html><head></head><body><pre style="word-wrap: break-word; white-space: pre-wrap;">-- +----------------------------------------------------------------------------+

-- | Jeffrey M. Hunter |

-- | jhunter@idevelopment.info |

-- | www.idevelopment.info |

-- |----------------------------------------------------------------------------|

-- | Copyright (c) 1998-2012 Jeffrey M. Hunter. All rights reserved. |

-- |----------------------------------------------------------------------------|

-- | DATABASE : Oracle |

-- | FILE : dba\_compare\_schemas.sql |

-- | CLASS : Database Administration |

-- | PURPOSE : This script can be used by developers and DBAs to compare two |

-- | Oracle schemas. This script will generate a report of all |

-- | object discrepancies between two Oracle database schemas. |

-- | |

-- | This script has been tested on the following Oracle database |

-- | versions: 7.3, 8, 8i, 9i, 10g, 11g. |

-- | |

-- | At this time, the following schema object types (and |

-- | attributes) are not compared and reported on within the |

-- | detailed discrepancy sections. Most of them, however, will |

-- | appear in the "Summary" section of the report. |

-- | |

-- | - Comments (On either tables nor columns.) |

-- | - Partitions (Introduced in Oracle8) |

-- | - Object types (Introduced in Oracle8) |

-- | - Nested tables (Introduced in Oracle8) |

-- | - Dimensions (Introduced in Oracle8i) |

-- | - Cluster definitions |

-- | - Auditing metadata |

-- | - Index organized tables (Introduced in Oracle8i) |

-- | - Temporary tables (Introduced in Oracle8i) |

-- | - Snapshots (Also known as materialized views |

-- | in Oracle8 and higher. Also no |

-- | details on snapshot logs and |

-- | refresh groups will be |

-- | generated.) |

-- | - New schema attributes (Introduced in Oracle 9i) |

-- | |

-- | NOTE : As with any code, ensure to test this script in a development |

-- | environment before attempting to run it in production. |

-- +----------------------------------------------------------------------------+

SET PAGESIZE 50000

SET LINESIZE 256

PROMPT

PROMPT +------------------------------------------------------------------------+

PROMPT | COMPARE SCHEMA SCRIPT |

PROMPT |------------------------------------------------------------------------|

PROMPT | |

PROMPT | USAGE |

PROMPT | -----------------------------------------------------------------------|

PROMPT | This SQL script should be run while connected to the Oracle database |

PROMPT | as one of the schemas you would like to compare. You will be prompted |

PROMPT | to enter the Oracle username, password, and Oracle Net Service Name of |

PROMPT | the second (remote) schema you would like to compare against. Lastly, |

PROMPT | you will be asked for the filename of the report you would like this |

PROMPT | script to create for all generated discrepancies. (You can hit [ENTER] |

PROMPT | to accept the default file name.) |

PROMPT | |

PROMPT | NOTE |

PROMPT | -----------------------------------------------------------------------|

PROMPT | The following database objects will be created for use by this script. |

PROMPT | |

PROMPT | [\*] Database Link (remote\_schema\_link) |

PROMPT | [\*] Table (schema\_compare\_temp) |

PROMPT | [\*] PL/SQL Procedure (getLongText) |

PROMPT | [\*] PL/SQL Procedure (getLongText2) |

PROMPT | |

PROMPT | These objects will be dropped at the end of this script. |

PROMPT +------------------------------------------------------------------------+

PROMPT

SET TERMOUT OFF;

COLUMN local\_conn\_info NEW\_VALUE local\_conn\_info NOPRINT;

SELECT 'You are currently connected to the [' ||

sys\_context('USERENV', 'INSTANCE\_NAME') || '] instance as the [' ||

sys\_context('USERENV', 'SESSION\_USER') || '] user.' local\_conn\_info

FROM dual;

SET TERMOUT ON;

PROMPT +------------------------------------------------------------------------+

PROMPT | LOCAL CONNECTION INFORMATION |

PROMPT |------------------------------------------------------------------------|

PROMPT | &amp;local\_conn\_info

PROMPT +------------------------------------------------------------------------+

PROMPT

ACCEPT a1 CHAR PROMPT "Hit &lt;ENTER&gt; to continue or CTL-C to exit this script ... ";

PROMPT

REM +---------------------------------------------------------------------------+

REM | PROMPT USER FOR USERNAME, PASSWORD, AND ORACLE NET SERVICE NAME. |

REM +---------------------------------------------------------------------------+

ACCEPT schema CHAR PROMPT "Enter USERNAME for remote schema: "

ACCEPT password CHAR PROMPT "Enter PASSWORD for remote schema: " HIDE

ACCEPT tns\_name CHAR PROMPT "Enter ORACLE NET SERVICE NAME for remote schema: "

REM +---------------------------------------------------------------------------+

REM | CREATE TEMPORARY DATABASE LINK. |

REM +---------------------------------------------------------------------------+

SET FEEDBACK OFF

SET VERIFY OFF

SET TRIMSPOOL ON

CREATE DATABASE LINK remote\_schema\_link

CONNECT TO &amp;schema IDENTIFIED BY &amp;password

USING '&amp;tns\_name'

/

REM +---------------------------------------------------------------------------+

REM | CONFIGURE A DEFAULT REPORT FILE NAME FOR THIS SCRIPT RUN. THE USER WILL |

REM | BE PROMPTED TO ENTER AN ALTERNATIVE TO THIS DEFAULT. |

REM +---------------------------------------------------------------------------+

SET TERMOUT OFF;

COLUMN dflt\_name NEW\_VALUE dflt\_name NOPRINT;

SELECT 'compare\_' ||

lower(user) || '\_' ||

lower('&amp;schema') || '\_' ||

lower('&amp;tns\_name') dflt\_name

FROM dual;

SET TERMOUT ON;

PROMPT +------------------------------------------------------------------------+

PROMPT | SPECIFY THE DISCREPANCY REPORT FILE NAME |

PROMPT |------------------------------------------------------------------------|

PROMPT | The default report file name is &amp;dflt\_name..lst

PROMPT | |

PROMPT | To use this name, press [ENTER] to continue, otherwise enter an |

PROMPT | alternative. |

PROMPT +------------------------------------------------------------------------+

PROMPT

SET HEADING OFF;

COLUMN report\_name new\_value report\_name NOPRINT;

SELECT

'Using the report name: ' || nvl('&amp;&amp;report\_name','&amp;dflt\_name')

, nvl('&amp;&amp;report\_name','&amp;dflt\_name') || '.lst' report\_name

FROM sys.dual;

spool &amp;report\_name;

SET HEADING ON;

REM +---------------------------------------------------------------------------+

REM | PRINT OUT DATE AND TIME AND OTHER REPORT HEADER INFORMATION. |

REM +---------------------------------------------------------------------------+

SELECT SUBSTR(RPAD(TO\_CHAR(sysdate, 'DD-MON-YYYY HH24:MI:SS'), 25), 1, 25) "Report Date and Time"

FROM dual;

COLUMN local\_schema FORMAT a45 HEADING "Local Schema" TRUNC

COLUMN remote\_schema FORMAT a45 HEADING "Remote Schema" TRUNC

SELECT

user || '@' || c.global\_name local\_schema

, a.username || '@' || b.global\_name remote\_schema

FROM

user\_users@remote\_schema\_link a

, global\_name@remote\_schema\_link b

, global\_name c

WHERE

rownum = 1;

SET FEEDBACK OFF

SET TERMOUT OFF

COLUMN object\_name FORMAT a40 HEADING 'Object Name'

COLUMN object\_type FORMAT a40 HEADING 'Object Type'

COLUMN obj\_count FORMAT 999,999,999 HEADING 'Object Count'

PROMPT

SET HEADING OFF

SET FEEDBACK OFF

SELECT '+----------------------------------------------------------------------+' || chr(10) ||

'| OBJECT SUMMARY |' || chr(10) ||

'+----------------------------------------------------------------------+'

FROM dual;

SET HEADING ON

SET FEEDBACK ON

PROMPT

PROMPT ========================================================

PROMPT Objects missing from local schema - (Summary)

PROMPT ========================================================

SELECT

object\_type

, count(\*) obj\_count

FROM

(select

object\_type

, decode( object\_type

, 'INDEX', decode(substr(object\_name, 1, 5), 'SYS\_C', 'SYS\_C', object\_name)

, 'LOB' , decode(substr(object\_name, 1, 7), 'SYS\_LOB', 'SYS\_LOB', object\_name)

, object\_name)

from user\_objects@remote\_schema\_link

minus

select

object\_type

, decode( object\_type

, 'INDEX', DECODE(SUBSTR(object\_name, 1, 5), 'SYS\_C', 'SYS\_C', object\_name)

, 'LOB', DECODE(SUBSTR(object\_name, 1, 7), 'SYS\_LOB', 'SYS\_LOB', object\_name),

object\_name)

from user\_objects

)

GROUP BY object\_type

ORDER BY object\_type;

PROMPT

PROMPT

PROMPT ========================================================

PROMPT Extraneous objects in local schema - (Summary)

PROMPT ========================================================

SELECT

object\_type

, count(\*) obj\_count

FROM

(select

object\_type

, DECODE( object\_type

, 'INDEX', DECODE (SUBSTR (object\_name, 1, 5), 'SYS\_C', 'SYS\_C', object\_name)

, 'LOB', DECODE (SUBSTR (object\_name, 1, 7), 'SYS\_LOB', 'SYS\_LOB', object\_name)

, object\_name)

from user\_objects

where object\_type != 'DATABASE LINK'

or object\_name NOT LIKE 'REMOTE\_SCHEMA\_LINK.%'

minus

select

object\_type

, DECODE( object\_type

, 'INDEX', DECODE (SUBSTR (object\_name, 1, 5), 'SYS\_C', 'SYS\_C', object\_name)

, 'LOB', DECODE (SUBSTR (object\_name, 1, 7), 'SYS\_LOB', 'SYS\_LOB', object\_name)

, object\_name)

from user\_objects@remote\_schema\_link

)

GROUP BY object\_type

ORDER BY object\_type;

PROMPT

SET HEADING OFF

SET FEEDBACK OFF

SELECT '+----------------------------------------------------------------------+' || chr(10) ||

'| PRIVILEGE DIFFERENCES |' || chr(10) ||

'+----------------------------------------------------------------------+'

FROM dual;

SET HEADING ON

SET FEEDBACK ON

COLUMN granted\_role FORMAT a30 HEADING 'Granted Role'

COLUMN default\_role FORMAT a22 HEADING 'Default Role'

COLUMN os\_granted FORMAT a11 HEADING 'O/S Granted'

COLUMN owner FORMAT a30 HEADING 'Owner'

COLUMN table\_name FORMAT a30 HEADING 'Table Name'

COLUMN schema FORMAT a7 HEADING 'Schema'

COLUMN grantee FORMAT a30 HEADING 'Grantee'

COLUMN privilege FORMAT a40 HEADING 'Privilege'

COLUMN grantable FORMAT a10 HEADING 'Grantable?'

COLUMN admin\_option FORMAT a13 HEADING 'Admin Option?'

PROMPT

PROMPT ========================================================

PROMPT Role privilege discrepancies

PROMPT ========================================================

(

SELECT

granted\_role

, 'Remote' schema

, admin\_option

, default\_role

, os\_granted

FROM

user\_role\_privs@remote\_schema\_link

MINUS

SELECT

granted\_role

, 'Remote' schema

, admin\_option

, default\_role

, os\_granted

FROM

user\_role\_privs

)

UNION ALL

(

SELECT

granted\_role

, 'Local' schema

, admin\_option

, default\_role

, os\_granted

FROM

user\_role\_privs

MINUS

SELECT

granted\_role

, 'Local' schema

, admin\_option

, default\_role

, os\_granted

FROM

user\_role\_privs@remote\_schema\_link

)

ORDER BY 1, 2;

PROMPT

PROMPT ========================================================

PROMPT System privilege discrepancies

PROMPT ========================================================

(

SELECT

privilege

, 'Remote' schema

, admin\_option

FROM

user\_sys\_privs@remote\_schema\_link

MINUS

SELECT

privilege

, 'Remote' schema

, admin\_option

FROM

user\_sys\_privs

)

UNION ALL

(

SELECT

privilege

, 'Local' schema

, admin\_option

FROM

user\_sys\_privs

MINUS

SELECT

privilege

, 'Local' schema

, admin\_option

FROM

user\_sys\_privs@remote\_schema\_link

)

ORDER BY 1, 2;

PROMPT

PROMPT ========================================================

PROMPT Object-level grant discrepancies

PROMPT ========================================================

(

SELECT

owner

, table\_name

, 'Remote' schema

, grantee

, privilege

, grantable

FROM

user\_tab\_privs@remote\_schema\_link

WHERE

(owner, table\_name) IN (

select owner, object\_name

from all\_objects

)

MINUS

SELECT

owner

, table\_name

, 'Remote' schema

, grantee

, privilege

, grantable

FROM user\_tab\_privs

)

UNION ALL

(

SELECT

owner

, table\_name

, 'Local' schema

, grantee

, privilege

, grantable

FROM

user\_tab\_privs

WHERE

(owner, table\_name) IN (

select owner, object\_name

from all\_objects@remote\_schema\_link

)

MINUS

SELECT

owner

, table\_name

, 'Local' schema

, grantee

, privilege

, grantable

FROM

user\_tab\_privs@remote\_schema\_link

)

ORDER BY 1, 2, 3;

PROMPT

SET HEADING OFF

SET FEEDBACK OFF

SELECT '+----------------------------------------------------------------------+' || chr(10) ||

'| OBJECT DIFFERENCES |' || chr(10) ||

'+----------------------------------------------------------------------+'

FROM dual;

SET HEADING ON

SET FEEDBACK ON

PROMPT

PROMPT ========================================================

PROMPT Objects missing from local schema

PROMPT ========================================================

SELECT

DECODE( object\_type

, 'INDEX', DECODE(SUBSTR(object\_name, 1, 5), 'SYS\_C', 'SYS\_C', object\_name)

, 'LOB', DECODE(SUBSTR(object\_name, 1, 7), 'SYS\_LOB', 'SYS\_LOB', object\_name)

, object\_name) object\_name

, object\_type

FROM user\_objects@remote\_schema\_link

MINUS

SELECT

DECODE( object\_type

, 'INDEX', DECODE(SUBSTR(object\_name, 1, 5), 'SYS\_C', 'SYS\_C', object\_name)

, 'LOB', DECODE(SUBSTR(object\_name, 1, 7), 'SYS\_LOB', 'SYS\_LOB', object\_name)

, object\_name) object\_name

, object\_type

FROM user\_objects

ORDER BY object\_type, object\_name;

PROMPT

PROMPT

PROMPT ========================================================

PROMPT Extraneous objects in local schema

PROMPT ========================================================

SELECT

DECODE( object\_type

, 'INDEX', DECODE(SUBSTR(object\_name, 1, 5), 'SYS\_C', 'SYS\_C', object\_name)

, 'LOB', DECODE(SUBSTR(object\_name, 1, 7), 'SYS\_LOB', 'SYS\_LOB', object\_name)

, object\_name) object\_name

, object\_type

FROM

user\_objects

WHERE

object\_type != 'DATABASE LINK'

OR object\_name NOT LIKE 'REMOTE\_SCHEMA\_LINK.%'

MINUS

SELECT

DECODE( object\_type

, 'INDEX', DECODE(SUBSTR(object\_name, 1, 5), 'SYS\_C', 'SYS\_C', object\_name)

, 'LOB', DECODE(SUBSTR(object\_name, 1, 7), 'SYS\_LOB', 'SYS\_LOB', object\_name)

, object\_name) object\_name

, object\_type

FROM

user\_objects@remote\_schema\_link

ORDER BY object\_type, object\_name;

PROMPT

PROMPT

PROMPT ========================================================

PROMPT Objects in local schema that are not valid

PROMPT ========================================================

SELECT object\_name, object\_type, status

FROM user\_objects

WHERE status != 'VALID'

ORDER BY object\_name, object\_type;

PROMPT

PROMPT

PROMPT ========================================================

PROMPT Objects in remote schema that are not valid

PROMPT ========================================================

SELECT object\_name, object\_type, status

FROM user\_objects@remote\_schema\_link

WHERE status != 'VALID'

ORDER BY object\_name, object\_type;

PROMPT

SET HEADING OFF

SET FEEDBACK OFF

SELECT '+----------------------------------------------------------------------+' || chr(10) ||

'| TABLE COLUMN DIFFERENCES |' || chr(10) ||

'+----------------------------------------------------------------------+'

FROM dual;

SET HEADING ON

SET FEEDBACK ON

PROMPT

PROMPT ========================================================

PROMPT Table columns missing from one schema

PROMPT (Discrepancies are not listed in column order)

PROMPT ========================================================

COLUMN table\_name FORMAT a30 HEADING 'Table Name'

COLUMN column\_name FORMAT a30 HEADING 'Column Name'

COLUMN mis FORMAT a17 HEADING 'Missing in Schema'

COLUMN schema FORMAT a7 HEADING 'Schema'

COLUMN nullable FORMAT a8 HEADING 'Nullable?'

COLUMN data\_type FORMAT a9 HEADING 'Data Type'

COLUMN data\_length FORMAT 9999 HEADING 'Length'

COLUMN data\_precision FORMAT 9999 HEADING 'Precision'

COLUMN data\_scale FORMAT 9999 HEADING 'Scale'

COLUMN default\_length FORMAT 9999 HEADING 'Length of Default Value'

(

SELECT

table\_name

, column\_name

, 'Local' mis

FROM user\_tab\_columns@remote\_schema\_link

WHERE table\_name IN (

select table\_name

from user\_tables

)

MINUS

SELECT

table\_name

, column\_name

, 'Local' mis

FROM user\_tab\_columns

)

UNION ALL

(

SELECT

table\_name

, column\_name

, 'Remote' mis

FROM user\_tab\_columns

WHERE table\_name IN (

select table\_name

from user\_tables@remote\_schema\_link

)

MINUS

SELECT

table\_name

, column\_name

, 'Remote' mis

FROM user\_tab\_columns@remote\_schema\_link

)

ORDER BY 1, 2;

PROMPT

PROMPT ========================================================

PROMPT Data type discrepancies for table columns that exist in

PROMPT both schemas

PROMPT ========================================================

(

SELECT

table\_name

, column\_name

, 'Remote' schema

, nullable

, data\_type

, data\_length

, data\_precision

, data\_scale

, default\_length

FROM user\_tab\_columns@remote\_schema\_link

WHERE (table\_name, column\_name) IN (

select table\_name, column\_name

from user\_tab\_columns

)

MINUS

SELECT

table\_name

, column\_name

, 'Remote' schema

, nullable

, data\_type

, data\_length

, data\_precision

, data\_scale

, default\_length

FROM user\_tab\_columns

)

UNION ALL

(

SELECT

table\_name

, column\_name

, 'Local' schema

, nullable

, data\_type

, data\_length

, data\_precision

, data\_scale

, default\_length

FROM user\_tab\_columns

WHERE (table\_name, column\_name) IN (

select table\_name, column\_name

from user\_tab\_columns@remote\_schema\_link

)

MINUS

SELECT

table\_name

, column\_name

, 'Local' schema

, nullable

, data\_type

, data\_length

, data\_precision

, data\_scale

, default\_length

FROM user\_tab\_columns@remote\_schema\_link

)

ORDER BY 1, 2, 3;

PROMPT

SET HEADING OFF

SET FEEDBACK OFF

SELECT '+----------------------------------------------------------------------+' || chr(10) ||

'| INDEX DIFFERENCES |' || chr(10) ||

'+----------------------------------------------------------------------+'

FROM dual;

SET HEADING ON

SET FEEDBACK ON

COLUMN index\_name FORMAT a30 HEADING 'Index Name'

COLUMN schema FORMAT a7 HEADING 'Schema'

COLUMN uniquenes HEADING 'Uniquenes'

COLUMN table\_name FORMAT a30 HEADING 'Table Name'

COLUMN column\_name FORMAT a30 HEADING 'Column Name'

COLUMN column\_position FORMAT 999 HEADING 'Order'

PROMPT

PROMPT ========================================================

PROMPT Index discrepancies for indexes that exist in both

PROMPT schemas

PROMPT ========================================================

(

SELECT

a.index\_name

, 'Remote' schema

, a.uniqueness

, a.table\_name

, b.column\_name

, b.column\_position

FROM

user\_indexes@remote\_schema\_link a

, user\_ind\_columns@remote\_schema\_link b

WHERE

a.index\_name IN (

select index\_name

from user\_indexes

)

AND b.index\_name = a.index\_name

AND b.table\_name = a.table\_name

MINUS

SELECT

a.index\_name

, 'Remote' schema

, a.uniqueness

, a.table\_name

, b.column\_name

, b.column\_position

FROM

user\_indexes a

, user\_ind\_columns b

WHERE

b.index\_name = a.index\_name

AND b.table\_name = a.table\_name

)

UNION ALL

(

SELECT

a.index\_name

, 'Local' schema

, a.uniqueness

, a.table\_name

, b.column\_name

, b.column\_position

FROM

user\_indexes a

, user\_ind\_columns b

WHERE

a.index\_name IN (

select index\_name

from user\_indexes@remote\_schema\_link

)

AND b.index\_name = a.index\_name

AND b.table\_name = a.table\_name

MINUS

SELECT

a.index\_name

, 'Local' schema

, a.uniqueness

, a.table\_name

, b.column\_name

, b.column\_position

FROM

user\_indexes@remote\_schema\_link a

, user\_ind\_columns@remote\_schema\_link b

WHERE

b.index\_name = a.index\_name

AND b.table\_name = a.table\_name

)

ORDER BY 1, 2, 6;

PROMPT

SET HEADING OFF

SET FEEDBACK OFF

SELECT '+----------------------------------------------------------------------+' || chr(10) ||

'| CONSTRAINT DIFFERENCES |' || chr(10) ||

'+----------------------------------------------------------------------+'

FROM dual;

SET HEADING ON

SET FEEDBACK ON

PROMPT

PROMPT ========================================================

PROMPT Constraint discrepancies for tables that exist in both

PROMPT schemas

PROMPT ========================================================

SET FEEDBACK OFF

CREATE TABLE schema\_compare\_temp (

database NUMBER(1)

, object\_name VARCHAR2(30)

, object\_text VARCHAR2(2000)

, hash\_value NUMBER

)

/

DECLARE

CURSOR c1 IS

SELECT constraint\_name, search\_condition

FROM user\_constraints

WHERE search\_condition IS NOT NULL;

CURSOR c2 IS

SELECT constraint\_name, search\_condition

FROM user\_constraints@remote\_schema\_link

WHERE search\_condition IS NOT NULL;

v\_constraint\_name VARCHAR2(30);

v\_search\_condition VARCHAR2(32767);

BEGIN

OPEN c1;

LOOP

FETCH c1 INTO v\_constraint\_name, v\_search\_condition;

EXIT WHEN c1%NOTFOUND;

v\_search\_condition := SUBSTR (v\_search\_condition, 1, 2000);

INSERT INTO schema\_compare\_temp (

database, object\_name, object\_text

) VALUES (

1, v\_constraint\_name, v\_search\_condition

);

END LOOP;

CLOSE c1;

OPEN c2;

LOOP

FETCH c2 INTO v\_constraint\_name, v\_search\_condition;

EXIT WHEN c2%NOTFOUND;

v\_search\_condition := SUBSTR (v\_search\_condition, 1, 2000);

INSERT INTO schema\_compare\_temp (

database, object\_name, object\_text

) VALUES (

2, v\_constraint\_name, v\_search\_condition

);

END LOOP;

CLOSE c2;

COMMIT;

END;

/

SET FEEDBACK ON

COLUMN constraint\_name FORMAT a30 HEADING 'Constraint|Name'

COLUMN schema FORMAT a7 HEADING 'Schema'

COLUMN constraint\_type FORMAT a10 HEADING 'Constraint|Type'

COLUMN table\_name FORMAT a30 HEADING 'Table|Name'

COLUMN r\_constraint\_name FORMAT a30 HEADING 'R Constraint|Name'

COLUMN delete\_rule FORMAT a10 HEADING 'Delete|Rule'

COLUMN status FORMAT a9 HEADING 'Status'

COLUMN object\_text FORMAT a20 HEADING 'Object|Text'

(

SELECT

REPLACE(TRANSLATE(a.constraint\_name,'012345678','999999999'), '9', NULL) constraint\_name

, 'Remote' schema

, a.constraint\_type

, a.table\_name

, a.r\_constraint\_name

, a.delete\_rule

, a.status

, b.object\_text

FROM

user\_constraints@remote\_schema\_link a

, schema\_compare\_temp b

WHERE

a.table\_name IN (

select table\_name

from user\_tables

)

AND b.database(+) = 2

AND b.object\_name(+) = a.constraint\_name

MINUS

SELECT

REPLACE(TRANSLATE(a.constraint\_name,'012345678','999999999'), '9', NULL) constraint\_name

, 'Remote' schema

, a.constraint\_type

, a.table\_name

, a.r\_constraint\_name

, a.delete\_rule

, a.status

, b.object\_text

FROM

user\_constraints a

, schema\_compare\_temp b

WHERE

b.database(+) = 1

AND b.object\_name(+) = a.constraint\_name

)

UNION ALL

(

SELECT

REPLACE(TRANSLATE(a.constraint\_name,'012345678','999999999'), '9', NULL) constraint\_name

, 'Local' schema

, a.constraint\_type

, a.table\_name

, a.r\_constraint\_name

, a.delete\_rule

, a.status

, b.object\_text

FROM

user\_constraints a

, schema\_compare\_temp b

WHERE

a.table\_name IN (

select table\_name

from user\_tables@remote\_schema\_link

)

AND b.database(+) = 1

AND b.object\_name(+) = a.constraint\_name

MINUS

SELECT

REPLACE(TRANSLATE(a.constraint\_name,'012345678','999999999'), '9', NULL) constraint\_name

, 'Local' schema

, a.constraint\_type

, a.table\_name

, a.r\_constraint\_name

, a.delete\_rule

, a.status

, b.object\_text

FROM

user\_constraints@remote\_schema\_link a

, schema\_compare\_temp b

WHERE

b.database(+) = 2

AND b.object\_name(+) = a.constraint\_name

)

ORDER BY 1, 4, 2;

PROMPT

SET HEADING OFF

SET FEEDBACK OFF

SELECT '+----------------------------------------------------------------------+' || chr(10) ||

'| SEQUENCE DIFFERENCES |' || chr(10) ||

'+----------------------------------------------------------------------+'

FROM dual;

SET HEADING ON

SET FEEDBACK ON

PROMPT

PROMPT ========================================================

PROMPT Sequence discrepancies

PROMPT ========================================================

COLUMN sequence\_name FORMAT a30 HEADING 'Sequence|Name'

COLUMN schema FORMAT a7 HEADING 'Schema'

COLUMN min\_value HEADING 'Min.|Value'

COLUMN max\_value HEADING 'Max.|Value'

COLUMN increment\_by HEADING 'Increment|By'

COLUMN cycle\_flag FORMAT a5 HEADING 'Cycle|Flag'

COLUMN order\_flag FORMAT a5 HEADING 'Order|Flag'

COLUMN cache\_size HEADING 'Cache|Size'

(

SELECT

sequence\_name

, 'Remote' schema

, min\_value

, max\_value

, increment\_by

, cycle\_flag

, order\_flag

, cache\_size

FROM

user\_sequences@remote\_schema\_link

MINUS

SELECT

sequence\_name

, 'Remote' schema

, min\_value

, max\_value

, increment\_by

, cycle\_flag

, order\_flag

, cache\_size

FROM

user\_sequences

)

UNION ALL

(

SELECT

sequence\_name

, 'Local' schema

, min\_value

, max\_value

, increment\_by

, cycle\_flag

, order\_flag

, cache\_size

FROM

user\_sequences

MINUS

SELECT

sequence\_name

, 'Local' schema

, min\_value

, max\_value

, increment\_by

, cycle\_flag

, order\_flag

, cache\_size

FROM

user\_sequences@remote\_schema\_link

)

ORDER BY 1, 2;

PROMPT

SET HEADING OFF

SET FEEDBACK OFF

SELECT '+----------------------------------------------------------------------+' || chr(10) ||

'| PRIVATE SYNONYM DIFFERENCES |' || chr(10) ||

'+----------------------------------------------------------------------+'

FROM dual;

SET HEADING ON

SET FEEDBACK ON

PROMPT

PROMPT ========================================================

PROMPT Private synonym discrepancies

PROMPT ========================================================

COLUMN synonym\_name FORMAT a30 HEADING 'Synonym|Name'

COLUMN schema FORMAT a7 HEADING 'Schema'

COLUMN table\_owner FORMAT a20 HEADING 'Table|Owner'

COLUMN table\_name FORMAT a30 HEADING 'Table|Name'

COLUMN db\_link FORMAT a25 HEADING 'DB|Link Name'

(

SELECT

synonym\_name

, 'Remote' schema

, table\_owner

, table\_name

, db\_link

FROM

user\_synonyms@remote\_schema\_link

MINUS

SELECT

synonym\_name

, 'Remote' schema

, table\_owner

, table\_name

, db\_link

FROM user\_synonyms

)

UNION ALL

(

SELECT

synonym\_name

, 'Local' schema

, table\_owner

, table\_name

, db\_link

FROM

user\_synonyms

MINUS

SELECT

synonym\_name

, 'Local' schema

, table\_owner

, table\_name

, db\_link

FROM

user\_synonyms@remote\_schema\_link

)

ORDER BY 1, 2;

PROMPT

SET HEADING OFF

SET FEEDBACK OFF

SELECT '+----------------------------------------------------------------------+' || chr(10) ||

'| PL/SQL DIFFERENCES |' || chr(10) ||

'+----------------------------------------------------------------------+'

FROM dual;

SET HEADING ON

SET FEEDBACK ON

PROMPT

PROMPT ========================================================

PROMPT Source code discrepancies for all packages, procedures,

PROMPT and functions that exist in both schemas

PROMPT (CASE SENSITIVE COMPARISON)

PROMPT ========================================================

COLUMN name FORMAT a30 HEADING 'Source|Name'

COLUMN type FORMAT a20 HEADING 'Source|Type'

COLUMN discrepancies FORMAT 999,999,999 HEADING 'Number|Discrepancies'

SELECT

name

, type

, COUNT(\*) discrepancies

FROM

( ( SELECT name, type, line, text

FROM user\_source@remote\_schema\_link

WHERE (name, type) IN (

SELECT object\_name, object\_type

FROM user\_objects

)

MINUS

SELECT name, type, line, text

FROM user\_source

)

UNION ALL

( SELECT name, type, line, text

FROM user\_source

WHERE (name, type) IN (

SELECT object\_name, object\_type

FROM user\_objects@remote\_schema\_link

)

MINUS

SELECT name, type, line, text

FROM user\_source@remote\_schema\_link

)

)

GROUP BY name, type

ORDER BY name, type;

PROMPT

PROMPT ========================================================

PROMPT Source code discrepancies for all packages, procedures,

PROMPT and functions that exist in both schemas

PROMPT (CASE INSENSITIVE COMPARISON)

PROMPT ========================================================

COLUMN name FORMAT a30 HEADING 'Source|Name'

COLUMN type FORMAT a20 HEADING 'Source|Type'

COLUMN discrepancies FORMAT 999,999,999 HEADING 'Number|Discrepancies'

SELECT

name

, type

, COUNT (\*) discrepancies

FROM

( ( SELECT name, type, line, UPPER(text)

FROM user\_source@remote\_schema\_link

WHERE (name, type) IN (

select object\_name, object\_type

from user\_objects

)

MINUS

SELECT name, type, line, UPPER(text)

FROM user\_source

)

UNION ALL

( SELECT name, type, line, UPPER(text)

FROM user\_source

WHERE (name, type) IN (

select object\_name, object\_type

from user\_objects@remote\_schema\_link

)

MINUS

SELECT name, type, line, UPPER(text)

FROM user\_source@remote\_schema\_link

)

)

GROUP BY name, type

ORDER BY name, type;

PROMPT

SET HEADING OFF

SET FEEDBACK OFF

SELECT '+----------------------------------------------------------------------+' || chr(10) ||

'| TRIGGER DIFFERENCES |' || chr(10) ||

'+----------------------------------------------------------------------+'

FROM dual;

SET HEADING ON

SET FEEDBACK ON

PROMPT

PROMPT ========================================================

PROMPT Trigger discrepancies

PROMPT ========================================================

SET FEEDBACK OFF

TRUNCATE TABLE schema\_compare\_temp

/

DECLARE

CURSOR c1 IS

SELECT trigger\_name, trigger\_body

FROM user\_triggers;

CURSOR c2 IS

SELECT trigger\_name, trigger\_body

FROM user\_triggers@remote\_schema\_link;

v\_trigger\_name VARCHAR2(30);

v\_trigger\_body VARCHAR2(32767);

v\_hash\_value NUMBER;

BEGIN

OPEN c1;

LOOP

FETCH c1 INTO v\_trigger\_name, v\_trigger\_body;

EXIT WHEN c1%NOTFOUND;

v\_trigger\_body := REPLACE(v\_trigger\_body, ' ', NULL);

v\_trigger\_body := REPLACE(v\_trigger\_body, CHR(9), NULL);

v\_trigger\_body := REPLACE(v\_trigger\_body, CHR(10), NULL);

v\_trigger\_body := REPLACE(v\_trigger\_body, CHR(13), NULL);

v\_trigger\_body := UPPER(v\_trigger\_body);

v\_hash\_value := dbms\_utility.get\_hash\_value(v\_trigger\_body, 1, 65536);

INSERT INTO schema\_compare\_temp (

database, object\_name, hash\_value

) VALUES (

1, v\_trigger\_name, v\_hash\_value

);

END LOOP;

CLOSE c1;

OPEN c2;

LOOP

FETCH c2 INTO v\_trigger\_name, v\_trigger\_body;

EXIT WHEN c2%NOTFOUND;

v\_trigger\_body := REPLACE(v\_trigger\_body, ' ', NULL);

v\_trigger\_body := REPLACE(v\_trigger\_body, CHR(9), NULL);

v\_trigger\_body := REPLACE(v\_trigger\_body, CHR(10), NULL);

v\_trigger\_body := REPLACE(v\_trigger\_body, CHR(13), NULL);

v\_trigger\_body := UPPER(v\_trigger\_body);

v\_hash\_value := dbms\_utility.get\_hash\_value(v\_trigger\_body, 1, 65536);

INSERT INTO schema\_compare\_temp (

database, object\_name, hash\_value

) VALUES (

2, v\_trigger\_name, v\_hash\_value

);

END LOOP;

CLOSE c2;

END;

/

SET FEEDBACK ON

COLUMN trigger\_name FORMAT a20 HEADING 'Trigger|Name'

COLUMN schema FORMAT a7 HEADING 'Schema'

COLUMN trigger\_type FORMAT a16 HEADING 'Trigger|Type'

COLUMN triggering\_event FORMAT a20 HEADING 'Triggering|Event'

COLUMN table\_name FORMAT a15 HEADING 'Table|Name'

COLUMN referencing\_names FORMAT a20 HEADING 'Referencing|Names'

COLUMN when\_clause FORMAT a20 HEADING 'When|Clause'

COLUMN status FORMAT a9 HEADING 'Status'

COLUMN hash\_value HEADING 'Hash Value'

( SELECT

a.trigger\_name

, 'Local' schema

, a.trigger\_type

, SUBSTR(a.triggering\_event, 1, 20) triggering\_event

, a.table\_name

, SUBSTR(a.referencing\_names, 1, 20) referencing\_names

, SUBSTR(a.when\_clause, 1, 20) when\_clause

, a.status

, b.hash\_value

FROM

user\_triggers a

, schema\_compare\_temp b

WHERE

b.object\_name(+) = a.trigger\_name

AND b.database(+) = 1

AND a.table\_name IN (

select table\_name

from user\_tables@remote\_schema\_link

)

MINUS

SELECT

a.trigger\_name

, 'Local' schema

, a.trigger\_type

, SUBSTR(a.triggering\_event, 1, 20) triggering\_event

, a.table\_name

, SUBSTR(a.referencing\_names, 1, 20) referencing\_names

, SUBSTR(a.when\_clause, 1, 20) when\_clause

, a.status

, b.hash\_value

FROM

user\_triggers@remote\_schema\_link a

, schema\_compare\_temp b

WHERE

b.object\_name(+) = a.trigger\_name

AND b.database(+) = 2

)

UNION ALL

(

SELECT

a.trigger\_name

, 'Remote' schema

, a.trigger\_type

, SUBSTR(a.triggering\_event, 1, 20) triggering\_event

, a.table\_name

, SUBSTR(a.referencing\_names, 1, 20) referencing\_names

, SUBSTR(a.when\_clause, 1, 20) when\_clause

, a.status

, b.hash\_value

FROM

user\_triggers@remote\_schema\_link a

, schema\_compare\_temp b

WHERE

b.object\_name(+) = a.trigger\_name

AND b.database(+) = 2

AND a.table\_name IN (

select table\_name

from user\_tables

)

MINUS

SELECT

a.trigger\_name

, 'Remote' schema

, a.trigger\_type

, SUBSTR(a.triggering\_event, 1, 20) triggering\_event

, a.table\_name

, SUBSTR(a.referencing\_names, 1, 20) referencing\_names

, SUBSTR(a.when\_clause, 1, 20) when\_clause

, a.status

, b.hash\_value

FROM

user\_triggers a

, schema\_compare\_temp b

WHERE

b.object\_name(+) = a.trigger\_name

AND b.database(+) = 1

)

ORDER BY 1, 2, 5, 3;

PROMPT

SET HEADING OFF

SET FEEDBACK OFF

SELECT '+----------------------------------------------------------------------+' || chr(10) ||

'| VIEW DIFFERENCES |' || chr(10) ||

'+----------------------------------------------------------------------+'

FROM dual;

SET HEADING ON

SET FEEDBACK ON

prompt

prompt ========================================================

prompt View discrepancies for views that exist in both

prompt schemas

prompt ========================================================

SET FEEDBACK OFF

SET LONG 32767

TRUNCATE TABLE schema\_compare\_temp

/

CREATE OR REPLACE FUNCTION getLongText ( p\_tname IN VARCHAR2

, p\_cname IN VARCHAR2

, p\_vname IN VARCHAR2) RETURN VARCHAR2

AS

l\_sql VARCHAR2(4000);

l\_cursor INTEGER DEFAULT dbms\_sql.open\_cursor;

l\_n NUMBER;

l\_long\_val VARCHAR2(4000);

l\_long\_len NUMBER;

l\_buflen NUMBER := 4000;

l\_curpos NUMBER := 0;

BEGIN

l\_sql := 'select ' || p\_cname || ' from ' || p\_tname || ' where UPPER(view\_name) = UPPER(:view\_name)';

DBMS\_SQL.PARSE( l\_cursor

, l\_sql

, DBMS\_SQL.NATIVE);

DBMS\_SQL.BIND\_VARIABLE(l\_cursor, ':view\_name', p\_vname);

DBMS\_SQL.DEFINE\_COLUMN\_LONG(l\_cursor, 1);

l\_n := DBMS\_SQL.EXECUTE(l\_cursor);

IF (DBMS\_SQL.FETCH\_ROWS(l\_cursor) &gt; 0)

THEN

DBMS\_SQL.COLUMN\_VALUE\_LONG( l\_cursor

, 1

, l\_buflen

, l\_curpos

, l\_long\_val

, l\_long\_len);

END IF;

DBMS\_SQL.CLOSE\_CURSOR(l\_cursor);

RETURN l\_long\_val;

END getLongText;

/

CREATE OR REPLACE FUNCTION getLongText2 ( p\_tname IN VARCHAR2

, p\_cname IN VARCHAR2

, p\_vname IN VARCHAR2) RETURN VARCHAR2

AS

l\_sql VARCHAR2(4000);

l\_cursor INTEGER DEFAULT dbms\_sql.open\_cursor;

l\_n NUMBER;

l\_long\_val VARCHAR2(4000);

l\_long\_len NUMBER;

l\_buflen NUMBER := 4000;

l\_curpos NUMBER := 0;

BEGIN

l\_sql := 'select ' || p\_cname || ' from ' || p\_tname || '@remote\_schema\_link where UPPER(view\_name) = UPPER(:view\_name)';

DBMS\_SQL.PARSE( l\_cursor

, l\_sql

, DBMS\_SQL.NATIVE);

DBMS\_SQL.BIND\_VARIABLE(l\_cursor, ':view\_name', p\_vname);

DBMS\_SQL.DEFINE\_COLUMN\_LONG(l\_cursor, 1);

l\_n := DBMS\_SQL.EXECUTE(l\_cursor);

IF (DBMS\_SQL.FETCH\_ROWS(l\_cursor) &gt; 0)

THEN

DBMS\_SQL.COLUMN\_VALUE\_LONG( l\_cursor

, 1

, l\_buflen

, l\_curpos

, l\_long\_val

, l\_long\_len);

END IF;

DBMS\_SQL.CLOSE\_CURSOR(l\_cursor);

RETURN l\_long\_val;

END getLongText2;

/

DECLARE

CURSOR c1 IS

SELECT view\_name, getLongText('USER\_VIEWS', 'TEXT', view\_name)

FROM user\_views;

CURSOR c2 IS

SELECT view\_name, getLongText2('USER\_VIEWS', 'TEXT', view\_name)

FROM user\_views@remote\_schema\_link;

v\_view\_name VARCHAR2(30);

v\_text VARCHAR2(32767);

v\_hash\_value NUMBER;

BEGIN

OPEN c1;

LOOP

FETCH c1 INTO v\_view\_name, v\_text;

EXIT WHEN c1%NOTFOUND;

v\_hash\_value := dbms\_utility.get\_hash\_value(v\_text, 1, 65536);

INSERT INTO schema\_compare\_temp (

database, object\_name, object\_text, hash\_value

) VALUES (

1, v\_view\_name, '[' || v\_text || ']', v\_hash\_value

);

END LOOP;

CLOSE c1;

OPEN c2;

LOOP

FETCH c2 INTO v\_view\_name, v\_text;

EXIT WHEN c2%NOTFOUND;

v\_hash\_value := dbms\_utility.get\_hash\_value(v\_text, 1, 65536);

INSERT INTO schema\_compare\_temp (

database, object\_name, object\_text, hash\_value

) VALUES (

2, v\_view\_name, '[' || v\_text || ']', v\_hash\_value

);

END LOOP;

CLOSE c2;

END;

/

SET FEEDBACK ON

COLUMN view\_name FORMAT a30 HEADING 'View|Name'

COLUMN schema FORMAT a7 HEADING 'Schema'

COLUMN hash\_value HEADING 'Hash Value'

(

SELECT

a.view\_name

, 'Local' schema

, b.hash\_value

FROM

user\_views a

, schema\_compare\_temp b

WHERE

b.object\_name(+) = a.view\_name

AND b.database(+) = 1

AND a.view\_name IN (

select view\_name

from user\_views@remote\_schema\_link

)

MINUS

SELECT

a.view\_name

, 'Local' schema

, b.hash\_value

FROM

user\_views@remote\_schema\_link a

, schema\_compare\_temp b

WHERE

b.object\_name(+) = a.view\_name

AND b.database(+) = 2

)

UNION ALL

(

SELECT

a.view\_name

, 'Remote' schema

, b.hash\_value

FROM

user\_views@remote\_schema\_link a

, schema\_compare\_temp b

WHERE

b.object\_name(+) = a.view\_name

AND b.database(+) = 2

AND a.view\_name IN (

select view\_name

from user\_views

)

MINUS

SELECT

a.view\_name

, 'Remote' schema

, b.hash\_value

FROM

user\_views a

, schema\_compare\_temp b

WHERE

b.object\_name(+) = a.view\_name

AND b.database(+) = 1

)

ORDER BY 1, 2;

PROMPT

SET HEADING OFF

SET FEEDBACK OFF

SELECT '+----------------------------------------------------------------------+' || chr(10) ||

'| JOB QUEUE DIFFERENCES |' || chr(10) ||

'+----------------------------------------------------------------------+'

FROM dual;

SET HEADING ON

SET FEEDBACK ON

PROMPT

PROMPT ========================================================

PROMPT Job queue discrepancies

PROMPT ========================================================

COLUMN what FORMAT a30 HEADING 'What'

COLUMN interval FORMAT a30 HEADING 'Interval'

COLUMN broken FORMAT a7 HEADING 'Broken?'

(

SELECT

what

, interval

, broken

, 'Remote' schema

FROM

user\_jobs@remote\_schema\_link

MINUS

SELECT

what

, interval

, broken

, 'Remote' schema

FROM

user\_jobs

)

UNION ALL

(

SELECT

what

, interval

, broken

, 'Local' schema

FROM

user\_jobs

MINUS

SELECT

what

, interval

, broken

, 'Local' schema

FROM

user\_jobs@remote\_schema\_link

)

ORDER BY 1, 2, 3;

PROMPT

SET HEADING OFF

SET FEEDBACK OFF

SELECT '+----------------------------------------------------------------------+' || chr(10) ||

'| DATABASE LINK DIFFERENCES |' || chr(10) ||

'+----------------------------------------------------------------------+'

FROM dual;

SET HEADING ON

SET FEEDBACK ON

PROMPT

PROMPT ========================================================

PROMPT Database link discrepancies

PROMPT ========================================================

COLUMN db\_link FORMAT a30 HEADING 'DB Link Name'

COLUMN schema FORMAT a7 HEADING 'Schema'

COLUMN username FORMAT a20 HEADING 'User Name'

COLUMN host FORMAT a20 HEADING 'Host'

(

SELECT

db\_link

, 'Remote' schema

, username

, host

FROM

user\_db\_links@remote\_schema\_link

MINUS

SELECT

db\_link

, 'Remote' schema

, username, host

FROM

user\_db\_links

)

UNION ALL

(

SELECT

db\_link

, 'Local' schema

, username, host

FROM

user\_db\_links

WHERE

db\_link NOT LIKE 'REMOTE\_SCHEMA\_LINK.%'

MINUS

SELECT

db\_link

, 'Local' schema

, username

, host

FROM

user\_db\_links@remote\_schema\_link

)

ORDER BY 1, 2;

SPOOL OFF

SET TERMOUT ON

PROMPT

PROMPT =============

PROMPT END OF REPORT

PROMPT =============

PROMPT

PROMPT Report output written to &amp;report\_name

PROMPT ==============================================================

SET FEEDBACK OFF

DROP TABLE schema\_compare\_temp;

DROP DATABASE LINK remote\_schema\_link;

DROP FUNCTION getLongText;

DROP FUNCTION getLongText2;

SET FEEDBACK 6

</pre></body></html>