Web Dynpro ABAP: ALV and Table in Popup Window

 Applies to:
 SAP ECC 6.0

 Summary
 Normally in ABAP, we come across the scenario of displaying ALV in popup. This article tells about displaying data both in ALV and table format in popup window using Webdynpro ABAP.

 Author: J.Jayanthi
 Company: Siemens IT Solutions and Services Pvt. Ltd.
 Created on: 11 October 2010

 Author Bio
 J.Jayanthi is a Certified ABAP consultant with HR ABAP knowledge.
Table of Contents

ALV in Web Dynpro ABAP........................................................................................................3
Prerequisites ................................................................................................................................3
Creating Web Dynpro .............................................................................................................4
  Component Controller ........................................................................................................4
  Designing View ..................................................................................................................7
  Embedding View ...............................................................................................................17
  Component Usages ............................................................................................................19
Creating Web Dynpro Application .......................................................................................20
  Code ....................................................................................................................................20
  Output ...............................................................................................................................22
Related Content ...................................................................................................................23
Disclaimer and Liability Notice ..............................................................................................24
**ALV in Web Dynpro ABAP**

ABAP Consultants are generally familiar in SAP List Viewer (ALV) display in popup.

We are going to see how to design popup window with ALV and Table display format. In this article, Main window contains the input field of Personnel Number with two buttons named Address and Details. Address button will display the address of the employee in Table format in popup window. Details button will display the details of the employee as ALV in Popup window.

**Prerequisites**

**Component**
The component is the central, reusable unit of the application project. You can create any number of views in a component and arrange them in any number of windows.

**Component Usages**
Web Dynpro components can be nested. This means that you can integrate any number of other, already existing components into a component.

**View**
The view is the smallest unit of a Web Dynpro application visible for the user. The layout elements and dialog elements - for example, tables, text fields, or buttons - required for the application are arranged in a view. The view contains a controller and a controller context in which the application data to be processed is stored in a hierarchical structure. This allows the linking of the graphical elements with the application data.

**Window**
A window is used to group multiple views and to specify the navigation between the views. A view can only be displayed by the browser if the view is embedded in a window.
Creating Web Dynpro

Go to SE80 and select Web Dynpro Comp./Intf. and provide the name (say ZZZ_JAYTEST7) to create. Then enter the description and choose the type as Web Dynpro Component.

Mention the Component Use as POPUP1 and Component as SALV_WD_TABLE in the Used Components tab in Web Dynpro (ZZZ_JAYTEST7).

This will create a Component Usages by name POPUP1.

Component Controller

Go to Component Controller and Right click the context. Then select Create Node.
Then select the fields required from the structure using Add Attribute from Structure button. Then Double click the node and then remove the dictionary structure, change the cardinality to 0:n. Uncheck the Initialization Lead selection.

Similarly create another node Details using PA0002 table and add attributes from structure and then delete the dictionary structure after that.
Create Attribute Pernr with properties as below.

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attribute Name</td>
<td>PERNR</td>
</tr>
<tr>
<td>Type assignment</td>
<td>Type</td>
</tr>
<tr>
<td>Type</td>
<td>PA0008-PERNR</td>
</tr>
<tr>
<td>Read-only</td>
<td></td>
</tr>
<tr>
<td>Default Value</td>
<td></td>
</tr>
<tr>
<td>Null Value</td>
<td></td>
</tr>
<tr>
<td>Input Help Mode</td>
<td>Automatic</td>
</tr>
<tr>
<td>Determined Input Help</td>
<td>PA0003</td>
</tr>
<tr>
<td>Type of Input Help</td>
<td>Check table</td>
</tr>
<tr>
<td>Formatting</td>
<td>Display Leading Zeros Default Value</td>
</tr>
</tbody>
</table>
Designing View

We are going to design three views (Main, POPUP, POPUP1). POPUP will be used for Table display and POPUP1 will be used for ALV display.

Go to the properties tab of Main view and then create as below.

In context tab, drag and drop the context which appears under component controller to view (Main is the view name).

The symbol in left side shows it is mapped.

Then in layout, set the properties for ROOTUIELEMENTCONTAINER as follows. Go to the layout in view and right click the ROOTUIELEMENTCONTAINER and then choose Insert element. Create a Label with ID Employee and then InputField with ID Pernr.
Choose the yellow icon in the right side of the input field PERNR properties and choose the value as above. Then create two buttons Address and Details with events ONCLICK for Address and ONDETAILSCLICK for Details.
After that, layout will appear as below.

Select the method ONACTIONONCLICK (For Address Button) in methods tab.

Use Web Dynpro code wizard to generate code automatically.

**Step a:**
Choose the radio button Read context for the attribute PERNR.

Keep the below generated code and delete the rest which is not required.
DATA lo_el_context TYPE REF TO if_wd_context_element.
DATA ls_context TYPE wd_this->element_context.
DATA lv_perrr LIKE ls_context-perrr.

Step b:
Again use Code Wizard as below for Address Node.

This will generate the code. Keep the below required code.

DATA lo_nd_address TYPE REF TO if_wd_context_node.
* Navigate from <CONTEXT> to <ADDRESS> via lead selection
lo_nd_address = wd_context->get_child_node( name = wd_this->wdctx_address ).

Step c:
To generate POPUP for Address details, do as below.

* Generate POPUP for Address Display
DATA lo_window_manager TYPE REF TO if_wd_window_manager.
DATA lo_api_component TYPE REF TO if_wd_component.
DATA lo_window TYPE REF TO if_wd_window.

lo_api_component = wd_comp_controller->wd_get_api().
lo_window_manager = lo_api_component->get_window_manager().
lo_window = lo_window_manager->create_window(
  window_name = 'POP',
  title = 'Address Details',
  close_in_any_case = abap_true,
  close_button = abap_true).

lo_window->open().

Step d:
Then select the data by normal ABAP statement (declare t_0006 as required) and bind the table to the address node.
Repeat the steps from a to d for Details button also in the event.

For the step c to generate popup ALV window for Details, do like below.

1. Generate Popup
   - Component Use: POPUP1
   - Window Name: TABLE

This will generate the code and change it as below.

```abap
DATA lo_window_manager TYPE REF TO if_wd_window_manager.
DATA lo_api_component TYPE REF TO if_wd_component.
DATA lo_window TYPE REF TO if_wd_window.

lo_api_component = wd_comp_controller->wd_get_api().
lo_window_manager = lo_api_component->get_window_manager().
lo_window = lo_window_manager->create_window_for_cmp_usage(
  interface_view_name = 'TABLE',
  component_usage_name = 'POPUP',
  title = 'Employee Details',
  close_in_any_case = absap_true,
  message_display_mode =
  if_wd_window->co_msg_display_mode_selected
).
lo_window->open().
```

Create another view say POPUP.

Design the properties as below.

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>popup</td>
</tr>
<tr>
<td>Created By</td>
<td>Jayaram K J</td>
</tr>
<tr>
<td>Created On</td>
<td>08.10.2010</td>
</tr>
<tr>
<td>Last Changed By</td>
<td>Jayaram K J</td>
</tr>
<tr>
<td>Changed On</td>
<td>11.10.2010</td>
</tr>
</tbody>
</table>

In context tab, drag and drop the context Address which appears under component controller to view (POPUP is the view name).
Go to Layout of this view and create a table by using Insert element in ROOTUIELEMENTCONTAINER.

For every column, in the text editor, select the value by choosing the yellow color icon in the right side as required.

Property | Value | Binding
---|---|---
ID | ADDRESS | 
Typ | Table | 
alignment | auto | 
enabled | | 
explanation | | 
length | 0 | 
passwordField | | 
readOnly | | 
state | Normal Item | 
textDirection | inherit | 
tooltip | | 
value | POPUP.ADDRESS.PERNR | 

© 2010 SAP AG
Similarly do for other columns.

We can change the text for each column as below.

Then right click the table and create binding.
Create another view POPUP1 for ALV display of employee details.
Design the properties of the view POPUP1 as below.
Do the mapping in the context by drag and drop the Details Node.

In layout, create ViewContainerUIElement by using Insert Element on right clicking the ROOTUIELEMENTCONTAINER.
**Embedding View**

Create two windows in addition to the default (ZZZ_JAYTEST7) window.

In the default window (ZZZ_JAYTEST7), embed Main view as below.

<table>
<thead>
<tr>
<th>Window</th>
<th>ZZZ_JAYTEST7</th>
<th>Active</th>
</tr>
</thead>
<tbody>
<tr>
<td>Properties</td>
<td>Window</td>
<td>Inbound Plugs</td>
</tr>
</tbody>
</table>

**Window-Struktur**

- ZZZ_JAYTEST7
  - MAIN
  - DEFAULT

In the window POP, embed table as below.

![Web Dynpro: Embed View](image)

- Component: ZZZ_JAYTEST7
- Window: POP
- View to Be Embedded: POPUP
- Component of View: ZZZ_JAYTEST7
- Component Use: ----

In the window POP1, embed the ALV as below.

![Web Dynpro: Embed View](image)

- Window: POP
  - Properties | Window | Inbound Plugs | Outbound Plugs | Context | Attributes |
  - Window-Struktur
    - POP
      - POPUP
      - DEFAULT
<table>
<thead>
<tr>
<th>Component</th>
<th>ZZZ_JAYTEST7</th>
</tr>
</thead>
<tbody>
<tr>
<td>Window</td>
<td>POP1</td>
</tr>
<tr>
<td>View to Be Embedded</td>
<td>TABLE</td>
</tr>
<tr>
<td>Component of View</td>
<td>SALV_WD_TABLE</td>
</tr>
<tr>
<td>Component Use</td>
<td>POPUP1</td>
</tr>
</tbody>
</table>
Component Usages

Properties of Component usage should be as below.

Right click the component Usage (here with name POPUP1) and Create controller Usage.

Drag and drop the node (Details in right side) from Component Controller context to Data (in left side) in Controller Usage Context.
Creating Web Dynpro Application

Create Web Dynpro Application by right clicking the Webdynpro (ZZZ_JAYTEST7).

Right click the Web Dynpro component and activate.

Code

METHOD onactiononclick.
  
  * To get the value from input field Pernr
  DATA lo_el_context TYPE REF TO if_wd_context_element.
  DATA ls_context TYPE wd_this->element_context.
  DATA lv_pernr LIKE ls_context-perrr.
  * get element via lead selection
  lo_el_context = wd_context->get_element( ).
  * get single attribute
  lo_el_context->get_attribute( 
    EXPORTING
    name = 'PERNR'
    IMPORTING
    value = lv_pernr ).

  DATA lo_nd_address TYPE REF TO if_wd_context_node.
  * Navigate from <CONTEXT> to <ADDRESS> via lead selection
  lo_nd_address = wd_context->get_child_node( name = 
    wd_this->wdctx_address ).

  TYPES : BEGIN OF ty_0006,
    pernr TYPE pa0006-perrn,
    subty TYPE pa0006-subty,
    stras TYPE pa0006-stras,
    ort01 TYPE pa0006-ort01,
    pstlz TYPE pa0006-pstlz,
  END OF ty_0006.
  DATA t_0006 TYPE STANDARD TABLE OF ty_0006.

  SELECT pernr subty stras ort01 pstlz FROM pa0006 INTO TABLE t_0006
  WHERE pernr = lv_pernr.

  * Binding table to the node
  lo_nd_address->bind_table( t_0006 ).

  * Generate POPUP for Address Display
  DATA lo_window_manager TYPE REF TO if_wd_window_manager.
  DATA lo_api_component TYPE REF TO if_wd_component.
  DATA lo_window TYPE REF TO if_wd_window.

  lo_api_component = wd_comp_controller->wd_get_api( ).
  lo_window_manager = lo_api_component->get_window_manager( ).
  lo_window = lo_window_manager->create_window( 
    window_name = 'POP',
    title = 'Address Details',
    close_in_any_case = abap_true,
    close_button = abap_true ).

  lo_window->open( ).

ENDMETHOD.
method ONACTIONONDETAILSCLICK.

  * Read the input from the attribute Pernr
    DATA lo_el_context TYPE REF TO if_wd_context_element.
    DATA ls_context TYPE wd_this->element_context.
    DATA lv_pernr LIKE ls_context->pernr.
  * get element via lead selection
    lo_el_context = wd_context->get_element( ).
  * get single attribute
    lo_el_context->get_attribute( EXPORTING
      name = 'PERNR'
      IMPORTING
        value = lv_pernr ).
  * navigate from <CONTEXT> to <DETAILS> via lead selection
    lo_nd_details = wd_context->get_child_node( name = wd_this->#dcxt_details ).
  DATA lo_window_manager TYPE REF TO if_wd_window_manager.
  DATA lo_api_component TYPE REF TO if_wd_component.
  DATA lo_window TYPE REF TO if_wd_window.

  lo_api_component = wd_comp_controller->wd_get_api( ).
  lo_window_manager = lo_api_component->get_window_manager( ).
  lo_window = lo_window_manager->create_window_for_comp_usage( interface_view_name = 'TABLE'
    component_usage_name = 'POPUP1'
    title = 'Employee Details'
    close_in_any_case = abap_true
    message_display_mode =
    if_wd_window->co_msg_display_mode isSelected
  ).
  lo_window->open( ).

  types : begin of ty_0002,
    pernr type pa0002-pernr,
    begda type pa0002-begda,
    endda type pa0002-endda,
    nachn type pa0002-nachn,
    vorna type pa0002-vorna,
    midnm type pa0002-midnm,
    cname type pa0002-cname,
  end of ty_0002.

  data t_0002 type standard table of ty_0002.

  select pernr begda endda nachn
    vorna midnm cname from pa0002 into table t_0002
  where pernr = lv_pernr.

  lo_nd_details->bind_table( t_0002 ).

endmethod.
Output

When Address button is clicked, popup will appear as below.

![Address Details](image1)

When Details button is clicked, Popup will appear as below.

![Employee Details](image2)
Related Content

Web Dynpro: Coloring Table Conditionally
Web Dynpro: Column Coloring in ALV
Disclaimer and Liability Notice

This document may discuss sample coding or other information that does not include SAP official interfaces and therefore is not supported by SAP. Changes made based on this information are not supported and can be overwritten during an upgrade.

SAP will not be held liable for any damages caused by using or misusing the information, code or methods suggested in this document, and anyone using these methods does so at his/her own risk.

SAP offers no guarantees and assumes no responsibility or liability of any type with respect to the content of this technical article or code sample, including any liability resulting from incompatibility between the content within this document and the materials and services offered by SAP. You agree that you will not hold, or seek to hold, SAP responsible or liable with respect to the content of this document.