

```

*&-----*
*& Report   Zalv_interactivehierarchy                               *
*&                                               *
*&-----*
*& AS : ALV code for hierarchical sequential list display(single child)
*& List of SAP tables as the parent node with interactive features(as a pop up)
*& and List of fields as the child node with interactive features too(as a pop
*& up)
*&-----*

```

```
REPORT zalv_interactivehierarchy.
```

```
* type-pools declarations for alv and icon
TYPE-POOLS: slis,icon.
```

```
*structure declaration for table details
```

```
TYPES : BEGIN OF ty_dd02l,
         tabname TYPE tabname,
         tabclass TYPE tabclass,
         expand,
       END OF ty_dd02l.
```

```
*internal table and wa decln for table details
```

```
DATA : it_dd02l TYPE STANDARD TABLE OF ty_dd02l,
       wa_dd02l TYPE ty_dd02l.
```

```
*structure declarations for field details
```

```
TYPES : BEGIN OF ty_dd03l,
         tabname TYPE tabname,
         fieldname TYPE fieldname,
         keyflag TYPE keyflag,
         checktable TYPE checktable,
         rollname TYPE rollname,
         datatype TYPE datatype_d,
         leng TYPE ddleng,
       END OF ty_dd03l.
```

```
* Internal table and wa decln for field details
```

```
DATA : it_dd03l TYPE STANDARD TABLE OF ty_dd03l,
       wa_dd03l TYPE ty_dd03l.
```

```
*Internal table and wa decln for tabletext details
```

```
DATA : it_dd02t TYPE STANDARD TABLE OF dd02t,
       wa_dd02t TYPE dd02t.
```

```
* data declarations for ALV
```

```
DATA : it_fieldcat TYPE slis_t_fieldcat_alv,
       wa_fieldcat TYPE slis_fieldcat_alv,
       it_layout TYPE slis_layout_alv,
       key TYPE slis_keyinfo_alv.
```

```
*Input the tables.User cannot enter a range but can enter any number
*of tables one by one in this select-options
SELECT-OPTIONS : s_table FOR wa_dd02l-tabname NO INTERVALS.
```

```
*initializatin event
INITIALIZATION.
```

```
*start-of-selection event
START-OF-SELECTION.
```

```
*subroutine to fetch the data from the tables
```

```

PERFORM fetch_tabledata.

*subroutine to build alv hierarchy output
  PERFORM hierarchyalv_build.

*&-----*
*&      Form  hierarchyalv_build
*&-----*
*      text
*-----*
* --> p1      text
* <-- p2      text
*-----*

FORM hierarchyalv_build .

*fieldcatalogue
  PERFORM build_fieldcat.

*layout
  PERFORM build_layout.

*key information for hierarchy
  PERFORM build_key.

*output
  PERFORM list_display.

ENDFORM.                " hierarchyalv_build

*&-----*
*&      Form  build_fieldcat
*&-----*
*      text
*-----*
* --> p1      text
* <-- p2      text
*-----*

FORM build_fieldcat .

  CLEAR wa_fieldcat.

  wa_fieldcat-col_pos = 1.
  wa_fieldcat-fieldname = 'TABNAME'.
  wa_fieldcat-tabname = 'IT_DD02L'.
  wa_fieldcat-seltext_m = 'Tablename'.
  wa_fieldcat-key = 'X'.
  wa_fieldcat-hotspot = 'X'.
  wa_fieldcat-emphasize = 'C610'.
  APPEND wa_fieldcat TO it_fieldcat.
  CLEAR wa_fieldcat.

  wa_fieldcat-col_pos = 2.
  wa_fieldcat-fieldname = 'FIELDNAME'.
  wa_fieldcat-tabname = 'IT_DD03L'.
  wa_fieldcat-key = 'X'.
  wa_fieldcat-hotspot = 'X'.
  wa_fieldcat-seltext_m = 'Field'.
  wa_fieldcat-emphasize = 'C510'.
  APPEND wa_fieldcat TO it_fieldcat.

```

```

CLEAR wa_fieldcat.

ENDFORM.                " build_fieldcat
*&-----*
*&      Form  build_layout
*&-----*
*      text
*-----*
* --> p1      text
* <-- p2      text
*-----*
FORM build_layout .
*to expand the header table for item details
  it_layout-expand_fieldname = 'EXPAND'.
  it_layout-window_titlebar = 'Interactive Hierarchical ALV'.
  it_layout-lights_tabname = 'IT_DD03L'.
  it_layout-colwidth_optimize = 'X'.

ENDFORM.                " build_layout
*&-----*
*&      Form  build_key
*&-----*
*      text
*-----*
* --> p1      text
* <-- p2      text
*-----*
FORM build_key .

*key infomation for the header and item table
  key-header01 = 'TABNAME'.
  key-item01 = 'TABNAME'.

ENDFORM.                " build_key
*&-----*
*&      Form  list_display
*&-----*
*      text
*-----*
* --> p1      text
* <-- p2      text
*-----*
FORM list_display .

*ALV output
CALL FUNCTION 'REUSE_ALV_HIERSEQ_LIST_DISPLAY'
  EXPORTING
    i_callback_program      = sy-cprog
    i_callback_user_command = 'USER_COMMAND'
    is_layout               = it_layout
    it_fieldcat             = it_fieldcat
    i_tabname_header       = 'IT_DD02L'
    i_tabname_item         = 'IT_DD03L'
    is_keyinfo              = key
  TABLES
    t_outtab_header        = it_dd02l
    t_outtab_item          = it_dd03l.

ENDFORM.                " list_display

*&-----*
*&      Form  fetch_tabledata

```

```

*&-----*
*      text
*-----*
* --> p1      text
* <-- p2      text
*-----*

```

FORM fetch_tabledata .

*select table data

```

SELECT tabname
       tabclass
FROM   dd021
INTO CORRESPONDING FIELDS OF TABLE it_dd021
WHERE  tabname IN s_table.

```

*select field data

```

IF it_dd021[] IS NOT INITIAL.
  SELECT tabname
         fieldname
         keyflag
         checktable
         rollname
         leng
         datatype
FROM     dd031
INTO CORRESPONDING FIELDS OF TABLE it_dd031
FOR ALL ENTRIES IN it_dd021
WHERE  tabname EQ it_dd021-tabname.
ENDIF.

```

*select table texts

```

IF it_dd021[] IS NOT INITIAL.

  SELECT tabname
         ddtext
FROM     dd02t
INTO CORRESPONDING FIELDS OF TABLE it_dd02t
FOR ALL ENTRIES IN it_dd021
WHERE  tabname EQ it_dd021-tabname
AND    ddlanguage = 'EN'.

ENDIF.

```

ENDFORM. " fetch_tabledata

```

*&-----*
*&      Form  user_command
*&-----*
*      text
*-----*

```

*User actions on ALV

```

FORM user_command USING r_ucomm TYPE sy-ucomm
                    rs_selfield TYPE slis_selfield.

```

```

CASE r_ucomm.
  WHEN '&IC1'.
    READ TABLE it_dd021 INDEX rs_selfield-tabindex INTO wa_dd021.
    IF sy-subrc = 0.

```

```

        PERFORM select_table.
        EXIT.
    ENDIF.
    READ TABLE it_dd03l INDEX rs_selfield-tabindex INTO wa_dd03l.
    IF sy-subrc = 0.
        PERFORM select_field.
        EXIT.
    ENDIF.
ENDCASE.
ENDFORM.                "user_command

*&-----*
*&      Form  select_table
*&-----*
*      text
*-----*
*  -->  p1          text
*  <--  p2          text
*-----*
*Subroutine to display the details of the parent node selected
FORM select_table .

    DATA : title(20),
            category(10),
            description(60),
            texts(40),
            text(60).

    title = wa_dd02l-tabname.
    category = wa_dd02l-tabclass.
    texts = 'The table category : '.

    CONCATENATE texts category INTO text separated by space.

    READ TABLE it_dd02t INTO wa_dd02t WITH KEY tabname = wa_dd02l-tabname.

    description = wa_dd02t-ddtext.

    CALL FUNCTION 'POPUP_FOR_INTERACTION'
        EXPORTING
            headline = 'Selection'
            text1     = title
            text2     = description
            text3     = text
            button_1 = 'Back to List'.

ENDFORM.                " select_table

*&-----*
*&      Form  select_FIELD
*&-----*
*      text
*-----*
*  -->  p1          text
*  <--  p2          text
*-----*
*Subroutine to display the details of the child node selected
FORM select_field .

    DATA : title(20),
            text1(35),
            table(10),

```

```

        text2(45),
        check(20),
        datatype(6),
        length(3),
        roll(8),
        key(10),
        flag(2),
        keytext(15),
        typedet(18),
        details(25),
        text3(15),
        text4(25),
        text5(15),
        text6(25),
        text7(15),
        text8(25).

text1 = 'The table for the selected field is'.
title = wa_dd031-fieldname.
table = wa_dd031-tabname.
check = wa_dd031-checktable.
datatype = wa_dd031-datatype.
length = wa_dd031-leng.
roll = wa_dd031-rollname.
key = 'Keyfield '.
flag = wa_dd031-keyflag.
typedet = 'Datatype & Length'.
text4 = 'Selected field is'.
text5 = 'Check table'.
text7 = 'Data Element'.

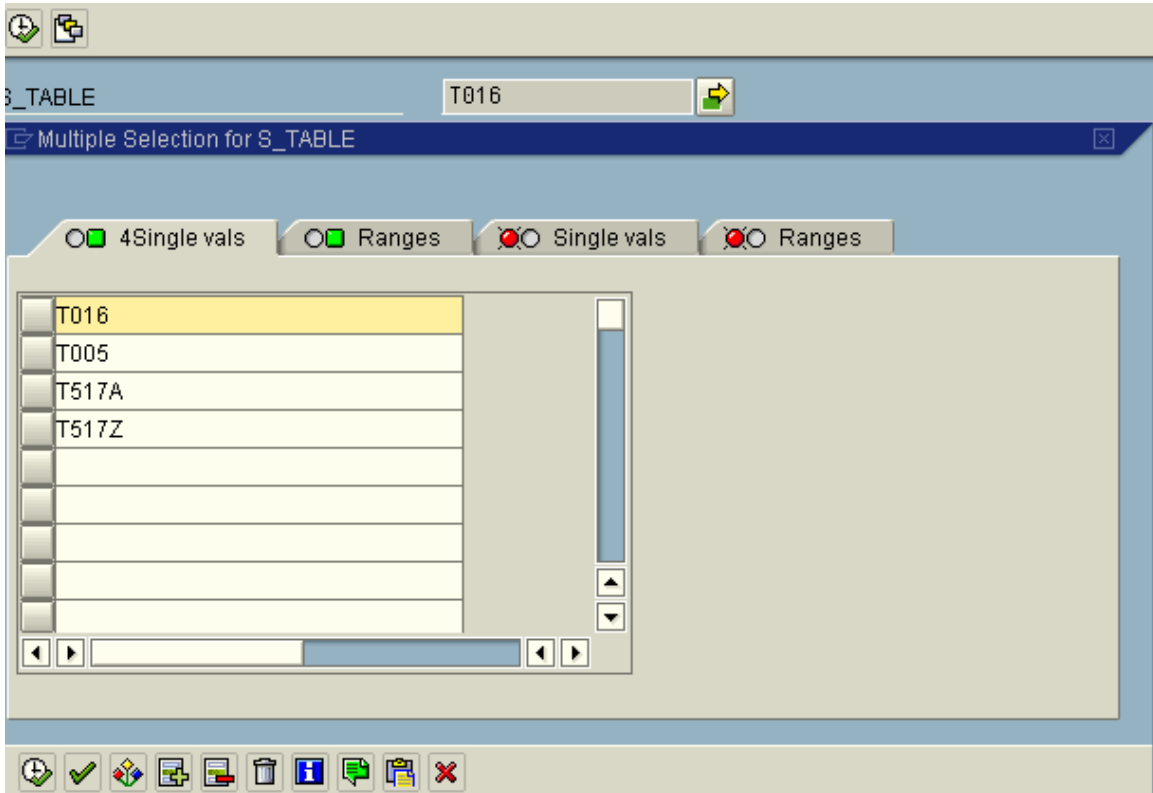
CONCATENATE text1 table INTO text2 separated by space.
CONCATENATE key flag INTO keytext separated by space.
concatenate typedet datatype length into details separated by space.
concatenate text3 title into text4 separated by space.
concatenate text5 check into text6 separated by space.
concatenate text7 roll into text8 separated by space.

CALL FUNCTION 'POPUP_FOR_INTERACTION'
  EXPORTING
    headline = 'Selection'
    text1     = text2
    text2     = text4
    text3     = text6
    text4     = details
    text5     = text8
    text6     = keytext
    button_1 = 'Back to List'.

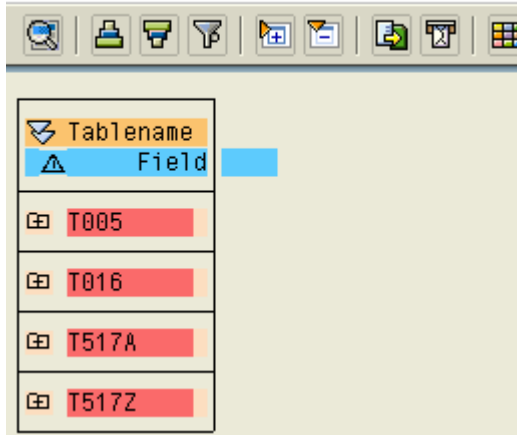
ENDFORM.                                " select_FIELD

```

Selection Screen



Interactive Hierarchical ALV



On clicking the Parent node, the selected table's details appear as



On expansion of the parent node we get the child node and on clicking a particular field in the child node, we get pop up containing the selected field.

