Step by Step Approach to Find the List of Inactive Objects in SAP BW

Applies to:

SAP BW 3.x & SAP BI Net Weaver 2004s. For more information, visit the EDW homepage.

Summary

This article explains the steps to find the list of inactive objects in SAP BW.

The program will get the list of Inactive Objects in the system

- Info Objects
- Info Providers
- ODS
- Open Hub
- Transformations
- Update Rules
- Data Transfer Process
- Transfer Structure
- MultiProvider
- Aggregates

Author: Vikram Srivastava

Company: Infosys Technologies Ltd

Created on: 21 March 2011

Author Bio

Vikram Srivastava is working as Technology Analyst with Infosys Technologies Limited. He has got rich experience on various BW Implementation/Support Projects in both SAP BW 3.5 and SAP BW 7.0.

Table of Contents

Scenario

In the article we will discuss the steps to find the list of inactive objects in SAP BW.

The program will get the list of Inactive Objects in the system

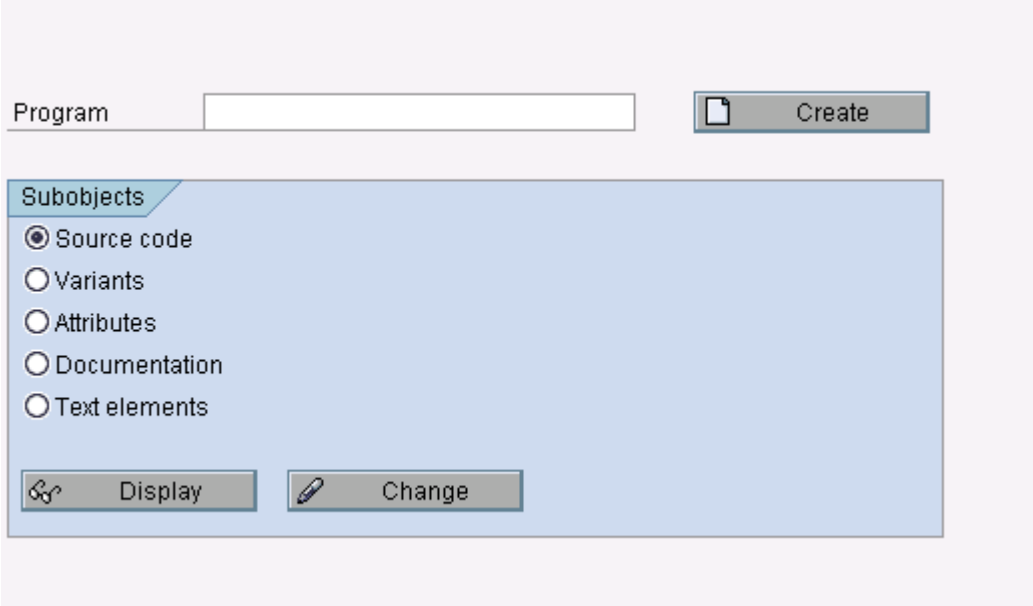
- Info Objects
- Info Providers
- ODS
- Open Hub
- Transformations
- Update Rules
- Data Transfer Process
- Transfer Structure
- MultiProvider
- Aggregates

This becomes very useful and handy in situations where we want to find out the list of inactive objects in a system before any critical activity like an upgrade or transport movement etc.

Step by Step Approach

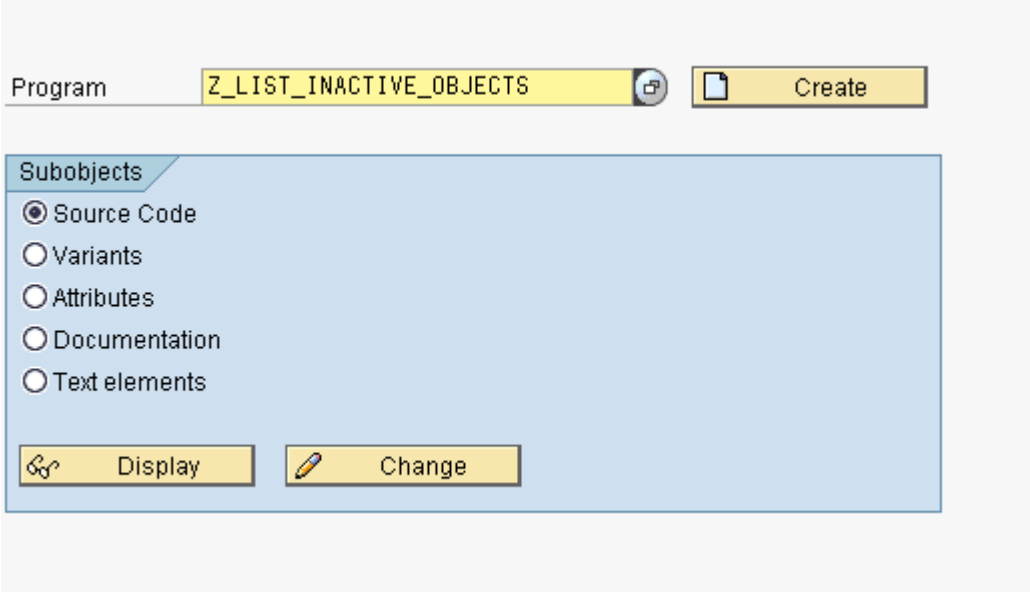
Step 1:

Go to TCode SE38



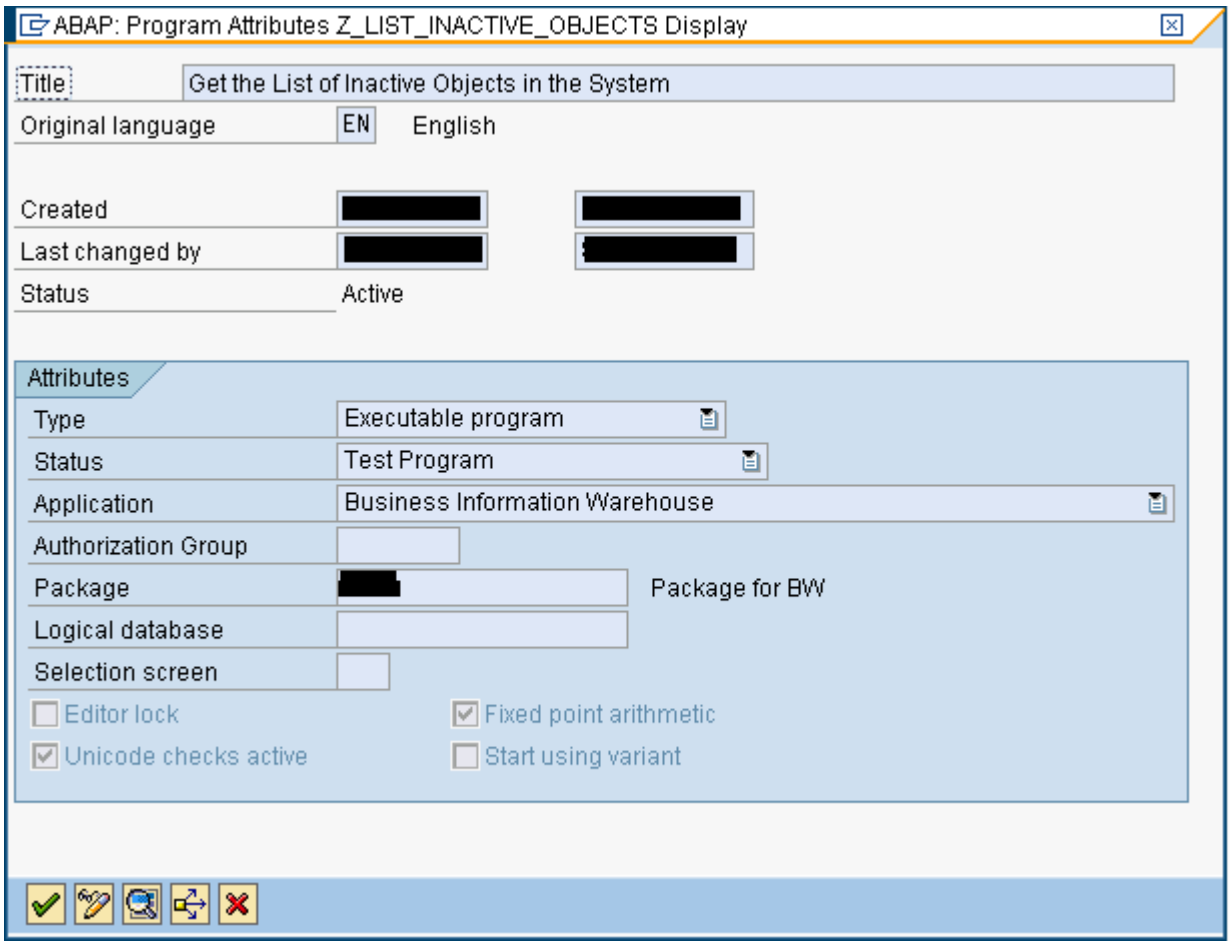
Step 2:

Enter the name of the program “Z_LIST_INACTIVE_OBJECTS”



Step 3:

Click on Create and specify the attributes as shown below and Click on Save



Step 4:

Write the program as mentioned below:

```
*&---------------------------------------------------------------------*
*& Report  Z_LIST_INACTIVE_OBJECTS
*&
*&---------------------------------------------------------------------*
*&
*&---------------------------------------------------------------------*

REPORT  Z_LIST_INACTIVE_OBJECTS MESSAGE-ID SY.

********START - Declarations ************************************

INCLUDE OLE2INCL.
TABLES:
  RSDCUBE,
  RSDIOBJ,
  RSDIOBJV.
DATA:          CHOOSE(10).
TYPES: RSD_S_TBHD LIKE RSDTBHD.   "table header.

TYPES: BEGIN OF I_S_IOBJ.
TYPES:   IOBJNM LIKE RSDIOBJ-IOBJNM,
  IOBJTP LIKE RSDIOBJ-IOBJTP,
     END OF I_S_IOBJ.
DATA: I_T_IOBJ TYPE STANDARD TABLE OF I_S_IOBJ.
DATA: I_WA_IOBJ TYPE I_S_IOBJ.

TYPES: BEGIN OF I_S_IOBJT.
TYPES:   IOBJNM LIKE RSDIOBJT-IOBJNM,
  TXTLG LIKE RSDIOBJT-TXTLG,
     END OF I_S_IOBJT.
DATA: I_T_IOBJT TYPE STANDARD TABLE OF I_S_IOBJT.
DATA: I_WA_IOBJT TYPE I_S_IOBJT.

TYPES: BEGIN OF I_S_CUBE.
TYPES:   INFOCUBE LIKE RSDCUBE-INFOCUBE,
  CUBETYPE LIKE RSDCUBE-CUBETYPE,
     END OF I_S_CUBE.
DATA: I_T_CUBE TYPE STANDARD TABLE OF I_S_CUBE.
DATA: I_WA_CUBE TYPE I_S_CUBE.

TYPES: BEGIN OF I_S_CUBET.
TYPES:   INFOCUBE LIKE RSDCUBET-INFOCUBE,
  TXTLG LIKE RSDCUBET-TXTLG,
     END OF I_S_CUBET.
DATA: I_T_CUBET TYPE STANDARD TABLE OF I_S_CUBET.
DATA: I_WA_CUBET TYPE I_S_CUBET.

TYPES: BEGIN OF I_S_TRFN.
TYPES:   TRANID LIKE RSTRAN-TRANID,
  SOURCENAME LIKE RSTRAN-SOURCENAME,
  TARGETNAME LIKE RSTRAN-TARGETNAME,
     END OF I_S_TRFN.
DATA: I_T_TRFN TYPE STANDARD TABLE OF I_S_TRFN.
DATA: I_WA_TRFN TYPE I_S_TRFN.

TYPES: BEGIN OF I_S_TRU.
```

```
TYPES:   TRANSTRU LIKE RSTS-TRANSTRU,
  LOGSYS LIKE RSTS-LOGSYS,
     END OF I_S_TRU.
DATA: I_T_TRU TYPE STANDARD TABLE OF I_S_TRU.
DATA: I_WA_TRU TYPE I_S_TRU.

TYPES: BEGIN OF I_S_UPDR.
TYPES:   UPDID LIKE RSUPDINFO-UPDID,
  ISOURCE LIKE RSUPDINFO-ISOURCE,
  INFOCUBE LIKE RSUPDINFO-INFOCUBE,
     END OF I_S_UPDR.
DATA: I_T_UPDR TYPE STANDARD TABLE OF I_S_UPDR.
DATA: I_WA_UPDR TYPE I_S_UPDR.

TYPES: BEGIN OF I_S_ODS.
TYPES:   ODSOBJECT LIKE RSDODSO-ODSOBJECT,
         ODSOTYPE LIKE RSDODSO-ODSOTYPE,
     END OF I_S_ODS.
DATA: I_T_ODS_A_D TYPE STANDARD TABLE OF I_S_ODS.
DATA: I_WA_ODS_A_D TYPE I_S_ODS.

DATA: I_T_ODS_D TYPE STANDARD TABLE OF I_S_ODS.
DATA: I_WA_ODS_D TYPE I_S_ODS.

DATA: I_T_ODS TYPE STANDARD TABLE OF I_S_ODS.
DATA: I_WA_ODS TYPE I_S_ODS.

TYPES: BEGIN OF I_S_ODST.
TYPES:   ODSOBJECT LIKE RSDODSOT-ODSOBJECT,
  TXTLG LIKE RSDODSOT-TXTLG,
     END OF I_S_ODST.
DATA: I_T_ODST TYPE STANDARD TABLE OF I_S_ODST.
DATA: I_WA_ODST TYPE I_S_ODST.

TYPES: BEGIN OF I_S_OHD.
TYPES:   OHDEST LIKE RSBOHDEST-OHDEST,
  DESTTYPE LIKE RSBOHDEST-DESTTYPE,
     END OF I_S_OHD.
DATA: I_T_OHD TYPE STANDARD TABLE OF I_S_OHD.
DATA: I_WA_OHD TYPE I_S_OHD.

TYPES: BEGIN OF I_S_OHDT.
TYPES:   OHDEST LIKE RSBOHDESTT-OHDEST,
  TXTLG LIKE RSBOHDESTT-TXTLG,
     END OF I_S_OHDT.
DATA: I_T_OHDT TYPE STANDARD TABLE OF I_S_OHDT.
DATA: I_WA_OHDT TYPE I_S_OHDT.

TYPES: BEGIN OF I_S_DTP.
TYPES:   DTP LIKE RSBKDTP-DTP,
SRC LIKE RSBKDTP-SRC,
TGT LIKE RSBKDTP-TGT,
  DTPTYPE LIKE RSBKDTP-DTPTYPE,
     END OF I_S_DTP.
DATA: I_T_DTP TYPE STANDARD TABLE OF I_S_DTP.
DATA: I_WA_DTP TYPE I_S_DTP.

TYPES: BEGIN OF I_S_AGGR.
TYPES:   AGGRUID LIKE RSDDAGGRDIR-AGGRUID,
```

```
  AGGRCUBE LIKE RSDDAGGRDIR-AGGRCUBE,
  INFOCUBE LIKE RSDDAGGRDIR-INFOCUBE,
     END OF I_S_AGGR.
DATA: I_T_AGGR TYPE STANDARD TABLE OF I_S_AGGR.
DATA: I_WA_AGGR TYPE I_S_AGGR.


DATA:
  H_APPL  TYPE OLE2_OBJECT,
  H_CELL  TYPE OLE2_OBJECT,
  H_WORK  TYPE OLE2_OBJECT,
  H_ZL TYPE OLE2_OBJECT,
  ILEN TYPE I,
  LJ TYPE I,
  TEMPLATE LIKE RLGRAP-FILENAME,
  TMPSTR(200),
  C_CHAR(19).

DATA: P_TIC LIKE  RLGRAP-FILENAME
            VALUE 'C:.xlt'.

DATA: LV_INDEX TYPE SY-TABIX.

********END - Declarations ****************************************


************************************************************************
START-OF-SELECTION.

  WRITE: /'If the template file is placed at', P_TIC. SKIP.
  WRITE: /'The program will get the list of Inactive Objects in the
system'.SKIP.
  WRITE: /'1. Info Objects'.
  WRITE: /'2. Info Providers'.
  WRITE: /'3. ODS'.
  WRITE: /'4. Open Hub'.
  WRITE: /'5. Transformations'.
  WRITE: /'6. Update Rules'.
  WRITE: /'7. Data Transfer Process'.
  WRITE: /'8. Transfer Structure'.
  WRITE: /'9. Multiproviders'.
  WRITE: /'10. Aggregates'.SKIP.
  WRITE: /'Choose CANCEL if you no not want this list or'.
  WRITE: 'file is not in the correct place else CONTINUE ...'.SKIP.
*  selection-screen pushbutton 25(15) CHOOSE.
  WRITE  /'CANCEL  ' COLOR 6.    "red
  CHOOSE = 'CANCEL'.
  HIDE CHOOSE.SKIP.
  WRITE / 'CONTINUE' COLOR 5.    "green
  CHOOSE = 'CONTINUE'.
  HIDE CHOOSE.SKIP.

AT LINE-SELECTION.

  CASE CHOOSE.
*...continue ...*
    WHEN 'CONTINUE'.
      WRITE: /'Save the Excel'.
*     PARAMETERS : P_IOBJ TYPE C AS CHECKBOX .
```

```
*...cancel...*
    WHEN 'CANCEL'.
      WRITE: 'Cancelled'.
      EXIT.
    WHEN OTHERS.
      EXIT.
  ENDCASE.
  CLEAR CHOOSE.
*******START- List of Inactive Infoproviders************************


  MOVE SY-DATUM TO C_CHAR.
  MOVE C_CHAR+4(2) TO TMPSTR+1(2).
  MOVE '-' TO TMPSTR+3(1).
  MOVE C_CHAR+6(2) TO TMPSTR+4(2).
  MOVE '-' TO TMPSTR+6(1).
  MOVE C_CHAR+0(4) TO TMPSTR+7(4).
  MOVE ' - ' TO TMPSTR+11(3).
  TEMPLATE = P_TIC.
  PERFORM PREPARE_EXCEL.
  TMPSTR+14(6) = 'ICB - '.
  TMPSTR+20(30) = 'Inactive Cubes'.
  CONDENSE TMPSTR.

  SELECT * FROM RSDCUBE
    INTO CORRESPONDING FIELDS OF TABLE I_T_CUBE
    WHERE OBJSTAT  = 'INA'
    AND OBJVERS NE 'D'.
    SORT I_T_CUBE BY INFOCUBE.
    DELETE ADJACENT DUPLICATES FROM I_T_CUBE.
    SORT I_T_CUBE BY CUBETYPE.
  SELECT * FROM RSDCUBET
    INTO CORRESPONDING FIELDS OF TABLE I_T_CUBET
    FOR ALL ENTRIES IN I_T_CUBE
    WHERE INFOCUBE = I_T_CUBE-INFOCUBE
    AND LANGU = 'EN'.
SORT I_T_CUBET BY INFOCUBE.
  CALL METHOD OF H_APPL 'Goto'
    EXPORTING
    #1 = 'C_Up'.
  PERFORM FILL_RANGE USING TMPSTR 'G_Cubedoc'.
*  if p_skip <> 'X'.
  LJ = 6.
  LOOP AT I_T_CUBE INTO I_WA_CUBE WHERE CUBETYPE NE 'A' and CUBETYPE NE 'M'.
    PERFORM FILL_CELL USING LJ  2 I_WA_CUBE-INFOCUBE.
    READ TABLE I_T_CUBET INTO I_WA_CUBET
    WITH KEY INFOCUBE = I_WA_CUBE-INFOCUBE
    BINARY SEARCH.
    IF SY-SUBRC = 0.
      PERFORM FILL_CELL USING LJ 3 I_WA_CUBET-TXTLG.
    ENDIF.
    CASE I_WA_CUBE-CUBETYPE.
      WHEN 'B'. TMPSTR = 'Standard InfoCube'.
      WHEN 'R'. TMPSTR = 'Remote InfoCube'.
      WHEN 'V'. TMPSTR = 'Virtual InfoProvider'.
    ENDCASE.
    PERFORM FILL_CELL USING LJ 4 TMPSTR.
    LJ = LJ + 1.
  ENDLOOP.
```

```
********END- List of Inactive Infoproviders************************

*******START- List of Inactive Aggregates************************


  MOVE SY-DATUM TO C_CHAR.
  MOVE C_CHAR+4(2) TO TMPSTR+1(2).
  MOVE '-' TO TMPSTR+3(1).
  MOVE C_CHAR+6(2) TO TMPSTR+4(2).
  MOVE '-' TO TMPSTR+6(1).
  MOVE C_CHAR+0(4) TO TMPSTR+7(4).
  MOVE ' - ' TO TMPSTR+11(3).
  TEMPLATE = P_TIC.

  TMPSTR+14(6) = 'AGGR - '.
  TMPSTR+20(30) = 'Inactive Aggregates'.
  CONDENSE TMPSTR.
SELECT * FROM RSDDAGGRDIR
    INTO CORRESPONDING FIELDS OF TABLE I_T_AGGR
    WHERE OBJSTAT  = 'INA'
    AND OBJVERS NE 'D'.
    SORT I_T_AGGR BY AGGRUID.
    DELETE ADJACENT DUPLICATES FROM I_T_AGGR.

  CALL METHOD OF H_APPL 'Goto'
    EXPORTING
    #1 = 'A_Up'.
  PERFORM FILL_RANGE USING TMPSTR 'G_Aggrdoc'.
*  if p_skip <> 'X'.
  LJ = 6.

  LOOP AT I_T_AGGR INTO I_WA_AGGR.
    PERFORM FILL_CELL USING LJ  2 I_WA_AGGR-AGGRCUBE.
      PERFORM FILL_CELL USING LJ 3 I_WA_AGGR-INFOCUBE.
    LJ = LJ + 1.
  ENDLOOP.
********END- List of Inactive Aggregates************************
*******START- List of Inactive Multiproviders************************


  MOVE SY-DATUM TO C_CHAR.
  MOVE C_CHAR+4(2) TO TMPSTR+1(2).
  MOVE '-' TO TMPSTR+3(1).
  MOVE C_CHAR+6(2) TO TMPSTR+4(2).
  MOVE '-' TO TMPSTR+6(1).
  MOVE C_CHAR+0(4) TO TMPSTR+7(4).
  MOVE ' - ' TO TMPSTR+11(3).
  TEMPLATE = P_TIC.

  TMPSTR+14(6) = 'MPRO - '.
  TMPSTR+20(30) = 'Inactive Multiproviders'.
  CONDENSE TMPSTR.

  CALL METHOD OF H_APPL 'Goto'
    EXPORTING
    #1 = 'M_Up'.
  PERFORM FILL_RANGE USING TMPSTR 'G_Mprodoc'.
*  if p_skip <> 'X'.
  LJ = 6.
```

```
  LOOP AT I_T_CUBE INTO I_WA_CUBE where CUBETYPE = 'M' .
    PERFORM FILL_CELL USING LJ  2 I_WA_CUBE-INFOCUBE.
    READ TABLE I_T_CUBET INTO I_WA_CUBET
    WITH KEY INFOCUBE = I_WA_CUBE-INFOCUBE
    BINARY SEARCH.
    IF SY-SUBRC = 0.
      PERFORM FILL_CELL USING LJ 3 I_WA_CUBET-TXTLG.
    ENDIF.


   TMPSTR = 'MultiProvider'.
    PERFORM FILL_CELL USING LJ 4 TMPSTR.
    LJ = LJ + 1.
  ENDLOOP.
********END- List of Inactive Multiproviders************************
********START - List of Inactive Infoobjects************************

  CLEAR TMPSTR.
  MOVE SY-DATUM TO C_CHAR.
  MOVE C_CHAR+4(2) TO TMPSTR+1(2).
  MOVE '-' TO TMPSTR+3(1).
  MOVE C_CHAR+6(2) TO TMPSTR+4(2).
  MOVE '-' TO TMPSTR+6(1).
  MOVE C_CHAR+0(4) TO TMPSTR+7(4).
  MOVE ' - ' TO TMPSTR+11(3).
  TEMPLATE = P_TIC.
  TMPSTR+14(6) = 'IOBJ - '.
  TMPSTR+20(30) = 'Inactive InfoObjects'.
  CONDENSE TMPSTR.

  SELECT * FROM RSDIOBJ
    INTO CORRESPONDING FIELDS OF TABLE I_T_IOBJ
    WHERE OBJSTAT  = 'INA'
    AND OBJVERS NE 'D'.

    SORT I_T_IOBJ BY IOBJNM.

    DELETE ADJACENT DUPLICATES FROM I_T_IOBJ.

    SORT I_T_IOBJ BY IOBJTP.

  SELECT * FROM RSDIOBJT
    INTO CORRESPONDING FIELDS OF TABLE I_T_IOBJT
    FOR ALL ENTRIES IN I_T_IOBJ
    WHERE IOBJNM = I_T_IOBJ-IOBJNM
    AND LANGU = 'EN'.

    SORT I_T_IOBJT BY IOBJNM.

  CALL METHOD OF H_APPL 'Goto'
    EXPORTING
    #1 = 'I_Up'.
  PERFORM FILL_RANGE USING TMPSTR 'G_Iobjdoc'.
*  if p_skip <> 'X'.
  LJ = 6.

  LOOP AT I_T_IOBJ INTO I_WA_IOBJ.
    PERFORM FILL_CELL USING LJ  2 I_WA_IOBJ-IOBJNM.
```

```
    READ TABLE I_T_IOBJT INTO I_WA_IOBJT
    WITH KEY IOBJNM = I_WA_IOBJ-IOBJNM
    BINARY SEARCH.
    IF SY-SUBRC = 0.
      PERFORM FILL_CELL USING LJ 3 I_WA_IOBJT-TXTLG.
    ENDIF.

    CASE I_WA_IOBJ-IOBJTP.

      WHEN 'CHA'. TMPSTR = 'Characteristic'.
      WHEN 'KYF'. TMPSTR = 'Key figure'.
      WHEN 'TIM'. TMPSTR = 'Time characteristic'.
      WHEN 'DPA'. TMPSTR = 'Data packet characteristic'.
      WHEN 'UNI'. TMPSTR = 'Unit of measurement'.
    ENDCASE.
    PERFORM FILL_CELL USING LJ 4 TMPSTR.
    LJ = LJ + 1.
  ENDLOOP.

********END- List of Inactive Infoobjects**************************

********START - List of Inactive Transformations**********************

  CLEAR TMPSTR.
  MOVE SY-DATUM TO C_CHAR.
  MOVE C_CHAR+4(2) TO TMPSTR+1(2).
  MOVE '-' TO TMPSTR+3(1).
  MOVE C_CHAR+6(2) TO TMPSTR+4(2).
  MOVE '-' TO TMPSTR+6(1).
  MOVE C_CHAR+0(4) TO TMPSTR+7(4).
  MOVE ' - ' TO TMPSTR+11(3).
  TEMPLATE = P_TIC.
  TMPSTR+14(6) = 'TRFN - '.
  TMPSTR+20(30) = 'Inactive Transformations'.
  CONDENSE TMPSTR.

  SELECT * FROM RSTRAN
    INTO CORRESPONDING FIELDS OF TABLE I_T_TRFN
    WHERE OBJSTAT  = 'INA'
    AND OBJVERS NE 'D'.

    SORT I_T_TRFN BY TRANID.

    DELETE ADJACENT DUPLICATES FROM I_T_TRFN.

  CALL METHOD OF H_APPL 'Goto'
  EXPORTING
    #1 = 'T_Up'.
  PERFORM FILL_RANGE USING TMPSTR 'G_Trfndoc'.
*  if p_skip <> 'X'.
  LJ = 6.

  LOOP AT I_T_TRFN INTO I_WA_TRFN.
    PERFORM FILL_CELL USING LJ  2 I_WA_TRFN-TRANID.
    PERFORM FILL_CELL USING LJ  3 I_WA_TRFN-SOURCENAME.
    PERFORM FILL_CELL USING LJ  4 I_WA_TRFN-TARGETNAME.
    LJ = LJ + 1.
```

```
  ENDLOOP.

********END- List of Inactive Transformations**************************

********START - List of Inactive Update Rules***********************

  CLEAR TMPSTR.
  MOVE SY-DATUM TO C_CHAR.
  MOVE C_CHAR+4(2) TO TMPSTR+1(2).
  MOVE '-' TO TMPSTR+3(1).
  MOVE C_CHAR+6(2) TO TMPSTR+4(2).
  MOVE '-' TO TMPSTR+6(1).
  MOVE C_CHAR+0(4) TO TMPSTR+7(4).
  MOVE ' - ' TO TMPSTR+11(3).
  TEMPLATE = P_TIC.
  TMPSTR+14(6) = 'UPDR - '.
  TMPSTR+20(30) = 'Inactive Update Rules'.
  CONDENSE TMPSTR.

  SELECT * FROM RSUPDINFO
    INTO CORRESPONDING FIELDS OF TABLE I_T_UPDR
    WHERE OBJSTAT  = 'INA'
    AND OBJVERS NE 'D'.

    SORT I_T_UPDR BY UPDID.

    DELETE ADJACENT DUPLICATES FROM I_T_UPDR.

  CALL METHOD OF H_APPL 'Goto'
    EXPORTING
    #1 = 'U_Up'.
  PERFORM FILL_RANGE USING TMPSTR 'G_Updrdoc'.
*  if p_skip <> 'X'.
  LJ = 6.

  LOOP AT I_T_UPDR INTO I_WA_UPDR.
    PERFORM FILL_CELL USING LJ  2 I_WA_UPDR-UPDID.
    PERFORM FILL_CELL USING LJ  3 I_WA_UPDR-ISOURCE.
    PERFORM FILL_CELL USING LJ  4 I_WA_UPDR-INFOCUBE.
    LJ = LJ + 1.
  ENDLOOP.

********END- List of Inactive Update Rules**************************

********START - List of Inactive ODS ******************************

  CLEAR TMPSTR.
  MOVE SY-DATUM TO C_CHAR.
  MOVE C_CHAR+4(2) TO TMPSTR+1(2).
  MOVE '-' TO TMPSTR+3(1).
  MOVE C_CHAR+6(2) TO TMPSTR+4(2).
  MOVE '-' TO TMPSTR+6(1).
  MOVE C_CHAR+0(4) TO TMPSTR+7(4).
  MOVE ' - ' TO TMPSTR+11(3).
  TEMPLATE = P_TIC.
  TMPSTR+14(6) = 'ODS - '.
  TMPSTR+20(30) = 'Inactive ODS'.
  CONDENSE TMPSTR.
```

```
  SELECT ODSOBJECT ODSOTYPE FROM RSDODSO
    INTO TABLE I_T_ODS_A_D
    WHERE OBJVERS NE 'D'.
SORT I_T_ODS_A_D BY ODSOBJECT.

DELETE ADJACENT DUPLICATES FROM I_T_ODS_A_D.

  SELECT ODSOBJECT FROM RSDODSOLOC
  INTO TABLE I_T_ODS
  WHERE OBJSTAT = 'INA'.

SORT I_T_ODS BY ODSOBJECT.

DELETE ADJACENT DUPLICATES FROM I_T_ODS.

SORT I_T_ODS BY ODSOTYPE.

  LOOP AT I_T_ODS INTO I_WA_ODS.
    LV_INDEX = SY-TABIX.
    READ TABLE I_T_ODS_A_D INTO I_WA_ODS_A_D
    WITH KEY ODSOBJECT = I_WA_ODS-ODSOBJECT
    BINARY SEARCH.
    IF SY-SUBRC <> 0.
      DELETE I_T_ODS WHERE ODSOBJECT = I_WA_ODS-ODSOBJECT.
    ENDIF.
  ENDLOOP.

  SELECT * FROM RSDODSOT
    INTO CORRESPONDING FIELDS OF TABLE I_T_ODST
    FOR ALL ENTRIES IN I_T_ODS
    WHERE ODSOBJECT = I_T_ODS-ODSOBJECT
    AND LANGU = 'EN'.

SORT I_T_ODST BY ODSOBJECT.

  CALL METHOD OF H_APPL 'Goto'
    EXPORTING
    #1 = 'OD_Up'.
  PERFORM FILL_RANGE USING TMPSTR 'G_Odsdoc'.
*  if p_skip <> 'X'.
  LJ = 6.

  LOOP AT I_T_ODS INTO I_WA_ODS.
    PERFORM FILL_CELL USING LJ  2 I_WA_ODS-ODSOBJECT.
    READ TABLE I_T_ODST INTO I_WA_ODST
   WITH KEY ODSOBJECT = I_WA_ODS-ODSOBJECT
   BINARY SEARCH.
    IF SY-SUBRC = 0.
      PERFORM FILL_CELL USING LJ 3 I_WA_ODST-TXTLG.
    ENDIF.
    CASE I_WA_ODS-ODSOTYPE.
      WHEN ' '. TMPSTR = 'Standard'.
      WHEN 'T'. TMPSTR = 'Direct Update'.
      WHEN 'W'. TMPSTR = 'Write-Optimized'.
    ENDCASE.
    PERFORM FILL_CELL USING LJ 4 TMPSTR.
    LJ = LJ + 1.
  ENDLOOP.
```

```
********END- List of Inactive ODS *********************************

********START- List of Inactive Open Hub************************
  MOVE SY-DATUM TO C_CHAR.
  MOVE C_CHAR+4(2) TO TMPSTR+1(2).
  MOVE '-' TO TMPSTR+3(1).
  MOVE C_CHAR+6(2) TO TMPSTR+4(2).
  MOVE '-' TO TMPSTR+6(1).
  MOVE C_CHAR+0(4) TO TMPSTR+7(4).
  MOVE ' - ' TO TMPSTR+11(3).
  TEMPLATE = P_TIC.
  TMPSTR+14(6) = 'OHD - '.
  TMPSTR+20(30) = 'Inactive Open Hub'.
  CONDENSE TMPSTR.

  SELECT * FROM RSBOHDEST
    INTO CORRESPONDING FIELDS OF TABLE I_T_OHD
    WHERE OBJSTAT  = 'INA'
    AND OBJVERS NE 'D'.

    SORT I_T_OHD BY OHDEST.

    DELETE ADJACENT DUPLICATES FROM I_T_OHD.

    SORT I_T_OHD BY DESTTYPE.

  SELECT * FROM RSBOHDESTT
    INTO CORRESPONDING FIELDS OF TABLE I_T_OHDT
    FOR ALL ENTRIES IN I_T_OHD
    WHERE OHDEST = I_T_OHD-OHDEST
    AND LANGU = 'EN'.

SORT I_T_OHDT BY OHDEST.

  CALL METHOD OF H_APPL 'Goto'
    EXPORTING
    #1 = 'O_Up'.
  PERFORM FILL_RANGE USING TMPSTR 'G_Ohddoc'.
*  if p_skip <> 'X'.
  LJ = 6.

  LOOP AT I_T_OHD INTO I_WA_OHD.
    PERFORM FILL_CELL USING LJ  2 I_WA_OHD-OHDEST.


    READ TABLE I_T_OHDT INTO I_WA_OHDT
    WITH KEY OHDEST = I_WA_OHD-OHDEST
    BINARY SEARCH.
    IF SY-SUBRC = 0.
      PERFORM FILL_CELL USING LJ 3 I_WA_OHDT-TXTLG.
    ENDIF.

    CASE I_WA_OHD-DESTTYPE.

      WHEN 'TAB'. TMPSTR = 'Database Table'.
      WHEN 'FILE'. TMPSTR = '.csv File'.
      WHEN 'RFC'. TMPSTR = 'RFC Interface'.
      WHEN 'LS'. TMPSTR = 'MultiProvider'.
```

```
      WHEN 'V'. TMPSTR = 'Logical System'.
    ENDCASE.
    PERFORM FILL_CELL USING LJ 4 TMPSTR.
    LJ = LJ + 1.
  ENDLOOP.
********END- List of Inactive Open Hub************************

********START- List of Inactive DTP*****************************
  MOVE SY-DATUM TO C_CHAR.
  MOVE C_CHAR+4(2) TO TMPSTR+1(2).
  MOVE '-' TO TMPSTR+3(1).
  MOVE C_CHAR+6(2) TO TMPSTR+4(2).
  MOVE '-' TO TMPSTR+6(1).
  MOVE C_CHAR+0(4) TO TMPSTR+7(4).
  MOVE ' - ' TO TMPSTR+11(3).
  TEMPLATE = P_TIC.
  TMPSTR+14(6) = 'DTP - '.
  TMPSTR+20(30) = 'Inactive DTP'.
  CONDENSE TMPSTR.

      SELECT A~DTP
            SRC
            TGT
        DTPTYPE
      INTO CORRESPONDING FIELDS OF TABLE I_T_DTP
      FROM RSBKDTPSTAT AS A
      INNER JOIN RSBKDTP AS B
      ON A~DTP = B~DTP WHERE A~OBJSTAT = 'INA'.


  CALL METHOD OF H_APPL 'Goto'
    EXPORTING
    #1 = 'D_Up'.
  PERFORM FILL_RANGE USING TMPSTR 'G_Dtpdoc'.
*  if p_skip <> 'X'.
  LJ = 6.

  LOOP AT I_T_DTP INTO I_WA_DTP.
    PERFORM FILL_CELL USING LJ  2 I_WA_DTP-DTP.
    PERFORM FILL_CELL USING LJ  3 I_WA_DTP-SRC.
    PERFORM FILL_CELL USING LJ  4 I_WA_DTP-TGT.

    CASE I_WA_DTP-DTPTYPE.
      WHEN ' '. TMPSTR = 'Standard (Can Be Scheduled)'.
      WHEN 'REAL'. TMPSTR = 'DTP for Real-Time Data Acquisition'.
      WHEN 'REMT'. TMPSTR = 'DTP for Direct Access'.
      WHEN 'EDTP'. TMPSTR = 'Error DTP'.
    ENDCASE.
    PERFORM FILL_CELL USING LJ 5 TMPSTR.
    LJ = LJ + 1.
  ENDLOOP.
********END- List of Inactive DTP*****************************
CLEAR TMPSTR.
  MOVE SY-DATUM TO C_CHAR.
  MOVE C_CHAR+4(2) TO TMPSTR+1(2).
  MOVE '-' TO TMPSTR+3(1).
  MOVE C_CHAR+6(2) TO TMPSTR+4(2).
  MOVE '-' TO TMPSTR+6(1).
  MOVE C_CHAR+0(4) TO TMPSTR+7(4).
```

```
  MOVE ' - ' TO TMPSTR+11(3).
  TEMPLATE = P_TIC.
  TMPSTR+14(6) = 'TRU - '.
  TMPSTR+20(30) = 'Inactive Transfer Structures'.
  CONDENSE TMPSTR.

  SELECT * FROM RSTS
    INTO CORRESPONDING FIELDS OF TABLE I_T_TRU
    WHERE OBJSTAT  = 'INA'
    AND OBJVERS NE 'D'.

    SORT I_T_TRU BY TRANSTRU.

    DELETE ADJACENT DUPLICATES FROM I_T_TRU.

    SORT I_T_TRU BY LOGSYS DESCENDING.

  CALL METHOD OF H_APPL 'Goto'
    EXPORTING
    #1 = 'S_Up'.
  PERFORM FILL_RANGE USING TMPSTR 'G_Trudoc'.
*  if p_skip <> 'X'.
  LJ = 6.

  LOOP AT I_T_TRU INTO I_WA_TRU.
    PERFORM FILL_CELL USING LJ  2 I_WA_TRU-LOGSYS.
    PERFORM FILL_CELL USING LJ  3 I_WA_TRU-TRANSTRU.
    LJ = LJ + 1.
  ENDLOOP.


PERFORM EXCEL_RELEASE.
********END- List of Inactive Transformations**************************


********START OF FORMS*********************************************

*---------------------------------------------------------------------*
*       FORM FILL_CELL                                                *
*---------------------------------------------------------------------*
*       ........                                                      *
*---------------------------------------------------------------------*
*  -->  I                                                             *
*  -->  J                                                             *
*  -->  VAL                                                           *
*---------------------------------------------------------------------*
FORM FILL_CELL USING I J VAL.
  IF NOT ( VAL IS INITIAL ).
    CALL METHOD OF H_APPL 'Cells' = H_ZL
      EXPORTING
      #1 = I
      #2 = J.
    SET PROPERTY OF H_ZL 'Value' = VAL .
  ENDIF.
ENDFORM.                    "fill_cell
*---------------------------------------------------------------------*
*       FORM PREPARE_EXCEL                                            *
*---------------------------------------------------------------------*
*       ........                                                      *
```

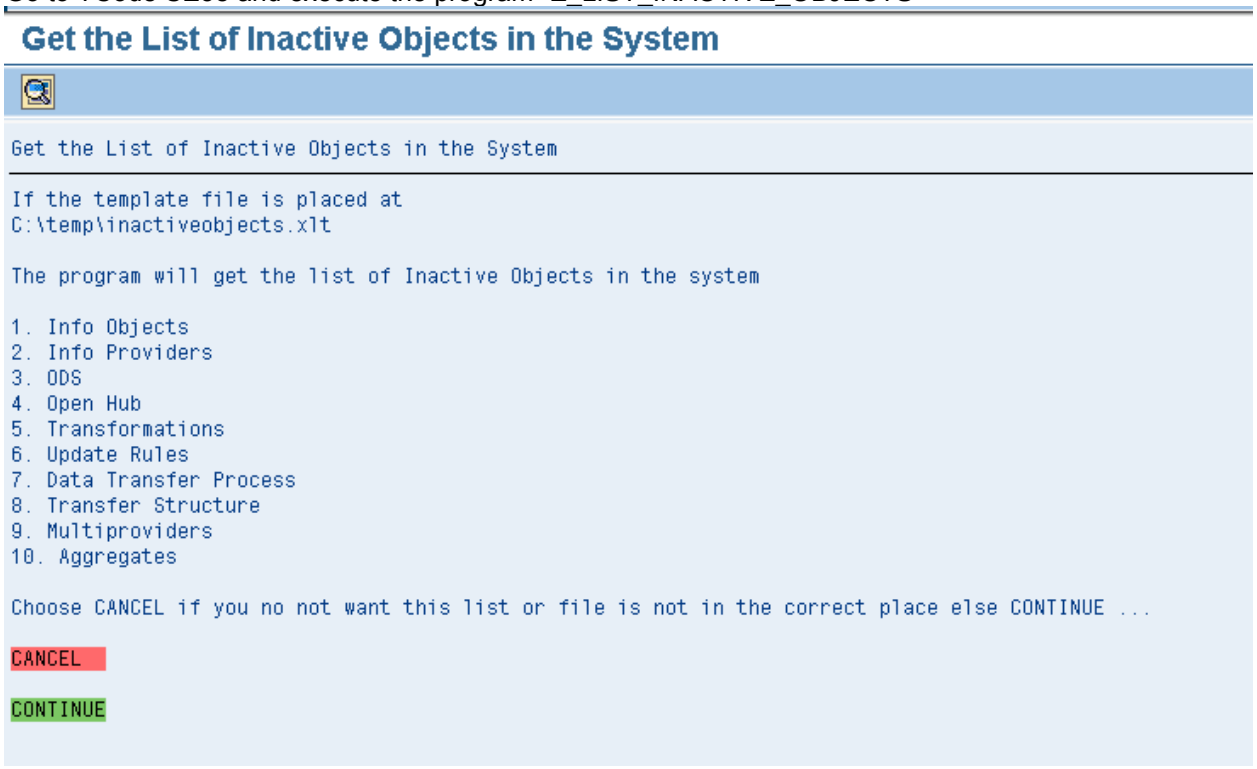
```
*---------------------------------------------------------------------*
FORM PREPARE_EXCEL.
  CREATE OBJECT H_APPL 'EXCEL.APPLICATION'.
  IF SY-SUBRC NE 0. MESSAGE I002 WITH SY-MSGLI. ENDIF.
  SET PROPERTY OF H_APPL  'VISIBLE' = 1.
  CALL METHOD OF H_APPL 'WORKBOOKS' = H_WORK.
  CALL METHOD OF H_WORK 'OPEN'
    EXPORTING
    #1 = TEMPLATE.
ENDFORM.                    "prepare_excel
*---------------------------------------------------------------------*
*       FORM EXCEL_RELEASE                                            *
*---------------------------------------------------------------------*
*       ........                                                      *
*---------------------------------------------------------------------*
FORM EXCEL_RELEASE.
  CALL METHOD OF H_APPL 'SendKeys'
    EXPORTING
    #1 = '^t'.
  CALL METHOD OF H_APPL 'SendKeys'
    EXPORTING
    #1 = '^x'.
  FREE OBJECT H_CELL.
  FREE OBJECT H_WORK.
  FREE OBJECT H_APPL.
ENDFORM.                    "excel_release
*---------------------------------------------------------------------*
*       FORM FILL_RANGE                                               *
*---------------------------------------------------------------------*
*       ........                                                      *
*---------------------------------------------------------------------*
*  -->  CVALUE                                                        *
*  -->  CRANGE                                                        *
*---------------------------------------------------------------------*
FORM FILL_RANGE USING
   CVALUE
   CRANGE.
  IF NOT ( CVALUE IS INITIAL ).
    CALL METHOD OF H_APPL 'Goto'
      EXPORTING
      #1 = CRANGE.
    GET PROPERTY OF  H_APPL 'ACTIVECELL' = H_CELL.
    SET PROPERTY OF  H_CELL 'VALUE'      = CVALUE.
  ENDIF.
ENDFORM.                    "fill_range


*******END OF FORMS*********************************************
```

Step 5:

Save and Activate

Using the program

Go to TCode SE38 and execute the program “Z_LIST_INACTIVE_OBJECTS”



This program writes the output in a xls for which the below template needs to be placed at:
C:\temp
The template name should be inactiveobjects.xlt

On executing this program and xls gets populated with the list of inactive objects.

Related Content

www.help.sap.com
www.sdn.sap.com
www.service.sap.com

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.