

Datenaustausch zwischen UIBBs

Nachzusehen im Paket /NVIAS/ST_ADMIN_WD in den Klassen:

- /NVIAS/CL_AD_ADH_SEARCH_F
- /NVIAS/CL_AD_ADH_SEARCH_LIST_F
- /NVIAS/CL_AD_ADH_SEARCH_UI_M (Das „M“ am Schluss steht immer für Model)

Für den Datenaustausch zwischen UIBBs wird ein Model benötigt.

Wichtig hierbei ist:

Es wird eine Klassenvariable (static) GO_INSTANCE benötigt, damit immer nur eine Instanz dieser Klasse existieren kann, die sich selbst als Bezugstyp referenziert.

Also z.B. hat die Klasse ZCL_Testklasse_M ein Static-Attribut GO_INSTANCE Type Ref To ZCL_Testklasse_M.

Außerdem muss hier eine Klassenmethode GET_INSTANCE mit einem Returning Parameter implementiert werden.

Beispiel:

```
*      IF go_instance IS INITIAL.  
*          CREATE OBJECT go_instance.  
*      ENDIF.  
*      ro_instance = go_instance.
```

Dadurch wird sichergestellt, dass zur Laufzeit genau ein Objekt dieser Klasse existiert.

Außerdem werden hier die Attribute gesetzt, die übergeben werden sollen.

UIBB1 (z.B. eine Suchmaske):

In den Methoden wird eine Methode „GET_MODEL“ benötigt.

Diese Methode hat einen Rückgabeparameter mit der Typisierung „Type Ref To“ mit Bezug zur Klasse des Models.

```
METHOD get_model.  
  
    ro_model = /nvias/cl_ad_adh_search_ui_m=>get_instance( ).
```

Man kann nun z.B. durch eine Setter-Methode die benötigten Daten dem Model übergeben.

/

UIBB2.

Auch hier wird wieder die Methode „GET_MODEL“ aufgerufen.

Die benötigten Daten können hier mittels einer Getter-Methode abgeholt werden (In der Get_Data-Methode).

Wichtig hierbei ist, dass man den Parameter `ev_data_changed = abap_true` setzt.
Sonst würden bei einer Liste z.B. die neuen Daten nicht angezeigt werden.