

Hiding Lines between Rows and Columns in an ALV



Applies to:

Webdynpro using ABAP. For more information, visit the [Web Dynpro ABAP homepage](#).

Summary

Displaying the output of an ALV in the form of cells is a basic thing, but the feature of hiding the lines between the rows and columns of an ALV is always handy.

Author: Arfathulla S.M.

Company: Infosys Technologies Limited

Created on: 19 February 2010

Author Bio

Arfathulla is working as ABAP technical consultant in Infosys technologies since 18 months and worked on the projects of webdynpro abap.

Table of Contents

Steps to create a component and displaying the output in alv with out lines.	3
Related Content	10
Disclaimer and Liability Notice.....	11

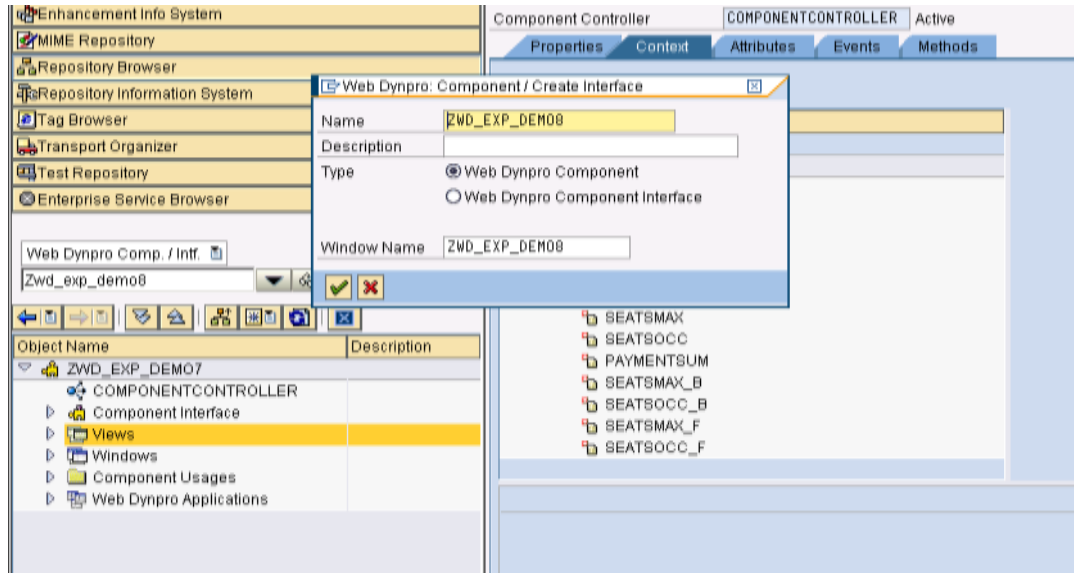
Pre-Requisites

In this article iam not going to explain the basic things .we have already seen how to edit an alv and colouring the cells of an alv.But, in this article iam explaining how to hide the lines between the rows and columns of an alv table in webdynpro abap.

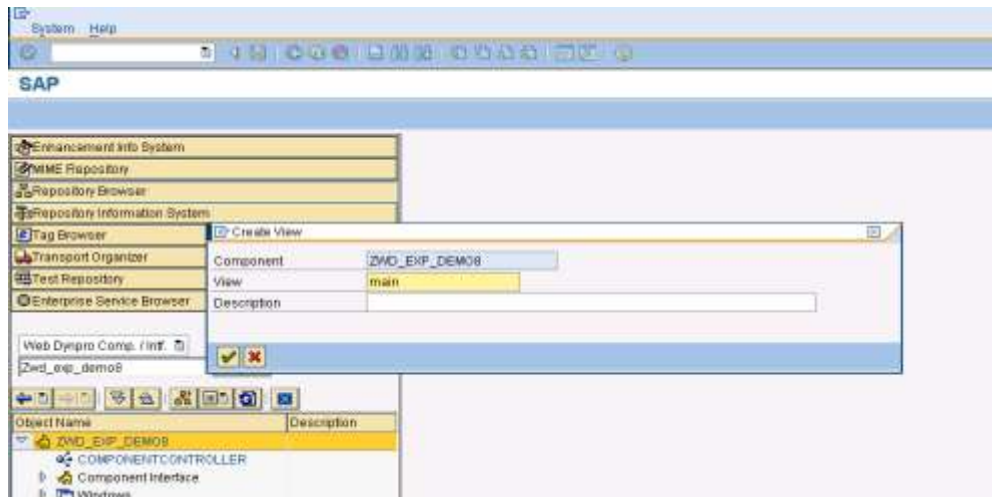
Basic concepts like creating the ui elements must be needed before going through this article.

Steps to create a component and displaying the output in alv without lines.

1)Create a webdynpro component in se80.

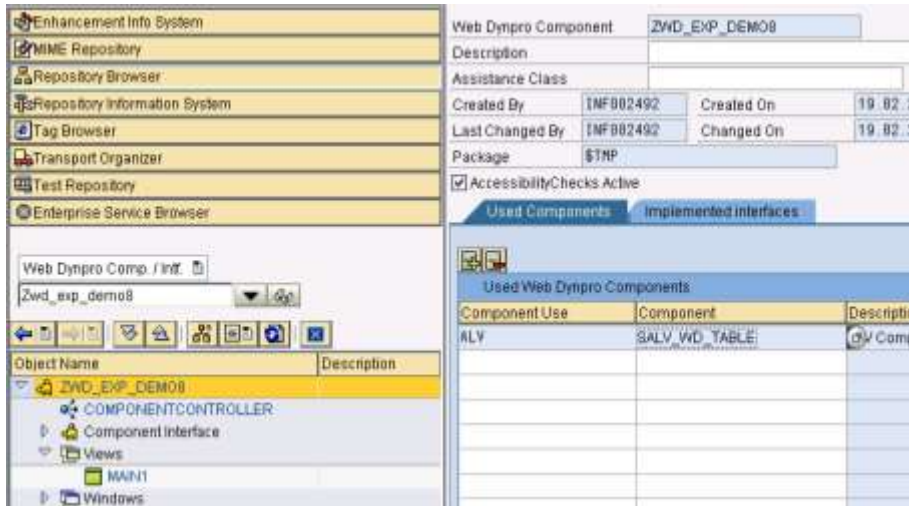


2)Create a view with the name main1(If your system doesn't create one automatically then you have to give right click on the component and create a view on your own).

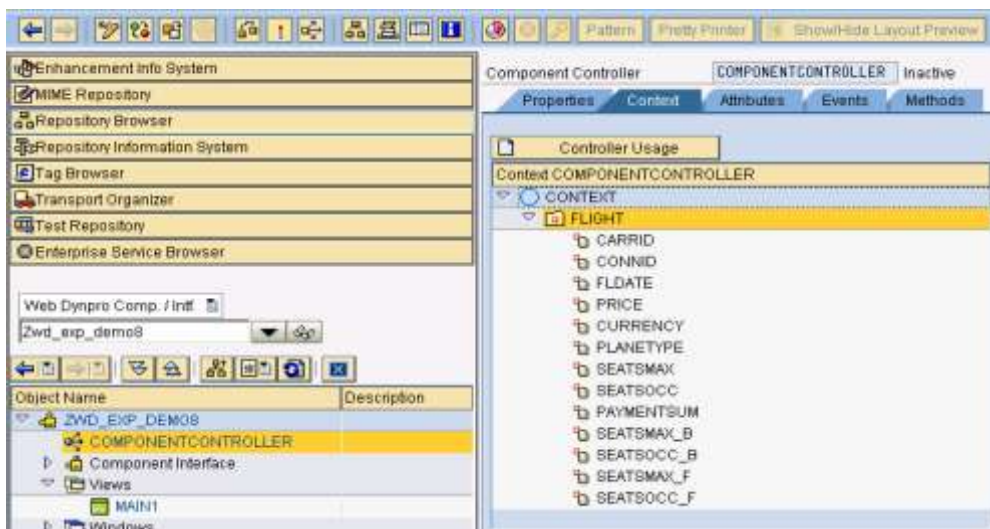


3)Declare the component SALV_WD_TABLE in the used components tab of the component.

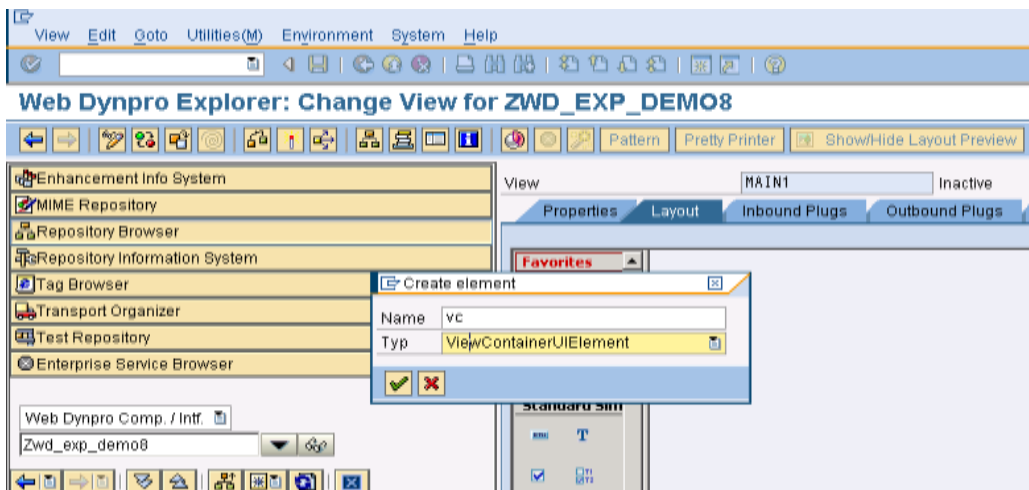
With the help of this component only we can see our output in an ALV.



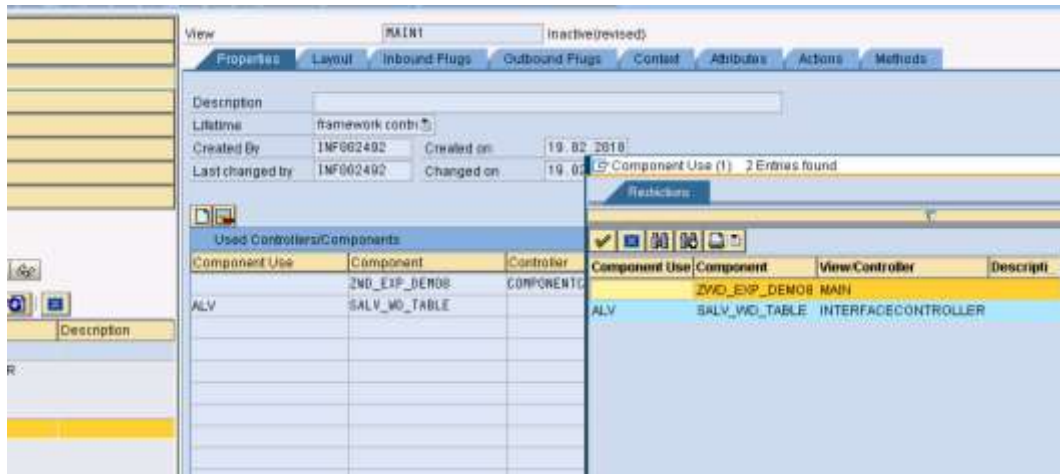
4) create a node with the name 'FLIGHT' and add all the attributes of the 'SFLIGHT' table.



5) In the layout tab of the view create a viewcontaineruelement in which we will display the alv table.



We have to declare the component usage in the properties tab of the window.



The following code should be written to fetch the details from the table 'SFLIGHT' in the method wddommodifyview of view main1.

Method WDDOMODIFYVIEW.

```

* create local data variable to access context information
DATA: CONTEXT_NODE TYPE REF TO IF_WD_CONTEXT_NODE.

* create table based on context structure
DATA: IT_flight TYPE STANDARD TABLE OF IF_MAIN1=>ELEMENT_flight,
      WA_flight LIKE LINE OF IT_flight.

*select data from required SAP table
SELECT * FROM SFLIGHT
      INTO TABLE IT_flight.

* bind data to context used by ALV
CONTEXT_NODE = WD_CONTEXT->GET_CHILD_NODE( NAME = 'FLIGHT').
CONTEXT_NODE->BIND_TABLE( IT_flight ).

* bind data to context used by ALV
CONTEXT_NODE = WD_CONTEXT->GET_CHILD_NODE( NAME = 'FLIGHT').
CONTEXT_NODE->BIND_TABLE( IT_FLIGHT ).

* Declare ALV config table
DATA: ALV_CONFIG_TABLE TYPE REF TO CL_SALV_WD_CONFIG_TABLE.
DATA: WD_TABLE_USAGE TYPE REF TO IF_WD_COMPONENT_USAGE.

* Create an instance of ALV component created ALV_COMP is usage name
WD_TABLE_USAGE = WD_THIS->WD_CPUSE_ALV( ).
IF WD_TABLE_USAGE->HAS_ACTIVE_COMPONENT( ) IS INITIAL.
  WD_TABLE_USAGE->CREATE_COMPONENT( ).
ENDIF.

* Get ALV component
DATA: WD_TABLE TYPE REF TO IWCI_SALV_WD_TABLE.

      WD_TABLE = WD_THIS->WD_CPIFC_ALV( ).
      |ALV_CONFIG_TABLE = WD_TABLE->GET_MODEL( ).

* Declare variable to store column details
DATA: COLUMN_SETTINGS TYPE REF TO IF_SALV_WD_COLUMN_SETTINGS,
      COLUMN          TYPE REF TO CL_SALV_WD_COLUMN.

```

```
COLUMN_SETTINGS ?= ALV_CONFIG_TABLE.
```

```
Endmethod.
```

The following is the code to get the feature of hiding the lines between rows and columns of an alv .

```
method WDDOINIT .

data lo_cmp_usage type ref to if_wd_component_usage.

lo_cmp_usage = wd_this->wd_cpuse_alv( ).
if lo_cmp_usage->has_active_component( ) is initial.
    lo_cmp_usage->create_component( ).
endif.

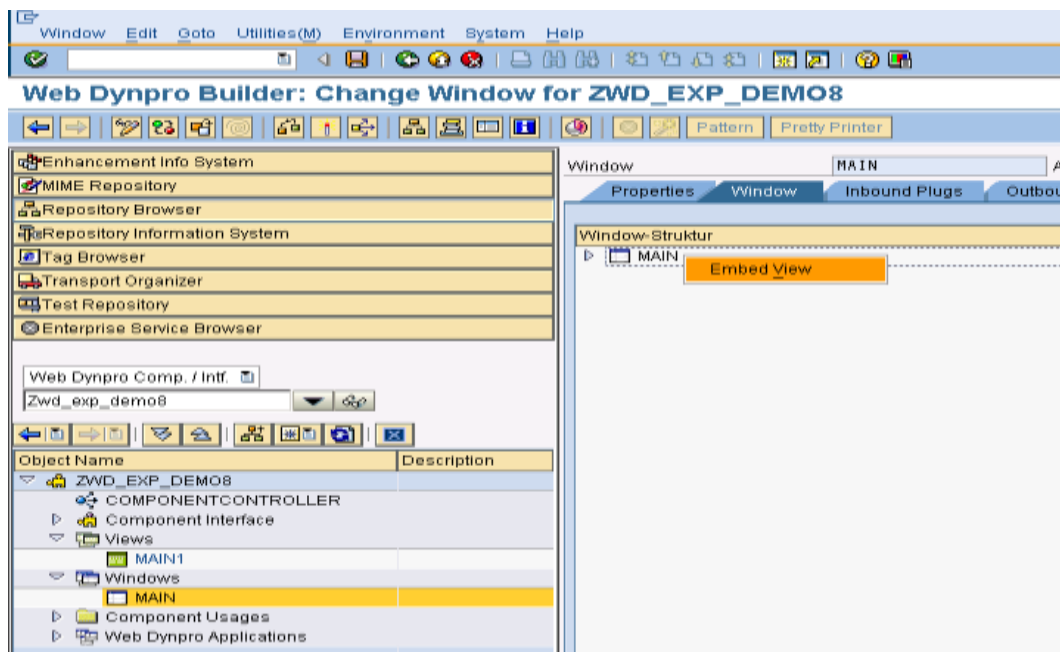
data lo_interfacecontroller type ref to iwci_salv_wd_table.
lo_interfacecontroller = wd_this->wd_cpifc_alv( ).

data lo_value type ref to cl_salv_wd_config_table.
lo_value = lo_interfacecontroller->get_model( ).

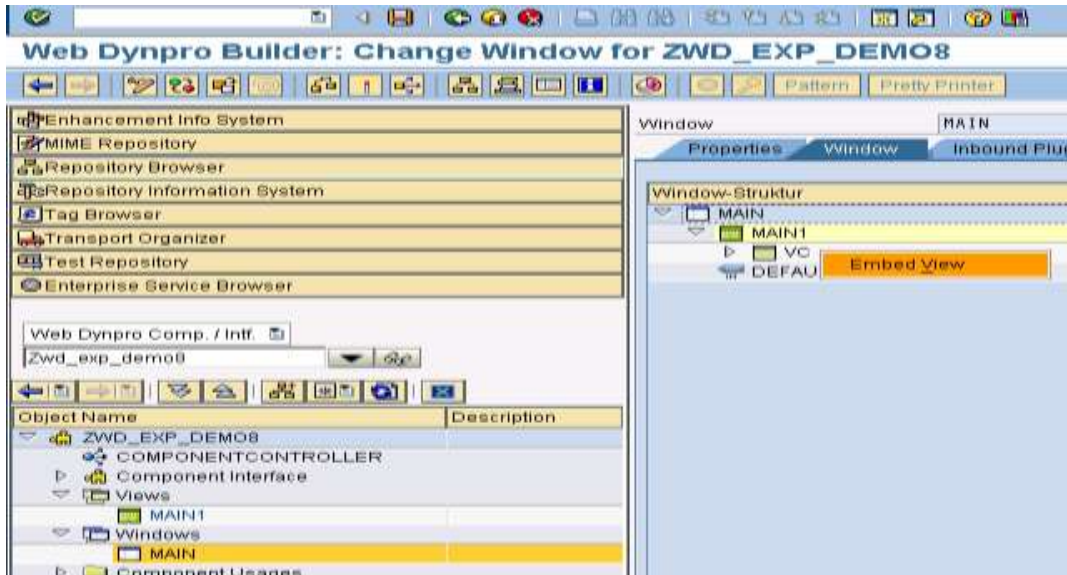
* hide the grid lines
lo_value->if_salv_wd_table_settings~SET_GRID_MODE( value = '01' ).

endmethod.
```

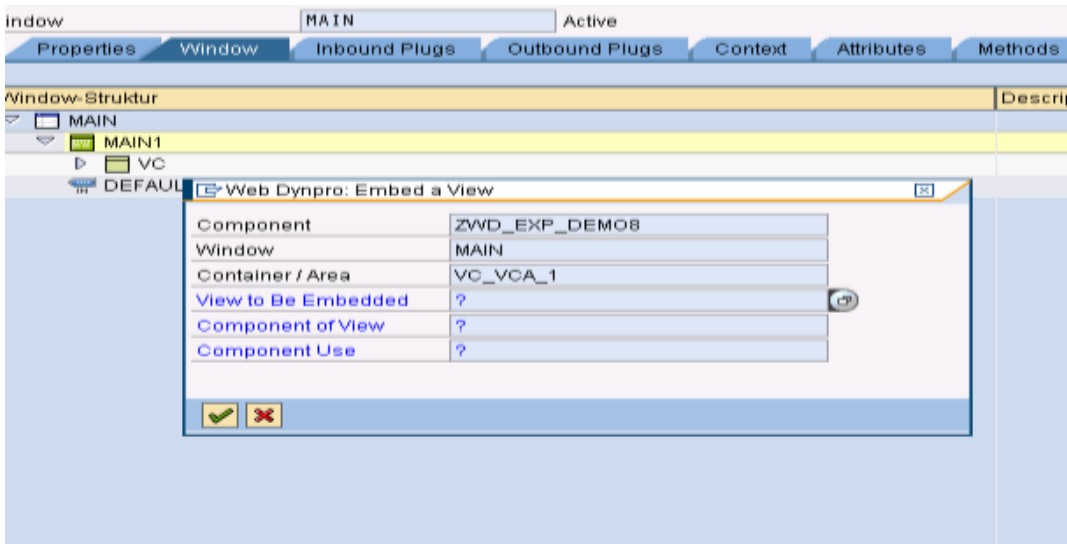
Now we have to embed the main1 view into the window.



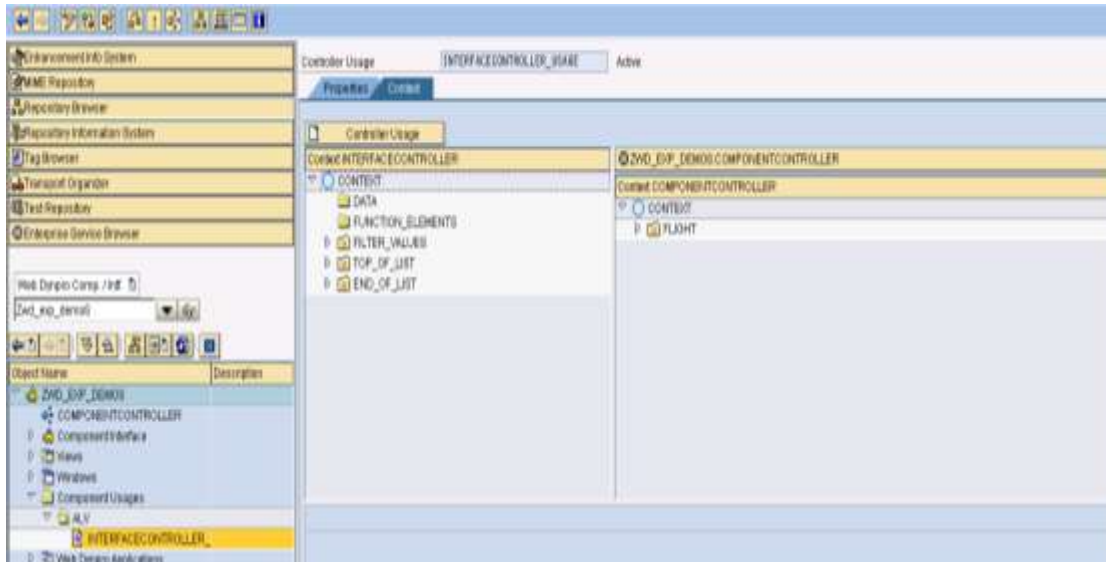
And in the viewcontaineruelement we have to embed the table view of the alv component.



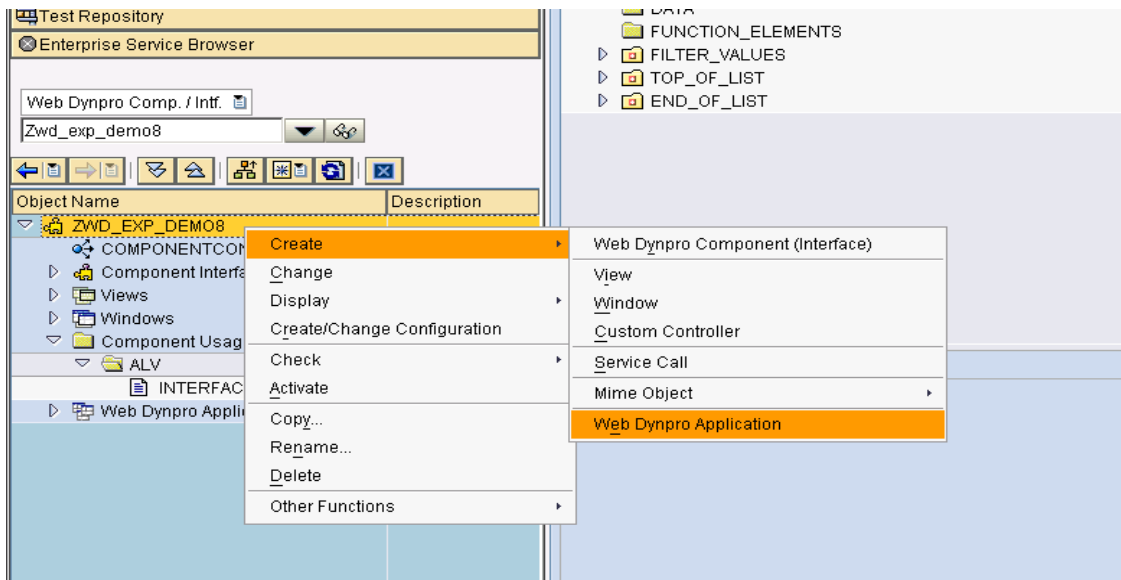
Give a f4 in the view to be embedded and select the table view of the alv component.



Now we have to define the mapping of the context flight to the data node of the interfacecontroller of the alv, To achieve this we have to navigate from component usage to ALV then interface controller. Component usage->ALV->interface controller.



Now we have to create the webdynpro application to test the view and to see the output.



Save the component and activate the component.

Right click on the webdynpro application and select 'test'.

The output must be like the following.

ID	No.	Date	Airfare	Cur.	Name	Capacity in economy class	Occupied economy class	Booking total	Capacity business class	Occupied business class
AA	0017	19.11.2008	422.94	USD	747-400	385	373	192,497.52	31	31
AA	0017	17.12.2008	422.94	USD	747-400	385	372	192,628.23	31	30
AA	0017	14.01.2009	422.94	USD	747-400	385	373	193,529.04	31	29
AA	0017	11.02.2009	422.94	USD	747-400	385	371	192,120.71	31	30
AA	0017	11.03.2009	422.94	USD	747-400	385	363	189,282.69	31	30
AA	0017	08.04.2009	422.94	USD	747-400	385	367	189,559.29	31	30
AA	0017	06.05.2009	422.94	USD	747-400	385	384	190,847.73	31	30
AA	0017	03.06.2009	422.94	USD	747-400	385	370	188,870.72	31	29
AA	0017	01.07.2009	422.94	USD	747-400	385	181	94,005.98	31	15
AA	0017	29.07.2009	422.94	USD	747-400	385	63	33,488.38	31	5

Related Content

http://help.sap.com/saphelp_nw04/helpdata/en/07/e7eb40c4f8712ae10000000a155106/content.htm

For more information, visit the [Web Dynpro ABAP homepage](#)

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.