




# **ix** Rückwirkende Änderung einzelner Werte in Datensätzen



UNITED PLANET INTREXX XTREME  
RELEASE 4.5







## Inhaltsverzeichnis

1. Einleitung .....	3
2. Vorgehensweise .....	6



### Schreibkonventionen

In diesem Handbuch werden Textstellen *kursiv* dargestellt, wenn sie sich auf Einstellungen in den abgebildeten Dialogen beziehen. Menüpunkte, die in Kontextmenüs erreichbar sind, sind immer auch über das Hauptmenü erreichbar. Hauptmenüpunkte werden nicht beschrieben, es sei denn, sie sind nicht über das Kontextmenü erreichbar. Eine Beschreibung der allgemeinen Hauptmenüpunkte finden Sie im Handbuch  *Center*. Programmiercode im Text wird in der Schriftart *Courier* dargestellt. Kontextmenüs können mit einem Klick mit der rechten Maustaste auf das beschriebene Element geöffnet werden.

<xtreme> bezeichnet im Folgenden Ihren Intrexx Installationspfad, unter Windows z.B.  *C:\xtreme\*, unter Linux z.B.  */opt/xtreme/*. Folgende Symbole werden für die Kennzeichnung von speziellen Informationen verwendet:

-  Wichtige Hinweise
-  Tipps und Hintergrundinformationen
-  Verweise auf weiterführende Informationen in einem Intrexx Xtreme Handbuch
-  Verzeichnisse
-  URLs
-  Schaltflächen in Dialogen oder Assistenten

### Vorkenntnisse

Für das Verständnis dieser Dokumentation sind keine speziellen Vorkenntnisse erforderlich. Hilfreiche Informationen finden Sie in den Intrexx Xtreme Handbüchern  *Setup* und  *Start*.

### 1. Einleitung

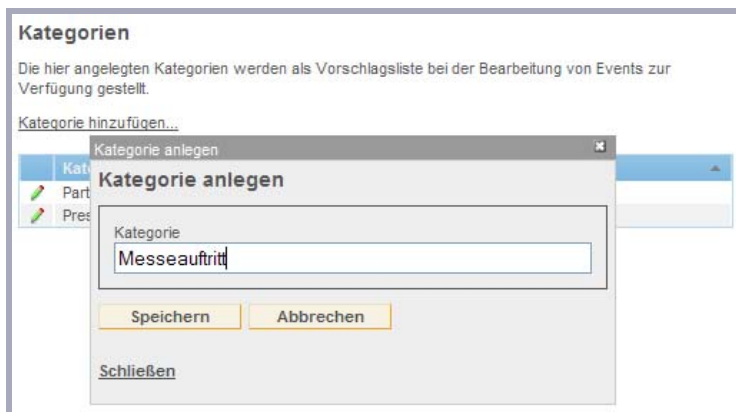
In einer Anwendung soll bei der Anlage eines Datensatzes, z. B. zu einem Event, aus einer Auswahlliste ein vorgegebener Wert, z. B. eine Kategorie, ausgewählt werden. Oft wird dies in Applikationen so umgesetzt, dass die vorzugegebenen Werte in einer eigenen Datengruppe Kategorie verwaltet werden. Sie werden dann durch Anlage einer Referenz in der Datengruppe bereitgestellt bzw. gespeichert, in der auf die vordefinierten Werte zugegriffen werden soll, in unserem Beispiel also in der Datengruppe Event. Ändert man nun z. B. die Bezeichnung in der referenzierten Datengruppe Kategorie, dann wirkt sich die Änderung auf alle Datensätze der Datengruppe Event aus, die zu dieser Kategorie gehören. Es werden also rückwirkend bereits angelegte Datensätze geändert, was jedoch nicht immer gewünscht ist.

Dieser Werkstattbeitrag zeigt Ihnen anhand eines einfachen Beispiels, wie Sie dem Benutzer die Möglichkeit geben, die Bezeichnung von Vorgabewerten zu verändern und dabei im Einzelfall zu entscheiden, inwieweit die Änderungen für bereits angelegte Datensätze übernommen werden sollen.

Folgende Varianten werden angeboten:

- die Änderungen des Vorgabewerts werden nicht auf bereits angelegte Datensätze übertragen,
- die Änderungen des Vorgabewerts werden für alle bereits angelegten Datensätze mit diesem Vorgabewert übernommen oder
- die Änderungen des Vorgabewerts werden für alle bereits angelegten Datensätze mit diesem Vorgabewert übernommen deren Beginndatum nach dem anzugebenden Datum liegt.

In der beiliegenden Beispielapplikation gibt es eine einfache Eingabeseite, über die Kategorien angelegt werden können, die als Vorgabewerte verwendet werden:

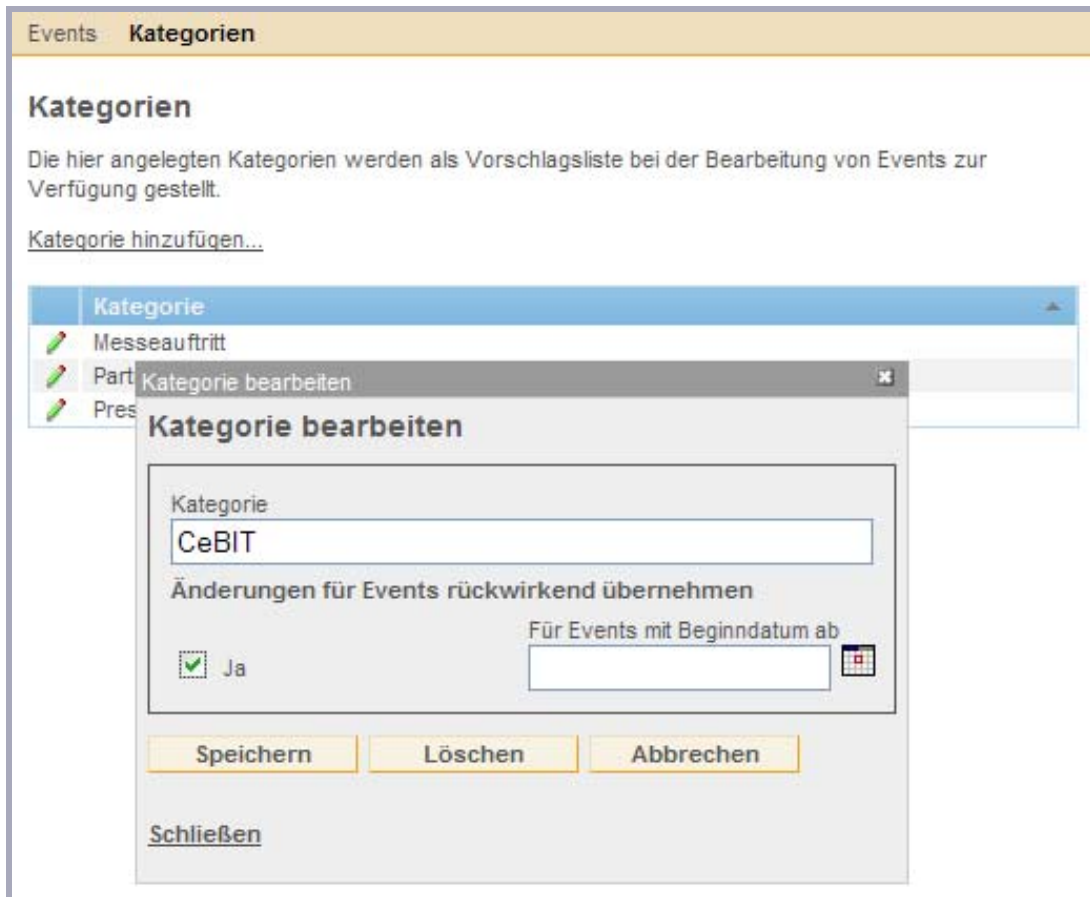


Über eine weitere Eingabeseite können Events erfasst werden, denen jeweils eine Kategorie zugeordnet wird:

Nach dem Speichern einiger Beispieldaten erhalten Sie in der Tabelle eine Ansicht ähnlich der folgenden:

Events		Kategorien		
Events				
Neues Event hinzufügen...				
	Von	Bis	Event	Kategorie
	09.10.2009 09:00	09.10.2009 17:00	Produktpräsentation 2009	Partnereinführung
	03.03.2009 07:00	08.03.2009 20:00	Messestand 2009	Messeauftritt
	10.10.2008 09:00	10.10.2008 17:00	Produktpräsentation 2008	Partnereinführung
	04.03.2008 07:00	09.03.2008 20:00	Standbeteiligung 2008	Messeauftritt
	05.10.2007 09:00	05.10.2007 17:00	Produktpräsentation 2007	Partnereinführung
	15.03.2007 07:00	21.03.2007 20:00	Standbeteiligung 2007	Messeauftritt

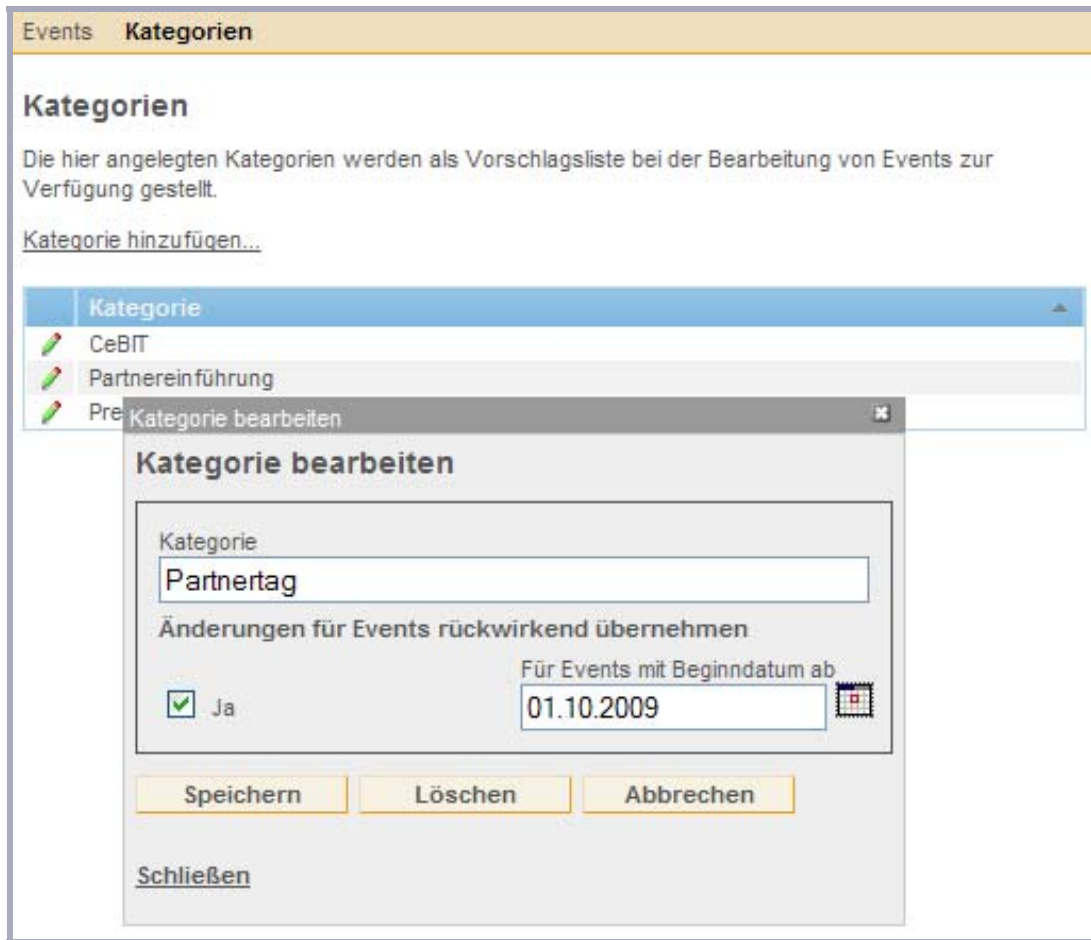
Nun soll in diesem Beispiel die Kategorie *Messeauftritt* für die bisher angelegten Events in *CeBIT* geändert werden, weil in Zukunft auch Auftritte auf anderen Messen in Planung sind, was über die Kategorie unterschieden werden soll. Da die Änderungen rückwirkend für die betreffenden Events übernommen werden soll, muss beim Bearbeiten der Kategorie die Checkbox *Ja* aktiviert werden:



Nach Speichern der Änderung sieht die Übersicht der Events wie folgt aus:

Von	Bis	Event	Kategorie
09.10.2009 09:00	09.10.2009 17:00	Produktpräsentation 2009	Partnereinführung
03.03.2009 07:00	08.03.2009 20:00	Messestand 2009	CeBIT
10.10.2008 09:00	10.10.2008 17:00	Produktpräsentation 2008	Partnereinführung
04.03.2008 07:00	09.03.2008 20:00	Standbeteiligung 2008	CeBIT
05.10.2007 09:00	05.10.2007 17:00	Produktpräsentation 2007	Partnereinführung
15.03.2007 07:00	21.03.2007 20:00	Standbeteiligung 2007	CeBIT

Wird zusätzlich zur Aktivierung der Checkbox zur Übernahme der Änderungen für bereits bestehende Events ein Datum eingegeben, werden die Änderungen nur für Events übernommen, deren Beginndatum an oder nach diesem Datum liegt. Ein denkbarer Anwendungsfall ist z. B. die Umbenennung der Veranstaltungsreihe *Partnereinführung* in *Partnertag* für alle Veranstaltungen ab dem 01. Oktober 2009:

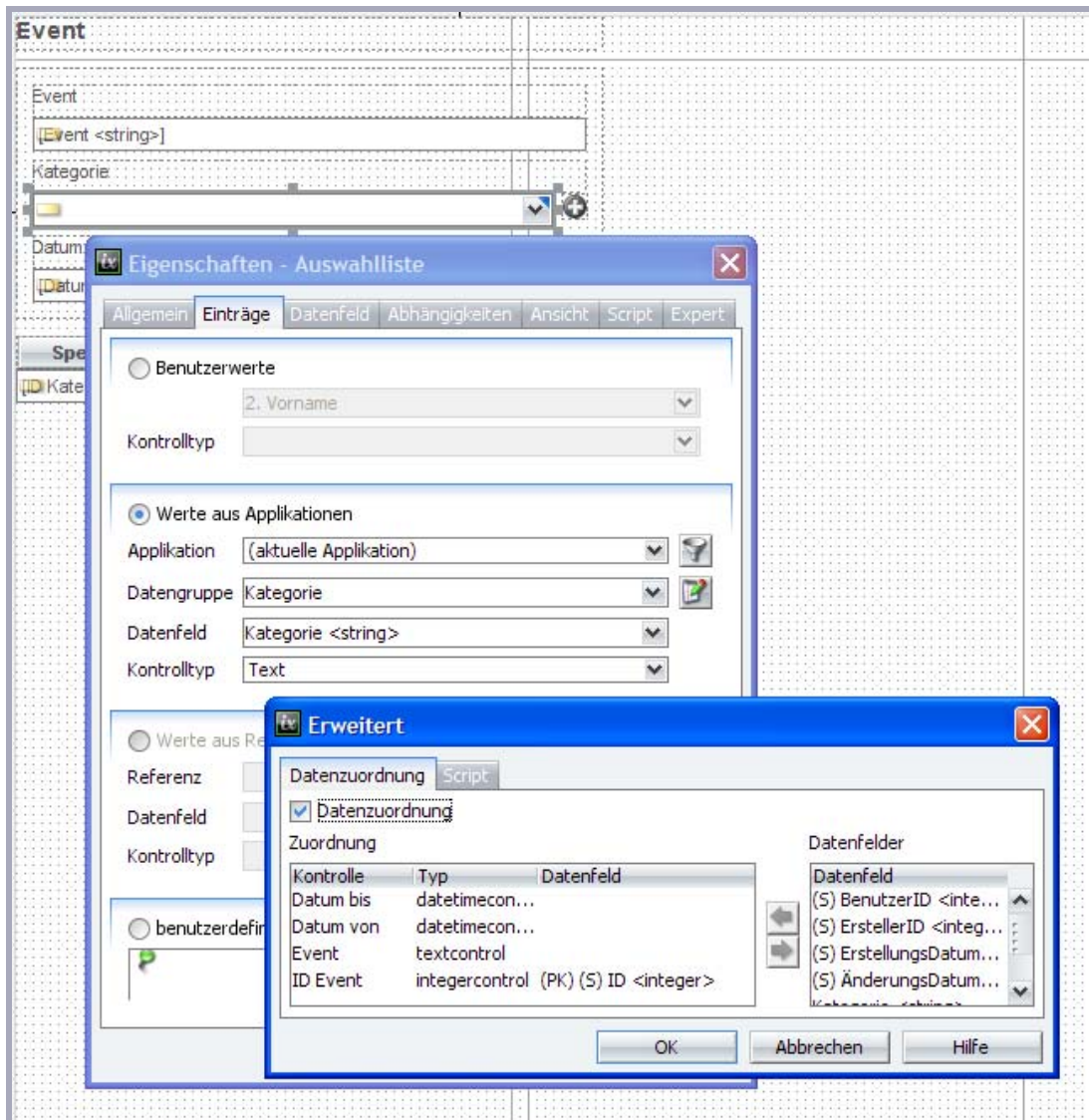


Nach Speichern der Änderung sieht die Übersicht der Events wie folgt aus:

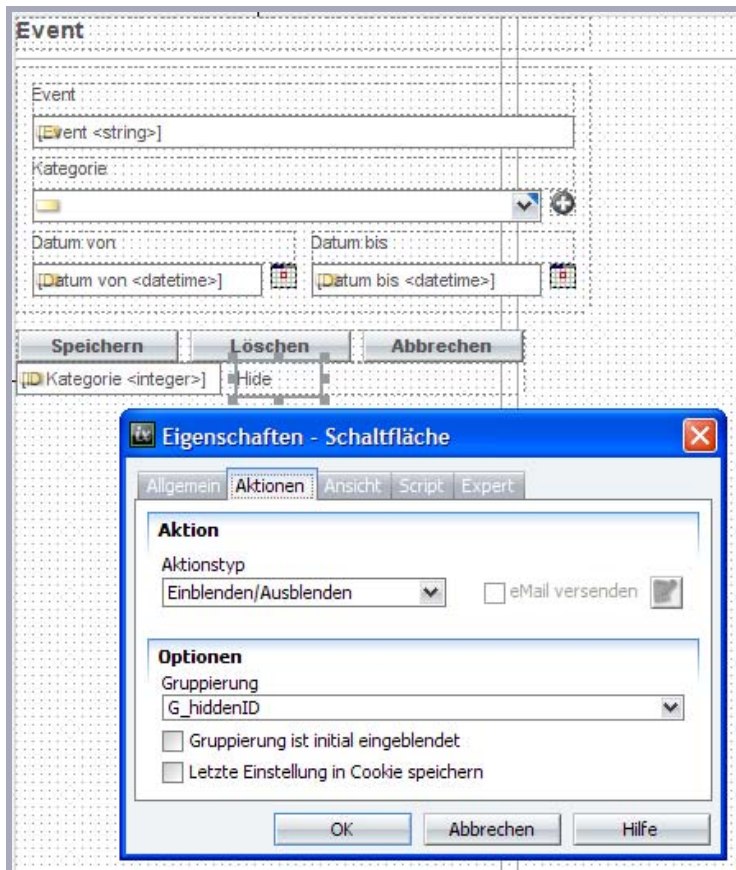
	Von	Bis	Event	Kategorie
	09.10.2009 09:00	09.10.2009 17:00	Produktpräsentation 2009	Partnertag
	03.03.2009 07:00	08.03.2009 20:00	Messestand 2009	CeBIT
	10.10.2008 09:00	10.10.2008 17:00	Produktpräsentation 2008	Partnereinführung
	04.03.2008 07:00	09.03.2008 20:00	Standbeteiligung 2008	CeBIT
	05.10.2007 09:00	05.10.2007 17:00	Produktpräsentation 2007	Partnereinführung
	15.03.2007 07:00	21.03.2007 20:00	Standbeteiligung 2007	CeBIT

2. **Vorgehensweise**

Öffnen Sie im Applikationsdesigner die Eingabeseite, auf der ein Vorgabewert aus einer anderen Datengruppe zugeordnet werden soll. Erstellen Sie eine Auswahlliste, die ihre Werte aus der Datengruppe bezieht, in der die Vorgabewerte verwaltet werden. Zugeordnet wird ein Datenfeld des Typs String, wenn es sich um eine Bezeichnung handelt. Legen Sie zusätzlich ein Eingabefeld für die ID des Datensatzes des gewählten Vorgabewerts an. Um dieses Feld zu füllen, klicken Sie bei der Auswahl der Einträge für die Liste auf *Erweitert* und konfigurieren die Datenzuordnung wie folgt:



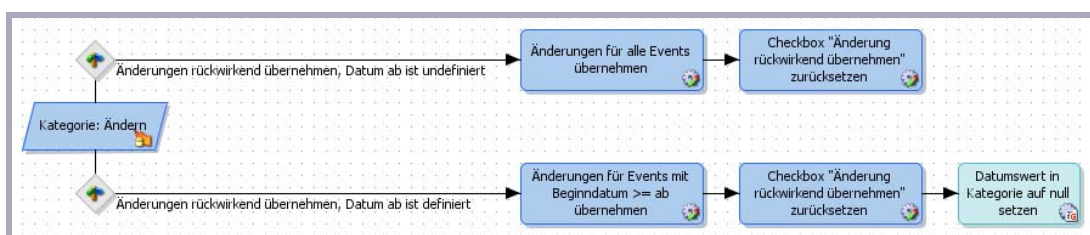
Legen Sie um das Feld für die ID eine Gruppierung, in der Sie zusätzlich eine Schaltfläche erstellen mit der Aktion *Einblenden/Ausblenden* für die umgebende Gruppierung. Damit ist das Feld versteckt.



Legen Sie auf der Seite zum Ändern der Vorgabewerte eine Checkbox an. Wird sie angehakt, sollen die Änderungen für bereits angelegte Datensätze übernommen werden. Zusätzlich ist ein Eingabefeld erforderlich für das Datum, ab dem die bereits angelegten Datensätze angepasst werden sollen.

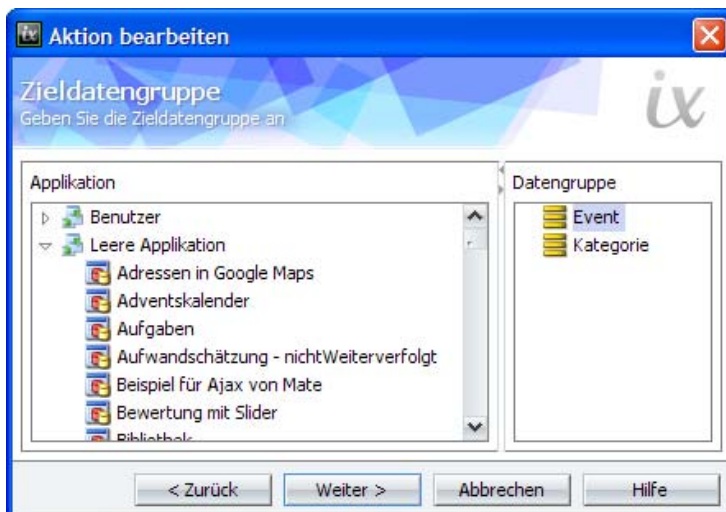
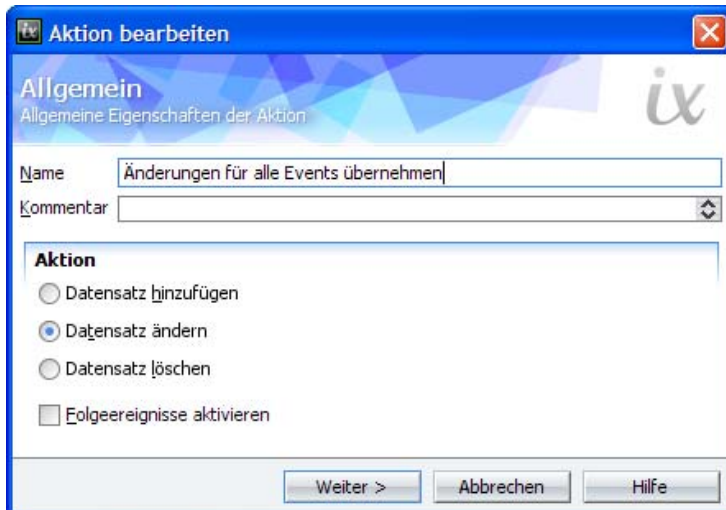


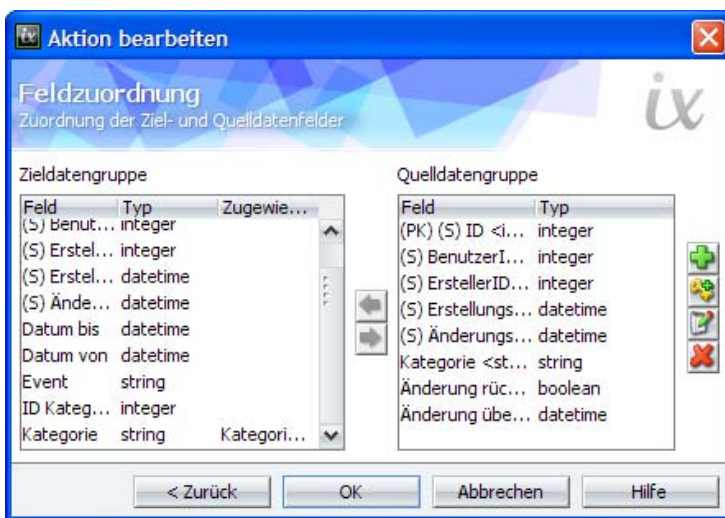
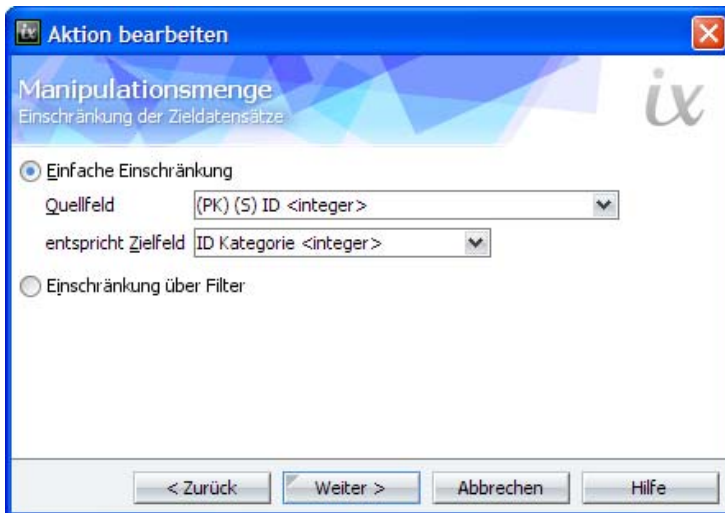
Die Übernahme geänderter Vorgabewerte für bereits angelegte Datensätze wird über einen Prozess gesteuert:



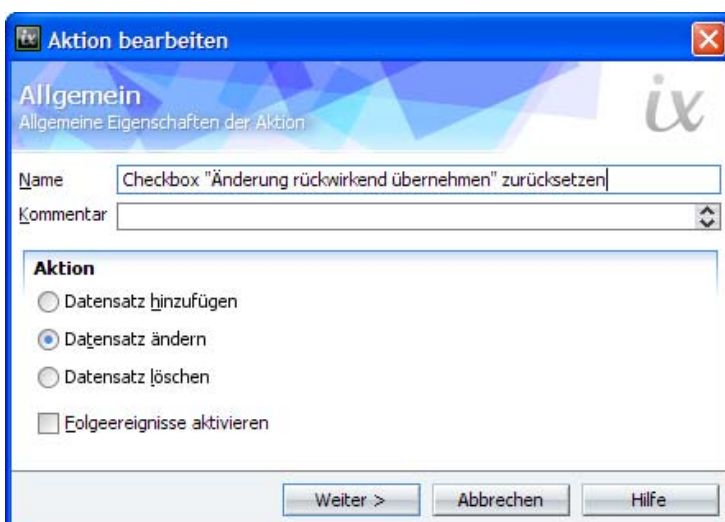
Auslösendes Datengruppenereignis ist hier die Änderung einer Kategorie, also des Vorgabewerts. In den Bedingungen wird geprüft, ob die Checkbox bei Änderung des Vorgabewerts aktiviert worden ist.

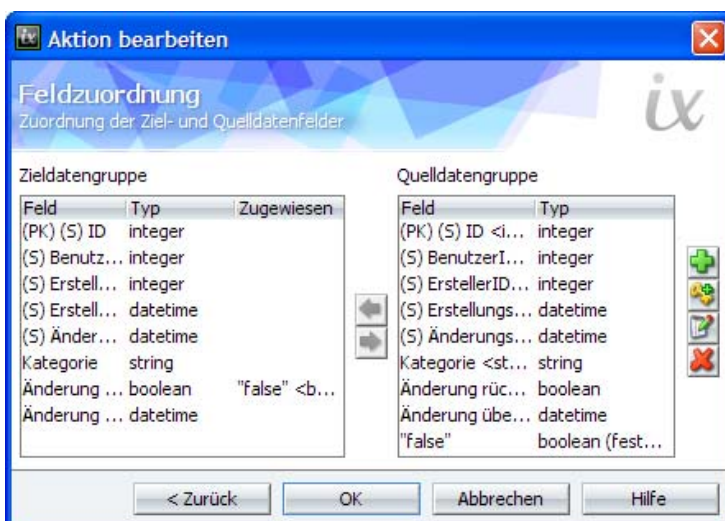
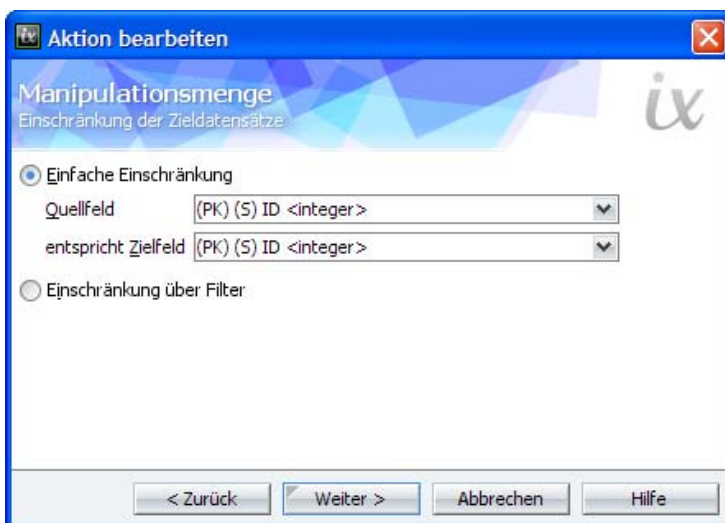
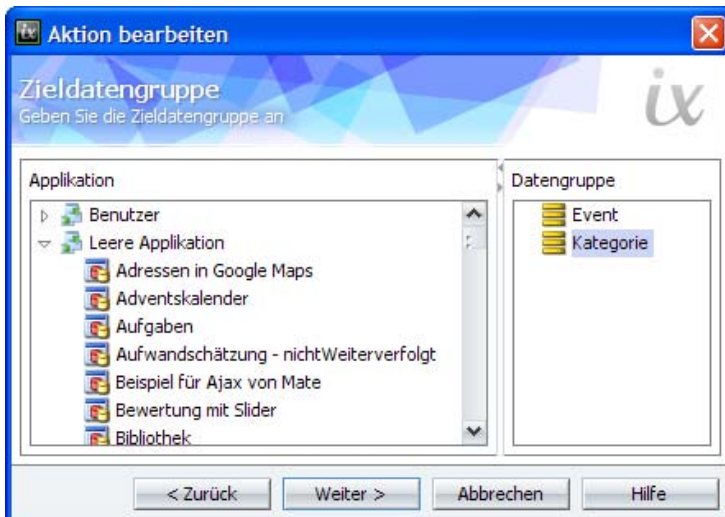
Ist dies der Fall und wurde kein Datum angegeben, ab dem die Änderungen auf bereits angelegte Datensätze übertragen werden sollen, wird der obere Zweig des Prozesses abgearbeitet. Hier wird im ersten Schritt eine Datengruppenaktion ausgeführt, die die Änderung für alle Datensätze (hier *Events*) übernimmt, die den Vorgabewert (hier *Kategorie*) enthalten. Die folgenden Screenshots zeigen die Konfiguration der Datengruppenaktion:





Mit **OK** schließen Sie die Konfiguration der Datengruppenaktion ab. Im nächsten Schritt wird die Checkbox im Datensatz des Vorgabewerts durch eine weitere Datengruppenaktion zurückgesetzt:





Wurde bei der Änderung des Vorgabewerts neben der Aktivierung der Checkbox ein Datum angegeben, kommt der untere Zweig des Prozesses zur Ausführung. Im ersten Schritt kommt hier die Datengruppenaktion *Änderungen für Events mit Beginndatum >= ab übernehmen* zur Ausführung. Die Aktion wird wie die oben beschriebene Datengruppenaktion *Änderungen für alle Events übernehmen* konfiguriert, nur dass hier die Einschränkung der Zieldatensätze über einen Filter erfolgt:

The screenshot shows a dialog box titled 'Aktion bearbeiten' with a close button in the top right corner. The main heading is 'Manipulationsmenge' with the subtitle 'Einschränkung der Zieldatensätze'. There are two radio button options: 'Einfache Einschränkung' (unselected) and 'Einschränkung über Filter' (selected). Under 'Einfache Einschränkung', there are two dropdown menus: 'Quellfeld' with the value '(PK) (S) ID <integer>' and 'entspricht Zielfeld' with the value '(PK) (S) ID <integer>'. At the bottom, there are four buttons: '< Zurück', 'Weiter >', 'Abbrechen', and 'Hilfe'.

The screenshot shows the same dialog box 'Aktion bearbeiten' but at the 'Filter' step. The subtitle is 'Filter für die Einschränkung der Zieldatensätze'. A tree view shows a collapsed 'Und' group containing two conditions: 'ID Kategorie = ID' and 'Datum von >= Änderung übernehmen für Events ab'. To the right of the tree view are four icons: a green plus sign, a grey minus sign, a trash can, and a red 'X' icon. At the bottom, there are four buttons: '< Zurück', 'Weiter >', 'Abbrechen', and 'Hilfe'.

Die anschließende Datengruppenaktion *Checkbox "Änderung rückwirkend übernehmen"* zurücksetzen im zweiten Schritt wird wie oben beschrieben konfiguriert.

In diesem Zweig des Prozesses muss abschließend das Datum zurückgesetzt werden, das bei Änderung des Vorgabewerts angegeben wurde. Dies erfolgt durch das folgende Groovy-Skript:

```
import de.uplanet.lucy.server.dataobjects.impl.ValueHolderFactory as VH

// Systemconnection
l_conn = g_dbConnections.systemConnection

// ID Category, datafield (PK) (S) ID
def l_vhCategoryID = g_record["0B55BAD86F3BC46E03514894736197CFD6C512D9"]

// Constants datagroup category and field "date from"
static final String STR_TABLE_CATEGORY = "XWS08_CATEGORY"
static final String STR_FIELD_DATE_FROM = "DT_APPLY_CHANGES"

def l_stmtUpdate = g_dbQuery.prepare(l_conn, "UPDATE ${STR_TABLE_CATEGORY}
SET ${STR_FIELD_DATE_FROM} = null WHERE LID = ?")

l_stmtUpdate.setInt(1, l_vhCategoryID)
l_stmtUpdate.executeUpdate()

return true
```

Alle Schritte können Sie auch anhand der beiliegenden Beispielapplikation nachvollziehen.

Viel Vergnügen bei der Übertragung in Ihre eigenen Applikationen.