Using Totals and Subtotals Feature of ALV Component in WD4A

**Applies to:**
Web Dynpro for ABAP  For more information, visit the [Web Dynpro ABAP homepage](http).

**Summary**
The tutorial describes how to configure ALV and how to make use of Totals and Subtotals feature provided by ALV component in Web Dynpro for ABAP.

**Author:** Manas Dua
**Company:** SAP Labs India Pvt. Ltd
**Created on:** 26 June 2009

**Author Bio**
Author works as a developer for Insurance and Sales Force Management project in Bangalore.
Table of Contents

Introduction .........................................................................................................................................................3
Getting Started ................................................................................................................................................3
Creating ALV Component Usage ..................................................................................................................4
Binding Data to ALV .......................................................................................................................................6
Creating Subtotals for ALV ..............................................................................................................................6
Testing the Application .....................................................................................................................................8
Related Content ...............................................................................................................................................9
Copyright ..........................................................................................................................................................10
Using Totals and Subtotals Feature of ALV Component in WD4A

Introduction

SAP List Viewer (ALV) is a flexible tool used to display data in a tabular format. The tool provides common list operations as standard functions and can be enhanced by self-defined options.

As the developer of the application, we have various methods at our disposal to define the appearance, function, and run-time behavior of this ALV output.

One such behavior which most developers have to define is Summing up the numeric columns based on some parameter like Currency.

In this particular article we'll see how an ALV can be configured by code to create Total and Subtotal Rows based on user requirements.

We'll be using data of SFLIGHT table and we'll be displaying total of Ticket Prices based on Ticket Currency.

Getting Started

- Create a sample web dynpro component (let's say ZTEST_ALV).
- Add a view to it (ALV_VIEW).
- In view ALV_VIEW create a view container UI element and assign a name to it (let's say ALV_CONT).
- Create a window (ALV_WINDOW) and embed ALV_VIEW to this window.
- Create a web dynpro application (ZTEST_WDA_ALV) for this newly created component ZTEST_ALV and mention Interface View as ALV_WINDOW and plug name as DEFAULT.
- Create a node in component controller named as TABLE_DATA with Dictionary Structure as SFLIGHT and cardinality as 0..n. Transfer all table columns in SFLIGHT so as to create attributes for node.
- To fill data we've created method FILL_DATA which selects all data from SFLIGHT and then we bind this fetched data to node TABLE_DATA. Call the method FILL_DATA in WDDOINIT of component controller. Following is the code for FILL_DATA method

```plaintext
method FILL_DATA .

DATA lo_nd_table_data TYPE REF TO if_wd_context_node.
DATA lt_table_data TYPE wd_this->Elements_table_data.

* get node reference
lo_nd_table_data = wd_context->get_child_node( name = wd_this->wdctx_table_data ).

* select all data from SFLIGHT
select * from sflight into table lt_table_data.

* bind data to node
lo_nd_table_data- >bind_table( new_items = lt_table_data set_initial_elements = abap_true ).

endmethod.
```
Creating ALV Component Usage

With basic customizing in place, next step would be to create a Component Usage for ALV.

- First we'll be defining a component usage for the ALV component SALV_WD_TABLE in our application component. For this go to your component ZTEST_ALV and in Used Components tab click on + button to create a component usage for type SALV_WD_TABLE. Give a name to your component usage under Component Use column (let's say MY_ALV).

<table>
<thead>
<tr>
<th>Web Dynpro Component</th>
<th>ZTEST_ALV</th>
<th>Active</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Test Application</td>
<td></td>
</tr>
<tr>
<td>Assistance Class</td>
<td>ZTEST_ASSISTANCE_MANAS</td>
<td></td>
</tr>
<tr>
<td>Created By</td>
<td>DUAM</td>
<td>Created On</td>
</tr>
<tr>
<td>Last Changed By</td>
<td>DUAM</td>
<td>Changed On</td>
</tr>
<tr>
<td>Original Lang.</td>
<td>EN</td>
<td>Package</td>
</tr>
</tbody>
</table>

Defining Usage of ALV Component

- Now we have to define the usage of this component in the properties of our view (ALV_VIEW). For this click on ALV_VIEW and in Properties tab under Used Controllers/ Components section click on Create button to select entries related to our defined component usage MY_ALV.

View: ALV_VIEW

<table>
<thead>
<tr>
<th>Properties</th>
<th>Layout</th>
<th>Inbound Plugs</th>
<th>Outbound Plugs</th>
<th>Context</th>
<th>Attributes</th>
<th>Actions</th>
</tr>
</thead>
</table>

Description: ALV View

Lifetime: 0 framework.con

Created By: DUAM

Created on: 16.06.2009

Last changed by: DUAM

Changed on: 29.06.2009

Defining Usage of MY_ALV component in View

- Once the component use is defined for view, next step would be to embed the ALV view into window's view container UI element. For this go to Window ALV_WINDOW and right click on view
Using Totals and Subtotals Feature of ALV Component in WD4A

container UI element (ALV_CONT) to embed a view. Select declared ALV component MY_ALV and with View Interface View entry as TABLE

Embedding the ALV view into View Container
Binding Data to ALV

To bind the data to ALV, click on Component Usages and select our used ALV component MY_ALV and expand it to view INTERFACECONTROLLER_USAGE node. To bind the data to ALV, drag the node TABLE_DATA from component controller and map it to DATA node of ALV context. This action will map the data from component controller context node to ALV Interface controller context node DATA.

Creating Subtotals for ALV

Once the data is bound, next step is to configure the ALV for creating subtotals.

For this create a method named as CREATE_SUBTOTAL and call it inside view WDDOINIT method.

Inside CREATE_SUBTOTAL we’ll be writing the code for creating subtotal of Price column based on Currency.

Following are the steps performed in CREATE_SUBTOTAL method –

- Getting the instance of ALV
- Getting all columns
- Looping at columns and declaring aggregation rule for PRICE column.
- For creating subtotal based on Currency create a sort rule for currency column.
Following is the code snipped for CREATE_SUBTOTAL method –

```
method CREATE_SUBTOTAL.

*This method creates subtotal for PRICE based on currency column

DATA: lo_cmp_usage            TYPE ref to if_wd_component_usage.
DATA: lr_salv_wd_table        TYPE REF TO iwci_salv_wd_table.
DATA: lr_column_settings      TYPE REF TO if_salv_wd_column_settings,
  lr_column               TYPE REF TO cl_salv_wd_column,
  lt_column               TYPE salv_wd_t_column_ref,
  ls_column               TYPE salv_wd_s_column_ref.
DATA: lr_function_settings    TYPE REF TO CL_SALV_WD_CONFIG_TABLE.
DATA: lr_field_settings       TYPE REF TO IF_SALV_WD_FIELD_SETTINGS,
  lr_field_curr           TYPE REF TO CL_SALV_WD_FIELD,
  lr_field_amnt           TYPE REF TO CL_SALV_WD_FIELD.
DATA: lv_aggr_rule            TYPE REF TO CL_SALV_WD_AGGR_RULE.
DATA: lr_sort_rule            TYPE REF TO CL_SALV_WD_SORT_RULE.

*create an instance of ALV component
lo_cmp_usage = wd_this->wd_cpuse_MY_ALV( ).
* if not initialized, then initialize
  if lo_cmp_usage->has_active_component( ) is initial.
    lo_cmp_usage->create_component( ).
  endif.
* get ALV component
lr_salv_wd_table = wd_this->wd_cpifc_MY_ALV( ).
lr_function_settings = lr_salv_wd_table->get_model( ).
* get reference to column settings
lr_column_settings ?= lr_function_settings.
* get all columns
lt_column = lr_column_settings->get_columns( ).
* loop at columns
  loop at lt_column into ls_column.
    CASE ls_column-id.
      when 'PRICE'.
        * for PRICE aggregate field
          CALL METHOD LR_FUNCTION_SETTINGS->IF_SALV_WD_FIELD_SETTINGS~GET_FIELD
            EXPORTING
              FIELDNAME = 'PRICE'
            RECEIVING
              VALUE = lr_field_amnt.
        * create aggregate rule as total
          CALL METHOD LR_FIELD_AMNT->IF_SALV_WD_AGGR~CREATE_AGGR_RULE
            EXPORTING
              AGGREGATION_TYPE = IF_SALV_WD_C_AGGR=>AGGRTYPE_TOTAL
            RECEIVING
              VALUE = lv_aggr_rule.
      when 'CURRENCY'.
        CALL METHOD LR_FUNCTION_SETTINGS->IF_SALV_WD_FIELD_SETTINGS~GET_FIELD
            EXPORTING
              FIELDNAME = 'CURRENCY'
            RECEIVING
              VALUE = lr_field_curr.
      * sub totals based on contract currency.
        CALL METHOD LR_FIELD_CURR->IF_SALV_WD_SORT~CREATE_SORT_RULE
          EXPORTING
            SORT_ORDER = IF_SALV_WD_C_SORT=>SORT_ORDER_ASCENDING
            GROUP_AGGREGATION = ABAP_TRUE
```
RECEIVING
VALUE = lr_sort_rule.

*hide others
  when OTHERS.
    ls_column-r_column->set_visible( if_wdl_core=>visibility_none ).
  endcase.
endloop.
endmethod.

Testing the Application

Test the application created above ZTEST_WDA_ALV to see the subtotals of Price column based on currency in ALV output.

ALV with Subtotals of Airfare based on Currency
Related Content

For more information, visit the User Interface Technology homepage.

For more information, visit the Web Dynpro ABAP homepage.