informatica_powercenter 资料库元数据查询

——Informatica PowerCenter 培训系列

TABLE OF CONTENTS

1 OVERVIEW

Below Steps are intended for informatica development team to check if their etl code is as per ETL Standards', developer team need to have read only access to informatica repository tables and Views. Please suffix your respective schema names for all your table / views names in below queries.

2 FOLDER

2.1 LIST FOLDER DETAILS

```
SELECT SUBJ_NAME,SUBJ_DESC FROM OPB_SUBJECT ORDER BY 1,2
```

2.2 LIST OF SHARED FOLDERS

```
 SELECT SUBJ_NAME,SUBJ_DESC FROM OPB_SUBJECT
WHERE IS_SHARED <>0
ORDER BY 1,2
```

2.3 LIST OF USERS AND GROUPS HAVING PRIVILEGES' ON FOLDERS

```
 SELECT subj.subj_name folder_name, user_group.NAME user_name,
DECODE (obj_access.user_type, 1, 'USER', 2, 'GROUP') TYPE,
CASE WHEN ((obj_access.permissions - (obj_access.user_id + 1)) IN (8, 16))THEN 'READ'
WHEN ((obj_access.permissions - (obj_access.user_id + 1)) IN (10, 20))THEN 'READ & EXECUTE'
WHEN ((obj_access.permissions - (obj_access.user_id + 1)) IN (12, 24))THEN 'READ & WRITE'
WHEN ((obj_access.permissions - (obj_access.user_id + 1)) IN (14, 28))THEN 'READ, WRITE & EXECUTE'
ELSE 'NO PERMISSIONS'
END permissions
FROM opb_object_access obj_access,opb_subject subj,opb_user_group user_group
WHERE obj_access.object_type = 29
AND obj_access.object_id = subj.subj_id
AND obj_access.user_id = user_group.ID
AND obj_access.user_type = user_group.TYPE
 --  and user_group.NAME not in ('Admin','READ_ONLY','Administrator','Administrators')
order by 1,2,3
```

3 SOURCE

3.1 LIST OF SOURCE TABLES

```
 SELECT
B.SUBJ_NAME,
C.DBDNAM,
D.DBTYPE_NAME,
A.SOURCE_NAME AS TABLE_NAME,
A.FILE_NAME SCHEMA_NAME,
A.OWNERNAME
FROM
OPB_SRC A,OPB_SUBJECT B, OPB_DBD C,OPB_MMD_DBTYPE D
WHERE A.SUBJ_ID = B.SUBJ_ID
AND A.DBDID = C.DBDID
AND C.DBTYPE = D.DBTYPE_ID
--AND A.SOURCE_NAME <> A.FILE_NAME
ORDER BY 1,2,3,4,5
```

3.2 LIST AND COUNT OF TABLES IN EACH FOLDER BY DB TYPE

```
SELECT
B.SUBJ_NAME,
D.DBTYPE_NAME,
count(*)
FROM
OPB_SRC A,OPB_SUBJECT B, OPB_DBD C,OPB_MMD_DBTYPE D
WHERE A.SUBJ_ID = B.SUBJ_ID
AND A.DBDID = C.DBDID
AND C.DBTYPE = D.DBTYPE_ID
--AND A.SOURCE_NAME <> A.FILE_NAME
group by B.SUBJ_NAME,D.DBTYPE_NAME
order by 1,2,3
```

3.3 LIST AND COUNT OF TABLES OVERALL USED

```
SELECT SOURCE_NAME, COUNT(SOURCE_NAME) FROM REP_TBL_MAPPING
GROUP BY SOURCE_NAME
ORDER BY 1,2 ASC
```

3.4 LIST OF SOURCE TABLES USED IN MAPPING

```
SELECT SUBJECT_AREA,SOURCE_NAME,MAPPING_NAME FROM REP_SRC_MAPPING
ORDER BY 1,2,3
```

3.5 LIST OF SOURCE TABLES USING AS SHORTCUTS

```
 SELECT DISTINCT
B.SUBJ_NAME,
C.DBDNAM,
D.DBTYPE_NAME,
A.SOURCE_NAME AS TABLE_NAME,
A.FILE_NAME SCHEMA_NAME,
A.OWNERNAME
FROM
OPB_SRC A,OPB_SUBJECT B, OPB_DBD C,OPB_MMD_DBTYPE D
WHERE A.SUBJ_ID = B.SUBJ_ID
AND A.DBDID = C.DBDID
AND C.DBTYPE = D.DBTYPE_ID
--AND A.SOURCE_NAME <> A.FILE_NAME
and A.SOURCE_NAME like 'sc_%'
ORDER BY 1,2,3,4,5
```

4 TARGET

4.1 LIST OF TARGET TABLES

```
 SELECT B.SUBJ_NAME,
A.TARGET_NAME,
DECODE(A.DBTYPE,
0,'VSAM',
1,'IMS',
2,'Sybase',
3,'Oracle',
4,'Informix',
5,'Microsoft SQL Server',
6,'DB2',
7,'Flat File',
8,'ODBC',
9,'SAP BW',
10,'PeopleSoft',
11,'SAP R/3',
12,'XML',
13,'MQSeries',
14,'Siebel',
15,'Teradata' ) as DB_TYPE
FROM
OPB_TARG A,
OPB_SUBJECT B
WHERE A.SUBJ_ID = B.SUBJ_ID
ORDER BY 1,2,3
```

4.2 LIST AND COUNT OF TABLES IN EACH FOLDER BY DB TYPE

```
 SELECT B.SUBJ_NAME,
DECODE(A.DBTYPE,
0,'VSAM',
1,'IMS',
2,'Sybase',
3,'Oracle',
4,'Informix',
5,'Microsoft SQL Server',
6,'DB2',
7,'Flat File',
8,'ODBC',
9,'SAP BW',
10,'PeopleSoft',
11,'SAP R/3',
12,'XML',
13,'MQSeries',
14,'Siebel',
15,'Teradata' ) as DB_TYPE,
count(*)
FROM
OPB_TARG A,
OPB_SUBJECT B
WHERE A.SUBJ_ID = B.SUBJ_ID
GROUP BY B.SUBJ_NAME,A.DBTYPE
ORDER BY 1,2
```

4.3 LIST AND COUNT OF TABLE OVERALL USED

```
SELECT SOURCE_NAME, COUNT(SOURCE_NAME) FROM REP_TBL_MAPPING
GROUP BY SOURCE_NAME
ORDER BY 1,2 ASC
```

5 TRANSFORMATION

5.1 LIST OF FILER TRANSFORMATIONS

```
SELECT SUBSTR(WIDGET_NAME,1,3), COUNT(WIDGET_NAME)
FROM REP_ALL_TRANSFORMS
WHERE WIDGET_TYPE_NAME = 'Filter'
GROUP BY SUBSTR(WIDGET_NAME,1,3)
```

5.2 LIST OF SEQUENCE TRANSFORMATIONS

```
SELECT DISTINCT SUBJECT_AREA, PARENT_WIDGET_NAME FROM REP_ALL_TRANSFORMS WHERE WIDGET_TYPE
_NAME ='Sequence' ORDER BY1,2
```

5.3 LIST OF TABLES USED AS LOOKUPS

```
SELECT DISTINCT
B.PARENT_SUBJECT_AREA AS FOLDER_NAME,
C.ATTR_VALUE AS TABLE_NAME,A.INSTANCE_NAME AS TRANSFORMATION_NAME, A.WIDGET_TYPE_NAME AST
RANSFORMATION_TYPE,B.MAPPING_NAME
FROM
REP_WIDGET_INST A INNER JOIN REP_ALL_MAPPINGS B ON A.MAPPING_ID = B.MAPPING_ID INNER JOIN
REP_WIDGET_ATTR C ON A.WIDGET_ID = C.WIDGET_ID
WHERE
C.ATTR_DESCRIPTION LIKE 'Lookup source table'
ORDER BY 1,2,3,4,5
```

5.4 LIST OF TRANSFORMATIONS USING SQL OVERRIDES

```
SELECT DISTINCT
d.subject_area AS Folder, d.mapping_name, a.widget_type_name AS Transformation_Type,
a.instance_name as Transformation_Name, b.attr_name, b.attr_value, c.session_name
FROM
REP_WIDGET_INST a, REP_WIDGET_ATTR b, REP_LOAD_SESSIONS c, REP_ALL_MAPPINGS d
WHERE b.widget_id = a. widget_id
AND b.widget_type = a. widget_type
AND b.widget_type in (3, 11)
AND c.mapping_id = a.mapping_id
AND d.mapping_id = a.mapping_id
AND b.attr_id= 1
AND b.attr_datatype=2 and b.attr_type=3
ORDER BY d.subject_area, d.mapping_name
```

5.5 LIST ALL TRANSFORMATIONS

```
 SELECT DISTINCT version_subject.subject_area "FOLDER_NAME",   version_props.object_name "OBJECT_NAME"
,
CASE
WHEN version_props.object_type = 1 THEN 'Source Definition'              ELSE CASE
WHEN version_props.object_type = 2 THEN 'Target Definition'              ELSE CASE
WHEN version_props.object_type = 3 THEN 'Source Qualifier'               ELSE CASE
WHEN version_props.object_type = 4 THEN 'Update Strategy'                ELSE CASE
WHEN version_props.object_type = 5 THEN 'Expression'                     ELSE CASE
WHEN version_props.object_type = 6 THEN 'Stored Procedure'               ELSE CASE
WHEN version_props.object_type = 7 THEN 'Sequence'                 ELSE CASE
WHEN version_props.object_type = 8 THEN 'External Procedure'             ELSE CASE
WHEN version_props.object_type = 9 THEN 'Aggregator'                     ELSE CASE
WHEN version_props.object_type = 10 THEN 'Filter'                                   ELSE CASE
WHEN version_props.object_type = 11 THEN 'Lookup Procedure'                ELSE CASE
WHEN version_props.object_type = 12 THEN 'Joiner'                           ELSE CASE
WHEN version_props.object_type = 13 THEN 'Procedure'               ELSE CASE
WHEN version_props.object_type = 14 THEN 'Normalizer'              ELSE CASE
WHEN version_props.object_type = 16 THEN 'Merger'                           ELSE CASE
WHEN version_props.object_type = 17 THEN 'Pivot'                      ELSE CASE
WHEN version_props.object_type = 18 THEN 'Session Obsolete'                ELSE CASE
WHEN version_props.object_type = 19 THEN 'Batch'                             ELSE CASE
WHEN version_props.object_type = 20 THEN 'Shortcut'               ELSE CASE
WHEN version_props.object_type = 21 THEN 'Mapping'                ELSE CASE
WHEN version_props.object_type = 26 THEN 'Rank'                   ELSE CASE
WHEN version_props.object_type = 27 THEN 'Star Schema'            ELSE CASE
WHEN version_props.object_type = 28 THEN 'Folder Version'         ELSE CASE
WHEN version_props.object_type = 29 THEN 'Folder'                           ELSE CASE
WHEN version_props.object_type = 30 THEN 'Cube'                   ELSE CASE
WHEN version_props.object_type = 31 THEN 'Dimension'              ELSE CASE
WHEN version_props.object_type = 32 THEN 'Level'                  ELSE CASE
WHEN version_props.object_type = 33 THEN 'Hierarchy'              ELSE CASE
WHEN version_props.object_type = 34 THEN 'Fact Table'             ELSE CASE
WHEN version_props.object_type = 35 THEN 'General Object'                    ELSE CASE
WHEN version_props.object_type = 36 THEN 'FTP Object'             ELSE CASE
WHEN version_props.object_type = 37 THEN 'Oracle External Loader Object'     ELSE CASE
WHEN version_props.object_type = 38 THEN 'Informix External Loader Object'             ELSE CASE
WHEN version_props.object_type = 39 THEN 'Sybase IQ External Loader Object'     ELSE CASE
WHEN version_props.object_type = 54 THEN 'Sybase IQ 12 External Loader Object'            ELSE CASE
WHEN version_props.object_type = 53 THEN 'Tera Data External Loader Object'     ELSE CASE
WHEN version_props.object_type = 40 THEN 'File Object'                  ELSE CASE
WHEN version_props.object_type = 41 THEN 'Server Object'              ELSE CASE
WHEN version_props.object_type = 42 THEN 'Database Object'                       ELSE CASE
WHEN version_props.object_type = 43 THEN 'Repository'                   ELSE CASE
WHEN version_props.object_type = 44 THEN 'Mapplet'                        ELSE CASE
WHEN version_props.object_type = 45 THEN 'Application Source Qualifier'        ELSE CASE
WHEN version_props.object_type = 46 THEN 'Input Transformation'              ELSE CASE
WHEN version_props.object_type = 47 THEN 'Output Transformation'             ELSE CASE
WHEN version_props.object_type = 50 THEN 'Advanced External Procedure'        ELSE CASE
WHEN version_props.object_type = 48 THEN 'Business Component Framework'   ELSE CASE
WHEN version_props.object_type = 49 THEN 'Business Component'               ELSE CASE
WHEN version_props.object_type = 51 THEN 'SAP Structure'              ELSE CASE
WHEN version_props.object_type = 52 THEN 'SAP Function'               ELSE CASE
WHEN version_props.object_type = 15 THEN 'Router'                           ELSE CASE
WHEN version_props.object_type = 55 THEN 'XML Source Qualifier'             ELSE CASE
WHEN version_props.object_type = 56 THEN 'MQ Source Qualifier'                ELSE CASE
WHEN version_props.object_type = 57 THEN 'MQ Connection Object'            ELSE CASE
WHEN version_props.object_type = 58 THEN 'Command'               ELSE CASE
WHEN version_props.object_type = 59 THEN 'Decision'                 ELSE CASE
WHEN version_props.object_type = 60 THEN 'Event Wait'               ELSE CASE
WHEN version_props.object_type = 61 THEN 'Event Raise'              ELSE CASE
WHEN version_props.object_type = 62 THEN 'Start'                    ELSE CASE
WHEN version_props.object_type = 63 THEN 'Abort'                             ELSE CASE
WHEN version_props.object_type = 64 THEN 'Stop'                       ELSE CASE
WHEN version_props.object_type = 65 THEN 'Email'                             ELSE CASE
WHEN version_props.object_type = 66 THEN 'Timer'                            ELSE CASE
WHEN version_props.object_type = 67 THEN 'Assignment'              ELSE CASE
WHEN version_props.object_type = 68 THEN 'Session'                           ELSE CASE
WHEN version_props.object_type = 69 THEN 'Scheduler'                ELSE CASE
WHEN version_props.object_type = 70 THEN 'Worklet'                   ELSE CASE
WHEN version_props.object_type = 71 THEN 'Workflow'                 ELSE CASE
WHEN version_props.object_type = 72 THEN 'SessionConfig'            ELSE CASE
WHEN version_props.object_type = 73 THEN 'Relational'               ELSE CASE
WHEN version_props.object_type = 74 THEN 'Application'              ELSE CASE
WHEN version_props.object_type = 75 THEN 'FTP'                       ELSE CASE
WHEN version_props.object_type = 76 THEN 'External Loader'                    ELSE CASE
WHEN version_props.object_type = 77 THEN 'Queue'                            ELSE CASE
WHEN version_props.object_type = 78 THEN 'Reader'                           ELSE CASE
WHEN version_props.object_type = 79 THEN 'Writer'                           ELSE CASE
WHEN version_props.object_type = 80 THEN 'Sorter'                           ELSE CASE
WHEN version_props.object_type = 81 THEN 'Vendor'                           ELSE CASE
WHEN version_props.object_type = 84 THEN 'App Multi-Group Source Qualifier'  ELSE CASE
WHEN version_props.object_type = 91 THEN 'Control'                           ELSE CASE
WHEN version_props.object_type = 92 THEN 'Transaction Control'              ELSE CASE
WHEN version_props.object_type = 97 THEN 'Custom Transformation'            ELSE CASE
WHEN version_props.object_type = 93 THEN 'Query'                            ELSE CASE
WHEN version_props.object_type = 94 THEN 'Deployment Group'                 ELSE CASE
WHEN version_props.object_type = 95 THEN 'Label'                            ELSE CASE
WHEN version_props.object_type = 96 THEN 'Deployed Deployment Group'        ELSE CASE
WHEN version_props.object_type = 98 THEN 'Server Grid'              ELSE CASE
WHEN version_props.object_type = 99 THEN 'Profiling Ruleset'                ELSE CASE
WHEN version_props.object_type = 100 THEN 'Template Extension'              ELSE CASE
WHEN version_props.object_type = 101 THEN 'Global Profile Resource'         ELSE CASE
WHEN version_props.object_type = 102 THEN 'Web Services Hub'                ELSE CASE
WHEN version_props.object_type = 103 THEN 'Lookup Extension'                ELSE CASE
WHEN version_props.object_type = 105 THEN 'Service Level'           ELSE CASE
WHEN version_props.object_type = 106 THEN 'User Defined Function' ELSE 'Shortcut'
  END     END     END     END     END     END     END     END     END     END     EN
D     END     END     END   END     END     END     END     END     END     END
END     END     END     END     END     END     END     END     END     END     END
END     END     END     END     END     END     END     END     END     END     END
    END     END     END     END     END     END     END     END     END     END     END
      END     END     END     END     END     END     END     END     END     END     E
ND     END     END     END     END     END     END     END     END     END     END
 END     END     END     END     END     END     END     END     END     END     END
    END     END     END     END     END     END

END "OBJECT_TYPE"
FROM rep_users version_users,
rep_version_props version_props,
rep_reposit_info version_reposit_info,
rep_subject version_subject
WHERE (version_props.user_id = version_users.user_id
AND version_props.object_id <> version_reposit_info.repository_id
AND version_props.subject_id = version_subject.subject_id
)
ORDER BY 3, 1
```

5.6 LIST ALL EXPRESSION TRANSFORMATIONS USING 'CONCAT' FUNCTION

```
SELECT DISTINCT REP_ALL_MAPPINGS.SUBJECT_AREA, REP_ALL_MAPPINGS.MAPPING_NAME,            REP_
WIDGET_INST.WIDGET_TYPE_NAMEAS TRANSFORMATION_TYPE,            REP_WIDGET_INST.INSTANCE_NA
MEAS TRANSFORMATION_NAME,            REP_WIDGET_FIELD.FIELD_NAME AS PORT_NAME,
CASE
 WHEN REP_WIDGET_FIELD.PORTTYPE = 1 THEN 'I'
 WHEN REP_WIDGET_FIELD.PORTTYPE = 2 THEN 'O'
 WHEN REP_WIDGET_FIELD.PORTTYPE = 3 THEN 'IO'
 WHEN REP_WIDGET_FIELD.PORTTYPE = 32 THEN 'V'
```

```
END AS PORT_TYPE,
REP_WIDGET_FIELD.EXPRESSION
FROM REP_WIDGET_INST, REP_WIDGET_FIELD, REP_ALL_MAPPINGS
WHERE REP_WIDGET_INST.WIDGET_ID = REP_WIDGET_FIELD.WIDGET_ID
AND REP_WIDGET_INST.MAPPING_ID = REP_ALL_MAPPINGS.MAPPING_ID
AND REP_WIDGET_INST.WIDGET_TYPE = 5
AND REP_WIDGET_FIELD.EXPRESSION LIKE '%CONCAT%'
ORDER BY 1
```

5.7 LIST OF ALL PORT DETAILS OF AN EXPRESSION TRANSFORMATIONS

```
SELECT S.SUBJ_NAME, W.WIDGET_NAME, F.FIELD_ID,F.FIELD_NAME,E.VERSION_NUMBER, E.EXPRESSION FROM
OPB_WIDGET W,OPB_SUBJECT S, OPB_WIDGET_FIELD F, OPB_WIDGET_EXPR R, OPB_EXPRESSION E
WHERE W.SUBJECT_ID=S.SUBJ_ID AND W.WIDGET_ID=F.WIDGET_ID
AND W.WIDGET_ID=R.WIDGET_ID AND F.FIELD_ID=R.OUTPUT_FIELD_ID
AND W.WIDGET_ID=E.WIDGET_ID AND R.EXPR_ID=E.EXPR_ID
AND W.VERSION_NUMBER = F.VERSION_NUMBER
AND F.VERSION_NUMBER = R.VERSION_NUMBER
AND R.VERSION_NUMBER = E.VERSION_NUMBER
AND W.IS_VISIBLE = 1
AND W.WIDGET_NAME LIKE 'EXP_%'
ORDER BY 1,2,3
```

5.8 LIST OF ALL EXPRESSION TRANSFORMATION PORT LINKS

```
 SELECT DISTINCT S.SUBJ_NAME, WF.INSTANCE_NAME ||'.'|| F.FIELD_NAME
FROM_NAME, F.FIELD_ORDER AS EXP_PORT_ORDER,
WT.INSTANCE_NAME ||'.'|| T.FIELD_NAME TO_NAME,   T.FIELD_ORDER
FROM OPB_WIDGET Z, OPB_WIDGET_INST WF, OPB_WIDGET_INST WT,
      OPB_WIDGET_FIELD F, OPB_WIDGET_FIELD T, OPB_WIDGET_DEP D, OPB_SUBJECT S
WHERE Z.SUBJECT_ID = S.SUBJ_ID
AND Z.IS_VISIBLE = 1
AND Z.WIDGET_ID = F.WIDGET_ID
AND Z.WIDGET_ID = WF.WIDGET_ID
AND Z.RU_VERSION_NUMBER = WF.VERSION_NUMBER
AND WF.REF_VERSION_NUMBER = F.VERSION_NUMBER
AND WF.VERSION_NUMBER = D.VERSION_NUMBER
AND WF.MAPPING_ID = D.MAPPING_ID
AND WF.INSTANCE_ID = D.FROM_INSTANCE_ID
AND F.FIELD_ID = D.FROM_FIELD_ID
AND D.TO_INSTANCE_ID = WT.INSTANCE_ID
AND D.TO_FIELD_ID = T.FIELD_ID
AND D.MAPPING_ID = WT.MAPPING_ID
AND D.VERSION_NUMBER = WT.VERSION_NUMBER
AND WT.WIDGET_ID = T.WIDGET_ID
AND WT.REF_VERSION_NUMBER = T.VERSION_NUMBER
--AND Z.WIDGET_NAME LIKE 'EXP_%'
AND S.SUBJ_NAME = :FOLDER_NAME
AND WF.INSTANCE_NAME = :EXP_NAME
ORDER BY 1,2,3
```

5.9 LIST OF LKP TRANSFORMATION PORT LINKS USED IN ALL MAPPINGS

```
SELECT DISTINCT OPB_SUBJECT.SUBJ_NAME,OPB_MAPPING.MAPPING_NAME,
OPB_WIDGET_FIELD.FIELD_NAME FIELD_NAME,
OPB_EXPRESSION.EXPRESSION EXPRESSION
FROM OPB_WIDGET_EXPR, OPB_EXPRESSION, OPB_WIDGET_FIELD, REP_FLD_DATAT
YPE, OPB_WIDGET,OPB_SUBJECT,OPB_WIDGET_INST,OPB_MAPPING
WHERE
OPB_WIDGET_FIELD.WIDGET_ID = OPB_WIDGET.WIDGET_ID
AND OPB_WIDGET.SUBJECT_ID = OPB_SUBJECT.SUBJ_ID
AND OPB_WIDGET_INST.WIDGET_ID = OPB_WIDGET.WIDGET_ID
AND OPB_MAPPING.MAPPING_ID = OPB_WIDGET_INST.MAPPING_ID
AND OPB_WIDGET_FIELD.VERSION_NUMBER = OPB_WIDGET.VERSION_NUMBER
AND OPB_WIDGET.IS_VISIBLE = 1
AND OPB_WIDGET_FIELD.WIDGET_ID= OPB_WIDGET_EXPR.WIDGET_ID
AND OPB_WIDGET_FIELD.FIELD_ID= OPB_WIDGET_EXPR.OUTPUT_FIELD_ID
AND OPB_WIDGET_EXPR.WIDGET_ID=OPB_EXPRESSION.WIDGET_ID
AND OPB_WIDGET_EXPR.EXPR_ID=OPB_EXPRESSION.EXPR_ID
AND OPB_EXPRESSION.LINE_NO = 1
AND OPB_WIDGET_EXPR.VERSION_NUMBER = OPB_EXPRESSION.VERSION_NUMBER
AND OPB_WIDGET_EXPR.VERSION_NUMBER = OPB_WIDGET_FIELD.VERSION_NUMBER
--AND OPB_SUBJECT.SUBJ_NAME = 'FOLDER_NAME'
--AND OPB_MAPPING.MAPPING_NAME = 'MAPPING_NAME'
AND UPPER(EXPRESSION) LIKE '%LKP_ACCT_B%'
ORDER BY 1,2
```

6 MAPPING

6.1 LIST MAPPING NAMES

```
 SELECT   SUBJECT_AREA, PARENT_MAPPING_NAME
    FROM REP_ALL_MAPPINGS
ORDER BY 1, 2
```

6.2 LIST TOTAL COUNT OF MAPPINGS

```
 SELECT   SUBJECT_AREA, COUNT(PARENT_MAPPING_NAME) AS TOTAL_MAPPINGS
    FROM REP_ALL_MAPPINGS
      GROUP BY SUBJECT_AREA
```

```
ORDER BY 1, 2
```

6.3 LIST LAST SAVED USER FOR A MAPPING

```
 SELECT REP_SUBJECT.SUBJECT_AREA "FOLDER",REP_VERSION_PROPS.OBJECT_NAME"MAPPING",REP_USERS.US
ER_NAME,REP_VERSION_PROPS.LAST_SAVED
FROM REP_USERS,REP_VERSION_PROPS,REP_SUBJECT
WHERE REP_USERS.USER_ID=REP_VERSION_PROPS.USER_ID
 AND REP_VERSION_PROPS.OBJECT_TYPE IN (21)
 --AND REP_SUBJECT.SUBJECT_AREA = 'FOLDER_NAME'
 AND REP_SUBJECT.SUBJECT_ID = REP_VERSION_PROPS.SUBJECT_ID
ORDER BY 1,2,3,4
```

6.4 LIST MAPPING PARAMETERS AND VARIABLES

```
select distinct rep_reposit_info.repository_name, rep_all_mappings.subject_area
as folder_name, rep_all_mappings.mapping_name as object_name,
case when opb_map_parmvar.pv_flag = 2 then 'Mapping Parameter' else
case when opb_map_parmvar.pv_flag = 3 then 'Mapping Variable' end
end as parameter_type, opb_map_parmvar.pv_name as parameter_name,
opb_map_parmvar.pv_default as parameter_value,
opb_map_parmvar.pv_desc as description
from rep_all_mappings, opb_map_parmvar, rep_reposit_info
where rep_all_mappings.mapping_id = opb_map_parmvar.mapping_id
```

6.5 LIST ALL THE MAPPINGS USING PARALLEL HINTS

```
SELECT S.SUBJ_NAME, M.MAPPING_NAME, W.WIDGET_NAME, A.WIDGET_ID, W.VERSION_NUMBER,
SUBSTR(A.ATTR_VALUE, 1, 60) ATTR_VALUE
FROM OPB_WIDGET_ATTR A, OPB_WIDGET W, OPB_SUBJECT S, OPB_WIDGET_INST I, OPB_MAPPING M
WHERE A.WIDGET_ID = W.WIDGET_ID
AND W.IS_VISIBLE = 1
AND A.VERSION_NUMBER = W.VERSION_NUMBER
AND A.WIDGET_TYPE IN(2, 3, 11) --Limit to Src/Tgt/Lkp Transformations
AND W.WIDGET_ID = I.WIDGET_ID
AND W.VERSION_NUMBER = I.VERSION_NUMBER
AND I.MAPPING_ID = M.MAPPING_ID
AND I.VERSION_NUMBER = M.VERSION_NUMBER
AND W.SUBJECT_ID = S.SUBJ_ID
AND UPPER(A.ATTR_VALUE) LIKE '%PARALLEL%'
```

7 MAPPLET

7.1 LIST MAPPLETS IN ALL FOLDERS

```
 select subject_area,mapplet_name from rep_all_mapplets
order by 1,2
```

7.2 LIST MAPPLET PARAMETERS AND VARIABLES

```
select distinct rep_reposit_info.repository_name, rep_all_mapplets.subject_area
as folder_name, rep_all_mapplets.mapplet_name as object_name,
case when opb_map_parmvar.pv_flag = 2 then 'Mapplet Parameter' else
case when opb_map_parmvar.pv_flag = 3 then 'Mapplet Variable' end
end as parameter_type, opb_map_parmvar.pv_name as parameter_name,
opb_map_parmvar.pv_default as parameter_value,
opb_map_parmvar.pv_desc as description
from rep_all_mapplets, rep_widget_inst, opb_mapping, opb_map_parmvar, rep_reposit_info
where rep_all_mapplets.mapplet_id=opb_mapping.mapping_id
and rep_widget_inst.widget_id=opb_mapping.ref_widget_id
and opb_mapping.mapping_id=opb_map_parmvar.mapping_id
and rep_widget_inst.widget_type=44
```

8 SESSION

8.1 LIST SESSION NAMES

```
 SELECT   SUBJECT_AREA, TASK_TYPE_NAME, TASK_NAME FROM REP_ALL_TASKS
   WHERE TASK_TYPE IN (68)
          --AND SUBJECT_AREA= 'ABC'
ORDER BY 1, 2, 3
```

8.2 LIST SAVE SESSION LOG COUNT

```
select distinct cc.subject_area,cc.task_name as session_name,bb.attr_value as Savesessionlog
from
(select   a.session_id,min(a.config_id) as config_id,a.attr_id from   rep_sess_config_parm a
where a.attr_id = '103' group by a.session_id,a.attr_id) aa,
(select session_id,config_id,attr_value from rep_sess_config_parm
where attr_id = '103') bb,
(select subject_area,task_name,task_id from rep_all_tasks ) cc
where aa.session_id = bb.session_id
and aa.config_id=bb.config_id
and bb.session_id = cc.task_id
and bb.attr_value not in (8,4)
                order by 1,2,3
```

8.3 LIST STOP ON ERRORS COUNT

```
 SELECT DISTINCT
A.SUBJECT_AREA,
A.TASK_NAME AS SESSION_NAME,
B.ATTR_VALUE AS STOPONERRORS
FROM
  REP_ALL_TASKS A ,
  REP_SESS_CONFIG_PARM   B
```

```
WHERE
  A.TASK_ID = B.SESSION_ID
  AND TASK_TYPE_NAME = 'Session' AND B.ATTR_ID = '202'
  --AND B.ATTR_VALUE NOT IN (1) --AND A.SUBJECT_AREA in ('ABC')
ORDER BY 1,2
```

8.4 LIST HARD CODED PATHS

```
SELECT DISTINCT
A.SUBJECT_AREA,
A.TASK_NAME AS SESSION_NAME,
B.FILE_NAME,
DIR_NAME
FROM
  REP_ALL_TASKS A ,
  OPB_SESS_FILE_VALS B
WHERE
      A.TASK_TYPE_NAME = 'Session'
     AND A.TASK_ID = B.SESSION_ID
     --AND A.SUBJECT_AREA IN ('ABC')
ORDER BY 1,2
```

8.5 LIST PARAMETER FILE PATHS

```
SELECT DISTINCT
      B.SUBJECT_AREA,
      B.TASK_NAME AS SES_WF_NAME,
      A.ATTR_VALUE AS PRM_FILE_PATH
FROM
       OPB_TASK_ATTR A,
       REP_ALL_TASKS B
WHERE
        A.ATTR_ID IN (1,4)
       AND A.TASK_ID = B.TASK_ID
       AND A.ATTR_VALUE LIKE '%.prm%'
ORDER BY 1,2 ASC
```

8.6 List session log names

```
 SELECT DISTINCT
      A.SUBJECT_AREA,
      A.WORKFLOW_NAME,
      A.SESSION_NAME,
      A.SESSION_INSTANCE_NAME,
      SUBSTR(A.SESSION_LOG_FILE,25,300) AS EXISTING_SESSLOGNAME
FROM
       REP_SESS_LOG A
```

```
WHERE
        SUBSTR(A.SESSION_LOG_FILE,25,300) != CONCAT(LOWER(A.SESSION_INSTANCE_NAME),'.log')
      ORDER BY 1,2,3
```

8.7 LIST COMMIT INTERVALS

```
 SELECT
       B.SUBJECT_AREA,
       B.TASK_NAME AS SESS_NAME,
       A.ATTR_VALUE AS COMMITINTERVEL
FROM
      OPB_TASK_ATTR A ,
       REP_ALL_TASKS B
WHERE
       A.ATTR_ID IN (14)
      AND A.ATTR_VALUE <> 10000
      AND A.TASK_ID = B.TASK_ID
      AND TASK_TYPE_NAME IN ('Session')
ORDER BY 1,2 ASC
```

8.8 LIST TOTAL SOURCE PARTITIONS

```
 SELECT
       B.SUBJECT_AREA,
       B.TASK_NAME AS SESS_NAME,
       A.ATTR_VALUE AS TOTAL_SOURCE_PARTITIONS
FROM
      OPB_TASK_ATTR A ,
       REP_ALL_TASKS B
WHERE
       A.ATTR_ID IN (12)
      AND A.TASK_ID = B.TASK_ID
      AND TASK_TYPE_NAME IN ('Session')
ORDER BY 1,2 ASC
```

8.9 LIST TOTAL TARGET PARTITIONS

```
SELECT
       B.SUBJECT_AREA,
       B.TASK_NAME AS SESS_NAME,
       A.ATTR_VALUE AS TOTAL_TARGET_PARTITIONS
FROM
      OPB_TASK_ATTR A ,
       REP_ALL_TASKS B
WHERE
       A.ATTR_ID IN (11)
      AND A.TASK_ID = B.TASK_ID
```

```
    AND TASK_TYPE_NAME IN ('Session')
ORDER BY 1,2 ASC
```

8.10 List DTM Buffer Size

```
SELECT
     B.SUBJECT_AREA,
     B.TASK_NAME AS SESS_NAME,
     A.ATTR_VALUE AS DTM_BUFFER_SIZE
FROM
    OPB_TASK_ATTR A ,
     REP_ALL_TASKS B
WHERE
     A.ATTR_ID IN (101)
    AND A.TASK_ID = B.TASK_ID
    AND TASK_TYPE_NAME IN ('Session')
ORDER BY 1,2 ASC
```

8.11 LIST COLLECT PERFORMANCE DATA

```
SELECT
     B.SUBJECT_AREA,
     B.TASK_NAME AS SESS_NAME,
     A.ATTR_VALUE AS COLLECT_PERFORMANCE_DATA
FROM
    OPB_TASK_ATTR A ,
     REP_ALL_TASKS B
WHERE
     A.ATTR_ID IN (102)
    AND A.TASK_ID = B.TASK_ID
    AND TASK_TYPE_NAME IN ('Session')
ORDER BY 1,2 ASC
```

8.12 List Incremental Aggregation

```
 SELECT
     B.SUBJECT_AREA,
     B.TASK_NAME AS SESS_NAME,
     A.ATTR_VALUE AS INCREMENTAL_AGGREGATION
FROM
    OPB_TASK_ATTR A ,
     REP_ALL_TASKS B
WHERE
```

```
    A.ATTR_ID IN (103)
    AND A.TASK_ID = B.TASK_ID
    AND TASK_TYPE_NAME IN ('Session')
ORDER BY 1,2 ASC
```

8.13 List Reinitialize aggregate cache

```
SELECT
    B.SUBJECT_AREA,
    B.TASK_NAME AS SESS_NAME,
    A.ATTR_VALUE AS REINITIALIZE_AGGREGATE CACHE
FROM
    OPB_TASK_ATTR A ,
    REP_ALL_TASKS B
WHERE
    A.ATTR_ID IN (104)
    AND A.TASK_ID = B.TASK_ID
    AND TASK_TYPE_NAME IN ('Session')
ORDER BY 1,2 ASC
```

8.14 LIST ENABLE HIGH PRECISION

```
SELECT
    B.SUBJECT_AREA,
    B.TASK_NAME AS SESS_NAME,
    A.ATTR_VALUE AS ENABLE_HIGH_PRECISION
FROM
    OPB_TASK_ATTR A ,
    REP_ALL_TASKS B
WHERE
    A.ATTR_ID IN (105)
    AND A.TASK_ID = B.TASK_ID
    AND TASK_TYPE_NAME IN ('Session')
ORDER BY 1,2 ASC
```

8.15 LIST SESSION RETRY ON DEADLOCK

```
SELECT
    B.SUBJECT_AREA,
    B.TASK_NAME AS SESS_NAME,
    A.ATTR_VALUE AS SESSION_ RETRYON_DEADLOCK
FROM
    OPB_TASK_ATTR A ,
    REP_ALL_TASKS B
WHERE
    A.ATTR_ID IN (106)
```

```
      AND A.TASK_ID = B.TASK_ID
      AND TASK_TYPE_NAME IN ('Session')
ORDER BY 1,2 ASC
```

8.16 LIST WRITE BACKWARD COMPATIBLE CHECK

```
 SELECT DISTINCT
       A.SUBJECT_AREA,A.TASK_NAME,
       DECODE (B.ATTR_VALUE,0,'TO BE CHECKED',1,'CHECKED') WRITEBACKWARDCOMPATIBLE
FROM
       REP_ALL_TASKS A ,
      OPB_TASK_ATTR B
WHERE
       A.TASK_ID = B.TASK_ID
      AND B.TASK_TYPE IN 68
      AND B.ATTR_ID = 17
      --AND B.ATTR_VALUE <> 1
ORDER BY 1,2,3
```

8.17 LIST OVER RIDE TRACING

```
 SELECT REP_REPOSIT_INFO.REPOSITORY_NAME,
        REP_ALL_TASKS.SUBJECT_AREA AS FOLDER_NAME,
        REP_ALL_TASKS.TASK_NAME AS SESSION_NAME,
       CASE
         WHEN REP_SESS_CONFIG_PARM.ATTR_VALUE = 0 THEN 'NONE'
         WHEN REP_SESS_CONFIG_PARM.ATTR_VALUE = 1 THEN 'TERSE'
         WHEN REP_SESS_CONFIG_PARM.ATTR_VALUE = 2 THEN 'NORMAL'
       WHEN REP_SESS_CONFIG_PARM.ATTR_VALUE = 3 THEN 'VERBOSE INITIALIZATION'
         WHEN REP_SESS_CONFIG_PARM.ATTR_VALUE = 4 THEN 'VERBOSE DATA'
       END AS OVERRIDE_TRACING
  FROM
       REP_REPOSIT_INFO,
     REP_ALL_TASKS,
     REP_SESS_CONFIG_PARM
 WHERE
      REP_ALL_TASKS.TASK_ID = REP_SESS_CONFIG_PARM.SESSION_ID
      AND REP_SESS_CONFIG_PARM.ATTR_ID = 204
      --AND REP_SESS_CONFIG_PARM.ATTR_VALUE NOT IN (0,2)
ORDER BY 1,2,3
```

8.18 LIST SAVE SESSION LOG BY

```
SELECT   A.SUBJECT_AREA,
                  A.TASK_NAME AS SESSION_NAME,
                  B.ATTR_NAME,
                 DECODE(B.ATTR_VALUE,1,'TIME STAMP','BY RUNS') AS SAVE_SESSION_LOG_BY
```

```
FROM
    REP_ALL_TASKS A ,
    REP_SESS_CONFIG_PARM   B
WHERE
  A.TASK_ID = B.SESSION_ID
  AND TASK_TYPE_NAME = 'Session'
  AND B.ATTR_ID IN ('102')
  --AND B.ATTR_VALUE <> 0
ORDER BY 1,2
```

8.19 LIST LOAD TYPE

```
 SELECT DISTINCT
      REP_LOAD_SESSIONS.SUBJECT_AREA AS FOLDER, REP_LOAD_SESSIONS.SESSION_NAME,
    --REP_SESS_WIDGET_CNXS.CNX_NAME AS CONNECTION_NAME,
    CASE WHEN OPB_EXTN_ATTR.ATTR_VALUE ='0' THEN 'NORMAL'
           WHEN OPB_EXTN_ATTR.ATTR_VALUE ='1' THEN 'BULK'
    END AS TARGET_LOAD_TYPE
FROM
    REP_LOAD_SESSIONS,
    REP_SESS_WIDGET_CNXS,
    OPB_EXTN_ATTR
WHERE REP_LOAD_SESSIONS.SESSION_ID=REP_SESS_WIDGET_CNXS.SESSION_ID
    AND REP_LOAD_SESSIONS.SESSION_ID=OPB_EXTN_ATTR.SESSION_ID
   AND OPB_EXTN_ATTR.ATTR_ID=3
   AND OPB_EXTN_ATTR.ATTR_VALUE BETWEEN '0' AND '1'
   AND REP_SESS_WIDGET_CNXS.READER_WRITER_TYPE='Relational Writer'
   --AND OPB_EXTN_ATTR.ATTR_VALUE ='1'
ORDER BY 1,2
```

8.20 LIST 'POST_SESSION_SUCCESS_COMMAND' IN SESSION

```
 SELECT DISTINCT C.SUBJ_NAME AS FOLDER ,A.TASK_NAME AS TASK, B.PM_VALUE AS COMMAND
FROM OPB_TASK A,OPB_TASK_VAL_LIST B, OPB_SUBJECT C
WHERE A.TASK_TYPE=58 AND A.TASK_NAME='post_session_success_command'
AND B.TASK_ID=A.TASK_ID AND B.SUBJECT_ID=C.SUBJ_ID
ORDER BY 1
```

8.21 LIST OF ALL THE EMAILS WITH ATTACHMENT

```
 SELECT DISTINCT
D.SUBJ_NAME AS FOLDER_NAME, C.WORKFLOW_NAME AS WORKFLOW_NAME,A.TASK_NAME AS TASK_NAME,
B.ATTR_VALUEAS VALUE
FROM
OPB_TASK A, OPB_TASK_ATTR B, REP_TASK_INST_RUN C, OPB_SUBJECT D
WHERE A.TASK_ID = B.TASK_ID
AND A.TASK_TYPE = B.TASK_TYPE
AND C.SUBJECT_ID = A.SUBJECT_ID
AND A.SUBJECT_ID = D.SUBJ_ID
AND A.TASK_TYPE = 65
AND B.ATTR_ID IN (2,3)
AND(B.ATTR_VALUE LIKE '%\%a%' ESCAPE '\' OR B.ATTR_VALUE LIKE '%\%g%' ESCAPE '\')
```

8.22 INVALID SESSIONS AND WORKFLOWS

```
 select opb_subject.subj_name, opb_task.task_name
from   opb_task,    opb_subject
where task_type in (68,71)
and is_valid = 0
and opb_subject.subj_id = opb_task.subject_id
order by 1,2
```

```
SELECT SUBJECT_AREA AS FOLDER_NAME,
DECODE(IS_REUSABLE,1,'Reusable',' ') || ' ' ||TASK_TYPE_NAME AS TASK_TYPE,
TASK_NAME AS OBJECT_NAME,
DECODE(IS_VALID,0,'INVALID OBJECT','VALID OBJECT') STATUS,
LAST_SAVED
FROM REP_ALL_TASKS
WHERE IS_VALID=0
AND IS_ENABLED=1
--AND CHECKOUT_USER_ID = 0 -- Comment out for V6
--AND is_visible=1 -- Comment out for V6
ORDER BY 1,2
```

9 TASKS

9.1 LIST COMMAND TASKS

```
 SELECT    SUBJECT_AREA, TASK_TYPE_NAME, TASK_NAME
     FROM REP_ALL_TASKS
   WHERE TASK_TYPE IN (58)
           --AND SUBJECT_AREA= 'ABC'
ORDER BY 1, 2, 3
```

9.2 LIST DECISION TASKS

```
 SELECT    SUBJECT_AREA, TASK_TYPE_NAME, TASK_NAME
     FROM REP_ALL_TASKS
   WHERE TASK_TYPE IN (59)
           --AND SUBJECT_AREA= 'ABC'
ORDER BY 1, 2, 3
```

9.3 LIST EVENT WAIT TASKS

```
SELECT   SUBJECT_AREA, TASK_TYPE_NAME, TASK_NAME
    FROM REP_ALL_TASKS
  WHERE TASK_TYPE IN (60)
        --AND SUBJECT_AREA= 'ABC'
ORDER BY 1, 2, 3
```

10 WORKLET

10.1 LIST WORKLET NAMES

```
 SELECT   SUBJECT_AREA, TASK_TYPE_NAME, TASK_NAME
    FROM REP_ALL_TASKS
  WHERE TASK_TYPE IN (70)
        --AND SUBJECT_AREA= 'ABC'
ORDER BY 1, 2, 3
```

10.2 LIST HIERARCHIES OF ALL WORKFLOWS AND ITS WORKLETS

```
 SELECT DISTINCT '/' || temp1.task_id AS path, temp1.task_name AS hierarchy_structure
FROM opb_task temp1, opb_subject temp2
WHERE temp1.subject_id = temp2.subj_id
AND temp1.task_type = 71
AND temp2.subj_name = 'FOLDER_NAME'

UNION ALL

SELECT DISTINCT temp1.path, temp1.task_name AS hierarchy_structure
FROM (SELECT opb_task_inst.workflow_id,opb_task_inst.task_id,opb_task_inst.instance_id,LEVEL depth,
SYS_CONNECT_BY_PATH(opb_task_inst.workflow_id ,'/') || '/' || opb_task_inst.task_id || '/' path,
LPAD (' ', 4 * LEVEL, ' ') || SYS_CONNECT_BY_PATH(opb_task_inst.instance_name ,'/') task_name
FROM opb_task_inst WHERE opb_task_inst.task_type IN (68,70)
START WITH workflow_id IN (SELECT task_id FROM opb_task WHERE task_type = 71)
CONNECT BY PRIOR opb_task_inst.task_id = opb_task_inst.workflow_id) temp1,
opb_task temp2, opb_subject temp3
WHERE temp2.subject_id = temp3.subj_id
AND temp2.task_id = SUBSTR(temp1.path,2, INSTR(temp1.path,'/', 1, 2) -2 )
AND temp3.subj_name = 'FOLDER_NAME'
ORDER BY path ASC
```

11 WORKFLOW

11.1 LIST WORKFLOW NAMES

```
 SELECT   SUBJECT_AREA, TASK_TYPE_NAME, TASK_NAME
    FROM REP_ALL_TASKS
  WHERE TASK_TYPE IN (71)
        --AND SUBJECT_AREA= 'ABC'
```

```
ORDER BY 1, 2, 3
```

11.2 List save workflow log count

```
SELECT DISTINCT
A.SUBJECT_AREA,
A.TASK_NAME AS WORKFLOW_NAME,
B.ATTR_VALUE AS SAVEWFLOG
FROM
REP_ALL_TASKS A ,
REP_TASK_ATTR   B
WHERE
A.TASK_ID = B.TASK_ID
AND B.ATTR_ID = '4'
AND B.TASK_TYPE = 71
--AND B.ATTR_VALUE NOT IN (8,4)
-- AND A.SUBJECT_AREA = 'ABC'
ORDER BY 1,2,3
```

11.3 LIST WORKFLOW LOG NAMES

```
 SELECT DISTINCT
        SUBJ_NAME,
        WORKFLOW_NAME,
        SUBSTR(LOG_FILE,23,300) AS EXISTING_WFLOGNAME
FROM
       OPB_WFLOW_RUN,
       OPB_SUBJECT
WHERE
        OPB_WFLOW_RUN.SUBJECT_ID = OPB_SUBJECT.SUBJ_ID
        AND SUBSTR(LOG_FILE,23,300) != CONCAT(LOWER(WORKFLOW_NAME),'.log')
ORDER BY 1,2
```

11.4 LIST WRITE BACKWARD COMPATIBLE CHECK

```
 SELECT DISTINCT
A.SUBJECT_AREA,A.TASK_NAME,
DECODE (B.ATTR_VALUE,0,'TO BE CHECKED',1,'CHECKED') WRITEBACKWARDCOMPATIBLE
FROM
     REP_ALL_TASKS A,
     OPB_TASK_ATTR B
WHERE
A.TASK_ID = B.TASK_ID
AND B.TASK_TYPE IN 71
AND B.ATTR_ID = 12
AND B.ATTR_VALUE <> 1
```

11.5 LIST FAIL_PARENT_IF_TASK_FAILS OBJECTS

```
 SELECT
   REPOSITORY,
   FOLDER_NAME,
   WORKFLOW_OR_WORKLET,
   TASK_TYPE,
   WORKLET_OR_SESSION,
   FAIL_PARENT_IF_TASK_FAILS
FROM
(SELECT DISTINCT
OPB_REPOSIT_INFO.REPOSITORY_NAME AS REPOSITORY,
OPB_SUBJECT.SUBJ_NAME AS FOLDER_NAME,
OPB_TASK.TASK_NAME AS WORKFLOW_OR_WORKLET,
DECODE(OPB_TASK_INST.TASK_TYPE,58,'COMMAND',59,'DECISION',60,'EVENT
WAIT',62,'START',65,'EMAIL',66,'TIMER',67,'ASSIGNMENT',68,'SESSION',70,'WORKLET',91,'CONTROL',NULL) TASK
_TYPE,
OPB_TASK_INST.INSTANCE_NAME AS WORKLET_OR_SESSION,
DECODE (BITAND (OPB_TASK_INST.BIT_OPTIONS, 17),17,'SELECTED','NOT
SELECTED') AS FAIL_PARENT_IF_TASK_FAILS
FROM OPB_TASK_INST,OPB_OBJECT_TYPE,OPB_TASK,
        OPB_SUBJECT, OPB_REPOSIT_INFO
WHERE OPB_TASK_INST.TASK_TYPE != 62
AND OPB_TASK_INST.TASK_TYPE = OPB_OBJECT_TYPE.OBJECT_TYPE_ID
AND OPB_TASK_INST.WORKFLOW_ID = OPB_TASK.TASK_ID
AND OPB_TASK_INST.VERSION_NUMBER = OPB_TASK.VERSION_NUMBER
AND OPB_TASK.SUBJECT_ID = OPB_SUBJECT.SUBJ_ID
AND OPB_TASK.UTC_CHECKIN <> 0
--AND OPB_SUBJECT.SUBJ_NAME NOT LIKE 'WA%'
 )
WHERE FAIL_PARENT_IF_TASK_FAILS <> 'SELECTED'
```

11.6 LIST FAIL_PARENT_IF_TASK_DONT_RUN OBJECTS

```
SELECT REPOSITORY,FOLDER_NAME,WORKFLOW_OR_WORKLET,TASK_TYPE,WORKLET_OR_SESSION,FAIL_PAREN
T_IF_TASK_DONT_RUN
FROM
            (SELECT DISTINCT
          OPB_REPOSIT_INFO.REPOSITORY_NAME AS REPOSITORY,
OPB_SUBJECT.SUBJ_NAME AS FOLDER_NAME, OPB_TASK.TASK_NAME AS WORKFLOW_OR_WORKLET,
DECODE(OPB_TASK_INST.TASK_TYPE,58,'COMMAND',59,'DECISION',60,'EVENT
WAIT',62,'START',65,'EMAIL',66,'TIMER',67,'ASSIGNMENT',68,'SESSION',70,'WORKLET',91,'CONTROL',NULL) TASK
_TYPE,OPB_TASK_INST.INSTANCE_NAMEAS WORKLET_OR_SESSION, DECODE(BITAND(OPB_TASK_INST.BIT_OPTI
ONS, 49),49,'SELECTED','NOT SELECTED') AS FAIL_PARENT_IF_TASK_DONT_RUN
      FROM OPB_TASK_INST, OPB_OBJECT_TYPE,OPB_TASK,OPB_SUBJECT,OPB_REPOSIT_INFO
      WHERE OPB_TASK_INST.TASK_TYPE != 62
    AND OPB_TASK_INST.TASK_TYPE = OPB_OBJECT_TYPE.OBJECT_TYPE_ID
    AND OPB_TASK_INST.WORKFLOW_ID = OPB_TASK.TASK_ID
    AND OPB_TASK_INST.VERSION_NUMBER = OPB_TASK.VERSION_NUMBER
    AND OPB_TASK.SUBJECT_ID = OPB_SUBJECT.SUBJ_ID
    AND OPB_TASK.UTC_CHECKIN <> 0  )
WHERE FAIL_PARENT_IF_TASK_DONT_RUN <> 'SELECTED'
ORDER BY 2
```

11.7 LIST IS_TASK_ENABLED OBJECTS

```
SELECT REPOSITORY,FOLDER_NAME,WORKFLOW_OR_WORKLET,TASK_TYPE,WORKLET_OR_SESSION,IS_TASK_EN
ABLED
FROM
      (SELECT DISTINCT
     OPB_REPOSIT_INFO.REPOSITORY_NAME AS REPOSITORY,
OPB_SUBJECT.SUBJ_NAME AS FOLDER_NAME, OPB_TASK.TASK_NAME AS WORKFLOW_OR_WORKLET,
DECODE(OPB_TASK_INST.TASK_TYPE,58,'COMMAND',59,'DECISION',60,'EVENT
WAIT',62,'START',65,'EMAIL',66,'TIMER',67,'ASSIGNMENT',68,'SESSION',70,'WORKLET',91,'CONTROL',
NULL) TASK_TYPE,OPB_TASK_INST.INSTANCE_NAMEAS WORKLET_OR_SESSION,
       DECODE (OPB_TASK_INST.IS_ENABLED,1, 'ENABLED','DISABLED') AS IS_TASK_ENABLED
      FROM OPB_TASK_INST,OPB_OBJECT_TYPE,OPB_TASK,OPB_SUBJECT,OPB_REPOSIT_INFO
      WHERE OPB_TASK_INST.TASK_TYPE != 62
             AND OPB_TASK_INST.TASK_TYPE = OPB_OBJECT_TYPE.OBJECT_TYPE_ID
            AND OPB_TASK_INST.WORKFLOW_ID = OPB_TASK.TASK_ID
            AND OPB_TASK_INST.VERSION_NUMBER = OPB_TASK.VERSION_NUMBER
            AND OPB_TASK.SUBJECT_ID = OPB_SUBJECT.SUBJ_ID
           AND OPB_TASK.UTC_CHECKIN <> 0 )
WHERE IS_TASK_ENABLED = 'DISABLED'
ORDER BY 2,3
```

11.8 LIST TREAT_INPUT_LINKS_AS OBJECTS

```
SELECT REPOSITORY,FOLDER_NAME,WORKFLOW_OR_WORKLET,TASK_TYPE,
WORKLET_OR_SESSION,TREAT_INPUT_LINKS_AS
FROM
(SELECT DISTINCT
OPB_REPOSIT_INFO.REPOSITORY_NAME AS REPOSITORY,
OPB_SUBJECT.SUBJ_NAME AS FOLDER_NAME,
OPB_TASK.TASK_NAME AS WORKFLOW_OR_WORKLET,
DECODE(OPB_TASK_INST.TASK_TYPE,58,'COMMAND',59,'DECISION',60,'EVENT WAIT',62,'START',65,'EMAIL',66,'TIMER',67,'ASSIGNMENT',68,'SESSION',70,'WORKLET',91,'CONTROL',NULL) TASK_TYPE,
OPB_TASK_INST.INSTANCE_NAME AS WORKLET_OR_SESSION,
DECODE (BITAND (OPB_TASK_INST.BIT_OPTIONS, 3),1,'AND',2, 'OR') AS TREAT_INPUT_LINKS_AS
FROM OPB_TASK_INST,OPB_OBJECT_TYPE,OPB_TASK,OPB_SUBJECT,OPB_REPOSIT_INFO WHERE OPB_TASK_INST.TASK_TYPE != 62
AND OPB_TASK_INST.TASK_TYPE = OPB_OBJECT_TYPE.OBJECT_TYPE_ID
AND OPB_TASK_INST.WORKFLOW_ID = OPB_TASK.TASK_ID
AND OPB_TASK_INST.VERSION_NUMBER = OPB_TASK.VERSION_NUMBER
AND OPB_TASK.SUBJECT_ID = OPB_SUBJECT.SUBJ_ID
AND OPB_TASK.UTC_CHECKIN <> 0
)
--WHERE TREAT_INPUT_LINKS_AS = 'OR'
ORDER BY 2,3
```

11.9 LIST ALL WORKFLOWS WHOSE SERVER IS NOT ASSIGNED

```
SELECT SUBJECT_AREA,WORKFLOW_NAME,SERVER_NAME
FROM REP_WORKFLOWS
WHERE SERVER_NAME IS NULL
```

11.10 LIST OF WORKFLOW RUN DETAILS

```
SELECT DISTINCT OPB_SUBJECT.SUBJ_NAME,OPB_WFLOW_RUN.WORKFLOW_NAME,
OPB_TASK_INST_RUN.INSTANCE_NAME,
TRUNC(OPB_TASK_INST_RUN.START_TIME) LOAD_DATE,
TO_CHAR(OPB_TASK_INST_RUN.START_TIME,'HH24:MI.SS') SESS_START_TIME,
TO_CHAR(OPB_TASK_INST_RUN.END_TIME,'HH24:MI.SS') SESS_END_TIME,
TRUNC((OPB_TASK_INST_RUN.END_TIME - OPB_TASK_INST_RUN.START_TIME) * 1440,2) DURATION_IN_MINS
FROM OPB_SUBJECT,OPB_TASK_INST_RUN,OPB_WFLOW_RUN,OPB_OBJECT_TYPE, OPB_SESS_TASK_LOG
WHERE

 OPB_SUBJECT.SUBJ_ID = OPB_TASK_INST_RUN.SUBJECT_ID
   AND OPB_WFLOW_RUN.WORKFLOW_ID = OPB_TASK_INST_RUN.WORKFLOW_ID
   AND OPB_WFLOW_RUN.WORKFLOW_RUN_ID = OPB_TASK_INST_RUN.WORKFLOW_RUN_ID
   AND OPB_TASK_INST_RUN.TASK_TYPE = OPB_OBJECT_TYPE.OBJECT_TYPE_ID
   AND OPB_OBJECT_TYPE.OBJECT_TYPE = 68
   AND OPB_WFLOW_RUN.WORKFLOW_ID = OPB_SESS_TASK_LOG.WORKFLOW_ID
   AND OPB_WFLOW_RUN.WORKFLOW_RUN_ID = OPB_SESS_TASK_LOG.WORKFLOW_RUN_ID
```

```
     AND OPB_TASK_INST_RUN.WORKLET_RUN_ID = OPB_SESS_TASK_LOG.WORKLET_RUN_ID
     AND OPB_TASK_INST_RUN.INSTANCE_ID = OPB_SESS_TASK_LOG.INSTANCE_ID
ORDER BY 1,2,4
```

12 CONNECTIONS

12.1 LIST OF CNXS USING ALTER IN ENV SQL

```
SELECT DISTINCT
D.OBJECT_NAME,
C.DB, C.USERNAME,
C.ATTR_VALUE AS ENVIRONMENT_SQL
FROM OPB_CNX D,
    (SELECT DISTINCT A.OBJECT_NAME CONN, B.OBJECT_ID,
                     A.CONNECT_STRING DB, A.USER_NAME USERNAME,
                     B.ATTR_VALUE FROM OPB_CNX A,
           (SELECT B.OBJECT_ID, B.OBJECT_SUBTYPE,
                    B.OBJECT_TYPE, B.ATTR_VALUE
                FROM OPB_CNX_ATTR B
                WHERE B.ATTR_ID = 11) B
       WHERE A.OBJECT_ID = B.OBJECT_ID
  AND B.ATTR_VALUE IS NOT NULL) C
 WHERE D.OBJECT_NAME = C.CONN
 ORDER BY 1
```

12.2 LIST OF CNXS USED IN SESSION LEVELS

```
 SELECT DISTINCT
A.SUBJECT_AREA,
A.TASK_NAME,
B.CNX_NAME
FROM
REP_ALL_TASKS A,
REP_SESS_WIDGET_CNXS B
WHERE
A.TASK_ID = B.SESSION_ID
ORDER BY 1,2,3
```

12.3 LIST LOTUS CONNECTION DETAILS

```
SELECT DISTINCT
a.object_name as connection_name,
case when a.OBJECT_SUBTYPE = 101 then 'Oracle'
when a.OBJECT_SUBTYPE = 104 then 'Microsoft SQL Server'
when a.OBJECT_SUBTYPE = 106 then 'ODBC'
when a.OBJECT_SUBTYPE = 0 then 'FTP'
when a.OBJECT_SUBTYPE = 404000 then 'Lotus Notes' end as type,
a.user_name as conn_user_name,
e.ServerHost,
e.DatabaseFilename,
b.user_name connection_owner,
case when user_type = '1' and d.user_id in (select id from opb_user_group y where y.type = 1)   then (select 'User
- '||x.user_namefromopb_users xwhere d.user_id=x.user_id)
when user_type = '2' and d.user_id in (select id from opb_user_group y where y.type = 2) then (select 'Group -
'||z.name fromopb_user_group zwhere d.user_id=z.id and z.type = 2)
when user_type = '3' and d.user_id in (select id from opb_user_group) then (select name from opb_user_group )
when d.user_id = '0' then 'Others'
end as CONN_USERS_LIST,
CASE   WHEN user_type = 1 THEN
CASE WHEN permissions = d.user_id + 15 THEN 'RWX'
WHEN permissions = d.user_id + 13 THEN 'RW'
WHEN permissions = d.user_id + 11 THEN 'RX'
WHEN permissions = d.user_id + 9  THEN 'R'
WHEN permissions = d.user_id + 7  THEN 'WX'
WHEN permissions = d.user_id + 5  THEN 'W'
WHEN permissions = d.user_id + 3  THEN 'X'
ELSE 'NULL' END
WHEN user_type = 2 THEN
CASE WHEN permissions = d.user_id + 29  THEN 'RWX'
WHEN permissions = d.user_id + 25  THEN 'RW'
WHEN permissions = d.user_id + 21  THEN 'RX'
                WHEN permissions = d.user_id + 17  THEN 'R'
                WHEN permissions = d.user_id + 13  THEN 'WX'
                WHEN permissions = d.user_id + 9   THEN 'W'
                WHEN permissions = d.user_id + 5   THEN 'X'
ELSE 'NULL'
END
WHEN user_type = 3
THEN CASE
WHEN permissions = d.user_id + 57 THEN 'RWX'
WHEN permissions = d.user_id + 39 THEN 'RW'
WHEN permissions = d.user_id + 41 THEN 'RX'
WHEN permissions = d.user_id + 33 THEN 'R'
WHEN permissions = d.user_id + 25 THEN 'WX'
WHEN permissions = d.user_id + 17 THEN 'W'
WHEN permissions = d.user_id + 9  THEN 'X'
```

```
ELSE 'NULL'
END
END PREVILIGES
FROM opb_cnx a, opb_users b, opb_groups c, opb_object_access d,
( Select   a.object_id,a.ServerHost,b.DatabaseFilename  from
                    (select object_id,Attr_value as ServerHost  from opb_cnx_attr where OBJECT_SUBTYPE   = 40400
0 and attr_id = 1 ) a,
(select object_id,Attr_value as DatabaseFilename  from opb_cnx_attr where OBJECT_SUBTYPE = 404000 and attr
_id = 2 ) b
where a.object_id = b.object_id) e
WHERE a.owner_id = b.user_id
AND a.object_id = d.object_id
AND d.object_id = e.object_id
and a.OBJECT_SUBTYPE = 404000
ORDER BY object_name
```

12.4 ODBC / SQL SERVER CONNECTION DETAILS

```
SELECT DISTINCT
a.object_name as connection_name,
a.user_name as conn_user_name,
--     a.connect_string,
b.user_name connection_owner,
--   b.user_desc AS conn_owner_desc,
case when user_type = '1' and d.user_id in (select user_id from opb_users) then (select 'User -
'||x.user_name from opb_users xwhered.user_id=x.user_id)
when user_type = '2' and d.user_id in (select group_id from opb_groups) then (select 'Group -
'||x.group_name from opb_groups xwhered.user_id=x.group_id)
when d.user_id = '0' then 'World' end as CONN_USERS_LIST,


       CASE
             WHEN user_type = 1 THEN
                                                                                 CASE WHEN permissions = d.
user_id + 15 THEN 'RWX'
                                                                             WHEN permissions = d.user_id + 1
3 THEN 'RW'
WHEN permissions = d.user_id + 11 THEN 'RX'
WHEN permissions = d.user_id + 9  THEN 'R'
WHEN permissions = d.user_id + 7  THEN 'WX'
WHEN permissions = d.user_id + 5  THEN 'W'
WHEN permissions = d.user_id + 3  THEN 'X'
                                                                       ELSE 'NULL'
```

```
                            END
        WHEN user_type = 2 THEN
CASE WHEN permissions = d.user_id + 29  THEN 'RWX'
WHEN permissions = d.user_id + 25  THEN 'RW'
WHEN permissions = d.user_id + 21  THEN 'RX'
WHEN permissions = d.user_id + 17  THEN 'R'
WHEN permissions = d.user_id + 13  THEN 'WX'
WHEN permissions = d.user_id + 9   THEN 'W'
WHEN permissions = d.user_id + 5   THEN 'X'
                                                          ELSE 'NULL'
             END
          WHEN user_type = 3
           THEN CASE
               WHEN permissions = d.user_id + 57 THEN 'RWX'
               WHEN permissions = d.user_id + 39 THEN 'RW'
               WHEN permissions = d.user_id + 41 THEN 'RX'
               WHEN permissions = d.user_id + 33 THEN 'R'
               WHEN permissions = d.user_id + 25 THEN 'WX'
               WHEN permissions = d.user_id + 17 THEN 'W'
               WHEN permissions = d.user_id + 9  THEN 'X'
               ELSE 'NULL'
             END
        END PREVILIGES
         FROM opb_cnx a, opb_users b, opb_groups c, opb_object_access d
      WHERE a.owner_id = b.user_id
       AND a.GROUP_ID = c.group_id
       AND a.object_id = d.object_id and d.object_type=73
                     and a.object_subtype = 106
       ORDER BY object_name
```

12.5 LIST OF SESSIONS USED BY A CONNECTION

```
SELECT DISTINCT C.SUBJECT_AREA, B.WORKFLOW_NAME,A.SESSION_INSTANCE_NAME, CONNECTION_NAME, CONNECT_STRING
FROM REP_SESSION_CNXS C , OPB_CNX,REP_SESS_LOG A,REP_WFLOW_RUN B
WHERE C.CONNECTION_ID=OPB_CNX.OBJECT_ID
AND C.SESSION_ID = A.SESSION_ID
AND A.WORKFLOW_ID=B.WORKFLOW_ID
AND CONNECTION_NAME IN
('CMX_NAME') ORDER BY 1,2
```

12.6 QUERY TO FETCH CONNECTION DETAILS USERS LIST AND PRIVILEGES’

```
SELECT DISTINCT
                a.object_name as connection_name,
                a.user_name as conn_user_name,
                a.connect_string,
                b.user_name connection_owner,
                b.user_desc AS conn_owner_desc,
               case when user_type = '1' and d.user_id in (select user_id from opb_users) then (select 'User -
'||x.user_namefrom opb_users xwhere d.user_id=x.user_id)
                    when user_type = '2' and d.user_id in (select group_id from opb_groups) then (select 'Group -
'||x.group_namefrom opb_groupsxwhere d.user_id=x.group_id)
                               when d.user_id = '0' then 'World' end as CONN_USERS_LIST,
--             d.user_type CONN_USER_TYPE,
          --    d.permissions,
      CASE
            WHEN user_type = 1
                THEN CASE
                     WHEN permissions = d.user_id + 15
                       THEN 'RWX'
                     WHEN permissions = d.user_id + 13
                       THEN 'RW'
                     WHEN permissions = d.user_id + 11
                       THEN 'RX'
                     WHEN permissions = d.user_id + 9
                       THEN 'R'
                     WHEN permissions = d.user_id + 7
                       THEN 'WX'
                     WHEN permissions = d.user_id + 5
                       THEN 'W'
                     WHEN permissions = d.user_id + 3
                            THEN 'X'
                     ELSE 'NULL'
                   END
              WHEN user_type = 2
                THEN CASE
                     WHEN permissions = d.user_id + 29
                       THEN 'RWX'
                     WHEN permissions = d.user_id + 25
                       THEN 'RW'
                     WHEN permissions = d.user_id + 21
                       THEN 'RX'
                     WHEN permissions = d.user_id + 17
                       THEN 'R'
                     WHEN permissions = d.user_id + 13
                       THEN 'WX'
                     WHEN permissions = d.user_id + 9
                       THEN 'W'
                     WHEN permissions = d.user_id + 5
                       THEN 'X'
                     ELSE 'NULL'
                   END
              WHEN user_type = 3
                THEN CASE
                     WHEN permissions = d.user_id + 57
                       THEN 'RWX'
                     WHEN permissions = d.user_id + 39
                       THEN 'RW'
                     WHEN permissions = d.user_id + 41
                       THEN 'RX'
                     WHEN permissions = d.user_id + 33
                       THEN 'R'
                     WHEN permissions = d.user_id + 25
                       THEN 'WX'
                     WHEN permissions = d.user_id + 17
                       THEN 'W'
                     WHEN permissions = d.user_id + 9
                       THEN 'X'
                     ELSE 'NULL'
                   END
            END PREVILIGES
        FROM opb_cnx a, opb_users b, opb_groups c, opb_object_access d
         WHERE a.owner_id = b.user_id
          AND a.GROUP_ID = c.group_id
          AND a.object_id = d.object_id and d.object_type=73
          ORDER BY 1
```

13 REPOSITORY

13.1 REPOSITORY INFO

```
SELECT DOMAIN_NAME, REPOSITORY_NAME,PCSF_DOMAIN AS DOMAIN, DB_USER FROM OPB_REPOSIT_INFO
```

13.2 LIST OF OBJECTS WHICH ARE NOT VALID

```
SELECT REPOSITORY,FOLDER_NAME,WORKFLOW_OR_WORKLET,TASK_TYPE,    WORKLET_OR_SESSION,IS_VALID
FROM
(SELECT DISTINCT
OPB_REPOSIT_INFO.REPOSITORY_NAME AS REPOSITORY,
OPB_SUBJECT.SUBJ_NAME AS FOLDER_NAME, OPB_TASK.TASK_NAME AS WORKFLOW_OR_WORKLET,
DECODE(OPB_TASK_INST.TASK_TYPE,58,'COMMAND',59,'DECISION',60,'EVENT
WAIT',62,'START',65,'EMAIL',66,'TIMER',67,'ASSIGNMENT',68,'SESSION',70,'WORKLET',91,'CONTROL',NULL) TASK
_TYPE,
OPB_TASK_INST.INSTANCE_NAME AS WORKLET_OR_SESSION,
DECODE (OPB_TASK_INST.IS_VALID,1,'VALID','NOT VALID') AS IS_VALID
FROM OPB_TASK_INST,OPB_OBJECT_TYPE,OPB_TASK,OPB_SUBJECT,OPB_REPOSIT_INFO
WHERE OPB_TASK_INST.TASK_TYPE != 62
AND OPB_TASK_INST.TASK_TYPE = OPB_OBJECT_TYPE.OBJECT_TYPE_ID
AND OPB_TASK_INST.WORKFLOW_ID = OPB_TASK.TASK_ID
AND OPB_TASK_INST.VERSION_NUMBER = OPB_TASK.VERSION_NUMBER
AND OPB_TASK.SUBJECT_ID = OPB_SUBJECT.SUBJ_ID
AND OPB_TASK.UTC_CHECKIN <> 0  )
WHERE IS_VALID = 'NOT VALID'
ORDER BY 2,3
```

13.3 LIST OF OBJECTS WHICH ARE FAILED IN LAST 5 DAYS

```
 SELECT Subject_Area AS Folder,
Session_Name,
Last_Error AS Error_Message,
DECODE (Run_Status_Code,3,'Failed',4,'Stopped',5,'Aborted') AS Status,
Actual_Start AS Start_Time,
Session_TimeStamp
FROM rep_sess_log
WHERE run_status_code != 1
AND TRUNC(Actual_Start) BETWEEN TRUNC(SYSDATE -5) AND TRUNC(SYSDATE)
order by 1,2
```

13.4 LIST WHERE ALL A TABLE IS USED

Sometimes you want to know if certain tables are listed in sql overrides of Source Qualifier or Lookup transformation. This helps you identifying dependencies. The query below will list folder, attribute type and sql override as output.

```
SELECT DISTINCT REP_ALL_MAPPINGS.SUBJECT_AREA, REP_ALL_MAPPINGS.MAPPING_NAME,REP_WIDGET_ATT
R.ATTR_NAME,REP_WIDGET_ATTR.ATTR_VALUE
FROM REP_WIDGET_ATTR, REP_WIDGET_INST, REP_ALL_MAPPINGS
WHERE REP_WIDGET_ATTR.WIDGET_ID = REP_WIDGET_INST.WIDGET_ID
AND REP_WIDGET_INST.MAPPING_ID = REP_ALL_MAPPINGS.MAPPING_ID
AND REP_WIDGET_ATTR.WIDGET_TYPE IN (3,11)
AND REP_WIDGET_ATTR.ATTR_ID = 1
AND REP_WIDGET_ATTR.ATTR_VALUE LIKE '%' || REPLACE('TABLE_NAME', '_', '/_') || '%' ESCAPE '/'
ORDER BY 1,2,3
```

13.5 LIST ALL SOURCE AND TARGET TABLES OF MAPPING

```
SELECT DISTINCT SUBJECT_AREA,SOURCE_NAME,TARGET_NAME,MAPPING_NAME FROM REP_TBL_MAPPING
ORDER BY 1,2,3,4
```

13.6 LIST COMMENTS OF ALL OBJECTS

```
 SELECT
B.SUBJECT_AREA AS FOLDER_NAME, A.OBJECT_NAME,A.COMMENTS,   A.VERSION_NUMBER
FROM
REP_VERSION_PROPS A, REP_SUBJECT B
WHERE B.SUBJECT_ID = A.SUBJECT_ID
AND A.COMMENTS IS NOT NULL
ORDER BY 1,2
```

15 GROUPS & USERS

Power Center Version 8 onwards Users and Groups details are stored in Domain database and in unreadable format. On execution of below attached scripts on domain database v_users and v_user_group views will be created and you can query on these views.

Steps to Follow:

Step 1 : Log into Domain db with create view, procedure & Function user Privileges
Step 2 : Execute below function

```
create or replace FUNCTION xblob_to_clob(l_blob BLOB) RETURN CLOB IS l_clob CLOB;
l_src_offset NUMBER;
l_dest_offset NUMBER;
l_blob_csid NUMBER := dbms_lob.default_csid;
v_lang_context NUMBER := dbms_lob.default_lang_ctx;
l_warning NUMBER;
l_amount NUMBER;
BEGIN

  IF dbms_lob.getlength(l_blob) > 0 THEN
    dbms_lob.createtemporary(l_clob,     TRUE);
    l_src_offset := 1;
    l_dest_offset := 1;
    l_amount := dbms_lob.getlength(l_blob);
    dbms_lob.converttoclob(l_clob,   l_blob,   l_amount,   l_src_offset,   l_dest_offset,   1,   v_l
ang_context,   l_warning);
    RETURN l_clob;
  ELSE
```

```
    l_clob := to_clob('');
    RETURN l_clob;
  END IF;

  dbms_lob.freetemporary(l_clob);
END;
/
```

Step 3 : Execute below sql to create v_user view

```
CREATE OR REPLACE FORCE VIEW "V_USERS" ("USER_ID", "USER_NAME", "NAMESPACE",
"FULL_NAME", "DESCRIPTION", "EMAIL", "PHONE", "READ_ONLY", "DISABLE") AS
  SELECT id user_id,
      extractvalue(xmltype(xblob_to_clob(metadata)),
              '/metadata:User/userName',
              'xmlns:common="/pcsf/common" xmlns:usermanagement="/pcsf/usermanagement"
xmlns:domainservice="/pcsf/domainservice" xmlns:logservice="/pcsf/logservice"
xmlns:domainbackup="/pcsf/domainbackup" xmlns:xsi="/2001/XMLSchema-instance"
xmlns:metadata="/pcsf/metadata" xmlns:xsd="/2001/XMLSchema"
xmlns:domainconfigservice="/pcsf/domainconfigservice" xmlns:alertservice="/pcsf/alertservice"
xmlns:licenseusage="/pcsf/licenseusage" xmlns:webserviceshub="/pcsf/webserviceshub"'
             )  userName,
      extractvalue(xmltype(xblob_to_clob(metadata)),
              '/metadata:User/nameSpace',
              'xmlns:common="/pcsf/common" xmlns:usermanagement="/pcsf/usermanagement"
xmlns:domainservice="/pcsf/domainservice" xmlns:logservice="/pcsf/logservice"
xmlns:domainbackup="/pcsf/domainbackup" xmlns:xsi="/2001/XMLSchema-instance"
xmlns:metadata="/pcsf/metadata" xmlns:xsd="/2001/XMLSchema"
xmlns:domainconfigservice="/pcsf/domainconfigservice" xmlns:alertservice="/pcsf/alertservice"
xmlns:licenseusage="/pcsf/licenseusage" xmlns:webserviceshub="/pcsf/webserviceshub"'
             )  nameSpace,
      extractvalue(VALUE(i),
              'info/fullName',
              'xmlns:common="/pcsf/common" xmlns:usermanagement="/pcsf/usermanagement"
xmlns:domainservice="/pcsf/domainservice" xmlns:logservice="/pcsf/logservice"
xmlns:domainbackup="/pcsf/domainbackup" xmlns:xsi="/2001/XMLSchema-instance"
xmlns:metadata="/pcsf/metadata" xmlns:xsd="/2001/XMLSchema"
xmlns:domainconfigservice="/pcsf/domainconfigservice" xmlns:alertservice="/pcsf/alertservice"
xmlns:licenseusage="/pcsf/licenseusage" xmlns:webserviceshub="/pcsf/webserviceshub"'
             ) fullName,
      extractvalue(VALUE(i),
              'info/description',
              'xmlns:common="/pcsf/common" xmlns:usermanagement="/pcsf/usermanagement"
xmlns:domainservice="/pcsf/domainservice" xmlns:logservice="/pcsf/logservice"
```

```
xmlns:domainbackup="/pcsf/domainbackup" xmlns:xsi="/2001/XMLSchema-instance"
xmlns:metadata="/pcsf/metadata" xmlns:xsd="/2001/XMLSchema"
xmlns:domainconfigservice="/pcsf/domainconfigservice" xmlns:alertservice="/pcsf/alertservice"
xmlns:licenseusage="/pcsf/licenseusage" xmlns:webserviceshub="/pcsf/webserviceshub"'
            ) description,
    extractvalue(VALUE(i),
            'info/email',
            'xmlns:common="/pcsf/common" xmlns:usermanagement="/pcsf/usermanagement"
xmlns:domainservice="/pcsf/domainservice" xmlns:logservice="/pcsf/logservice"
xmlns:domainbackup="/pcsf/domainbackup" xmlns:xsi="/2001/XMLSchema-instance"
xmlns:metadata="/pcsf/metadata" xmlns:xsd="/2001/XMLSchema"
xmlns:domainconfigservice="/pcsf/domainconfigservice" xmlns:alertservice="/pcsf/alertservice"
xmlns:licenseusage="/pcsf/licenseusage" xmlns:webserviceshub="/pcsf/webserviceshub"'
            ) email,
    extractvalue(VALUE(i),
            'info/phone',
            'xmlns:common="/pcsf/common" xmlns:usermanagement="/pcsf/usermanagement"
xmlns:domainservice="/pcsf/domainservice" xmlns:logservice="/pcsf/logservice"
xmlns:domainbackup="/pcsf/domainbackup" xmlns:xsi="/2001/XMLSchema-instance"
xmlns:metadata="/pcsf/metadata" xmlns:xsd="/2001/XMLSchema"
xmlns:domainconfigservice="/pcsf/domainconfigservice" xmlns:alertservice="/pcsf/alertservice"
xmlns:licenseusage="/pcsf/licenseusage" xmlns:webserviceshub="/pcsf/webserviceshub"'
            ) phone,
    extractvalue(VALUE(i),
            'info/readOnly',
            'xmlns:common="/pcsf/common" xmlns:usermanagement="/pcsf/usermanagement"
xmlns:domainservice="/pcsf/domainservice" xmlns:logservice="/pcsf/logservice"
xmlns:domainbackup="/pcsf/domainbackup" xmlns:xsi="/2001/XMLSchema-instance"
xmlns:metadata="/pcsf/metadata" xmlns:xsd="/2001/XMLSchema"
xmlns:domainconfigservice="/pcsf/domainconfigservice" xmlns:alertservice="/pcsf/alertservice"
xmlns:licenseusage="/pcsf/licenseusage" xmlns:webserviceshub="/pcsf/webserviceshub"'
            ) readOnly,
    extractvalue(VALUE(i),
            'info/disable',
            'xmlns:common="/pcsf/common" xmlns:usermanagement="/pcsf/usermanagement"
xmlns:domainservice="/pcsf/domainservice" xmlns:logservice="/pcsf/logservice"
xmlns:domainbackup="/pcsf/domainbackup" xmlns:xsi="/2001/XMLSchema-instance"
xmlns:metadata="/pcsf/metadata" xmlns:xsd="/2001/XMLSchema"
xmlns:domainconfigservice="/pcsf/domainconfigservice" xmlns:alertservice="/pcsf/alertservice"
xmlns:licenseusage="/pcsf/licenseusage" xmlns:webserviceshub="/pcsf/webserviceshub"'
            ) disable
FROM pcsf_user x,
    TABLE(
     xmlsequence(
            EXTRACT(
              xmltype(xblob_to_clob(x.metadata)),
              '/metadata:User/info',
              'xmlns:common="/pcsf/common" xmlns:usermanagement="/pcsf/usermanagement"
xmlns:domainservice="/pcsf/domainservice" xmlns:logservice="/pcsf/logservice"
xmlns:domainbackup="/pcsf/domainbackup" xmlns:xsi="/2001/XMLSchema-instance"
xmlns:metadata="/pcsf/metadata" xmlns:xsd="/2001/XMLSchema"
xmlns:domainconfigservice="/pcsf/domainconfigservice" xmlns:alertservice="/pcsf/alertservice"
xmlns:licenseusage="/pcsf/licenseusage" xmlns:webserviceshub="/pcsf/webserviceshub"')
      )
    ) i;
    /
```

Step 4 : Execute below sql to create v_user_group view

```
CREATE OR REPLACE VIEW V_USER_GROUP
(GROUP_ID, GROUP_NAME, USER_NAME)
AS
SELECT id group_id,
  extractvalue(xmltype(xblob_to_clob(metadata)),    '/metadata:Group/groupName',    'xmlns:comm
on="/pcsf/common" xmlns:usermanagement="/pcsf/usermanagement"
xmlns:domainservice="/pcsf/domainservice" xmlns:logservice="/pcsf/logservice"
xmlns:domainbackup="/pcsf/domainbackup" xmlns:xsi="/2001/XMLSchema-instance"
xmlns:metadata="/pcsf/metadata" xmlns:xsd="/2001/XMLSchema"
xmlns:domainconfigservice="/pcsf/domainconfigservice" xmlns:alertservice="/pcsf/alertservice"
xmlns:licenseusage="/pcsf/licenseusage" xmlns:webserviceshub="/pcsf/webserviceshub"')
group_name,
  extractvalue(VALUE(p),    'userRef/userName',    'xmlns:common="/pcsf/common"
xmlns:usermanagement="/pcsf/usermanagement" xmlns:domainservice="/pcsf/domainservice"
xmlns:logservice="/pcsf/logservice" xmlns:domainbackup="/pcsf/domainbackup"
xmlns:xsi="/2001/XMLSchema-instance" xmlns:metadata="/pcsf/metadata"
xmlns:xsd="/2001/XMLSchema" xmlns:domainconfigservice="/pcsf/domainconfigservice"
xmlns:alertservice="/pcsf/alertservice" xmlns:licenseusage="/pcsf/licenseusage"
xmlns:webserviceshub="/pcsf/webserviceshub"') AS
user_name
FROM pcsf_group x,
TABLE(xmlsequence(EXTRACT(xmltype(xblob_to_clob(x.metadata)),    '/metadata:Group/userRef',
'xmlns:common="/pcsf/common" xmlns:usermanagement="/pcsf/usermanagement"
xmlns:domainservice="/pcsf/domainservice" xmlns:logservice="/pcsf/logservice"
xmlns:domainbackup="/pcsf/domainbackup" xmlns:xsi="/2001/XMLSchema-instance"
xmlns:metadata="/pcsf/metadata" xmlns:xsd="/2001/XMLSchema"
xmlns:domainconfigservice="/pcsf/domainconfigservice" xmlns:alertservice="/pcsf/alertservice"
xmlns:licenseusage="/pcsf/licenseusage" xmlns:webserviceshub="/pcsf/webserviceshub"'))) p
```

```
/
```

15.1 USER , GROUP AND STATUS OF USER

```
select distinct 'DEV8_ABCD_REPO' as Repository ,(select name from pcsf_domain) domain_name, group_name,a.
user_name,description,
decode(disable,'true','Disabled','false','Enabled',NULL) as user_status
from v_users a, v_user_group b
where a.user_name = b.user_name
and namespace = 'Native'
order by 1,2,3,4
```

1.1 LIST OF WORKFLOWS CURRENTLY RUNNING

```
SELECT DISTINCT subject_area,   workflow_name, server_name,
                start_time, end_time, TO_CHAR (start_time, 'Day') weekday,
                TRUNC (TO_CHAR ((end_time - start_time) * 1440)
                 ) elapsedminutes,
            CASE
              WHEN run_status_code = '1'    THEN 'Succeeded'
              WHEN run_status_code = '2'    THEN 'Disabled'
              WHEN run_status_code = '3'    THEN 'Failed'
              WHEN run_status_code = '4'    THEN 'Stopped'
              WHEN run_status_code = '5'    THEN 'Aborted'
              WHEN run_status_code = '6'    THEN 'Running'
              WHEN run_status_code = '15'  THEN 'Terminated'
            END AS load_status
         FROM rep_wflow_run
        WHERE run_status_code = '6'
      ORDER BY 1,2,3
```

1.2 List of Database used in Repository

The database definition view provides a list of all database definitions in the repository. A database definition includes the source database names, flat file or RDBMS, and the folder where the database definition resides.

```
select subject_area,database_name,def_source from REP_DATABASE_DEFS
order by 1,2,3
```

1.3 List the name of the object, type, date and last saved

```
        SELECT c.subj_name, a.object_name, b.object_type_name,
  TO_DATE (a.last_saved, 'mm/dd/yyyyHH24:mi:ss') newdate
  FROM opb_version_props a, opb_object_type b, opb_subject c
 WHERE a.object_type = b.object_type_id
   AND a.subject_id = c.subj_id
   AND TO_DATE (a.last_saved, 'mm/dd/yyyyHH24:mi:ss') >
                  TO_DATE ('04/22/2012 00:00:00', 'mm/dd/yyyyHH24:mi:ss')
    ORDER BY newdate DESC;
```

1.4 Check Scheduled workflows using pmcmd command

```
pmcmd getservicedetails -sv etltst222_iserv -d domain_ etltst222-u Administrator -p Administrator –schedule
```

1.5 Check Scheduled workflows using queries

```
   SELECT DISTINCT subject_area, workflow_name FROM rep_workflows
 WHERE run_options = 8 AND end_options=2 ORDER BY 1;
```

Please use below values for RUN_OPTIONS & END_OPTIONS based on your requirement.

RUN_OPTIONS
The workflow schedule type. Records the following values for each schedule type:
1 = Run on demand.
2 = Run once.
4 = Run every DELTA_VALUE seconds.
8 = Customized repeat.
16 = Run on Integration Service initialization.
18 = Run on Integration Service initialization and run once.
20 = Run on Integration Service initialization and every DELTA_VALUE seconds.
24 = Run on Integration Service initialization and customized repeat.
32 = Run continuously.

END_OPTIONS
The stop condition option for the workflow schedule type. Records the following values for each stop condition option:
0 = End on a date.
1 = End after the number of runs stored in RUN_COUNT.
2 = Run forever.

1.6 List Target Tables used in session level

```
SELECT subject_area,session_name,session_instance_name,widget_nameas target_table_name,Type_name asTarget_TypeFROM REP_SESS_TBL_LOG
               order by 1,2,3
```

1.7 List Bad file dir name of the session

```
SELECT subject_area,session_name,session_instance_name,bad_file_location FROM REP_SESS_TBL_LOG
--where bad_file_location not like '$PMBadFileDir\%'
               order by 1,2,3
```

1.8 List Throughput of a session

```
SELECT subject_area,session_name,session_instance_name,start_time,end_time,throughput FROM REP_SESS_TBL_LOG
where throughput > 600
               order by 1,2,3
```

1.9 List ports in Expression (I,IO,V,O)

```
SELECT rep_all_mappings.subject_area, rep_all_mappings.mapping_name,

rep_widget_inst.widget_type_name AS transformation_type,
rep_widget_inst.instance_name AS transformation_name, rep_widget_field.field_name AS port_name,
CASE WHEN rep_widget_field.porttype=1 THEN 'I'
WHEN rep_widget_field.porttype=2 THEN 'O'
WHEN rep_widget_field.porttype=3 THEN 'IO'
WHEN rep_widget_field.porttype=32 THEN 'V'
END AS port_type, rep_widget_field.expression
FROM rep_widget_inst, rep_widget_field, rep_all_mappings
WHERE rep_widget_inst.widget_id=rep_widget_field.widget_id
AND rep_widget_inst.mapping_id=rep_all_mappings.mapping_id
AND rep_widget_inst.widget_type IN (5,9)
```

QUERY TO CHECK SESSION PROPERTIES

```
SELECT DISTINCT A.SUBJECT_AREA,A.TASK_NAME AS SESSION_NAME,'HARDCODED_
VALUE', B.FILE_NAME ||'-'|| DIR_NAMEas Value
FROM
REP_ALL_TASKS A ,
OPB_SESS_FILE_VALS B
WHERE
            B.DIR_NAME  NOT LIKE '%PMSourceFileDir%'
          AND B.DIR_NAME  NOT LIKE '%PMBadFileDir%'
          AND B.DIR_NAME  NOT LIKE '%PMTargetFileDir%'
          AND B.DIR_NAME  NOT LIKE '%PMExtProcDir%'
          AND A.TASK_TYPE_NAME = 'Session'
          AND A.TASK_ID = B.SESSION_ID



UNION ALL


SELECT DISTINCT
A.SUBJECT_AREA,
A.TASK_NAME AS WORKFLOW_NAME,'SAVEWFLOGRUNS',
B.ATTR_VALUE AS SAVEWFLOG
FROM
REP_ALL_TASKS A ,
REP_TASK_ATTR   B
WHERE
A.TASK_ID = B.TASK_ID
AND B.ATTR_ID = '4'
AND B.TASK_TYPE = 71
AND B.ATTR_VALUE NOT IN (8,4)

UNION ALL

select distinct cc.subject_area,cc.task_name as session_name,'SAVESESSLOGRUNS',
bb.attr_value as Savesessionlog
from
(select   a.session_id,min(a.config_id) as config_id,a.attr_id from   rep_sess_config_p
arm a
where a.attr_id = '103'
group by a.session_id,a.attr_id) aa,
(select session_id,config_id,attr_value from rep_sess_config_parm
where attr_id = '103'
) bb,
(select subject_area,task_name,task_id from rep_all_tasks ) cc
where aa.session_id = bb.session_id
and aa.config_id=bb.config_id
and bb.session_id = cc.task_id
and bb.attr_value not in (8,4)


UNION ALL


SELECT DISTINCT
A.SUBJECT_AREA,
A.TASK_NAME AS SESSION_NAME,'STOPON_ERRORS',
B.ATTR_VALUE AS STOPONERRORS
FROM
REP_ALL_TASKS A ,
REP_SESS_CONFIG_PARM   B
WHERE
A.TASK_ID = B.SESSION_ID
AND TASK_TYPE_NAME = 'Session'
AND B.ATTR_ID = '202'
AND B.ATTR_VALUE NOT IN (1)

UNION ALL

SELECT DISTINCT
B.SUBJECT_AREA,
B.TASK_NAME AS SES_WF_NAME,'PRMFILE_PATH',
A.ATTR_VALUE AS PRM_FILE_PATH
FROM
OPB_TASK_ATTR A,
REP_ALL_TASKS B
WHERE
A.ATTR_ID IN (1,4)
AND A.TASK_ID = B.TASK_ID
AND A.ATTR_VALUE LIKE '%.prm%'
AND A.ATTR_VALUE NOT LIKE '%$PMExtProcDir%'
AND A.ATTR_VALUE NOT LIKE '%$PMRootDir/prm%'

UNION ALL

SELECT DISTINCT
B.SUBJECT_AREA,
B.TASK_NAME AS SESS_WF_NAME,'COMMIT_INTERVEL',
A.ATTR_VALUE AS COMMITINTERVEL
FROM
OPB_TASK_ATTR A ,
REP_ALL_TASKS B
WHERE
A.ATTR_ID IN (14)
AND A.ATTR_VALUE <> 10000
AND A.TASK_ID = B.TASK_ID
AND TASK_TYPE_NAME IN ('Session')

UNION ALL


SELECT DISTINCT
A.SUBJECT_AREA,A.TASK_NAME,'SESS_WRTBKCMP',
DECODE (B.ATTR_VALUE,0,'TO BE
CHECKED',1,'CHECKED') WRITEBACKWARDCOMPATIBLE
FROM
REP_ALL_TASKS A ,
OPB_TASK_ATTR B
WHERE
A.TASK_ID = B.TASK_ID
AND B.TASK_TYPE IN 68
AND B.ATTR_ID = 17
AND B.ATTR_VALUE <> 1

UNION ALL

SELECT DISTINCT
A.SUBJECT_AREA,A.TASK_NAME,'WF_WRTBKCMP',
DECODE (B.ATTR_VALUE,0,'TO BE
CHECKED',1,'CHECKED') WRITEBACKWARDCOMPATIBLE
FROM
REP_ALL_TASKS A,
OPB_TASK_ATTR B
WHERE
A.TASK_ID = B.TASK_ID
AND B.TASK_TYPE IN 71
AND B.ATTR_ID = 12
AND B.ATTR_VALUE <> 1

UNION ALL
```

```
SELECT DISTINCT REP_ALL_TASKS.SUBJECT_AREA AS FOLDER_NAME,
       REP_ALL_TASKS.TASK_NAME AS SESSION_NAME,'VERBOSE_CHK',
      CASE
        WHEN REP_SESS_CONFIG_PARM.ATTR_VALUE = 0 THEN 'NONE'
        WHEN REP_SESS_CONFIG_PARM.ATTR_VALUE = 1 THEN 'TERSE'
        WHEN REP_SESS_CONFIG_PARM.ATTR_VALUE = 2 THEN 'NORMAL'
      WHEN REP_SESS_CONFIG_PARM.ATTR_VALUE = 3 THEN 'VERBOSE INITIALIZATION'
        WHEN REP_SESS_CONFIG_PARM.ATTR_VALUE = 4 THEN 'VERBOSE DATA'
      END AS OVERRIDE_TRACING
 FROM
REP_REPOSIT_INFO,
REP_ALL_TASKS,
REP_SESS_CONFIG_PARM
 WHERE
REP_ALL_TASKS.TASK_ID = REP_SESS_CONFIG_PARM.SESSION_ID
AND REP_SESS_CONFIG_PARM.ATTR_ID = 204
AND REP_SESS_CONFIG_PARM.ATTR_VALUE NOT IN (0,2)

UNION ALL

SELECT DISTINCT
A.SUBJECT_AREA,
A.TASK_NAME AS SESSION_NAME,
'SAVESESSLOGBY',
DECODE(B.ATTR_VALUE,1,'TIME STAMP','BY RUNS') AS SAVE_SESSION_LOG_BY
FROM
REP_ALL_TASKS A ,
REP_SESS_CONFIG_PARM   B
WHERE
A.TASK_ID = B.SESSION_ID
AND TASK_TYPE_NAME = 'Session'
AND B.ATTR_ID IN ('102')
AND B.ATTR_VALUE <> 0


UNION ALL


select distinct b.subject_area,b.task_name as sess_wf_name,'WFSESSLOGNAME',a.attr_value as sess_wf_log_name
from OPB_TASK_ATTR a , rep_all_tasks b
where a.ATTR_ID in (2,3)
```

```
and a.attr_value like '%.log%'
and a.task_id = b.task_id
and b.task_name||'.log' != a.attr_value



UNION ALL

SELECT DISTINCT
          B.SUBJECT_AREA,
          B.TASK_NAME AS SESS_NAME,
        CASE
        WHEN A.ATTR_ID=101 THEN 'DTM_BUFFER_SIZE'
        WHEN A.ATTR_ID=102 THEN 'COLLECT_PERFORMANCE_DATA'
        WHEN A.ATTR_ID=103 THEN     'INCREMENTAL_AGGREGATION'
        WHEN A.ATTR_ID=104 THEN     'REINITIALIZE_AGGREGATE CACHE'
        WHEN A.ATTR_ID=105 THEN     'ENABLE_HIGH_PRECISION'
        WHEN A.ATTR_ID=106 THEN     'SESSION_ RETRYON_DEADLOCK'
        WHEN A.ATTR_ID=107 THEN     'PUSHDOWN_OPTIMIZATION'
        WHEN A.ATTR_ID=108 THEN 'WRT_PREF_DATA_TO_REPO'
        WHEN A.ATTR_ID=109 THEN 'ALLOW_TEMP_VIEW_PUSHDOWN'
        WHEN A.ATTR_ID=110 THEN 'ALLOW_TEMP_SEQ_PUSHDOWN'
        WHEN A.ATTR_ID=111 THEN 'ALLOW_PUSHDOWN_INCOMP_CNX'

                END AS TYPE,
          A.ATTR_VALUE AS COLLECT_PERFORMANCE_DATA
FROM
        OPB_TASK_ATTR A ,
          REP_ALL_TASKS B
WHERE
          A.ATTR_ID IN (101,102,103,104,105,106,107,108,109,110,111)
        AND A.TASK_ID = B.TASK_ID
        AND B.TASK_TYPE_NAME IN ('Session')
        AND A.ATTR_VALUE NOT IN ('Auto','0')
        ORDER BY 1,2,3 ASC
```