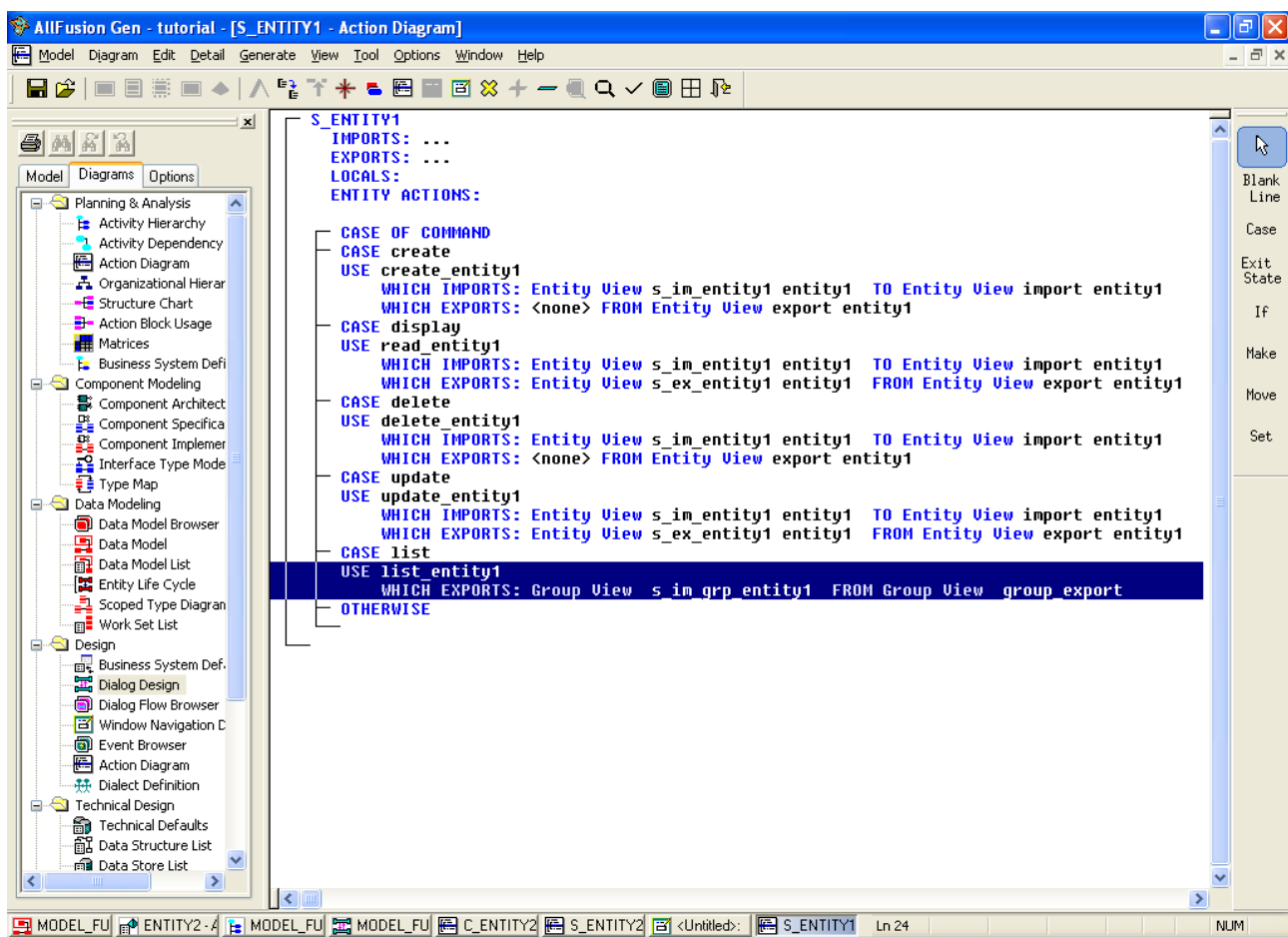


# 11. Events

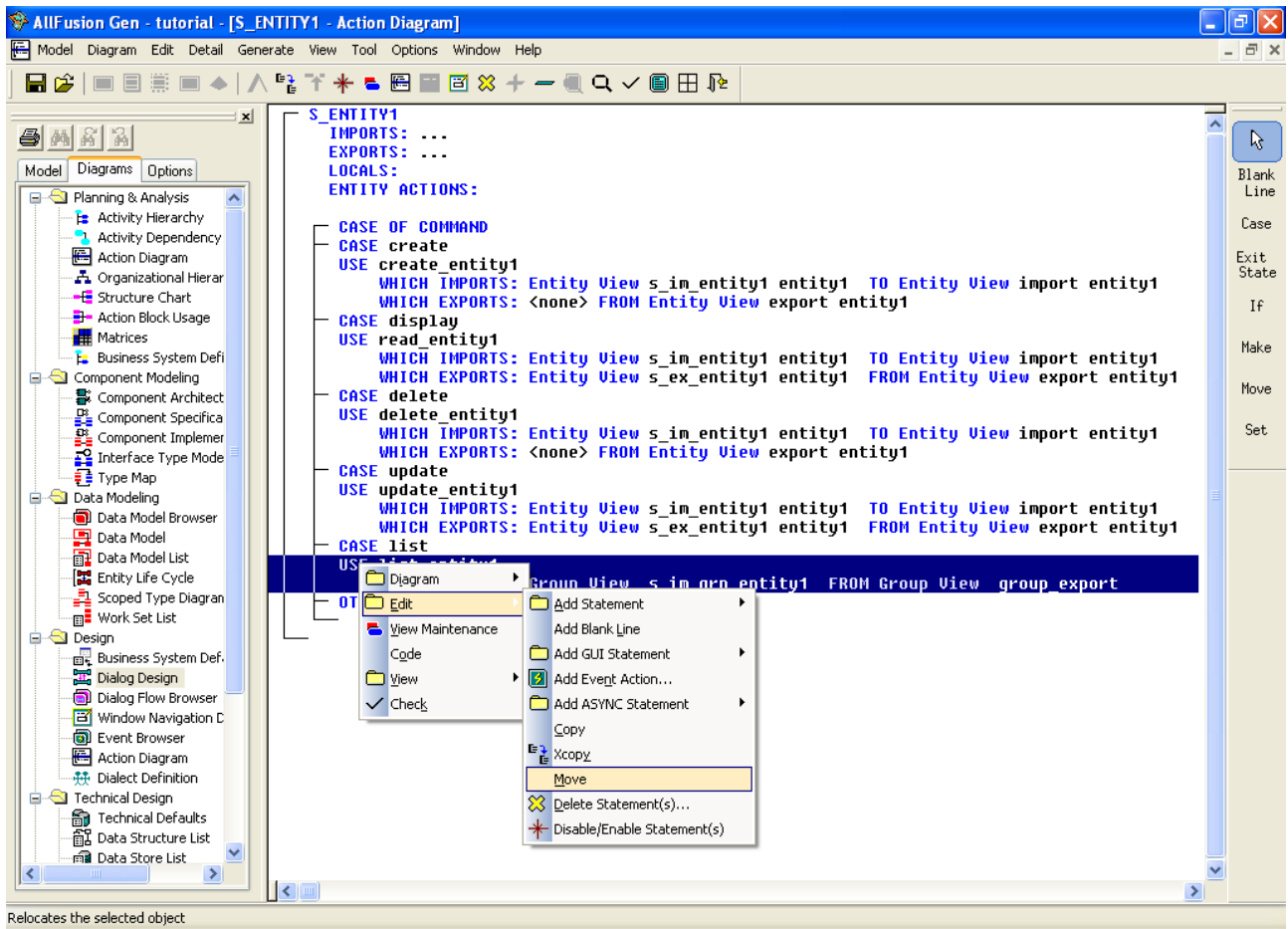
Nachdem Sie nun die grundlegenden Mechaniken kennengelernt haben, werden wir uns in diesem Kapitel mit den Steuerevents in GEN beschäftigen.

Zunächst gibt es aber noch eine Ergänzung zu Ihrem bisherigen Programm. Es ist Ihnen vielleicht beim Testen Ihres Programms aufgefallen, dass die Listbox sich wieder leert, wenn Sie einen anderen Button als "List" drücken.

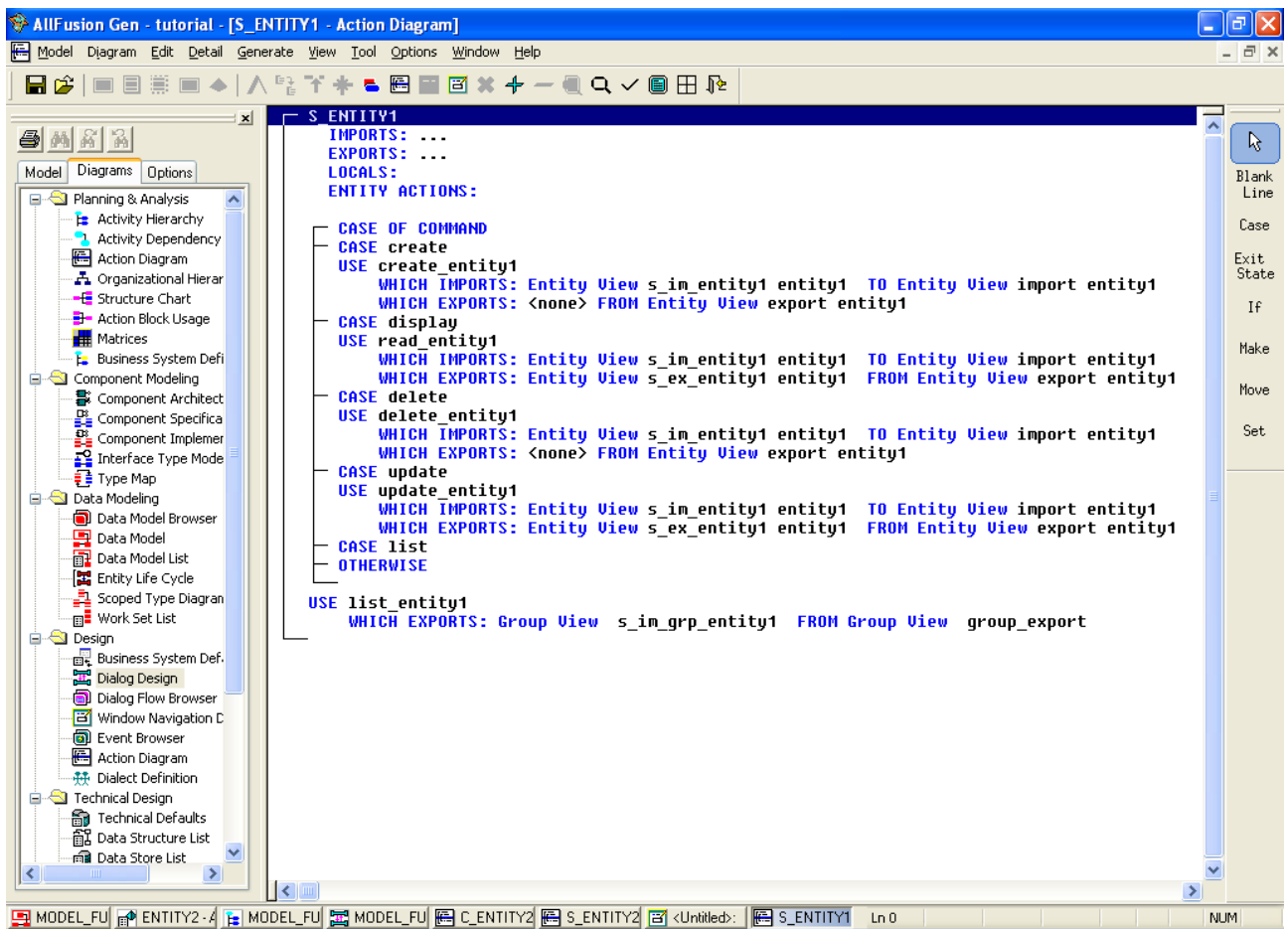
Das hängt damit zusammen wie die Daten im Server verarbeitet werden. Wenn Sie z.B. Create ausführen, bleibt die Group View im Server Export leer und wird auch so zurückgegeben. Dadurch wird die Liste im Client von einer leeren Liste überschrieben. Damit die Listbox a) stets gefüllt ist und b) immer aktuell ist (nach dem Löschen/Ändern/Erstellen von Daten) und b), müssen Sie nur eine Kleinigkeit in den Serverdialogen ändern.



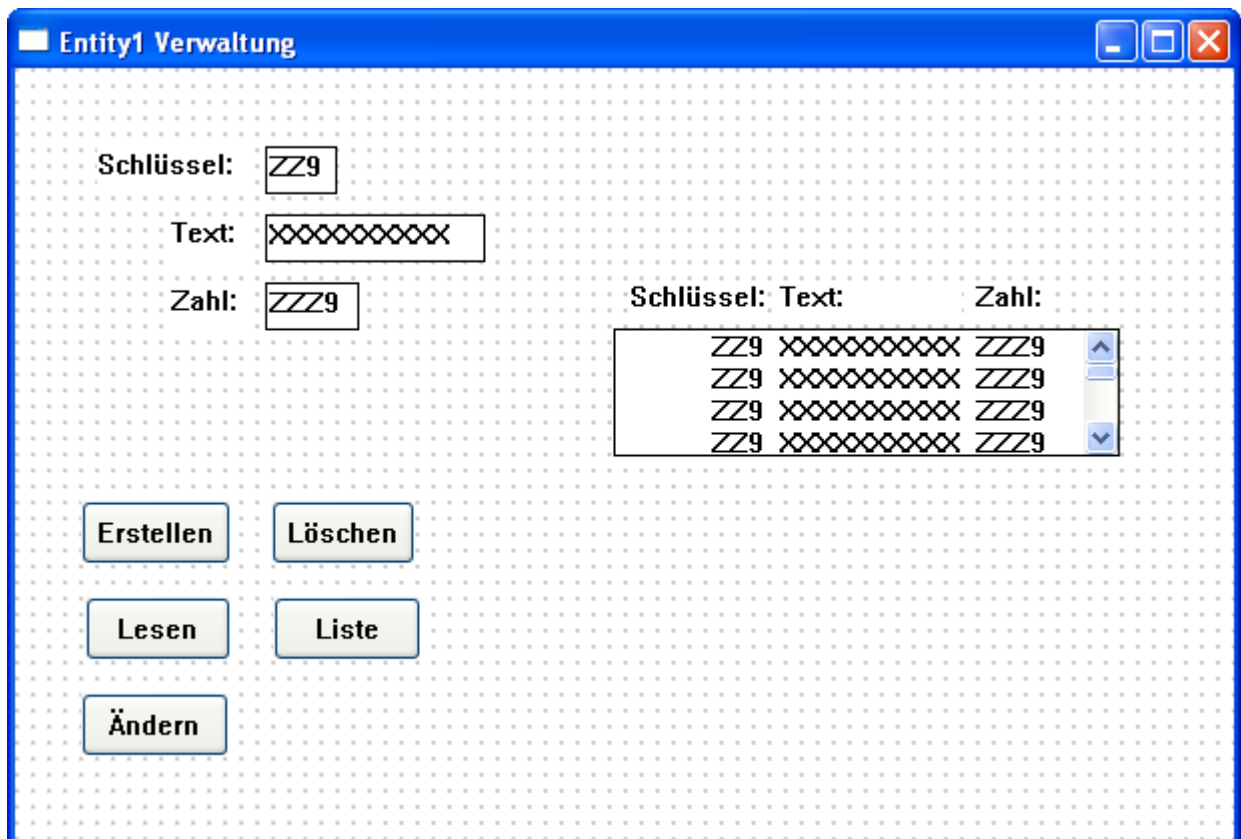
Das "USE list\_entity1" befüllt die Group View nur mit den Daten aus der Datenbank, wenn der Befehl "list" übermittelt wird. Damit die Group View nach jedem Aufruf des Servers befüllt wird, setzen Sie den USE Befehl einfach außerhalb der Case-Abfrage.



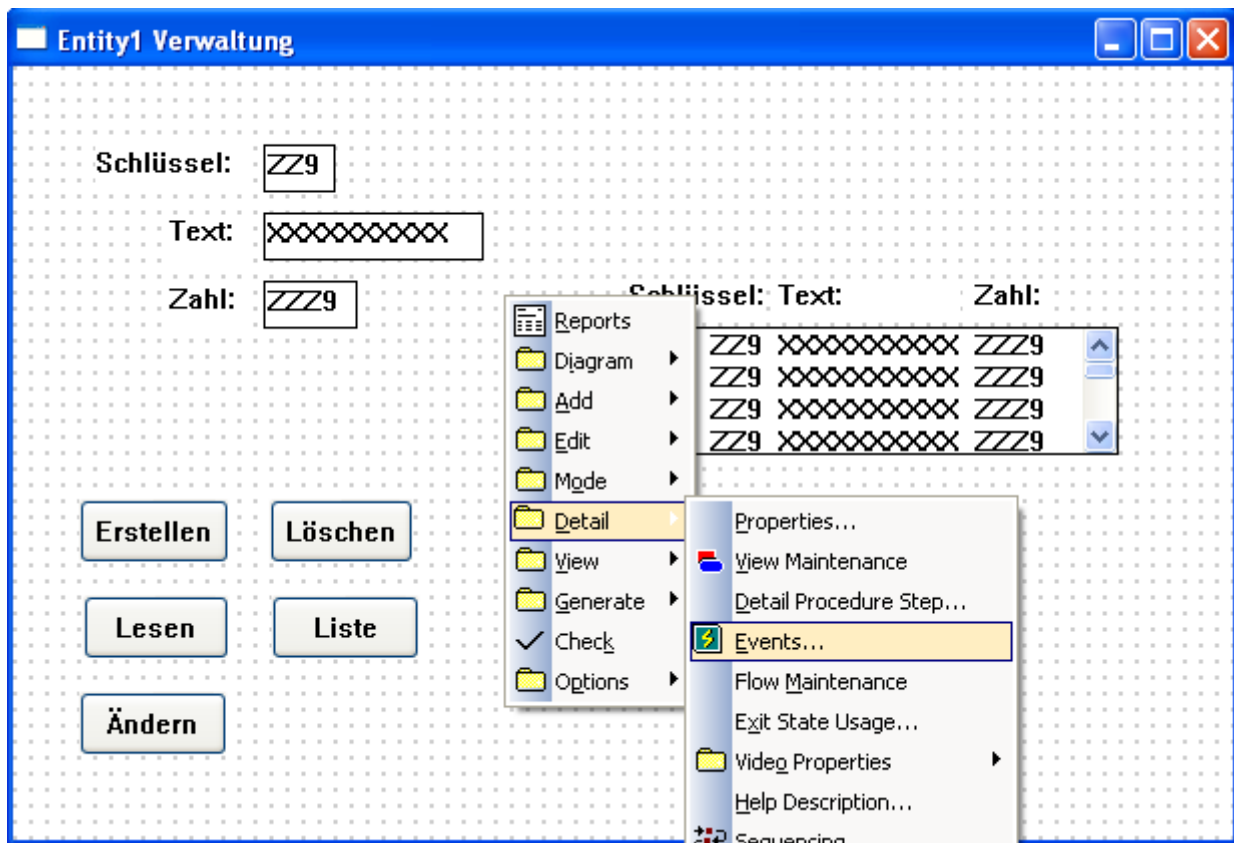
Über "Edit -> Move" können Sie einzelne Statements im Action Diagram verschieben. Markieren Sie das komplette Statement "USE list\_..." und verschieben Sie es unterhalb der Case-Abfrage.



Probieren Sie nun aus, ob Ihr Programm sich richtig verhält.



Was noch auffällt ist, dass beim Starten des Programms die Liste noch leer ist und Sie erst einmal auf den "Liste"-Button drücken müssen. Dies lässt sich über ein simples Event beheben.



Über einen Doppel-Rechtsklick oder über die Menüleiste oben im Programm, gehen Sie zu Detail – Events. Beachten Sie dabei, dass die Events jetzt für das vorher markierte Fensterelement gelten. Markieren Sie also zunächst den Hintergrund der Oberfläche (einfach auf eine freie Fläche klicken).

Es wird sich nun das Eventverwaltungs Fenster öffnen.

**Event Processing**

Window: Entity1 Verwaltung

Defined Events

Event Type	Action Name	Command

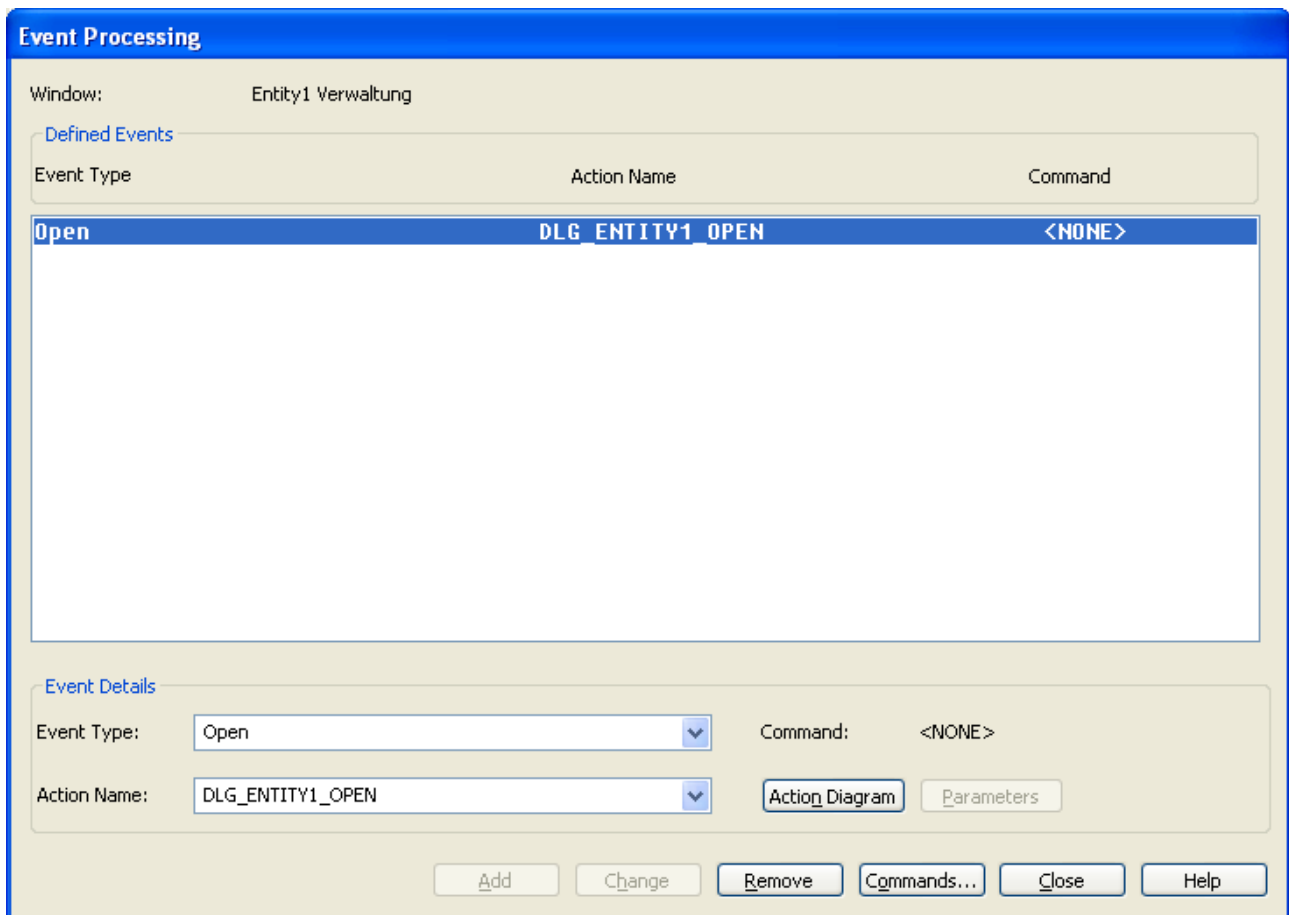
Event Details

Event Type:  Command: <NONE>

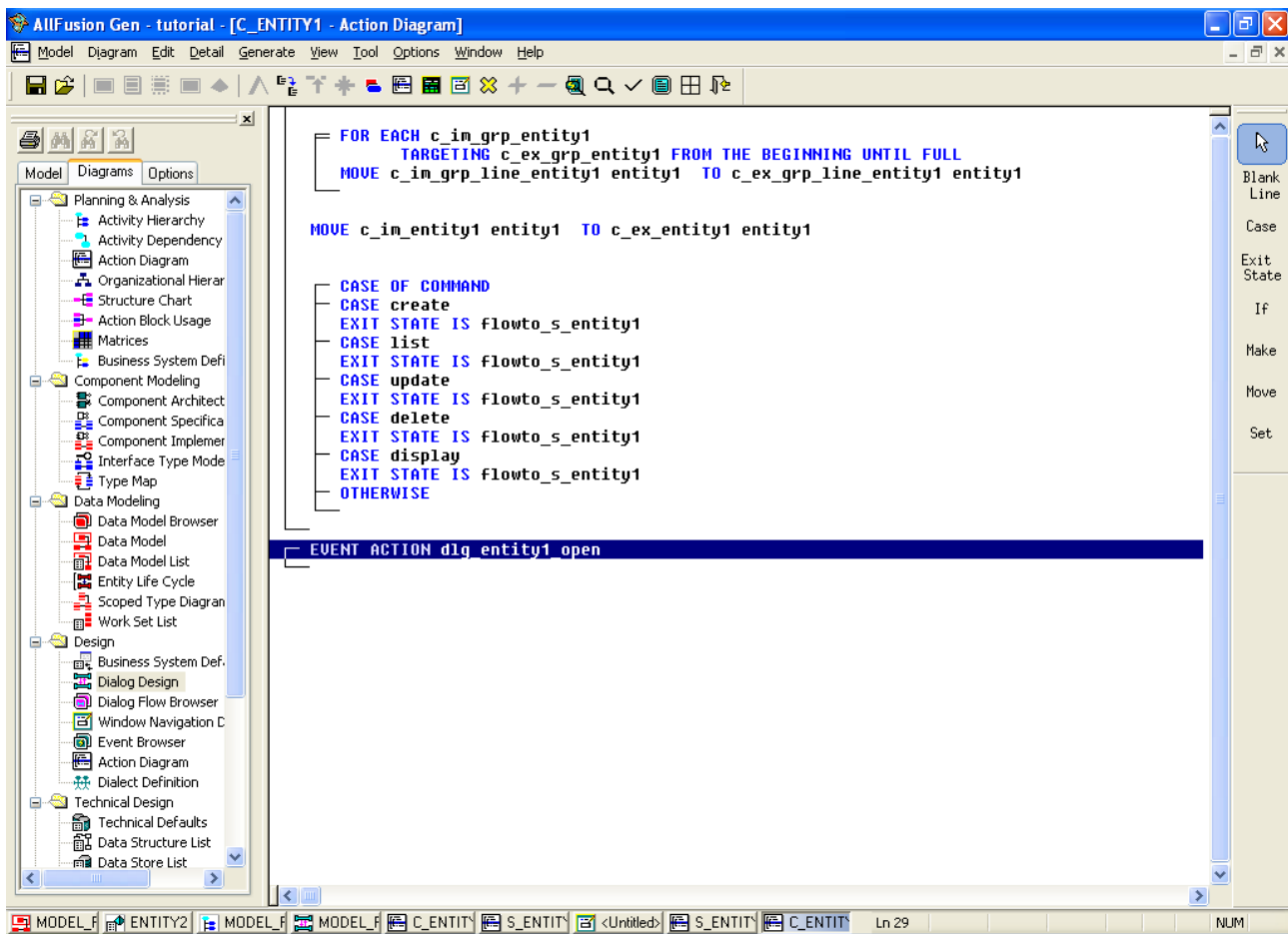
Action Name:

Hier sehen Sie eine (leere) Liste mit allen Events die für das von Ihnen ausgewählte Fensterelement gelten. Event Type beschreibt das Event, welches behandelt wird, Action Name ist der Name einer Unterfunktion im Action Diagram Ihres Client Dialogs. Über Command kann auch noch wie bei den Buttons ein Command gesetzt werden.

Wählen Sie bei Event Type nun "Open" aus. Sie können bei Action Name einen eigenen Namen vergeben. Tun Sie das nicht, wird einer generiert. Drücken Sie anschließend auf Add.



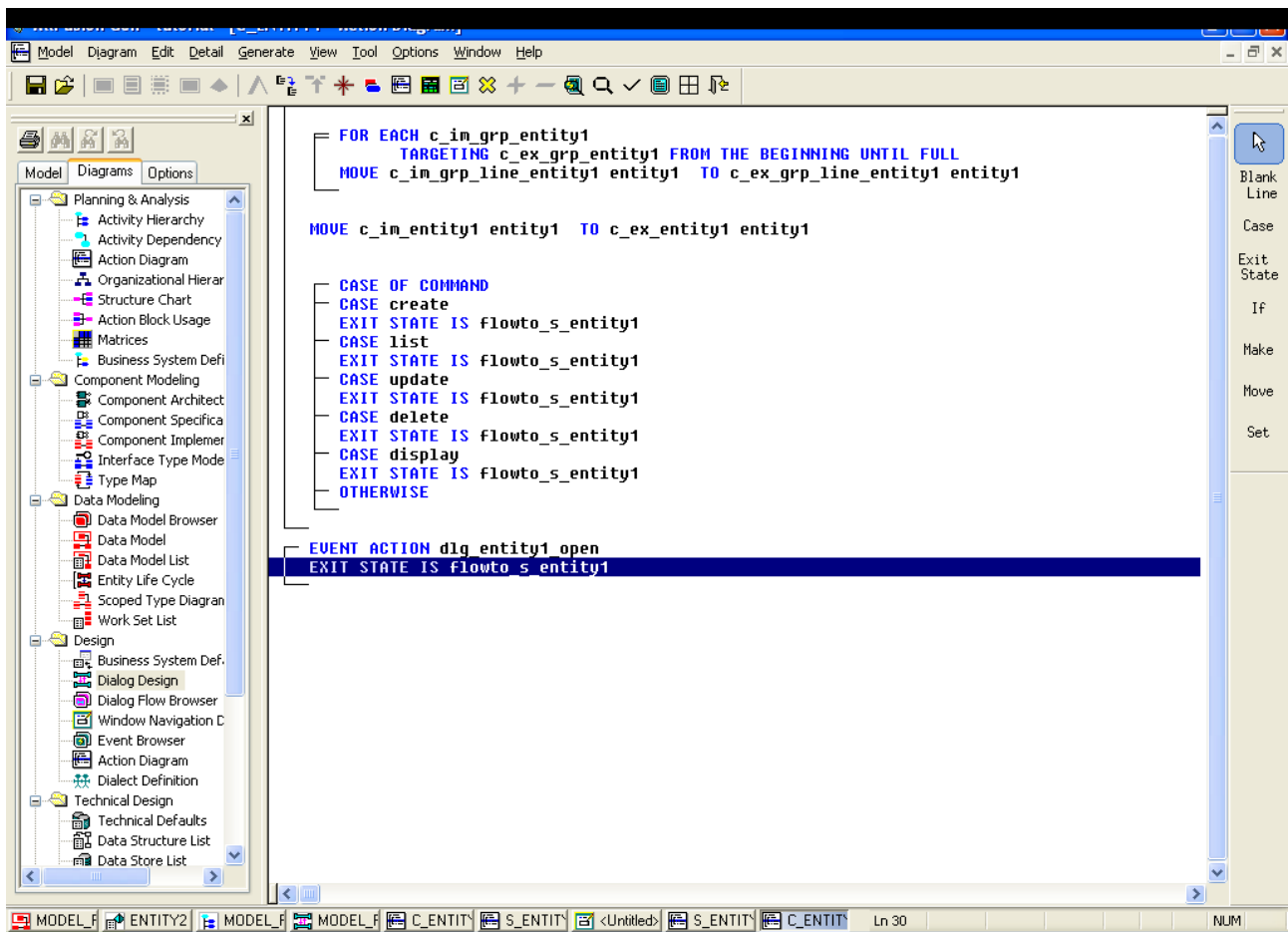
Sie haben erfolgreich ein Event erstellt. Wenn jetzt dieses Fenster geöffnet wird, wird die "Funktion" DLG\_ENTITY1\_OPEN ausgeführt. Was diese genau ist, können Sie sich über den Button "Action Diagram" anzeigen lassen.



Die "Event Actions" befinden sich unterhalb des normalen Action Blocks ihres Client Dialogs. Sie werden nur ausgeführt, wenn das zugehörige Event eintritt.

An dieser Stelle wollen Sie jetzt dafür sorgen, dass beim Start des Programms eine aktuelle Group View/Liste vom Server in die Listbox geschrieben wird. Dies lässt sich recht einfach realisieren:

Fügen Sie der Event Action ein "Exit State is" Statement mit dem Exit State, den Sie für den Flow vom Client zum Server benutzen, hinzu.

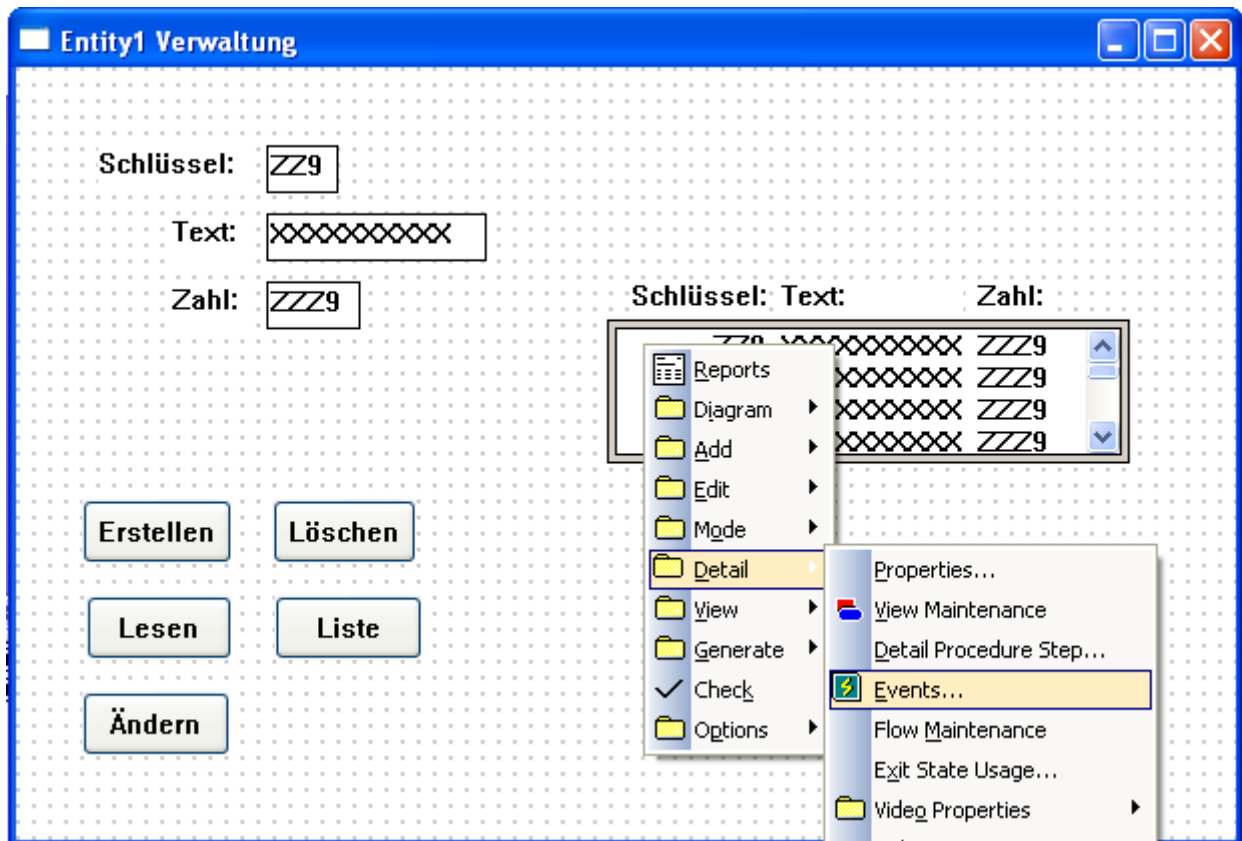


Nun wird beim Start des Programms sofort ein Flow zum Server aufgerufen, welcher eine Group View mit den aktuellen Daten der Datenbank zurückliefert, da der "List" Befehl ja nun stets im Server ausgeführt wird, wenn dieser aktiv wird.

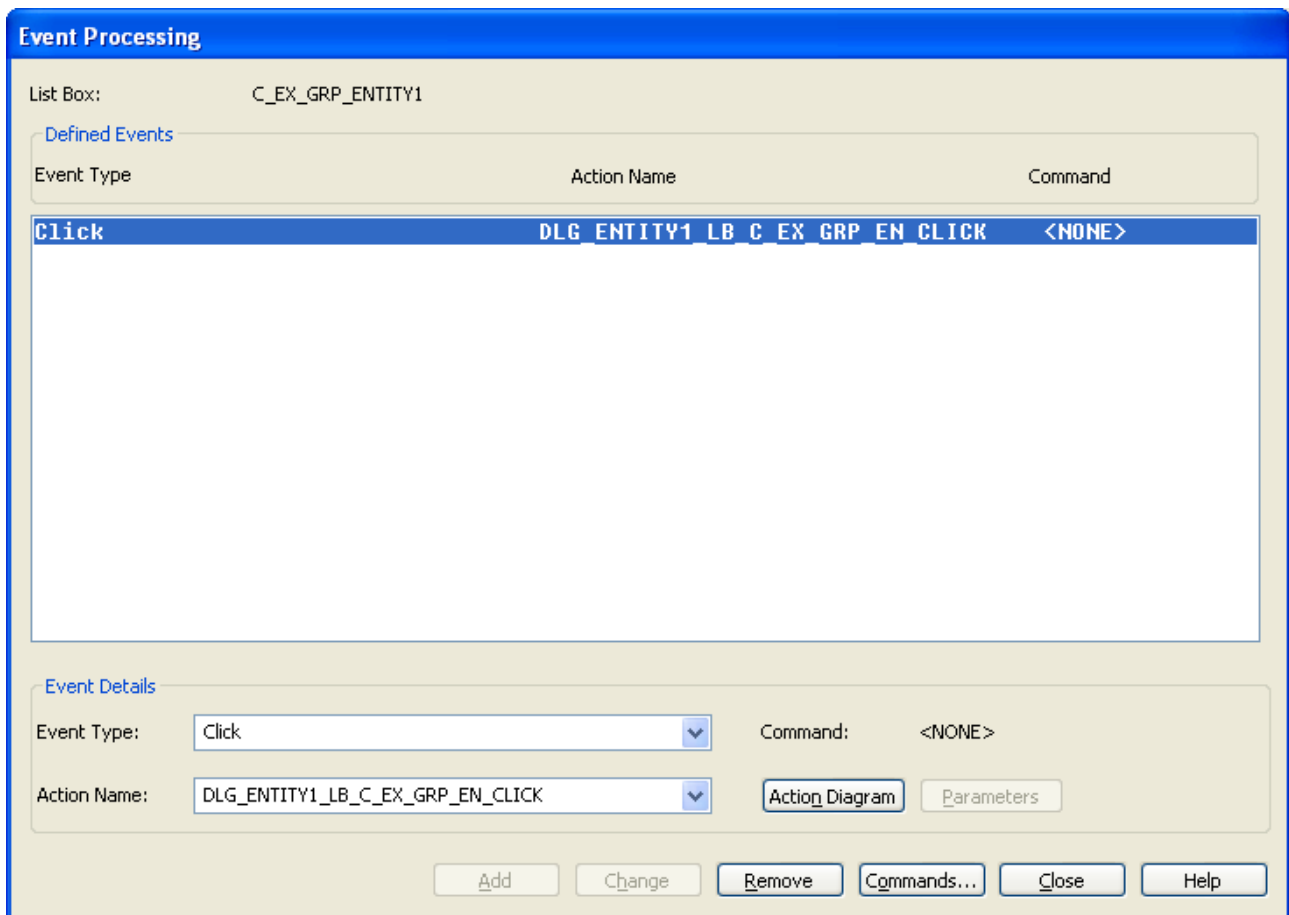
Probieren Sie die Änderung aus.



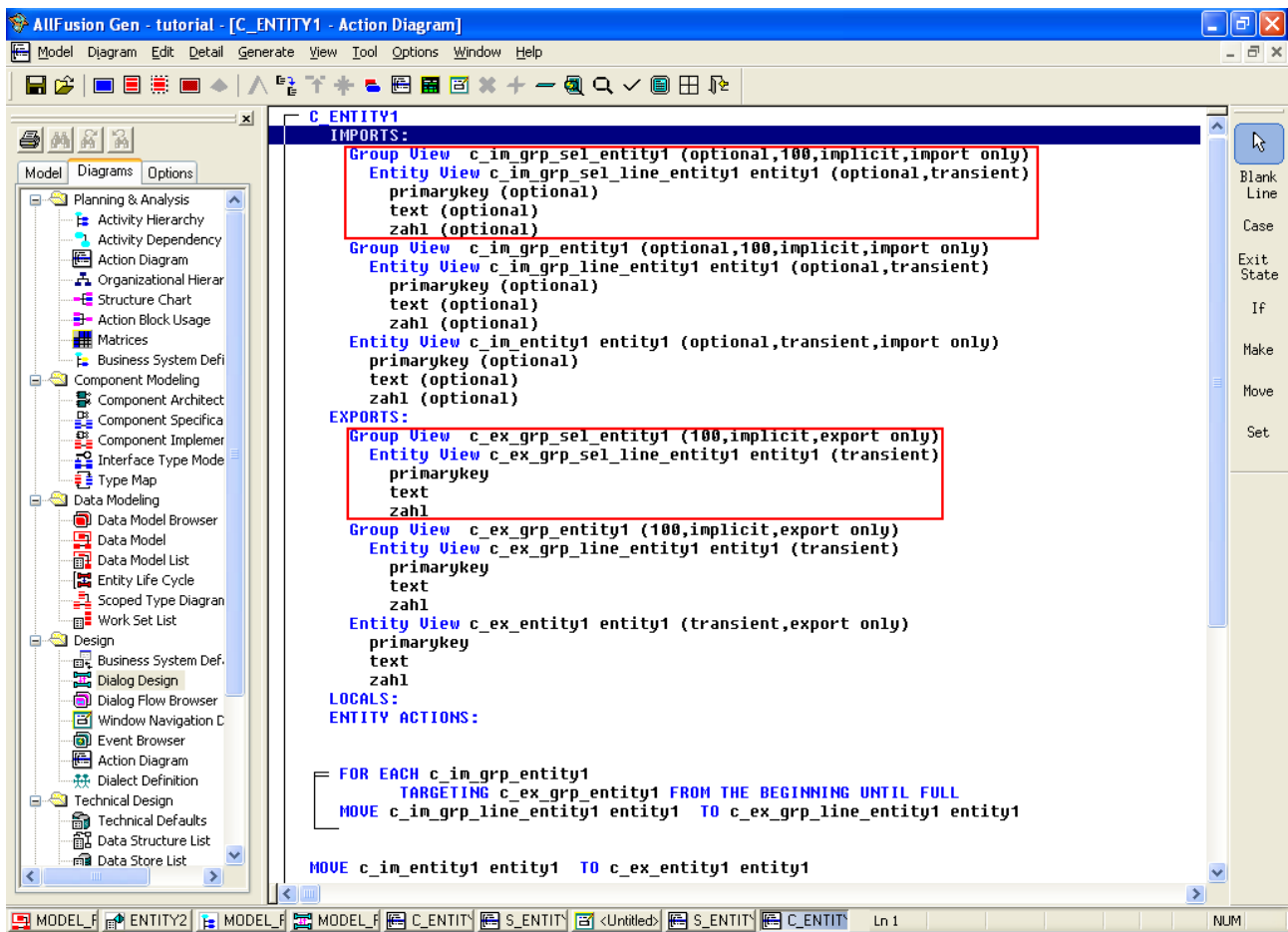
Als nächstes werden Sie ein komplexeres Event implementieren. Das Ziel: Durch einen Klick oder Doppelklick auf die Listbox soll der ausgewählte Eintrag in die Textboxen eingetragen werden, damit Sie zum Beispiel diesen Eintrag verändern können.



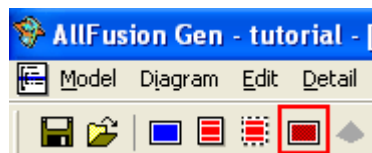
Wählen Sie zunächst die Listbox auf Ihrem Fenster aus und öffnen Sie die Eventverwaltung.



Legen Sie anschließend ein Event an für den Event Typ "Click" oder "DoubleClick". Öffnen Sie anschließend das Action Diagram Ihres Clients.



Damit Sie erkennen können, welche der Zeilen in der Listbox ausgewählt wurde, benötigen Sie ein zusätzliches Attribut in der Group View. Da dadurch jedoch Inkompatibilitäten mit dem Server entstehen würden, legen Sie besser zwei weitere Group Views an. Geben Sie ihnen einen Namen, der den Zweck erkennen lässt (hier das Zusatzprädikat "sel" für "selection").



Anschließend fügen Sie eine Work View (markierter Button) in beide Group Views hinzu. Dies sind bestimmte Views die vom System bereitgestellt werden und unter anderem einen "Selection Character" beinhalten, der für die Listenverarbeitung wichtig ist.

**Add Import Work View**

Name

is  used as input.

Supports entity actions (persistent)

Lock required on entry

Used as both input and output

Initialize on every entry

Generate attribute state flags

Entity Types

<b>wrk set</b>	<b>ASYNC_REQUEST</b>
<b>wrk set</b>	<b>DLG_ENTITY1</b>
<b>wrk set</b>	<b>DLG_ENTITY2</b>
<b>wrk set</b>	<b>IEF_SUPPLIED</b>

OK Back Search... Description... Cancel Help

Wählen Sie zunächst das Work Set "IEF\_SUPPLIED".

**Add Import Work View**

Name

is  used as input.

Supports entity actions (persistent)

Lock required on entry

Used as both input and output

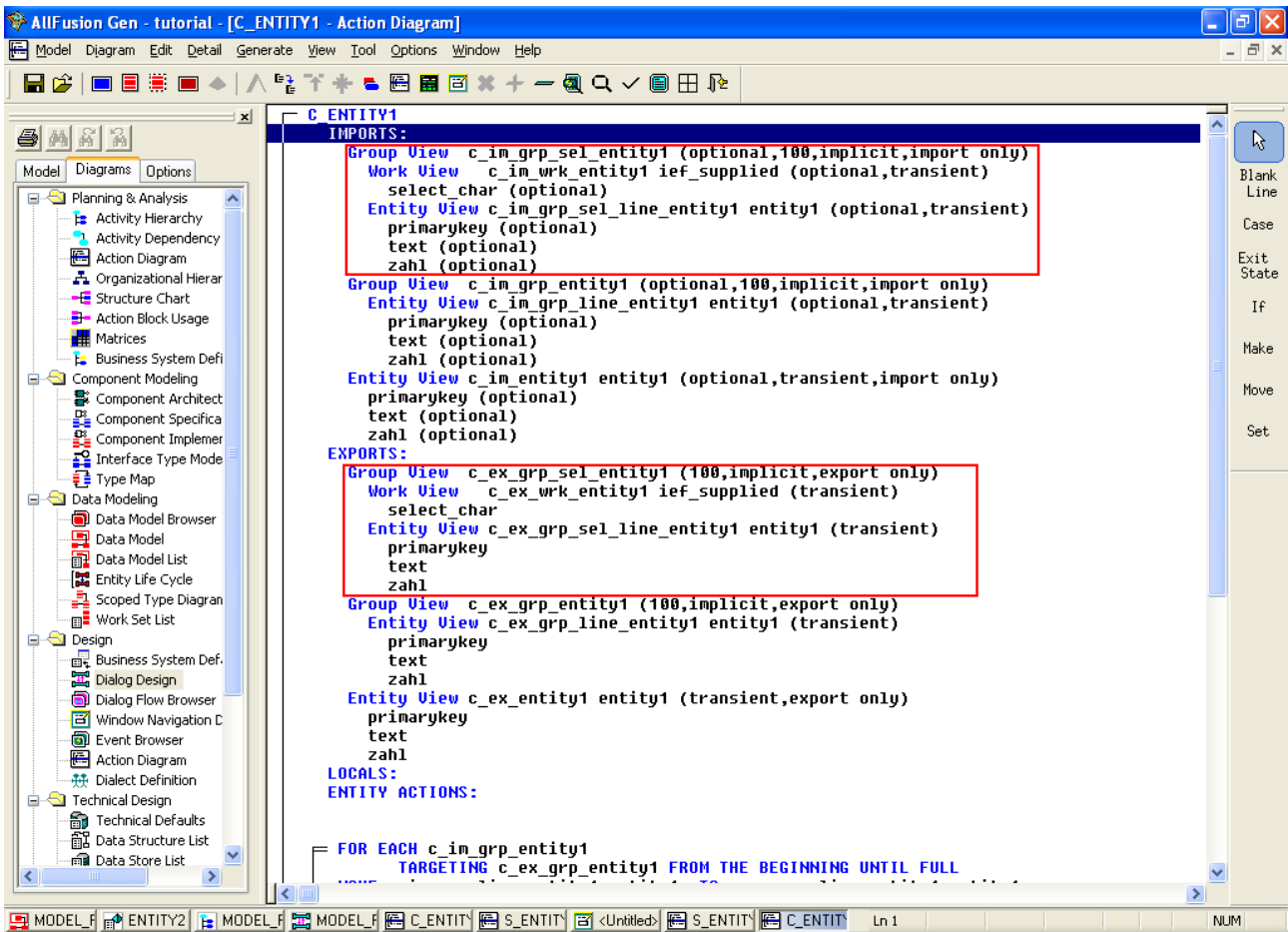
Initialize on every entry

Generate attribute state flags

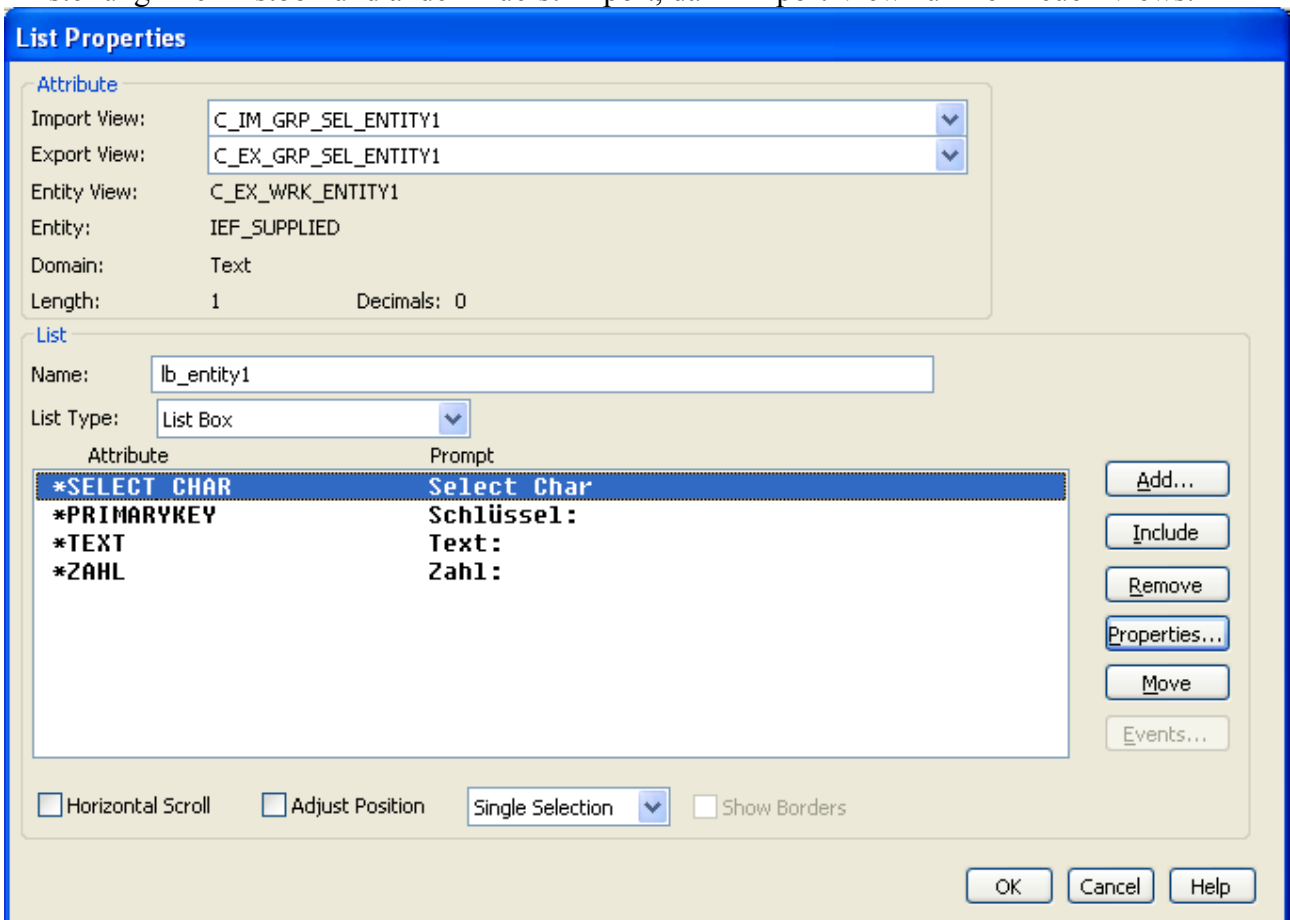
Work Attributes

wrk set	IEF SUPPLIED
attr	SELECT CHAR
attr	ACTION_ENTRY
attr	COMMAND
attr	COUNT
attr	TOTAL_REAL
attr	TOTAL_CURRENCY
attr	TOTAL_INTEGER
attr	PERCENTAGE
attr	AVERAGE_REAL
attr	AVERAGE_CURRENCY
attr	AVERAGE_INTEGER
attr	FLAG
attr	SUBSCRIPT

Anschließend wählen Sie das Attribut "Select\_Char" aus.



Ihre neuen Views sollten nun in etwa wie hier markiert aussehen. Als nächstes öffnen Sie die Einstellung ihrer Listbox und ändern zuerst Export, dann Import View zu ihren neuen Views.



Nun ist der Selection Character auch in der Liste der Attribute zu sehen. Wählen Sie diesen nun an und klicken Sie auf "Properties".

**List Box Field Properties**

**Attribute**

Import View: (none)  
Export View: C\_EX\_WRK\_ENTITY1  
Entity: IEF\_SUPPLIED  
Name: SELECT\_CHAR  
Domain: Text  
Length: 1      Decimals: 0

**Field**

Prompt: Select Char  
Prompt Align: Left  
Edit Pattern: <Default Edit Pattern>  
Edit Align: Default

Selection Indicator

OK    Cancel    Help

Setzen Sie dann das Häkchen bei "Selection Indicator".

**List Properties**

**Attribute**

Import View: C\_IM\_GRP\_SEL\_ENTITY1  
Export View: C\_EX\_GRP\_SEL\_ENTITY1  
Entity View:  
Entity:  
Domain:  
Length: 0      Decimals: 0

**List**

Name: lb\_entity1  
List Type: List Box

Attribute	Prompt
<b>dSELECT CHAR</b>	<b>Select Char</b>
<b>*PRIMARYKEY</b>	<b>Schlüssel:</b>
<b>*TEXT</b>	<b>Text:</b>
<b>*ZAHL</b>	<b>Zahl:</b>

Horizontal Scroll     Adjust Position     Single Selection     Show Borders

Add...  
Include  
Remove  
Properties...  
Move  
Events...

OK    Cancel    Help

Anschließend können Sie den SELECT\_CHAR aus der Liste der anzuzeigenden Attribute entfernen (Remove). Es sollte nun ein "d" vor dem Namen stehen (hidden).

Nun muss die Group View noch befüllt werden. Legen Sie ein neues "For each" an, das auch die normale Import Group View benutzt, aber diesmal in die Selection Group View kopiert.

```
FOR EACH c_im_grp_entity1
  TARGETING c_ex_grp_sel_entity1 FROM THE BEGINNING UNTIL FULL
  MOVE c_im_grp_line_entity1 entity1 TO c_ex_grp_sel_line_entity1 entity1

FOR EACH c_im_grp_entity1
  TARGETING c_ex_grp_entity1 FROM THE BEGINNING UNTIL FULL
  MOVE c_im_grp_line_entity1 entity1 TO c_ex_grp_line_entity1 entity1

MOVE c_im_entity1 entity1 TO c_ex_entity1 entity1

CASE OF COMMAND
```

Alternativ können Sie beide For Each Schleifen jetzt auch zusammenlegen.

```
FOR EACH c_im_grp_entity1
  TARGETING c_ex_grp_sel_entity1 FROM THE BEGINNING UNTIL FULL
  AND TARGETING c_ex_grp_entity1 FROM THE BEGINNING UNTIL FULL
  MOVE c_im_grp_line_entity1 entity1 TO c_ex_grp_sel_line_entity1 entity1
  MOVE c_im_grp_line_entity1 entity1 TO c_ex_grp_line_entity1 entity1

MOVE c_im_entity1 entity1 TO c_ex_entity1 entity1

CASE OF COMMAND
```

Sie können jetzt schon ausprobieren, ob das Programm läuft. Es sollte äußerlich keinerlei Unterschied festzustellen sein.

Nun jedoch zum Event. Gehen Sie zur Event Action der Listbox, indem Sie einfach im Client Action Diagram runterscrollen.

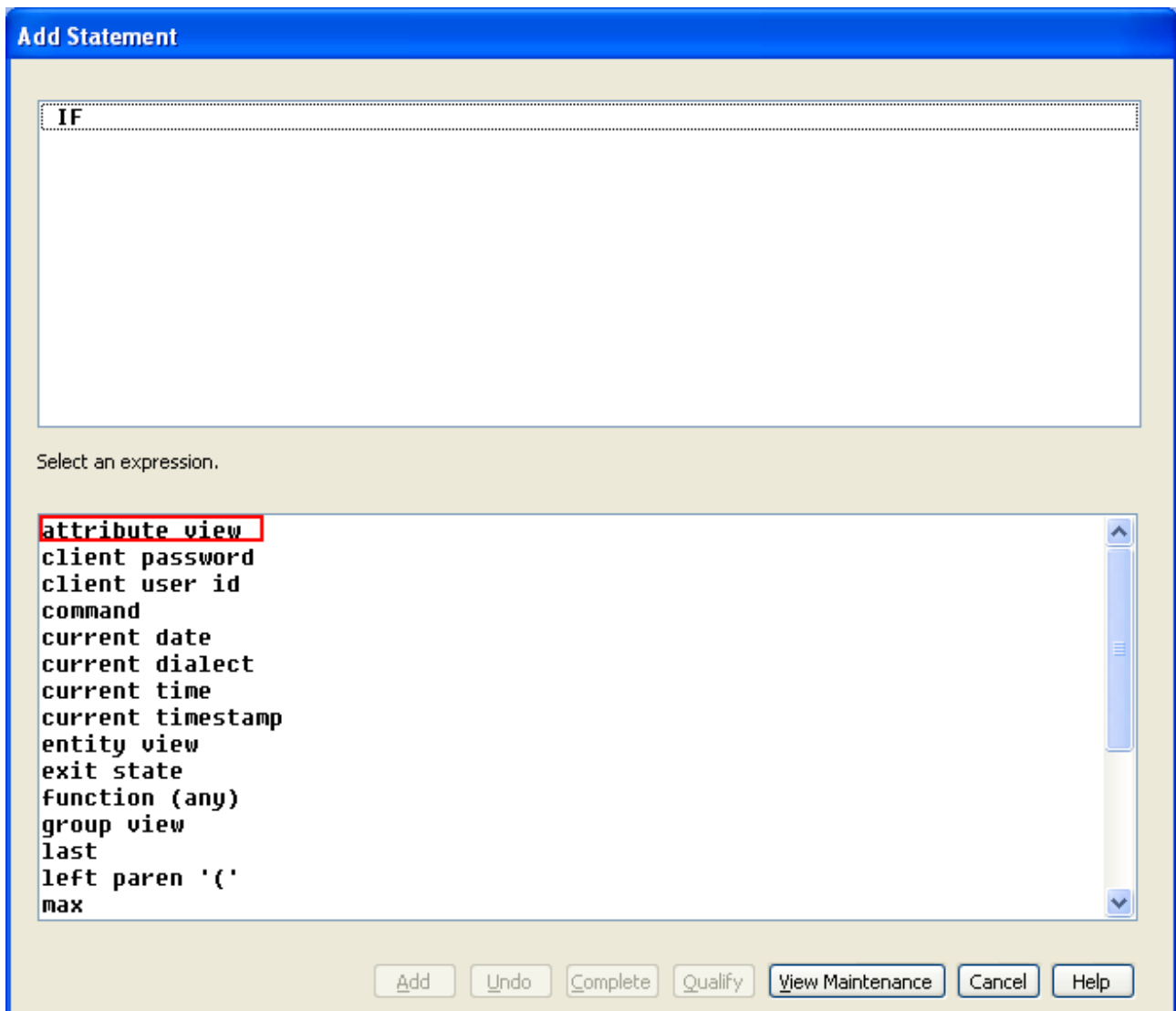
Zunächst brauchen Sie eine "For Each"-Schleife **ohne** "Targeting", da dies nur ein lesender Zugriff auf die Liste wird.

```
EVENT ACTION dlg_entity1_lb_c_ex_grp_en_click
  FOR EACH c_im_grp_sel_entity1
```

Es wird nun jedes Element der Group View durchlaufen. Um zu überprüfen, welche der Zeilen ausgewählt wurde, müssen Sie nur die Zeile finden, in welcher der Selection Char **nicht** leer ist. Die selektierte Zeile wird an dieser Stelle nämlich ein Symbol zur Markierung haben.

Fügen Sie nun ein If-Statement in die "For Each"-Schleife hinzu.





Wählen Sie nun "attribute view" aus. (Sie wollen ein Attribut untersuchen).

**Add Statement**

IF

Select an entity view.

```
c_ex_entity1 entity1  
c_im_entity1 entity1  
c_im_qrp_sel_line entity1 entity1  
c_im_wrk_entity1 ief_supplied
```

Add Undo Complete Qualify View Maintenance Cancel Help

Nun wählen Sie die View aus, welche Sie überprüfen wollen. Dies sollte die Work View sein.

**Add Statement**

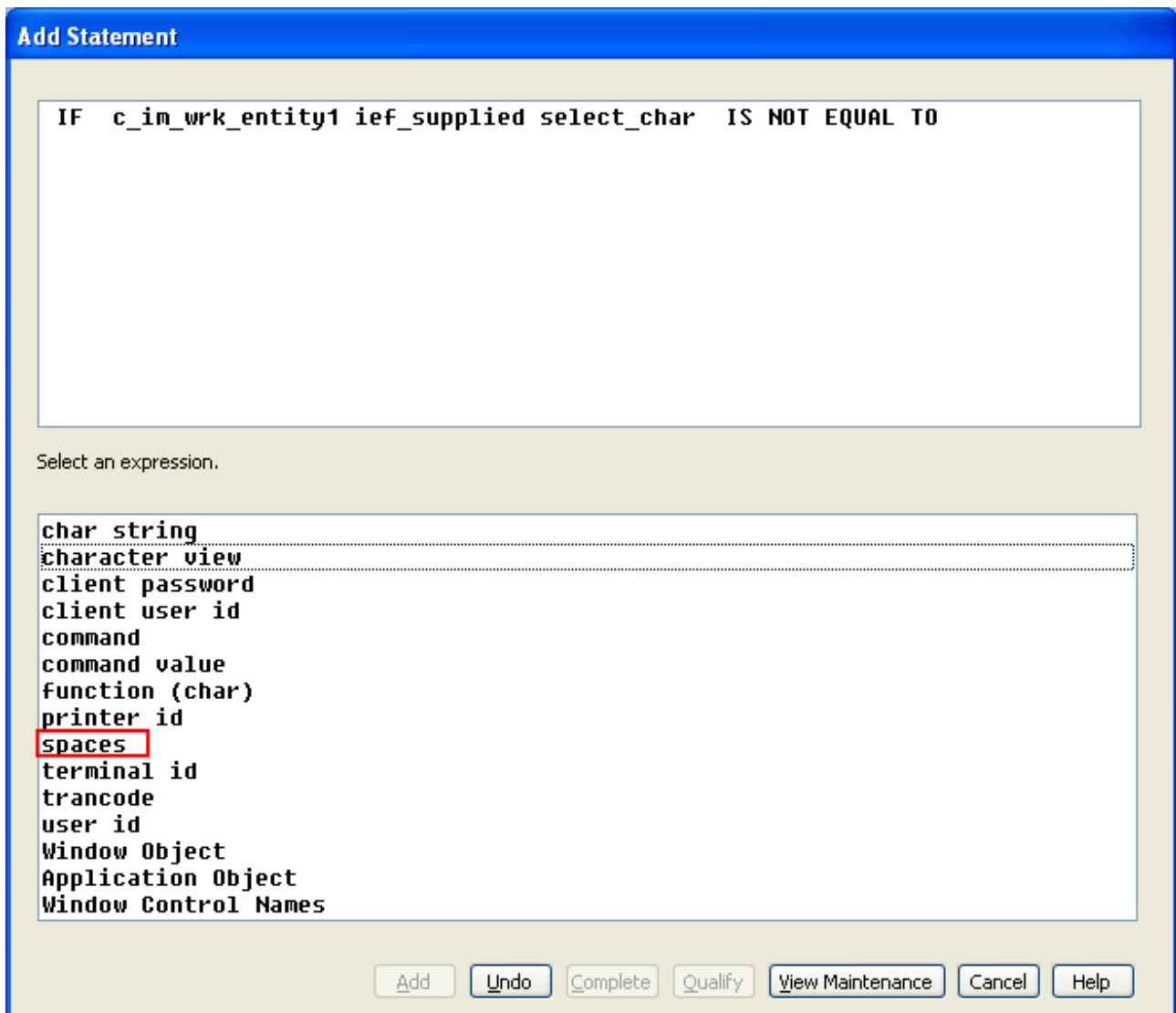
```
IF c_im_wrk_entity1 ief_supplied select_char
```

Select a relational operator.

- IS EQUAL TO
- IS NOT EQUAL TO**
- IS EQUIVALENT TO
- IS NOT EQUIVALENT TO
- IS GREATER THAN
- IS LESS THAN
- IS GREATER OR EQUAL TO
- IS LESS OR EQUAL TO
- IS VALID

Add Undo Complete Qualify View Maintenance Cancel Help

Als nächstes kommt ein Vergleichsoperator. Da Sie überprüfen wollen, welcher "select\_char" **nicht** leer ist, wählen Sie "is not equal to".



Zum Schluss wählen Sie als zweiten Ausdruck der Vergleichsoperation "spaces".

Diese If-Bedingung ist dann wahr, wenn select\_char ungleich einem leeren String ist.

Drücken Sie dann auf "Add".

```

EVENT ACTION dlg_entity1_lb_c_ex_grp_en click
├── FOR EACH c_im_grp_sel_entity1
│   └── IF c_im_wrk_entity1 ief_supplied select_char IS NOT EQUAL TO SPACES
└──

```

Wenn nun diese Bedingung ausgelöst wird, soll der Eintrag, der an dieser Stelle in der Group View liegt, in die Textboxen eingetragen werden, bzw. in die Entity Views, welche die Textboxen steuern.

```
EVENT ACTION dlg_entity1 lb c_ex_grp_en click
FOR EACH c_im_grp_sel_entity1
  IF c_im_wrk_entity1 ief_supplied select_char IS NOT EQUAL TO SPACES
  MOVE c_im_grp_sel_line_entity1 entity1 TO c_ex_entity1 entity1
```

Fertig. Sie könnten theoretisch hinter das Move ein "Exit" setzen, um die Schleife nicht unnötig weiter zu durchlaufen.

Probieren Sie Ihr Programm nun aus.