

How to Assign Different Cell Editors for a Column in ALV for Web Dynpro ABAP



Applies to:

SAP Net Weaver 2004s. Web Dynpro for ABAP. For more information, visit the [Web Dynpro ABAP homepage](#).

Summary

This document explains how to assign different cell editors for a column in ALV in Web dynpro ABAP based on certain conditions.

Author: Pooja Patodia

Company: Infosys Technologies

Created on: 14 March 2011

Author Bio

Pooja Patodia is working at Infosys Technologies Limited. She is working on Web Dynpro ABAP and Adobe Interactive forms.

Table of Contents

Create ALV and Assign Different Cell Editors to a Column.....	3
Steps.....	3
Result.....	11
Related Content.....	12
Disclaimer and Liability Notice.....	13

Create ALV and Assign Different Cell Editors to a Column

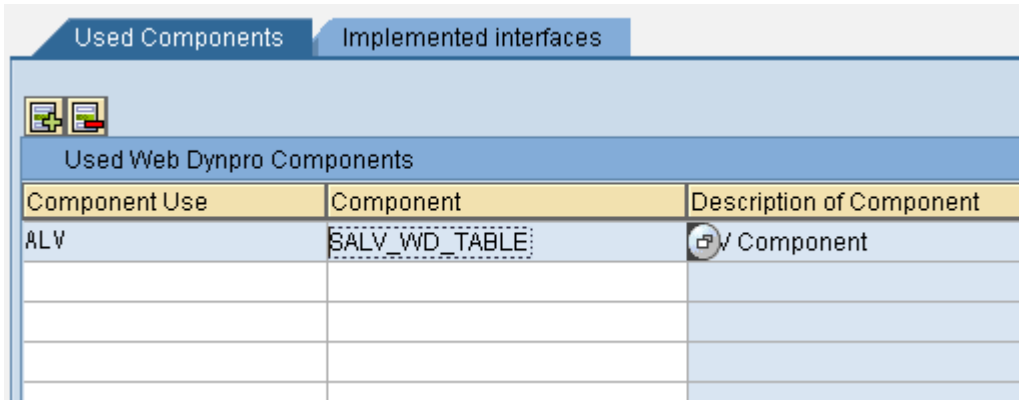
This document explains how to assign different cell editors to ALV in Web Dynpro ABAP. Textview is the cell editor for a column by default. But sometimes there are conditions where we need to display different cell editors for a column in different rows.

In the example below we are going to display the flight details in an ALV. If the airline id is 'AA' we would like to display a hyperlink in that column otherwise an inputfield.

Using different cell editors for different rows of the same column can be achieved using cell variants. For information on cell variants please visit [Cell Variants for Web Dynpro ABAP](#)

Steps

1. Create a new Web Dynpro Component 'ZALV_FLIGHT_DETAILS' with the following details.
View: FLIGHT_DETAILS_VIEW
Window: ZALV_FLIGHT_DET_WIN
2. Go to Component usage tab of the web dynpro component and declare 'SALV_WD_TABLE' as a component usage.

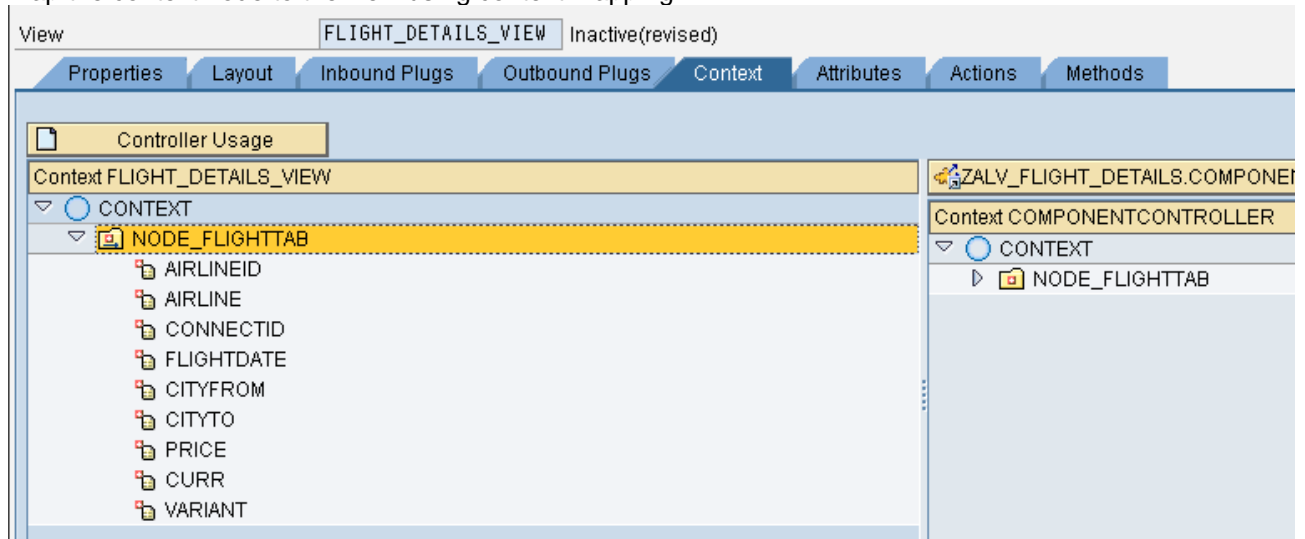


3. Create a node 'NODE_FLIGHTTAB' with cardinality 0..n in component controller for holding the flight details.
4. Create following attributes under the node.

Attribute Name	Type
AIRLINEID	BAPISFLDAT-AIRLINEID
AIRLINE	BAPISFLDAT-AIRLINE
CONNECTID	BAPISFLDAT-CONNECTID
FLIGHTDATE	BAPISFLDAT-FLIGHTDATE
CITYFROM	BAPISFLDAT-CITYFROM
CITYTO	BAPISFLDAT-CITYTO

PRICE	BAPISFLDAT-PRICE
CURR	BAPISFLDAT-CURR
VARIANT	STRING

- Map the context node to the view using context mapping.



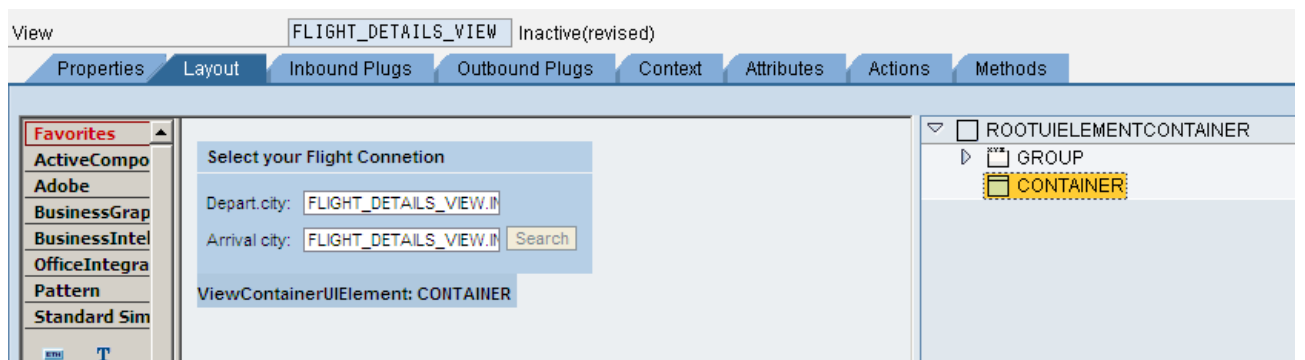
- Create a node 'INPUT_NODE' with cardinality 1..1 in view controller for holding the input details.

- Create the following attributes under the node.

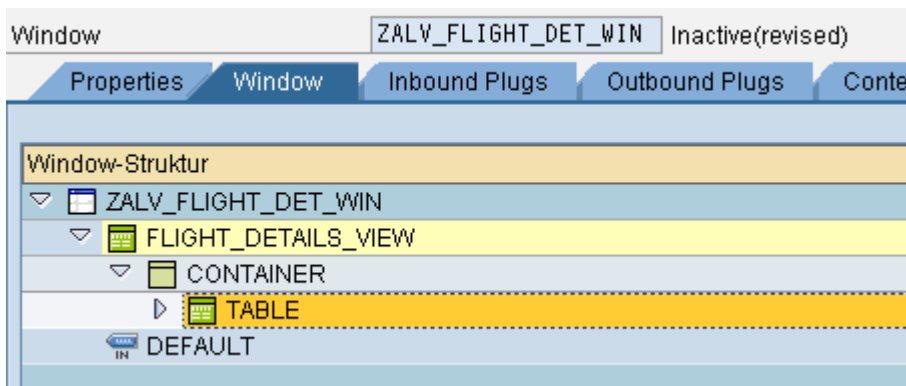
Attribute Name	Type
CITYFROM	SPFLI-CITYFROM
CITYTO	SPFLI-CITYTO

- Go to the layout tab of the view and create appropriate UI for the user to enter the city from and city to details and click on search button.

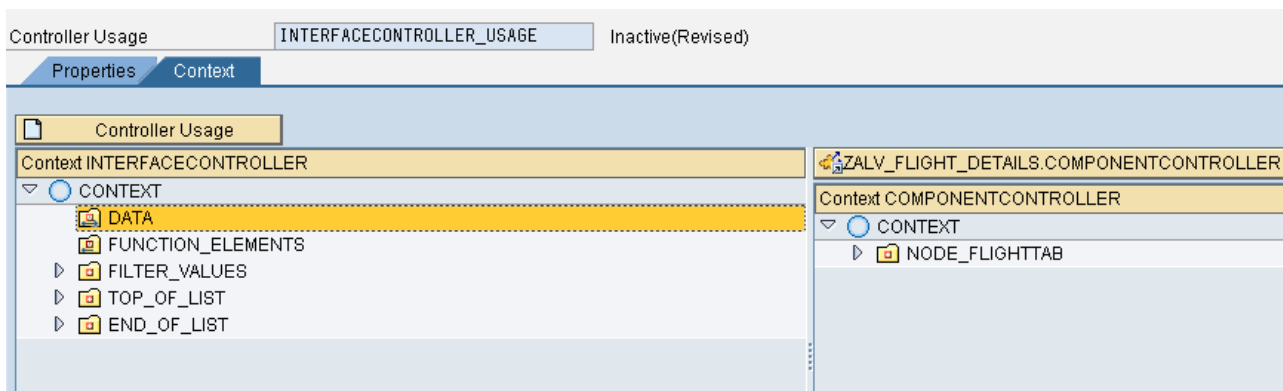
- Create a VIEWCONTAINERUIELEMENT in order to display the flight details in ALV.



- Go to the Window and embed 'TABLE' view from ALV component usage in the view container.



- Go to interface controller usage ALV and do the reverse context mapping for the DATA node.



- Create an event handler for SEARCH button in order to search the flight based on the input provided by the user.

Read the input provided by the user and get the data from the backend.

```

DATA:
  node_input_node          TYPE REF TO if_wd_context_node,
  elem_input_node         TYPE REF TO
if_wd_context_element,
  stru_input_node         TYPE
if_flight_details_view=>element_input_node ,
  item_cityfrom           LIKE stru_input_node-cityfrom,
  item_cityto             LIKE stru_input_node-cityto.

DATA: node_node_flighttab          TYPE REF TO
if_wd_context_node,
      ls_from          TYPE bapisfldst,
      ls_to           TYPE bapisfldst,
      lt_flights      TYPE TABLE OF bapisfldat,
      ls_flights      TYPE bapisfldat.

DATA: lt_final TYPE if_flight_details_view=>elements_node_flighttab,
      ls_final TYPE if_flight_details_view=>element_node_flighttab.

** Read the input provided by the user **

* navigate from <CONTEXT> to <INPUT_NODE> via lead selection
node_input_node = wd_context->get_child_node( name =
if_flight_details_view=>wdctx_input_node ).

* get element via lead selection
elem_input_node = node_input_node->get_element( ).

* get single attribute
elem_input_node->get_attribute(
  EXPORTING
    name = `CITYFROM`
  IMPORTING
    value = item_cityfrom ).

* get single attribute
elem_input_node->get_attribute(
  EXPORTING
    name = `CITYTO`
  IMPORTING
    value = item_cityto ).

```

```

* Fill the structures ls_from and ls_to
ls_from-city = item_cityfrom.
ls_to-city = item_cityto .

* Call the function BAPI_FLIGHT_GETLIST
CALL FUNCTION 'BAPI_FLIGHT_GETLIST'
EXPORTING
  destination_from = ls_from
  destination_to = ls_to
TABLES
  flight_list = lt_flights.

```

Populate the context attribute 'VARIANT' based on airline id.

```

* Assign cell variant based on airline id

LOOP AT lt_flights INTO ls_flights.
  MOVE-CORRESPONDING ls_flights TO ls_final.
  IF ls_flights-airlineid = 'AA'.
    ls_final-variant = 'LA'.
  ELSE.
    ls_final-variant = 'IN'.
  ENDIF.
  APPEND ls_final TO lt_final.
ENDLOOP.

```

Bind the data to context node in order to display it in ALV.

```

* navigate from <CONTEXT> to <NODE_FLIGHTTAB> via lead selection
node_node_flighttab = wd_context->get_child_node( name =
if_flight_details_view=>wdctx_node_flighttab ).

* Bind the data to the node NODE_FLIGHTTAB
node_node_flighttab->bind_elements( lt_final ).

```

13. Create controller usage in the view in order to get the ALV reference.

View: FLIGHT_DETAILS_VIEW Inactive(revised)

Properties | Layout | Inbound Plugs | Outbound Plugs | Context | Attributes | Action

Description:

Lifetime: framework contrc

Created By: IDADMIN Created on: 14.03.2011

Last changed by: IDADMIN Changed on: 14.03.2011

Used Controllers/Components

Component Use	Component	Controller	Description
	ZALV_FLIGHT_DETAILS	COMPONENTCONTROLLER	
ALV	SALV_WD_TABLE		ALV Component
ALV	SALV_WD_TABLE	INTERFACECONTROLLER	

14. Create an attribute in the view ALV_CONFIG_TABLE of type CL_SALV_WD_CONFIG_TABLE.

View: FLIGHT_DETAILS_VIEW Inactive

Properties | Layout | Inbound Plugs | Outbound Plugs | Context | Attributes | Actions | Mett

Attribute	RefTo	Associated Type	Description
WD_CONTEXT	<input checked="" type="checkbox"/>	IF_WD_CONTEXT_NODE	Reference to Local Controller Context
WD_THIS	<input checked="" type="checkbox"/>	IF_FLIGHT_DETAILS_VIEW	Self-Reference to Local Controller Inter
WD_COMP_CONTROLLER	<input checked="" type="checkbox"/>	IG_COMPONENTCONTROLLER	Reference to Component Controller
ALV_CONFIG_TABLE	<input checked="" type="checkbox"/>	CL_SALV_WD_CONFIG_TABLE	

15. Define the cell variants input field and link to action and set them for the ALV.

```
* create an instance of ALV component
DATA: l_ref_cmp_usage TYPE REF TO if_wd_component_usage.

l_ref_cmp_usage = wd_this->wd_cpuse_alv( ).
IF l_ref_cmp_usage->has_active_component( ) IS INITIAL.
  l_ref_cmp_usage->create_component( ).
ENDIF.
```



```
* get ALV component
DATA:
  lr_salv_wd_table TYPE REF TO iwci_salv_wd_table.

lr_salv_wd_table = wd_this->wd_cpifc_alv( ).
wd_this->alv_config_table = lr_salv_wd_table->get_model( ).

* set visible row count
DATA:
  lr_table_settings TYPE REF TO if_salv_wd_table_settings.

lr_table_settings ?= wd_this->alv_config_table.
lr_table_settings->set_visible_row_count( '10' ).

* setting cell variants
DATA:
  lr_column_settings TYPE REF TO if_salv_wd_column_settings.

DATA: lr_input_field TYPE REF TO cl_salv_wd_uie_input_field.
DATA: lr_column_airline TYPE REF TO cl_salv_wd_column.
DATA: lr_cv TYPE REF TO cl_salv_wd_cv_standard.
DATA: lr_la TYPE REF TO cl_salv_wd_uie_link_to_action.

lr_column_settings ?= wd_this->alv_config_table.

lr_column_airline = lr_column_settings->get_column( 'AIRLINE' ).

* Add cell variant input field
CREATE OBJECT lr_cv.

CALL METHOD lr_cv->set_key
  EXPORTING
    value = 'IN'.

CREATE OBJECT lr_input_field EXPORTING value_fieldname = 'AIRLINE'.

CALL METHOD lr_cv->set_editor
  EXPORTING
    value = lr_input_field.
```

```
CALL METHOD lr_column_airline->add_cell_variant
EXPORTING
  r_cell_variant = lr_cv.
* Add cell variantlink to action

CREATE OBJECT lr_cv.

CALL METHOD lr_cv->set_key
EXPORTING
  value = 'LA'.

CREATE OBJECT lr_la.

CALL METHOD lr_la->set_text
EXPORTING
  value = 'Details'.

CALL METHOD lr_cv->set_editor
EXPORTING
  value = lr_la.

CALL METHOD lr_column_airline->add_cell_variant
EXPORTING
  r_cell_variant = lr_cv.

CALL METHOD lr_column_airline->set_sel_cell_variant_fieldname
EXPORTING
  value = 'VARIANT'.

CALL METHOD wd_this->alv_config_table->if_salv_wd_table_settings~set_read_only
EXPORTING
  value = abap_false.
```

16. Save and activate the component.
17. Create a web dynpro application and execute it.

Result

In the below ALV Airline is shown as an hyperlink as the airline id is "AA".

Select your Flight Connetion

Depart.city:

Arrival city:

View [Standard View] Filter Settings

	Airline ↕	Airline ↕	Flight No. ↕	Date ↕	Depart.cit ↕	Dest. ↕	Price ↕	Currency ↕	VARIANT ↕
	AA	Details	0017	14.06.2006	NEW YORK	SAN FRANCISCO	422,9400	USD	LA
	AA	Details	0017	12.07.2006	NEW YORK	SAN FRANCISCO	422,9400	USD	LA
	AA	Details	0017	09.08.2006	NEW YORK	SAN FRANCISCO	422,9400	USD	LA
	AA	Details	0017	06.09.2006	NEW YORK	SAN FRANCISCO	422,9400	USD	LA
	AA	Details	0017	04.10.2006	NEW YORK	SAN FRANCISCO	422,9400	USD	LA
	AA	Details	0017	01.11.2006	NEW YORK	SAN FRANCISCO	422,9400	USD	LA
	AA	Details	0017	29.11.2006	NEW YORK	SAN FRANCISCO	422,9400	USD	LA
	AA	Details	0017	27.12.2006	NEW YORK	SAN FRANCISCO	422,9400	USD	LA
	AA	Details	0017	24.01.2007	NEW YORK	SAN FRANCISCO	422,9400	USD	LA
	AA	Details	0017	21.02.2007	NEW YORK	SAN FRANCISCO	422,9400	USD	LA

Row 1 of 28

In the below ALV Airline is shown as an hyperlink if the airline id is "AA" and otherwise the input field is displayed.

Select your Flight Connetion

Depart.city:

Arrival city:

View [Standard View] Filter Settings

	Airline ↕	Airline ↕	Flight No. ↕	Date ↕	Depart.cit ↕	Dest. ↕	Price ↕	Currency ↕	VARIANT ↕
	AA	Details	0017	21.03.2007	NEW YORK	SAN FRANCISCO	422,9400	USD	LA
	AA	Details	0017	18.04.2007	NEW YORK	SAN FRANCISCO	422,9400	USD	LA
	AA	Details	0017	16.05.2007	NEW YORK	SAN FRANCISCO	422,9400	USD	LA
	AA	Details	0017	13.06.2007	NEW YORK	SAN FRANCISCO	422,9400	USD	LA
	DL	Delta Airlines	1699	12.06.2006	NEW YORK	SAN FRANCISCO	422,9400	USD	IN
	DL	Delta Airlines	1699	10.07.2006	NEW YORK	SAN FRANCISCO	422,9400	USD	IN
	DL	Delta Airlines	1699	07.08.2006	NEW YORK	SAN FRANCISCO	422,9400	USD	IN
	DL	Delta Airlines	1699	04.09.2006	NEW YORK	SAN FRANCISCO	422,9400	USD	IN
	DL	Delta Airlines	1699	02.10.2006	NEW YORK	SAN FRANCISCO	422,9400	USD	IN
	DL	Delta Airlines	1699	30.10.2006	NEW YORK	SAN FRANCISCO	422,9400	USD	IN

Row 11 of 28

Related Content

[Creating a Simple ALV](#)

[Cell Variants for Web Dynpro ABAP](#)

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.