### 16.1 Beispiel

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

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

```
End Sub
```

### 16.2 Beschreibung

Dieses Event wird ausgelöst nachdem ein Command ausgeführt wird.

Ab Build < 283 verfügbar.

## 17 BeforeExecuteCommand

### 17.1 Beispiel

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

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

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

```
        blnCancel=True
```

```
    End If
```

```
End Sub
```

### 17.2 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.

## 18 Fill

(siehe auch „Hinweis FlowFact 2014“)

### 18.1 Beispiel

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

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

```
End Sub
```

```
`ePut:
```

```
Global Const CLR_AND_GET_BUFFER = 3
```

```
Global Const PUT_BUFFER = 1
```

```
Global Const GET_BUFFER = 0
```

### 18.2 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.



## 19 Callback\_AfterFlowFactStart

### 19.1 Beispiel

```
Sub Callback_AfterFlowFactStart ()
```

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

```
End Sub
```

### 19.2 Beschreibung

Dieses Event wird gefeuert nachdem FlowFact vollständig geladen ist.

Ab Build 360 verfügbar.

## 20 PaneActivate

### 20.1 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
```

### 20.2 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.

## 21 PaneDeActivate

### 21.1 Beispiel

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

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

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

```
    End If
```

```
End Sub
```

### 21.2 Beschreibung

Dieses Event wird ausgelöst sobald ein Fenster deaktiviert wurde.

Ab Build < 283 verfügbar.

## 22 PanelInitialize

### 22.1 Beispiel

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

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

```
        ' Init-Code
```

```
    End If
```

```
End Sub
```

### 22.2 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.

## 23 X\_MergeFieldGet

(siehe auch „Hinweis FlowFact 2014“)

### 23.1 Beispiel

Function X\_MergeFieldGet(ByVal strMergefield, rsDatensatz, Byref  
blnSuccess)

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

```
        x_mergefieldget = "X_Platzhaltername"
```

```
    End If
```

```
End Function
```

### 23.2 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-, Anfragendatensatz usw.)

Ab Build < 283 verfügbar.

## 24 X\_MergeFieldFillList

### 24.1 Beispiel

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

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

```
End Sub
```

### 24.2 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.

## 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 Datensatz 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  
bInCancel)
```

```
    MsgBox "Ein Optionsfeld wird gleich geändert"  
    bInCancel = 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 `bInCancel = True` wird verhindert, dass dieses Optionsfeld geöffnet und somit geändert werden kann.

## 30 EventGetStandardPermission

### 30.1 Beispiel

```
Sub EventGetStandardPermission(ByVal strTableName, ByVal 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, ByVal oXmlDoc, ByVal  
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 Mail

### 32.1 MessageReceived

Dieses Event wird aufgerufen, bevor der Maildienst eine eingehende Mail speichert.

Parameter:

- rsAkt (Datensatz der Aktivität der E-Mail)
- rsAccount (Datensatz des Mailkontos, an den die Mail ging)
- true (ist immer true)
- blnCancel (mit true wird das Speichern der Mail abgebrochen)

### 32.2 MessageProcessed

Sobald eine eingehende Mail auf der Datenbank gespeichert wurde, wird dieses Event aufgerufen. Die Merkmale auf der erstellten Aktivität sind noch nicht vorhanden.

Parameter:

- Msg (DartMail.Message mit dem Inhalt der E-Mail)
- rsAccount (Datensatz, des Mailkontos, an den die Mail ging)
- rsAkt (Datensatz der gespeicherten Aktivität)
- AktMkm (Merkmale, die auf die Aktivität gesetzt werden sollen)
- blnCancel (mit true wird das setzen der Merkmale unterbunden)

### 32.3 MessageSend

Dieses Event wird aufgerufen, bevor eine Mail aus FlowFact CRM versendet wird.

Parameter:

- strSmtpServer (Smtp Server)
- msg (DartMail.Message mit dem Inhalt der E-Mail)
- rs (Datensatz der E-Mail-Aktivität)

- rsBen (Datensatz des versendenden Benutzers [Besitzer der E-Mail])
- blnCancel (mit true wird der Versand der Mail unterbunden)

### 32.4 MessageSendError

Sobald beim Versenden einer Mail in FlowFact CRM ein Sendefehler festgestellt wird, wird dieses Event aufgerufen.

Parameter:

- strSmtpServer (Smtp Server)
- msg (DartMail.Message mit dem Inhalt der E-Mail)
- rs (Datensatz der E-Mail-Aktivität)
- rsBen (Datensatz des versendenden Benutzers [Besitzer der E-Mail])
- errNumber (Fehlernummer [Err.Number])
- strErrMsg (Fehlerbeschreibung [Err.Description])
- rsErrAkt (Datensatz der von FlowFact CRM erstellten Fehleraktivität)
- blnCancel (mit true wird der Vorgang abgebrochen)



## 33 Webserver – Events

Pfad: \FlowFact\system\Services\Websever\\*.vbs

### 33.1 Callback\_BeforeTransfer

Dieses Event wird vor der Übertragung zu einem Webserver aufgerufen.

Parameter:

- rsBoersen (Datensatz der Börse)
- strTempPath (Pfad zu den temporären Dateien)
- vCancel (mit true kann der Vorgang abgebrochen werden)

### 33.2 Callback\_AfterTransfer

Dieses Event wird nach der Übertragung zu einem Webserver aufgerufen.

Parameter:

- rsBoersen (Datensatz der Börse)
- strPhpFielPipeList (Liste mit „|“ getrennt aller übertragenen PHP Dateien)



## 34 OpenImmo – Events

Pfad: \FlowFact\system\Services\OpenImmo\ \*.vbs

### 34.1 Callback\_OpenImmoPicInit

Dieses Event wird bei der Vorbereitung jedes Bildes ausgeführt.

Parameter:

- rsBoersen (Datensatz der Börse)
- rsObj (Datensatz des aktuellen Objektes)
- vMMDateiKurz (Dateiname bei KurzBildern)
- vMMDateiLang (Dateiname bei LangBildern)
- vMaxCounter (Maximale Anzahl der Bilder)
- vCancel (mit true kann der Vorgang abgebrochen werden)
- vText (Textmeldung für Börsenübertragung)

### 34.2 Callback\_OpenImmoPicGet

Dieses Event wird beim Zugriff auf jedes Bild ausgeführt.

Parameter:

- rsBoersen (Datensatz der Börse)
- rsObj (Datensatz des aktuellen Objektes)
- lngNr (Nummer des Bildes)
- vPathAndFileName (Dateiname des Bildes)
- vTitel (Titel des Bildes)
- vCancel (mit true wird der Vorgang abgebrochen)
- vText (Textmeldung für Börsenübertragung)

### 34.3 Callback\_OpenImmoPicCopy

Dieses Event wird für jedes Bild vor dem Kopieren in den temporären Ordner ausgeführt.

Parameter:

- rsBoersen (Datensatz der Börse)
- rsObj (Datensatz des aktuellen Objektes)
- vFileName1 (Quelldateipfad)
- vFileName2 (Zieldateipfad)
- vGroesse (Größe des Zielbildes in Pixel)
- vBreite (Breite des Zielbildes in Pixel)
- vCopyRight ()
- vCancel (mit true wird der Vorgang abgebrochen)
- vText (Textmeldung für Börsenübertragung)

### 34.4 Callback\_BeforeXmlSave

Dieses Event wird aufgerufen, bevor die OpenImmo XML Datei gespeichert wird.

Parameter:

- rsBoersen (Datensatz der Börse)
- varXmlDoc (XMLDatei als Zeichenkette)
- varPath (Pfad der Zieldatei)
- varFileName (Name der Zieldatei)

### 34.5 Callback\_BeforeTransfer

Dieses Event wird aufgerufen, bevor die Übertragung zur Börse gestartet wird.

Parameter:

- rsBoersen (Datensatz der Börse)
- blnIsFtp (Ist die Übertragung eine via FTP)
- strPath ()
- strFileName ()

- `varFileFilter ()`

### **Rückgabewert**

Dieses Event kann einen String zurückgeben.

Fängt der Rückgabewert mit „>>>>“ an, dann wird die Übertragung mit einem Fehler abgebrochen, sonst wird der Text im Send.Log ausgegeben. Somit können Informationen aus einem Script in die Ausgabe der Übertragung geschrieben werden.

### **34.6 Callback\_AfterTransfer**

Dieses Event wird aufgerufen, sobald die Übertragung abgeschlossen wurde.

Parameter:

- `rsBoersen` (Datensatz der Börse)
- `blnIsFtp` (Ist die Übertragung eine via FTP)
- `blnTransferOk` (war die Übertragung erfolgreich)

### **Rückgabewert**

Dieses Event kann einen String zurückgeben.

Fängt der Rückgabewert mit „>>>>“ an, dann wird die Übertragung mit einem Fehler abgebrochen, sonst wird der Text im Send.Log ausgegeben. Somit können Informationen aus einem Script in die Ausgabe der Übertragung geschrieben werden.

### **34.7 Callback\_GetGeolocation**

Dieses Event wird aufgerufen, sobald die Geokoordinaten von FlowFact beim GeoWebservice abgefragt werden.

Bsp.: `Callback_GetGeolocation(ByRef strStrasse, ByRef strLand, ByRef strPLZ, ByRef strOrt, ByRef strLG, ByRef strBG)`

Es können alle Parameter verändert werden. Falls Längen- und Breitengrad (strLG, strBG) ein Wert zugewiesen wird, dann wird der GeoWebservice-Aufruf übersprungen und es werden dann diese Werte genommen.

## 35 IS24Rest Events

Pfad: \FlowFact\system\Services\IS24Rest\ \*.vbs

### 35.1 Callback\_BeforeTransfer

Dieses Event wird aufgerufen, bevor die Übertragung zur Börse gestartet wird.

Parameter:

- rsBoersen (Datensatz der Börse)

#### Rückgabewert

Dieses Event kann einen String zurückgeben.

Fängt der Rückgabewert mit „>>>>“ an, dann wird die Übertragung mit einem Fehler abgebrochen, sonst wird der Text im Send.Log ausgegeben. Somit können Informationen aus einem Script in die Ausgabe der Übertragung geschrieben werden.

### 35.2 Callback\_AfterTransfer

Dieses Event wird aufgerufen, sobald die Übertragung abgeschlossen wurde.

Parameter:

- rsBoersen (Datensatz der Börse)
- blnTransferOk (war die Übertragung bisher erfolgreich)

#### Rückgabewert

Dieses Event kann einen String zurückgeben.

Fängt der Rückgabewert mit „>>>>“ an, dann wird die Übertragung mit einem Fehler abgebrochen, sonst wird der Text im Send.Log ausgegeben. Somit

können Informationen aus einem Script in die Ausgabe der Übrerragung geschrieben werden.

## 36 Hinweis FlowFact 2014

Betroffene Events:

BeforePaneRSChanged  
PaneRSChanged  
DetailChanged  
AfterDetailChanged  
Fill  
X\_MergeFieldGet  
Callback\_NewAppointment  
CanDelete

Wird in einem Eventskript der Datensatz im aktuellen Fenster neu geladen, dann wird intern mit dem alten Datensatz weiter gearbeitet.  
Das hat u.a. zur Folge, dass der "Lock-Mechanismus" meldet: "Der Datensatz ist nicht mehr aktuell, die Ansicht wird jetzt neu geladen."

Lösung:

- 1) Das Recordset wird ab Build 4,56 als "ByRef" übergeben.
- 2) Das Recordset muss im Script noch einmal neu geladen werden, wenn es
  - a) mit UptateRecordset gespeichert wurde.
  - b) ein anderer Datensatz geladen wurde.

```
Application.Panes("F_OBJ").control.om 4,0  
'Refresh der Ansicht, damit Meldung nicht kommt
```

**Workaround nur im Obj-Fenster:**

```
Application.Panes("F_OBJ").control.ctlEditDetailsChangeAbort = true
```



## 37 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](mailto:info@flowfact.de)  
[www.flowfact.de](http://www.flowfact.de)