

Below are the scripts for adding multiple associations for what are Contacts and Projects in the 'standard' DayBack file - I am using these for Referrals and Cases respectively.

You need the UniqueList custom function ( <http://www.briandunning.com/cf/789> ), tweaks to the DBk\_EventSummaryCalc and DBk\_WebViewSource field definitions and additional entries in the Set Variable \$\$sc\_CustomEventActions step (line 29) in the Load Source Settings at Startup... script

### Changes to definition for DBk\_EventSummaryCalc field (in green)

Let ( [

//===== Edit the lines below to reflect the fields in your events table.  
=====

```
s = Summary ; // the title of your event
desc = Description ; // your event's description or notes field
Ref = Case(not IsEmpty(id_contact); "Referral(s): " & Substitute
( List(SampleContacts::zkSerialNump) ; "¶";, "")) ;
cse = Case(not IsEmpty(id_Project); "Case(s): " & Substitute
( List(SampleProjects::zkSerialNump) ; "¶";, "")) ;
res = ziResources ;
cont = DBk_Unused ; // your contact's name (optional)
ph = DBk_Unused ; // your contact's phone number (optional)
em = DBk_Unused ; // your contact's email (optional)
```

//===== You shouldn't have to edit below this line =====

```
d = Left ( desc ; 300 )
```

```
];
```

// Use the list function to concatenate a number of different fields together

```
Substitute ( List ( s ; Ref ; cse ; cont ; ph ; em ; d ; res ) ; "¶" ; "\n" )
```

```
)
```

```
*/
```

### Changes to field definition of DBk\_WebViewSource field (in green)

Let ( [

//===== Assign your event fields here =====

sc\_sourceNumber = 1 ; // The source number as defined in the "Source No" layout  
sc\_id = id ; // the unique ID or primary key of your event record. Often, it's serial number.

sc\_startDate = DateStart ;

sc\_startTime = TimeStart ;

sc\_endDate = DateEnd ;

sc\_endTime = TimeEnd ;

sc\_summary = Summary ;

sc\_summaryCalc = DBk\_EventSummaryCalc ; //Read only

sc\_description = Description ;

sc\_resource = Resource ;

sc\_status = Status ;

sc\_contactID = id\_contact ;

sc\_contactName = Substitute ( List(SampleContacts::zkSerialNump) ; "¶";, " ) ;

sc\_projectID = id\_Project ;

sc\_projectName = Substitute ( List(SampleProjects::zkSerialNump) ; "¶";, " ) ;

### **Changes to Load Source Settingsat Startup..... script**

List (

"Add Referral to Event, Add Referral to Event" ;

"Add Case to Event, Add Case to Event" ;

"Remove Referral from Event, Remove Referral from Event" ;

"Remove Case from Event, Remove Case from Event" ;

""

)

And you get

11:00 - 12:45  
Test Referral action  
Referral(s): 2, 1008  
Meeting Room 1,  
Christin Streczywilk

Test Referral action

Description

☐ All Day

StartWednesday, 9 Sep, 201511:00

End1 hour 45 minutes12:45

ResourceMeeting Room 1, Christin Strec...

StatusCompleted Chargeable

Referral2, 1008 | ➔

Casenone

Delete

Save & Close

⚙

Call your own scripts

Add Referral to Event

Add Case to Event

Remove Referral from Event

Remove Case from Event

Close

?

## DayBack Scripts: Calendar: Custom Action Examples: Add Referral to Event

---

```
If [ not IsEmpty ($eventID) or not IsEmpty (Get(ScriptParameter)) ]  
    Set Variable [ $eventID; Value:Case(IsEmpty($eventID); Get(ScriptParameter) ; $eventID) ]  
    Set Field [ GLOBALS::inputNumber1; "" ]  
    Show Custom Dialog [ Title: "Add Referral to event"; Message: "Enter the Referral Number to be added"; Default Button:  
        "Add", Commit: "Yes"; Button 2: "Cancel", Commit: "No"; Button 3: "Add More", Commit: "No"; Input #1: GLOBALS::  
        inputNumber1, "Referral Number" ]  
    Set Variable [ $action; Value:Choose( Get(LastMessageChoice)-1; "add"; "cancel";"more") ]  
    If [ $action="cancel" ]  
        Exit Script [ ]  
    End If  
    #check to see if an active and valid referral number  
    Set Variable [ $refUID; Value:ExecuteSQL ( "Select zkUIDp from Referrals where zkSerialNump=? and zkIsActive=0" ; "" ; "" ;  
        GLOBALS::inputNumber1) ]  
    If [ IsEmpty ( $refUID ) ]  
        Show Custom Dialog [ Title: "Alert"; Message: GLOBALS::inputNumber1 & " is not an active Referral"; Default Button:  
            "OK", Commit: "Yes" ]  
        Set Variable [ $action; Value:"more" ]  
    Else  
        Set Field [ SELECTOR::UIDSelectedCalendarEvent; $eventID ]  
        Set Field [ Selected_CalendarEvent::id_contact; UniqueList ( Selected_CalendarEvent::id_contact & ¶ & $refUID ; 0 ) ]  
    End If  
    If [ $action="more" ]  
        Perform Script [ "Refresh Calendar Event (Data ; {EventID ; RenderID ; QueryID ; Operation ; ForceRefresh} )";  
            Parameter: # ( "sc_EventID" ; $EventID ) &  
                # ( "sc_ShowEvent" ; True ) &  
                # ( "sc_Operation" ; "reload" ) ]  
    Else  
        Perform Script [ "Add Referral to Event"; Parameter: $eventid ]  
    End If  
End If
```

Add Case to Event is a duplicate with changes to the validation, messages and Set Field target

## DayBack Scripts: Calendar: Custom Action Examples: Remove Referral from Event

---

```
If [ not IsEmpty ($eventID) ]  
    Set Field [ GLOBALS::inputNumber1; "" ]  
    Show Custom Dialog [ Title: "Remove Referral from event"; Message: "Enter the Referral Number to be remove"; Default  
        Button: "Remove", Commit: "Yes"; Button 2: "Cancel", Commit: "No"; Input #1: GLOBALS::inputNumber1, "Referral  
        Number" ]  
    Set Variable [ $action; Value:Choose( Get(LastMessageChoice)-1; "add"; "cancel";"more") ]  
    If [ $action="cancel" ]  
        Exit Script [ ]  
    End If  
    #Check if Referral is included in the event  
    Set Variable [ $refUID; Value:ExecuteSQL ( "Select zkUIDp from Referrals where zkSerialNump=? and zkIsActive=0" ; "" ; "" ;  
        GLOBALS::inputNumber1) ]  
    Set Field [ SELECTOR::UIDSelectedCalendarEvent; $eventID ]  
    If [ IsEmpty ( FilterValues ( Selected_CalendarEvent::id_contact ; $refUID )) ]  
        Show Custom Dialog [ Title: "Alert"; Message: "Referral " & GLOBALS::inputNumber1 & " is not included in this event";  
            Default Button: "OK", Commit: "Yes" ]  
  
    Else  
        Set Field [ Selected_CalendarEvent::id_contact; list.AddorRemove ( $refUID ; Selected_CalendarEvent::id_contact ) ]  
    End If  
    Perform Script [ "Refresh Calendar Event (Data ; {EventID ; RenderID ; QueryID ; Operation ; ForceRefresh} )"; Parameter: #  
        ( "sc_EventID" ; $EventID ) &  
        # ( "sc_ShowEvent" ; True ) &  
        # ( "sc_Operation" ; "reload" ) ]  
End If
```

Remove Case from Event is a duplicate with changes to the validation, messages and Set Field target