Auto Refreshing ALV Grid using Web Dynpro ABAP

**Applies to:**
SAP Netweaver 2004s, Web Dynpro ABAP

**Summary**
This code sample explains how to create a simple ALV grid that refreshes automatically at specified interval of time.

**Author:** Kathirvel Balakrishnan

**Company:** Wipro Technologies

**Created on:** 02 November 2006

**Author Bio**
Kathirvel Balakrishnan is working as an ABAP Consultant for Wipro Technologies. His areas of interests are ABAP (Object, Web Dynpro), Java, VBA and PHP Programming.
Table of Contents

Applies to: ........................................................................................................................................ 1
Summary.......................................................................................................................................... 1
Author Bio ........................................................................................................................................ 1
Prerequisite...................................................................................................................................... 3
Sample Screen Shots ...................................................................................................................... 3
Step 1 – Create a Web Dynpro Component .................................................................................... 4
Step 2 – Create nodes in the Component Controller ..................................................................... 4
Step 3 – Create a methods to get details ....................................................................................... 5
Step 4 – Creating Views .................................................................................................................. 6
Step 5 - Assigning Views to Window .............................................................................................. 8
Step 6 – Set the Date to ALV ......................................................................................................... 9
Related Content............................................................................................................................... 9
Disclaimer and Liability Notice ....................................................................................................... 10
Prerequisite
This example requires basic knowledge about Web Dynpro ABAP and ALV. Lot many well documented examples are available in SDN which can be easily searched. A few of them have been given in the related content section.

Sample Screen Shots
The ALV grid refreshes at the specified timed intervals and picks the data at that point of time. Hence this is very useful in designing real time applications that require frequent updates.

The ALV grid before automatic refreshing,

The ALV Grid after automatic refresh
### Step 1 – Create a Web Dynpro Component

Create a Web Dynpro component and add the ALV Component Table to it.

<table>
<thead>
<tr>
<th>Web Dynpro Component</th>
<th>ZKB_ALV_REFRESH</th>
<th>Active</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Refresh ALV Grid Automatically</td>
<td></td>
</tr>
<tr>
<td>Assistance Class</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Created By</td>
<td>EKATHIRVEL</td>
<td></td>
</tr>
<tr>
<td>Created On</td>
<td>01.11.2005</td>
<td></td>
</tr>
<tr>
<td>Last Changed By</td>
<td>EKATHIRVEL</td>
<td></td>
</tr>
<tr>
<td>Changed On</td>
<td>02.11.2005</td>
<td></td>
</tr>
<tr>
<td>Package</td>
<td>$TMP</td>
<td></td>
</tr>
<tr>
<td>Accessibility/Checks Active</td>
<td>[ ]</td>
<td></td>
</tr>
</tbody>
</table>

### Used Web Dynpro Components

<table>
<thead>
<tr>
<th>Component Use</th>
<th>Component</th>
<th>Description of Component</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALV</td>
<td>SALV_WD_TABLE</td>
<td>ALV Component</td>
</tr>
</tbody>
</table>

### Step 2 – Create nodes in the Component Controller
Now double click the Component Controller and navigate to the Context Tab. Create two Nodes under that as shown below. You can refer the articles in the reference for help.

The ALV Grid after automatic refresh

**Step 3 – Create a methods to get details**

The source code for the method is given below,

```plaintext
method get_carriers.

* fill context node "node_flighttab"

  data: node_search type ref to if_wd_context_node,
  node_carriers type ref to if_wd_context_node,
  node_carrid type ref to if_wd_context_element,
  p_carrid type if_componentcontroller=>element_carriers,
```
ls_where(72) type c,
lt_where like table of ls_where,
lt_scarr type table of scarr.

* navigate from <CONTEXT> to <NODE_FLIGHT> via lead selection
node_search = wd_context->get_child_node( name = `SEARCH`).

* get element via lead selection
node_carrid = node_search->get_element().

* get all declared attributes
node_carrid->get_static_attributes(
  importing
    static_attributes = p_carrid).

* create where condition
if not p_carrid-carrid eq ' '.
  concatenate 'CARRID = '' p_carrid-carrid '' ' into ls_where.
append ls_where to lt_where.
endif.

* read data from database
select * from scarr
  into table lt_scarr
  where (lt_where).

* navigate from <CONTEXT> to <NODE_FLIGHT> via lead selection
node_carriers = wd_context->get_child_node( name = `CARRIERS`).

* fill context node
node_carriers->bind_table( lt_scarr).

endmethod.

Step 4 – Creating Views
Create a view named SEARCHVIEW and map the node SEARCH from the component controller to it.

Now add the Label, InputField and button SEARCH as shown below. Hint: Use the Create Container Form context menu option in the ROOTUIELEMENTCONTAINER. Add the event TIMED_SEARCH to the OnAction event of the button SEARCH.
Create a view named LAYOUT and add two view containers SEARCH and TABLE to it. Also add a TimedTrigger to it.

Now do the below given changes,

Set the layout property of ROOTUIELEMENTCONTAINER to MatrixLayout.

Set the Layout Data property of the Viewcontainers and TimedTrigger to MatrixHeadData.
Set the Visible property of TimedTrigger to none and create an Event TIMED_SEARCH for onAction event. Set the delay property as required. This will be used as the refresh interval for the ALV grid.

```
METHOD onactiontimed_search .
  wd_comp_controller->get_carriers( ).
ENDMETHOD.
```

**Step 5 - Assigning Views to Window**

Now the views are to be assigned to the window. The following steps need to be performed in order:

1. First double click the window and navigate to the Window tab.
2. Drag and drop the LAYOUT view into the window.
3. Expand the LAYOUT tree structure.
4. Embed the SEARCHVIEW to the SEARCH ViewContainer.
5. Embed the ALV TABLE view to the TABLE ViewContainer. The final structure will be as shown below.
Step 6 – Set the Date to ALV

Now navigate to the INTERFACECONTROLLER_USAGE and double click that. (Web Dynpro Component -> Components Usage -> ALV -> INTERFACECONTROLLER_USAGE). Now drag and drop the CARRIERS node in the context view of the component controller to the DATA node in the InterfaceController Context.

Create a Web Dynpro application and Test the program. The ALV will refresh automatically at regular intervals. If a select condition is specified, the same will be considered and the data will selected appropriately in the next refresh.

Related Content

- SAP List Viewer in Web Dynpro - Simple Example for Using ALV
- SAP List Viewer in Web Dynpro - Programming the ALV Configuration Model
- SAP List Viewer in Web Dynpro - Editing ALV
- SAP List Viewer in Web Dynpro - Using Events with ALV Tables
- SAP List Viewer in Web Dynpro - Using ALV with a Dynamic Context Node
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.