INFORMATICA REPOSITORY QUERIES - PART I

TABLE OF CONTENTS

1        Overview

2        FOLDER

2.1        List folder details

2.2        List of shared folders  
2.3        List of Users and Groups having Privileges on Folders

3        SOURCE

3.1        List of source tables

3.2        List and count of tables in each folder by db type

3.3        List and count of tables overall used  
3.4        List of source tables used in mappings  
3.5        List of Sources tables using as Shortcuts

4        TARGET

4.1        List of Target Tables

4.2        List and count of tables in each folder by db type

4.3        List and count of table overall used

5        TRANSFORMATION

5.1        List of filer transformations

5.2        List of Sequence transformations

5.3        List of tables used as lookups

5.4        List of transformations using sql overrides

5.5        List all transformations

5.6        List all Expression transformations using ‘concat’ function  
5.7        List of all port details of an Expression transformations  
5.8        List of all Expression transformation port links  
5.9        List of LKP transformation port links used in mappings

6        MAPPING

6.1        List mapping names

6.2        List total count of mappings

6.3        List last saved user for a mapping

6.4        List Mapping parameters and variables  
6.5        List all Mappings using PARALLEL hints

7        MAPPLET

7.1        List Mapplets in all folders

7.2        List Mapplet parameters and variables

8        SESSION

8.1        List session names

8.2        List save session log count

8.3        List stop on errors count

8.4        List hardcoded paths

8.5        List parameter file paths

8.6        List session log names

8.7        List commit intervals

8.8        List total source partitions

8.9        List total target partitions

8.10     List DTM Buffer Size

8.11     List collect performance data

8.12     List Incremental Aggregation

8.13     List Reinitialize aggregate cache

8.14     List Enable high precision

8.15     List Session retry on deadlock

8.16     List write backward compatible check

8.17     List over ride tracing

8.18     List save session log by

8.19     List load type

8.20     List 'post\_session\_success\_command' in session  
8.21     List of all emails with attachment  
8.22     List Invalid Sessions and Workflows

9        TASKS

9.1        List command tasks

9.2        List decision tasks

9.3        List Event Wait tasks

10     WORKLET

10.1     List worklet names

10.2     List hierarchies of all workflows and its worklets

11     WORKFLOW

11.1     List workflow names

11.2     List save workflow log count

11.3     List workflow log names

11.4     List write backward compatible check

11.5     List fail\_parent\_if\_task\_fails objects

11.6     List fail\_parent\_if\_task\_dont\_run objects

11.7     List is\_task\_enabled objects

11.8     List treat\_input\_links\_as objects

11.9     List all workflows whose server is not assigned

11.10  List of workflow run details

12     CONNECTIONS

12.1     List of cnxs using alter in env sql

12.2     List of cnxs used in session levels

12.3     List Lotus connection details

12.4     ODBC / SQL Server Connection details

12.5     List of sessions used by a connection  
12.6     List all Connections with User and Privileges

13     REPOSITORY

13.1     Repository Info

13.2     List of objects which are Not Valid

13.3     List of objects which are failed in last 5 days

13.4     List where all a table is used

13.5     List all source and target tables of mapping  
13.6     List comments of all object

14     MISLENIOUS

14.1     Query to find list of objects saved by last user  
  
15     GROUPS & USERS  
15.1  List User, Groups and status

**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 ASTRANSFORMATION\_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         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         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 "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\_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 = 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**

**Query 1 :-**

 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

**Query 2 :-**

select

repository.REPOSITORY\_NAME as repository\_name,

mappings.SUBJECT\_AREA as Folder\_name,

mappings.mapping\_name ,

Base.Instance\_name as Transformation\_name,

base.widget\_type as Transformation\_Type,

base.field\_name as PORT\_NAME,

base.field\_id as PORT\_ORDER,

base.expr as Formula\_Used

from

[Informatica\_2].[dbo].[V\_PC8X\_EXPR\_PARSE] base,

[Informatica\_2].[dbo].[REP\_ALL\_MAPPINGS] mappings,

[Informatica\_2].[dbo].[REP\_REPOSIT\_INFO] repository

where

mappings.mapping\_id=base.mapping\_id and

mappings.mapping\_version\_number=base.mapping\_vers\_num and

base.REPOSITORY\_ID = repository.REPOSITORY\_ID and

mappings.mapping\_Version\_status=0 and mappings.SUBJECT\_AREA = 'User99'

**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\_DATATYPE, 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.USER\_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

[INFORMATICA REPOSITORY QUERIES - PART II](http://aambarish.blogspot.com/2012/04/informatica-repository-queries-part-ii.html)

INFORMATICA REPOSITORY QUERIES - PART II

**TABLE OF CONTENTS**

[INFORMATICA REPOSITORY QUERIES - PART I](http://aambarish.blogspot.com/2012/01/normal-0-false-false-false-en-us-x-none.html)

11     WORKFLOW

11.1     List workflow names

11.2     List save workflow log count

11.3     List workflow log names

11.4     List write backward compatible check

11.5     List fail\_parent\_if\_task\_fails objects

11.6     List fail\_parent\_if\_task\_dont\_run objects

11.7     List is\_task\_enabled objects

11.8     List treat\_input\_links\_as objects

11.9     List all workflows whose server is not assigned

11.10  List of workflow run details

12     CONNECTIONS

12.1     List of cnxs using alter in env sql

12.2     List of cnxs used in session levels

12.3     List Lotus connection details

12.4     ODBC / SQL Server Connection details

12.5     List of sessions used by a connection  
12.6     List all Connections with User and Privileges

13     REPOSITORY

13.1     Repository Info

13.2     List of objects which are Not Valid

13.3     List of objects which are failed in last 5 days

13.4     List where all a table is used

13.5     List all source and target tables of mapping

13.6     List Comments of all objects

14     MISLENIOUS  
  
           14.1     Query to find list of all objects saved by what user   
  
  
  
  
  
  
  
**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\_PARENT\_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\_NAME AS WORKLET\_OR\_SESSION, DECODE (BITAND (OPB\_TASK\_INST.BIT\_OPTIONS, 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\_ENABLED

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\_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\_name fromopb\_users x where 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 z where 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  = 404000 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 x whered.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 x whered.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 + 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

                                            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\_name from 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\_name from opb\_groupsx where 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\_ATTR.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