```
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 ------------------------+
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
```

```
&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('&&report_name','&dflt_name')
   , nvl('&&report_name','&dflt_name') ||
'.lst' report_name
FROM sys.dual;
spool &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
```

```sql
REPLACE(TRANSLATE(a.constraint_name,'012345678','999999
999'), '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','999999
999'), '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','999999
999'), '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','999999
999'), '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
```

```sql
       , 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>
```