

**Update WHRSC Customer Base in EWITS 2.0**

**Updating WHRSC Customer Base in EWITS 2.0**

# Purpose of the Document

##### This document describes the step by step process to update the list of customers for WHRSC in the HHS\_WHRSC\_HR schema to be used by EWITS 2.0.

##### Whenever there is any update in the list of customers, WHRSC will release the finalized list of Customers and respective Admin Codes as shown below. [Admin Codes by WHRSC Supported Program.xls]

##### 

##### Every morning, a batch program [whrsc-interface] imports the admin codes, their description, respective opdivs and staffdivs from HHS\_HR.ADMINISTRATIVE\_CODE table into HHS\_WHRSC\_HR.ORGS table as part of importing CapHR data from the global schema into WHRSC schema. An update script is then executed to update the customers (IC) in the ORGS table. This script is also executed for updating customers in CHR\_EMPLOYEE\_INFO table.

##### A DBA (with Admin privileges) should log into the EWITS 2.0 database as HHS\_WHRSC\_HR user and follow the steps listed below:

1. Open and edit the CAPHR\_DATA\_PKS Oracle package.
2. If there are any changes in the Admin Codes, edit the cursor CUR\_ADMIN\_CODE as shown below. If there are no changes in the list of admin codes, go to next step.

--------------------------------------------------------

--CURSOR: CUR\_ADMIN\_CODE

--DESCRIPTION: Fetches records from the ADMINISTRATIVE\_CODE table

--based on the Admin Codes for WHRSC in 2018

--------------------------------------------------------

CURSOR CUR\_ADMIN\_CODE

IS

SELECT ADMIN\_CODE,

ADMIN\_CODE\_DESC,

OPDIV,

OPDIV\_NAME,

STAFFDIV,

STAFFDIV\_NAME

FROM HHS\_HR.ADMINISTRATIVE\_CODE

WHERE OPDIV IN ('ACF','ACL','AHRQ','ASA','ASFR','ASL','ASPA','ASPE','ASPR','DAB','IEA','IOS','OASH','OCIO','OCR','OGA','OGC','ONC','OSSI','PSC','SAMHSA')

AND **ADMIN\_CODE LIKE 'AX%' OR ADMIN\_CODE LIKE 'AB' OR ADMIN\_CODE LIKE 'AA%' OR ADMIN\_CODE LIKE 'ABC%' OR ADMIN\_CODE LIKE 'ABE%' OR ADMIN\_CODE LIKE 'AC%' OR ADMIN\_CODE LIKE 'AE%' OR ADMIN\_CODE LIKE 'AG%' OR ADMIN\_CODE LIKE 'AH%' OR ADMIN\_CODE LIKE 'AJ%' OR ADMIN\_CODE LIKE 'AJG%' OR ADMIN\_CODE LIKE 'AL%' OR ADMIN\_CODE LIKE 'AM%' OR ADMIN\_CODE LIKE 'AN%' OR ADMIN\_CODE LIKE 'AP%' OR ADMIN\_CODE LIKE 'AQ%' OR ADMIN\_CODE LIKE 'AR%' OR ADMIN\_CODE LIKE 'AT%' OR ADMIN\_CODE LIKE 'B%' OR ADMIN\_CODE LIKE 'E%' OR ADMIN\_CODE LIKE 'K%' OR ADMIN\_CODE LIKE 'M%' OR ADMIN\_CODE LIKE 'P%' OR ADMIN\_CODE LIKE 'AD%';**

TYPE TYP\_ADMIN\_CODE IS TABLE OF CUR\_ADMIN\_CODE%ROWTYPE

INDEX BY PLS\_INTEGER;

ADMINCODES TYP\_ADMIN\_CODE;

-----------------------------------------------------------------------------------------------------------

-----------------------------------------------------------------------------------------------

1. Edit the update script in the INSERT\_ADMIN\_CODES procedure as shown below based on the latest list of customers and admin codes release by WHRSC.

---------------------------------------------------------

--PROCEDURE: INSERT\_ADMIN\_CODES

---------------------------------------------------------

PROCEDURE INSERT\_ADMIN\_CODES

AS

BEGIN

OPEN CUR\_ADMIN\_CODE;

FETCH CUR\_ADMIN\_CODE BULK COLLECT INTO ADMINCODES;

CLOSE CUR\_ADMIN\_CODE;

IF ADMINCODES.COUNT > 0 THEN

DBMS\_OUTPUT.PUT\_LINE('COUNT: ' || ADMINCODES.COUNT);

--Truncate ORGS and CHR\_WGI tables

EXECUTE IMMEDIATE 'TRUNCATE TABLE HHS\_WHRSC\_HR.ORGS';

EXECUTE IMMEDIATE 'TRUNCATE TABLE HHS\_WHRSC\_HR.CHR\_WGI';

--FORALL i IN ADMINCODES.FIRST..ADMINCODES.LAST SAVE EXCEPTIONS

FOR i IN ADMINCODES.FIRST.. ADMINCODES.LAST LOOP

--Insert record into ORGS table

BEGIN

INSERT INTO HHS\_WHRSC\_HR.ORGS

( ADMIN\_CODE,

ADMIN\_TITLE,

ORG\_INITS)

VALUES

( ADMINCODES(i).ADMIN\_CODE,

ADMINCODES(i).ADMIN\_CODE\_DESC,

ADMINCODES(i).OPDIV\_NAME );

--Insert record into CHR\_WGI table

INSERT INTO HHS\_WHRSC\_HR.CHR\_WGI

( DEPT\_ID,

NAME )

VALUES

( ADMINCODES(i).ADMIN\_CODE,

ADMINCODES(i).STAFFDIV\_NAME );

EXCEPTION

WHEN OTHERS THEN

SP\_ERROR\_LOG();

END;

END LOOP;

**--Update ORGS table with IC for ADMIN\_CODE**

**UPDATE ORGS SET IC = CASE**

**WHEN ADMIN\_CODE LIKE 'AA%' OR ADMIN\_CODE LIKE 'AB' OR ADMIN\_CODE LIKE 'AX%' THEN 'IOS'**

**WHEN ADMIN\_CODE LIKE 'ABC%' OR ADMIN\_CODE LIKE 'AD%'THEN 'IEA'**

**WHEN ADMIN\_CODE LIKE 'ABE%' THEN 'OSSI'**

**WHEN ADMIN\_CODE LIKE 'AC%' THEN 'OASH'**

**WHEN ADMIN\_CODE LIKE 'AE%' THEN 'ASPE'**

**WHEN ADMIN\_CODE LIKE 'AG%' THEN 'OGC'**

**WHEN ADMIN\_CODE LIKE 'AH%' THEN 'DAB'**

**WHEN ADMIN\_CODE LIKE 'AJ%' AND ADMIN\_CODE NOT LIKE 'AJG%' THEN 'ASA'**

**WHEN ADMIN\_CODE LIKE 'AJG%' THEN 'OCIO'**

**WHEN ADMIN\_CODE LIKE 'AL%' THEN 'ASL'**

**WHEN ADMIN\_CODE LIKE 'AM%' THEN 'ASFR'**

**WHEN ADMIN\_CODE LIKE 'AN%' THEN 'ASPR'**

**WHEN ADMIN\_CODE LIKE 'AP%' THEN 'ASPA'**

**WHEN ADMIN\_CODE LIKE 'AQ%' THEN 'OGA'**

**WHEN ADMIN\_CODE LIKE 'AR%' THEN 'ONC'**

**WHEN ADMIN\_CODE LIKE 'AT%' THEN 'OCR'**

**WHEN ADMIN\_CODE LIKE 'B%' THEN 'ACL'**

**WHEN ADMIN\_CODE LIKE 'E%' THEN 'AHRQ'**

**WHEN ADMIN\_CODE LIKE 'K%' THEN 'ACF'**

**WHEN ADMIN\_CODE LIKE 'M%' THEN 'SAMHSA'**

**WHEN ADMIN\_CODE LIKE 'P%' THEN 'PSC'**

**ELSE ' ' END;**

COMMIT;

END IF;

END INSERT\_ADMIN\_CODES;

----------------------------------------------------------------------------------

----------------------------------------------------------------------------------

1. In the same CAPHR\_DATA\_PKS Oracle package, edit the update script in the INSERT\_CHR\_EMP\_INFO procedure as shown below based on the latest list of customers and admin codes release by WHRSC.

---------------------------------------------------------

--PROCEDURE: INSERT\_CHR\_EMP\_INFO

--DESCRIPTION :

---------------------------------------------------------

PROCEDURE INSERT\_CHR\_EMP\_INFO

AS

V\_JOB\_IDX NUMBER;

V\_MAX\_GRADE VARCHAR2(30);

BEGIN

OPEN CUR\_CAPHR\_JOB\_OPENINGS;

FETCH CUR\_CAPHR\_JOB\_OPENINGS

BULK COLLECT INTO V\_TBL\_JOB\_OPENINGS;

CLOSE CUR\_CAPHR\_JOB\_OPENINGS;

IF V\_TBL\_JOB\_OPENINGS.COUNT > 0 THEN

DBMS\_OUTPUT.PUT\_LINE('COUNT: ' || V\_TBL\_JOB\_OPENINGS.COUNT);

FOR i IN V\_TBL\_JOB\_OPENINGS.FIRST .. V\_TBL\_JOB\_OPENINGS.LAST LOOP

SELECT MAX(POS.GRADE) INTO V\_MAX\_GRADE FROM HHS\_HR.PS\_HRS\_JO\_POSN JOP,HHS\_HR.PS\_HRS\_JOB\_OPENING JO,HHS\_HR.PS\_POSITION\_DATA POS

WHERE JO.HRS\_JOB\_OPENING\_ID = JOP.HRS\_JOB\_OPENING\_ID

AND POS.POSITION\_NBR = JOP.POSITION\_NBR

AND JO.HRS\_JOB\_OPENING\_ID = V\_TBL\_JOB\_OPENINGS(i).HRS\_JOB\_OPENING\_ID ;

IF NOT V\_TBL\_JOBS\_CACHE.EXISTS(V\_TBL\_JOB\_OPENINGS(i).HRS\_JOB\_OPENING\_ID) THEN --If job opening id does not exist

--Create a cache to store job opening id and grade, index on job opening id

V\_TBL\_JOBS\_CACHE(V\_TBL\_JOB\_OPENINGS(i).HRS\_JOB\_OPENING\_ID).ID := V\_TBL\_JOB\_OPENINGS(i).HRS\_JOB\_OPENING\_ID;

V\_JOB\_IDX := V\_TBL\_INS\_EMP\_INFO.COUNT;

--Add to collection for insert

V\_TBL\_INS\_EMP\_INFO(V\_JOB\_IDX).ID := V\_TBL\_JOB\_OPENINGS(i).HRS\_JOB\_OPENING\_ID;

V\_TBL\_INS\_EMP\_INFO(V\_JOB\_IDX).DEPT\_ID := V\_TBL\_JOB\_OPENINGS(i).GVT\_RECR\_OFFICE;

V\_TBL\_INS\_EMP\_INFO(V\_JOB\_IDX).JOB\_TITLE := V\_TBL\_JOB\_OPENINGS(i).GVT\_ORG\_TTL\_DESCR;

V\_TBL\_INS\_EMP\_INFO(V\_JOB\_IDX).PAY\_PLAN := V\_TBL\_JOB\_OPENINGS(i).GVT\_PAY\_PLAN;

V\_TBL\_INS\_EMP\_INFO(V\_JOB\_IDX).OCC\_SERIES := V\_TBL\_JOB\_OPENINGS(i).GVT\_OCC\_SERIES;

V\_TBL\_INS\_EMP\_INFO(V\_JOB\_IDX).GRADE := V\_MAX\_GRADE;

V\_TBL\_INS\_EMP\_INFO(V\_JOB\_IDX).DATE\_ADDED := SYSDATE;

V\_TBL\_INS\_EMP\_INFO(V\_JOB\_IDX).DATE\_JOB\_REQ\_APPROVED := V\_TBL\_JOB\_OPENINGS(i).AUTHORIZATION\_DT;

V\_TBL\_INS\_EMP\_INFO(V\_JOB\_IDX).CAPHR\_COMMENTS := V\_TBL\_JOB\_OPENINGS(i).HE\_COMMENTS;

V\_TBL\_INS\_EMP\_INFO(V\_JOB\_IDX).CAN\_NUMBER := V\_TBL\_JOB\_OPENINGS(i).ACCT\_CD;

END IF;

END LOOP;

END IF;

IF V\_TBL\_INS\_EMP\_INFO.COUNT > 0 THEN

--SP\_INSERT\_CHR\_EMP\_INFO(V\_TBL\_INS\_EMP\_INFO);

FOR i IN V\_TBL\_INS\_EMP\_INFO.FIRST.. V\_TBL\_INS\_EMP\_INFO.LAST LOOP

BEGIN

INSERT INTO HHS\_WHRSC\_HR.CHR\_EMPLOYEE\_INFO

(ID,

DEPT\_ID,

JOB\_TITLE,

PAY\_PLAN,

OCC\_SERIES,

GRADE,

DATE\_ADDED,

DATE\_JOB\_REQ\_APPROVED,

CAPHR\_COMMENTS,

CAN\_NUMBER)

VALUES

(V\_TBL\_INS\_EMP\_INFO(i).ID,

V\_TBL\_INS\_EMP\_INFO(i).DEPT\_ID,

V\_TBL\_INS\_EMP\_INFO(i).JOB\_TITLE,

V\_TBL\_INS\_EMP\_INFO(i).PAY\_PLAN,

V\_TBL\_INS\_EMP\_INFO(i).OCC\_SERIES,

V\_TBL\_INS\_EMP\_INFO(i).GRADE,

V\_TBL\_INS\_EMP\_INFO(i).DATE\_ADDED,

V\_TBL\_INS\_EMP\_INFO(i).DATE\_JOB\_REQ\_APPROVED,

V\_TBL\_INS\_EMP\_INFO(i).CAPHR\_COMMENTS,

V\_TBL\_INS\_EMP\_INFO(i).CAN\_NUMBER);

EXCEPTION

WHEN OTHERS THEN

SP\_ERROR\_LOG();

END;

END LOOP;

**-- UPDATE HHS\_WHRSC\_HR.CHR\_EMPLOYEE\_INFO**

**-- SET IC = (SELECT O.IC FROM HHS\_WHRSC\_HR.ORGS O WHERE O.ADMIN\_CODE = DEPT\_ID);**

**UPDATE CHR\_EMPLOYEE\_INFO SET IC = CASE**

**WHEN DEPT\_ID LIKE 'AA%' OR DEPT\_ID LIKE 'AB' OR DEPT\_ID LIKE 'AX%' THEN 'IOS'**

**WHEN DEPT\_ID LIKE 'ABC%' OR DEPT\_ID LIKE 'AD%'THEN 'IEA'**

**WHEN DEPT\_ID LIKE 'ABE%' THEN 'OSSI'**

**WHEN DEPT\_ID LIKE 'AC%' THEN 'OASH'**

**WHEN DEPT\_ID LIKE 'AE%' THEN 'ASPE'**

**WHEN DEPT\_ID LIKE 'AG%' THEN 'OGC'**

**WHEN DEPT\_ID LIKE 'AH%' THEN 'DAB'**

**WHEN DEPT\_ID LIKE 'AJ%' AND DEPT\_ID NOT LIKE 'AJG%' THEN 'ASA'**

**WHEN DEPT\_ID LIKE 'AJG%' THEN 'OCIO'**

**WHEN DEPT\_ID LIKE 'AL%' THEN 'ASL'**

**WHEN DEPT\_ID LIKE 'AM%' THEN 'ASFR'**

**WHEN DEPT\_ID LIKE 'AN%' THEN 'ASPR'**

**WHEN DEPT\_ID LIKE 'AP%' THEN 'ASPA'**

**WHEN DEPT\_ID LIKE 'AQ%' THEN 'OGA'**

**WHEN DEPT\_ID LIKE 'AR%' THEN 'ONC'**

**WHEN DEPT\_ID LIKE 'AT%' THEN 'OCR'**

**WHEN DEPT\_ID LIKE 'B%' THEN 'ACL'**

**WHEN DEPT\_ID LIKE 'E%' THEN 'AHRQ'**

**WHEN DEPT\_ID LIKE 'K%' THEN 'ACF'**

**WHEN DEPT\_ID LIKE 'M%' THEN 'SAMHSA'**

**WHEN DEPT\_ID LIKE 'P%' THEN 'PSC'**

**ELSE ' ' END;**

COMMIT;

END IF;

END INSERT\_CHR\_EMP\_INFO;

-------------------------------------------------------------------------------

-------------------------------------------------------------------------------

This concludes the process to update the WHRSC customers in EWITS 2.0.

***Note: Column IC refers to the Customer***