|  |  |
| --- | --- |
| **Date:** | **Dec 15, 2011** |
| **Remedy Request ID:** | **Not Applicable** |
| **Production Build ID:** | **P.1.115.0** |
| Build Description : |  |

**Section 1 –**

|  |  |  |  |
| --- | --- | --- | --- |
| Type of Build : | * Hot Fix | * Full |  |
| Database Team Responsible Party: | SACWIS DBA’s | | |

**Section 2 – Technical Solution**

|  |  |
| --- | --- |
| Location of Solution Documentation: | R:\sacwis\Database\DBAWorkArea\DBDevelopment\P.1.115.0 |
| Solution Documentation Folder Name: | Prod\Docs |
| Software/SQL File Name: | sacwis\_prod\_db\_p.1.115. |

|  |  |
| --- | --- |
| List new table(s)/View/MView be added: |  |
| List new column(s) be added / modified: | **Add 0 New Table(s)**  **Added 21 Column(s) to 6 Existing Table(s)**  ALTER TABLE INTAKE\_PARTICIPANT ADD (school\_name\_text VARCHAR2 (50) NULL,  school\_grade\_text VARCHAR2 (25) NULL,  employment\_text VARCHAR2 (40) NULL,  phone1\_text VARCHAR2 (10) NULL,  phone1\_type VARCHAR2 (25) NULL,  phone2\_text VARCHAR2 (10) NULL,  phone2\_type VARCHAR2 (25) NULL,  phone3\_text VARCHAR2 (10) NULL,  phone3\_type VARCHAR2 (25) NULL,  safety\_hazard\_text VARCHAR2 (4000) NULL,  race2\_code VARCHAR2 (30) NULL,  hispanic\_latino\_code VARCHAR2 (30) NULL)  ALTER TABLE CASE\_PLAN ADD (recommended\_revision\_nbr NUMBER NULL,  finalized\_date DATE NULL,  recommended\_flag NUMBER NULL)  ALTER TABLE CASE\_PLAN\_PARTICIPANT ADD(recommended\_agncy\_lglstat\_code VARCHAR2(30) NULL,  recommended\_plcmnt\_type\_code VARCHAR2(30) NULL)  ALTER TABLE VISITATION\_PLAN ADD(finalized\_date DATE NULL)  ALTER TABLE HEARING\_SEGMENT ADD(hearing\_status\_code VARCHAR2(30) NULL,  status\_reason\_code VARCHAR2(30) NULL)  ALTER TABLE RULING ADD(comments\_text VARCHAR2(4000) NULL)  **Modified 3 Existing Table(s)**  ALTER TABLE PSA\_INCOMING\_AGENCY MODIFY (contact\_email\_text VARCHAR2 (320))  ALTER TABLE case\_plan MODIFY ( recommended\_flag NOT NULL)  ALTER TABLE CP\_PART\_PLACEMENT\_DESC MODIFY (placement\_setting\_id NULL) |
| List existing table(s) to be deleted: | **Droped 0 Table(s)** |
| List existing column(s) to be deleted: | **Deleted Columns from 0 Existing Table(s)** |

|  |  |
| --- | --- |
| Application Ref Data / Seed Data | **Some Ref & Seed Data Updates and Modifies:**  **Adds:**  **REF\_DATA**  INSERT INTO ref\_data  (ref\_data\_id, domain\_code,  ref\_data\_code, short\_desc, long\_desc, sort\_value, inactive\_flag,  created\_by, created\_date, modified\_by,  modified\_date,  ref\_data\_config\_id,  group\_code  )  VALUES (ref\_data\_id\_seq.NEXTVAL, 'Ruling Received',  'NUNCPROTUNC', 'Nunc Pro Tunc', 'Nunc Pro Tunc', 39, 0,  100, TO\_DATE (SYSDATE, 'MM/DD/YYYY'), 100,  TO\_DATE (SYSDATE, 'MM/DD/YYYY'),  (SELECT ref\_data\_config\_id  FROM ref\_data\_config  WHERE category\_name = 'Ruling Received'  AND domain\_code = 'Ruling Received'),  '00'  )  INSERT INTO ref\_data  (ref\_data\_id, domain\_code,  ref\_data\_code, short\_desc, long\_desc, sort\_value, inactive\_flag,  created\_by, created\_date, modified\_by,  modified\_date,  ref\_data\_config\_id,  group\_code  )  VALUES (ref\_data\_id\_seq.NEXTVAL, 'Ruling Received',  'NUNCPROTUNCINTEREST', 'Nunc Pro Tunc - Initial Best Interest', 'Nunc Pro Tunc - Initial Best Interest', 39, 0,  100, TO\_DATE (SYSDATE, 'MM/DD/YYYY'), 100,  TO\_DATE (SYSDATE, 'MM/DD/YYYY'),  (SELECT ref\_data\_config\_id  FROM ref\_data\_config  WHERE category\_name = 'Ruling Received'  AND domain\_code = 'Ruling Received'),  '00'  )  INSERT INTO ref\_data  (ref\_data\_id, domain\_code,  ref\_data\_code, short\_desc, long\_desc, sort\_value, inactive\_flag,  created\_by, created\_date, modified\_by,  modified\_date,  ref\_data\_config\_id,  group\_code  )  VALUES (ref\_data\_id\_seq.NEXTVAL, 'Ruling Received',  'NUNCPROTUNCEFFORTS', 'Nunc Pro Tunc - Initial Reasonable Efforts', 'Nunc Pro Tunc - Initial Reasonable Efforts', 39, 0,  100, TO\_DATE (SYSDATE, 'MM/DD/YYYY'), 100,  TO\_DATE (SYSDATE, 'MM/DD/YYYY'),  (SELECT ref\_data\_config\_id  FROM ref\_data\_config  WHERE category\_name = 'Ruling Received'  AND domain\_code = 'Ruling Received'),  '00'  )  INSERT INTO ref\_data  (ref\_data\_id, domain\_code,  ref\_data\_code, short\_desc, long\_desc, sort\_value, inactive\_flag,  created\_by, created\_date, modified\_by,  modified\_date,  ref\_data\_config\_id,  group\_code  )  VALUES (ref\_data\_id\_seq.NEXTVAL, 'Ruling Received',  'RESTRAINPROTECTORDERNOCONTACT', 'Restraining/Protection Orders/No Contact', 'Restraining/Protection Orders/No Contact', 52, 0,  100, TO\_DATE (SYSDATE, 'MM/DD/YYYY'), 100,  TO\_DATE (SYSDATE, 'MM/DD/YYYY'),  (SELECT ref\_data\_config\_id  FROM ref\_data\_config  WHERE category\_name = 'Ruling Received'  AND domain\_code = 'Ruling Received'),  '00'  )  INSERT INTO ref\_data  (ref\_data\_id, domain\_code,  ref\_data\_code, short\_desc, long\_desc, sort\_value, inactive\_flag,  created\_by, created\_date, modified\_by,  modified\_date,  ref\_data\_config\_id,  group\_code  )  VALUES (ref\_data\_id\_seq.NEXTVAL, 'Ruling Received',  'VISITATION', 'Visitation', 'Visitation', 80, 0,  100, TO\_DATE (SYSDATE, 'MM/DD/YYYY'), 100,  TO\_DATE (SYSDATE, 'MM/DD/YYYY'),  (SELECT ref\_data\_config\_id  FROM ref\_data\_config  WHERE category\_name = 'Ruling Received'  AND domain\_code = 'Ruling Received'),  '00'  )  INSERT INTO ref\_data  (ref\_data\_id, domain\_code,  ref\_data\_code, short\_desc, long\_desc, sort\_value, inactive\_flag,  created\_by, created\_date, modified\_by,  modified\_date,  ref\_data\_config\_id,  group\_code  )  VALUES (ref\_data\_id\_seq.NEXTVAL, 'Ruling Received',  'TPRDADDESERTED', 'TPR Father - Deserted Child/Safe Haven Baby', 'TPR Father - Deserted Child/Safe Haven Baby', 65, 0,  100, TO\_DATE (SYSDATE, 'MM/DD/YYYY'), 100,  TO\_DATE (SYSDATE, 'MM/DD/YYYY'),  (SELECT ref\_data\_config\_id  FROM ref\_data\_config  WHERE category\_name = 'Ruling Received'  AND domain\_code = 'Ruling Received'),  '00'  )  INSERT INTO ref\_data  (ref\_data\_id, domain\_code,  ref\_data\_code, short\_desc, long\_desc, sort\_value, inactive\_flag,  created\_by, created\_date, modified\_by,  modified\_date,  ref\_data\_config\_id,  group\_code  )  VALUES (ref\_data\_id\_seq.NEXTVAL, 'Ruling Received',  'TPRDADINVOLUNTARY', 'TPR Father - Involuntary due to CAN or Dependency', 'TPR Father - Involuntary due to CAN or Dependency', 66, 0,  100, TO\_DATE (SYSDATE, 'MM/DD/YYYY'), 100,  TO\_DATE (SYSDATE, 'MM/DD/YYYY'),  (SELECT ref\_data\_config\_id  FROM ref\_data\_config  WHERE category\_name = 'Ruling Received'  AND domain\_code = 'Ruling Received'),  '00'  )  INSERT INTO ref\_data  (ref\_data\_id, domain\_code,  ref\_data\_code, short\_desc, long\_desc, sort\_value, inactive\_flag,  created\_by, created\_date, modified\_by,  modified\_date,  ref\_data\_config\_id,  group\_code  )  VALUES (ref\_data\_id\_seq.NEXTVAL, 'Ruling Received',  'TPRMOMDESERTED', 'TPR Mother - Deserted Child/Safe Haven Baby', 'TPR Mother - Deserted Child/Safe Haven Baby', 71, 0,  100, TO\_DATE (SYSDATE, 'MM/DD/YYYY'), 100,  TO\_DATE (SYSDATE, 'MM/DD/YYYY'),  (SELECT ref\_data\_config\_id  FROM ref\_data\_config  WHERE category\_name = 'Ruling Received'  AND domain\_code = 'Ruling Received'),  '00'  )  INSERT INTO ref\_data  (ref\_data\_id, domain\_code,  ref\_data\_code, short\_desc, long\_desc, sort\_value, inactive\_flag,  created\_by, created\_date, modified\_by,  modified\_date,  ref\_data\_config\_id,  group\_code  )  VALUES (ref\_data\_id\_seq.NEXTVAL, 'Ruling Received',  'TPRMOMINVOLUNTARY', 'TPR Mother - Involuntary due to CAN or Dependency', 'TPR Mother - Involuntary due to CAN or Dependency', 72, 0,  100, TO\_DATE (SYSDATE, 'MM/DD/YYYY'), 100,  TO\_DATE (SYSDATE, 'MM/DD/YYYY'),  (SELECT ref\_data\_config\_id  FROM ref\_data\_config  WHERE category\_name = 'Ruling Received'  AND domain\_code = 'Ruling Received'),  '00'  )  INSERT INTO REF\_DATA (REF\_DATA\_ID,PARENT\_REF\_DATA\_CODE,DOMAIN\_CODE,REF\_DATA\_CODE,SHORT\_DESC,LONG\_DESC,SORT\_VALUE,INACTIVE\_FLAG,CREATED\_BY,  created\_date,modified\_by,modified\_date,ref\_data\_config\_id,group\_code) values  (REF\_DATA\_ID\_seq.nextVal,null,'ReasonHearingNotHeld', 'ADTNDOCNEED','Additional documentation needed','Additional documentation needed',10,0,100,sysdate,100,sysdate,  (select ref\_data\_config\_id from ref\_data\_config where domain\_code='ReasonHearingNotHeld'),'00')  INSERT INTO REF\_DATA (REF\_DATA\_ID,PARENT\_REF\_DATA\_CODE,DOMAIN\_CODE,REF\_DATA\_CODE,SHORT\_DESC,LONG\_DESC,SORT\_VALUE,INACTIVE\_FLAG,CREATED\_BY,  created\_date,modified\_by,modified\_date,ref\_data\_config\_id,group\_code) values  (ref\_data\_id\_seq.nextval,null,'ReasonHearingNotHeld', 'ADTNTIMEREQ','Additional time requested to complete previous order','Additional time requested to complete previous order',20,0,100,sysdate,100,sysdate,  (select ref\_data\_config\_id from ref\_data\_config where domain\_code='ReasonHearingNotHeld'),'00')  INSERT INTO REF\_DATA (REF\_DATA\_ID,PARENT\_REF\_DATA\_CODE,DOMAIN\_CODE,REF\_DATA\_CODE,SHORT\_DESC,LONG\_DESC,SORT\_VALUE,INACTIVE\_FLAG,CREATED\_BY,  created\_date,modified\_by,modified\_date,ref\_data\_config\_id,group\_code) values  (REF\_DATA\_ID\_seq.nextVal,null,'ReasonHearingNotHeld', 'ATTORNEYCONFLICT','Attorney Schedule Conflict','Attorney Schedule Conflict',30,0,100,sysdate,100,sysdate,  (select ref\_data\_config\_id from ref\_data\_config where domain\_code='ReasonHearingNotHeld'),'00')  INSERT INTO REF\_DATA (REF\_DATA\_ID,PARENT\_REF\_DATA\_CODE,DOMAIN\_CODE,REF\_DATA\_CODE,SHORT\_DESC,LONG\_DESC,SORT\_VALUE,INACTIVE\_FLAG,CREATED\_BY,  created\_date,modified\_by,modified\_date,ref\_data\_config\_id,group\_code) values  (REF\_DATA\_ID\_seq.nextVal,null,'ReasonHearingNotHeld', 'COURTSCHCONFLICT','Court Schedule Conflict','Court Schedule Conflict',40,0,100,sysdate,100,sysdate,  (select ref\_data\_config\_id from ref\_data\_config where domain\_code='ReasonHearingNotHeld'),'00')  INSERT INTO REF\_DATA (REF\_DATA\_ID,PARENT\_REF\_DATA\_CODE,DOMAIN\_CODE,REF\_DATA\_CODE,SHORT\_DESC,LONG\_DESC,SORT\_VALUE,INACTIVE\_FLAG,CREATED\_BY,  created\_date,modified\_by,modified\_date,ref\_data\_config\_id,group\_code) values  (REF\_DATA\_ID\_seq.nextVal,null,'ReasonHearingNotHeld', 'NTFCNOTCOMP','Notification/Service Not complete','Notification/Service Not complete',50,0,100,sysdate,100,sysdate,  (select ref\_data\_config\_id from ref\_data\_config where domain\_code='ReasonHearingNotHeld'),'00')  INSERT INTO REF\_DATA (REF\_DATA\_ID,PARENT\_REF\_DATA\_CODE,DOMAIN\_CODE,REF\_DATA\_CODE,SHORT\_DESC,LONG\_DESC,SORT\_VALUE,INACTIVE\_FLAG,CREATED\_BY,  created\_date,modified\_by,modified\_date,ref\_data\_config\_id,group\_code) values  (REF\_DATA\_ID\_seq.nextVal,null,'ReasonHearingNotHeld', 'PRNTNOTAVAIL','Parent(s) not available','Parent(s) not available',60,0,100,sysdate,100,sysdate,  (select ref\_data\_config\_id from ref\_data\_config where domain\_code='ReasonHearingNotHeld'),'00')  INSERT INTO REF\_DATA (REF\_DATA\_ID,PARENT\_REF\_DATA\_CODE,DOMAIN\_CODE,REF\_DATA\_CODE,SHORT\_DESC,LONG\_DESC,SORT\_VALUE,INACTIVE\_FLAG,CREATED\_BY,  created\_date,modified\_by,modified\_date,ref\_data\_config\_id,group\_code) values  (REF\_DATA\_ID\_seq.nextVal,null,'ReasonHearingNotHeld', 'PARTYEMERGENCY','Party III/Emergency Situation','Party III/Emergency Situation',70,0,100,sysdate,100,sysdate,  (select ref\_data\_config\_id from ref\_data\_config where domain\_code='ReasonHearingNotHeld'),'00')  INSERT INTO REF\_DATA (REF\_DATA\_ID,PARENT\_REF\_DATA\_CODE,DOMAIN\_CODE,REF\_DATA\_CODE,SHORT\_DESC,LONG\_DESC,SORT\_VALUE,INACTIVE\_FLAG,CREATED\_BY,  created\_date,modified\_by,modified\_date,ref\_data\_config\_id,group\_code) values  (ref\_data\_id\_seq.nextval,null,'ReasonHearingNotHeld', 'STAFFAVAIL','Staff Reassignment/Changes/Availability','Staff Reassignment/Changes/Availability',80,0,100,sysdate,100,sysdate,  (select ref\_data\_config\_id from ref\_data\_config where domain\_code='ReasonHearingNotHeld'),'00')  INSERT INTO REF\_DATA (REF\_DATA\_ID,PARENT\_REF\_DATA\_CODE,DOMAIN\_CODE,REF\_DATA\_CODE,SHORT\_DESC,LONG\_DESC,SORT\_VALUE,INACTIVE\_FLAG,CREATED\_BY,  created\_date,modified\_by,modified\_date,ref\_data\_config\_id,group\_code) values  (ref\_data\_id\_seq.nextval,null,'HearingStatus', 'REFILED','Dismissed-Refiled','Dismissed-Refiled',70,0,100,sysdate,100,sysdate,  (select ref\_data\_config\_id from ref\_data\_config where domain\_code='HearingStatus'),'00')  INSERT INTO REF\_DATA (REF\_DATA\_ID,PARENT\_REF\_DATA\_CODE,DOMAIN\_CODE,REF\_DATA\_CODE,SHORT\_DESC,LONG\_DESC,SORT\_VALUE,INACTIVE\_FLAG,CREATED\_BY,  created\_date,modified\_by,modified\_date,ref\_data\_config\_id,group\_code) values  (ref\_data\_id\_seq.nextval,null,'HearingStatus', 'VACATED','Vacated','Vacated',80,0,100,sysdate,100,sysdate,  (select ref\_data\_config\_id from ref\_data\_config where domain\_code='HearingStatus'),'00')    INSERT INTO REF\_DATA (REF\_DATA\_ID,PARENT\_REF\_DATA\_CODE,DOMAIN\_CODE,REF\_DATA\_CODE,SHORT\_DESC,LONG\_DESC,SORT\_VALUE,INACTIVE\_FLAG,CREATED\_BY,  CREATED\_DATE,MODIFIED\_BY,MODIFIED\_DATE,REF\_DATA\_CONFIG\_ID,GROUP\_CODE) values  (REF\_DATA\_ID\_seq.nextVal,'','PlanCategory', 'TRADITIONAL','Traditional Plan Category','Traditional Plan Category',10,0,100,sysdate,100,sysdate,  (select ref\_data\_config\_id from ref\_data\_config where domain\_code='PlanCategory'),'00')  INSERT INTO REF\_DATA (REF\_DATA\_ID,PARENT\_REF\_DATA\_CODE,DOMAIN\_CODE,REF\_DATA\_CODE,SHORT\_DESC,LONG\_DESC,SORT\_VALUE,INACTIVE\_FLAG,CREATED\_BY,  CREATED\_DATE,MODIFIED\_BY,MODIFIED\_DATE,REF\_DATA\_CONFIG\_ID,GROUP\_CODE) values  (REF\_DATA\_ID\_SEQ.NEXTVAL,'','PlanCategory', 'RECOMMENDED','Recommended Plan Category','Recommended Plan Category',20,0,100,sysdate,100,sysdate,  (select ref\_data\_config\_id from ref\_data\_config where domain\_code='PlanCategory'),'00')  INSERT INTO REF\_DATA (REF\_DATA\_ID,PARENT\_REF\_DATA\_CODE,DOMAIN\_CODE,REF\_DATA\_CODE,SHORT\_DESC,LONG\_DESC,SORT\_VALUE,INACTIVE\_FLAG,CREATED\_BY,  CREATED\_DATE,MODIFIED\_BY,MODIFIED\_DATE,REF\_DATA\_CONFIG\_ID,GROUP\_CODE) values  (REF\_DATA\_ID\_seq.nextVal,'RECOMMENDED','VisitationPlanStatus', 'LINKEDRECOMMENDED','Linked Recommended','Linked Recommended',10,0,100,sysdate,100,sysdate,  (select ref\_data\_config\_id from ref\_data\_config where domain\_code='VisitationPlanStatus'),'00')  INSERT INTO REF\_DATA (REF\_DATA\_ID,PARENT\_REF\_DATA\_CODE,DOMAIN\_CODE,REF\_DATA\_CODE,SHORT\_DESC,LONG\_DESC,SORT\_VALUE,INACTIVE\_FLAG,CREATED\_BY,  CREATED\_DATE,MODIFIED\_BY,MODIFIED\_DATE,REF\_DATA\_CONFIG\_ID,GROUP\_CODE) values  (REF\_DATA\_ID\_SEQ.NEXTVAL,'RECOMMENDED','VisitationPlanStatus', 'RECOMMENDED','Recommended','Recommended',20,0,100,sysdate,100,sysdate,  (select ref\_data\_config\_id from ref\_data\_config where domain\_code='VisitationPlanStatus'),'00')    insert into ref\_data (ref\_data\_id, parent\_ref\_data\_code, domain\_code, ref\_data\_code, short\_desc, long\_desc, sort\_value, inactive\_flag,  created\_by, created\_date, modified\_by, modified\_date, ref\_data\_config\_id, group\_code)  values (ref\_data\_id\_seq.nextval, null, 'taskType', 'WAIVER', 'Waiver', 'Waiver', 37, 0,  100, sysdate, 100, sysdate, (select ref\_data\_config\_id from ref\_data\_config where domain\_code = 'taskType'), '00')  **REF\_DATA\_CONFIG**  insert into ref\_data\_config(ref\_data\_config\_id,category\_name,domain\_code,parent\_domain\_code,has\_child\_flag,  sql\_statement,class\_text,category\_desc,state\_exclusive\_flag,created\_by,created\_date,modified\_by,modified\_date)  values(ref\_data\_config\_id\_seq.nextval,'ReasonHearingNotHeld','ReasonHearingNotHeld','','',  'select REF\_DATA\_CODE as "lookup\_code", SHORT\_DESC as "short\_description", LONG\_DESC as "long\_description" from REF\_DATA where DOMAIN\_CODE = ''ReasonHearingNotHeld'' ORDER BY REF\_DATA\_CODE',  'us.oh.state.odjfs.sacwis.framework.rda.ReferenceVO','',0,100,sysdate,100,sysdate)  insert into ref\_data\_config(ref\_data\_config\_id,category\_name,domain\_code,parent\_domain\_code,has\_child\_flag,  sql\_statement,class\_text,category\_desc,state\_exclusive\_flag,created\_by,created\_date,modified\_by,modified\_date)  values(ref\_data\_config\_id\_seq.nextval,'PlanCategory','PlanCategory','','',  'select REF\_DATA\_CODE as "lookup\_code", SHORT\_DESC as "short\_description", LONG\_DESC as "long\_description" from REF\_DATA where DOMAIN\_CODE = ''PlanCategory'' ORDER BY REF\_DATA\_CODE',  'us.oh.state.odjfs.sacwis.framework.rda.ReferenceVO','',0,100,sysdate,100,sysdate)  **UC\_ACTION**  insert into uc\_action (uc\_action\_id, business\_function\_id, name, uc\_action\_code, action\_mode)  values (uc\_action\_id\_seq.nextval, (select business\_function\_id from business\_function where business\_function\_code = 'RM04a'),  'RM04a\_APPROVE\_PROVIDER\_WAIVER\_VARIANCE', 'RM04a\_APPROVE\_PROVIDER\_WAIVER\_VARIANCE', 'UPDATE')  **ACTION\_ROLE**  insert into action\_role (uc\_action\_id, role\_id, action\_role\_id)  values ((select uc\_action\_id from uc\_action where uc\_action\_code = 'RM04a\_VIEW\_PROVIDER\_WAIVER\_VARIANCE'),  (select role\_id from role where role\_code = 'STATE\_LICENSING\_SPECIALIST'), action\_role\_id\_seq.nextval);  insert into action\_role (uc\_action\_id, role\_id, action\_role\_id)  values ((select uc\_action\_id from uc\_action where uc\_action\_code = 'RM04a\_ADD\_PROVIDER\_WAIVER\_VARIANCE'),  (select role\_id from role where role\_code = 'STATE\_CERTIFICATION\_WORKER'), action\_role\_id\_seq.nextval);  insert into action\_role (uc\_action\_id, role\_id, action\_role\_id)  values ((select uc\_action\_id from uc\_action where uc\_action\_code = 'RM04a\_ADD\_PROVIDER\_WAIVER\_VARIANCE'),  (select role\_id from role where role\_code = 'STATE\_LICENSING\_SPECIALIST'), action\_role\_id\_seq.nextval);  insert into action\_role (uc\_action\_id, role\_id, action\_role\_id)  values ((select uc\_action\_id from uc\_action where uc\_action\_code = 'RM04a\_EDIT\_PROVIDER\_WAIVER\_VARIANCE'),  (select role\_id from role where role\_code = 'STATE\_CERTIFICATION\_WORKER'), action\_role\_id\_seq.nextval);  insert into action\_role (uc\_action\_id, role\_id, action\_role\_id)  values ((select uc\_action\_id from uc\_action where uc\_action\_code = 'RM04a\_EDIT\_PROVIDER\_WAIVER\_VARIANCE'),  (select role\_id from role where role\_code = 'STATE\_LICENSING\_SPECIALIST'), action\_role\_id\_seq.nextval);  insert into action\_role (uc\_action\_id, role\_id, action\_role\_id)  values ((select uc\_action\_id from uc\_action where uc\_action\_code = 'RM04a\_DELETE\_PROVIDER\_WAIVER\_VARIANCE'),  (select role\_id from role where role\_code = 'STATE\_CERTIFICATION\_WORKER'), action\_role\_id\_seq.nextval);  insert into action\_role (uc\_action\_id, role\_id, action\_role\_id)  values ((select uc\_action\_id from uc\_action where uc\_action\_code = 'RM04a\_DELETE\_PROVIDER\_WAIVER\_VARIANCE'),  (select role\_id from role where role\_code = 'STATE\_LICENSING\_SPECIALIST'), action\_role\_id\_seq.nextval);  insert into action\_role (uc\_action\_id, role\_id, action\_role\_id)  values ((select uc\_action\_id from uc\_action where uc\_action\_code = 'RM04a\_REVIEW\_PROVIDER\_WAIVER\_VARIANCE'),  (select role\_id from role where role\_code = 'STATE\_LICENSING\_SPECIALIST'), action\_role\_id\_seq.nextval);  insert into action\_role (uc\_action\_id, role\_id, action\_role\_id)  values ((select uc\_action\_id from uc\_action where uc\_action\_code = 'RM04a\_APPROVE\_PROVIDER\_WAIVER\_VARIANCE'),  (select role\_id from role where role\_code = 'STATE\_CERTIFICATION\_WORKER'), action\_role\_id\_seq.nextval);  insert into action\_role (uc\_action\_id, role\_id, action\_role\_id)  values ((select uc\_action\_id from uc\_action where uc\_action\_code = 'RM04a\_APPROVE\_PROVIDER\_WAIVER\_VARIANCE'),  (select role\_id from role where role\_code = 'STATE\_LICENSING\_SPECIALIST'), action\_role\_id\_seq.nextval);  **SECURITY\_PROFILE**  insert into security\_profile (security\_profile\_id, role\_id, business\_function\_id, profile\_name, profile\_desc,  state\_exclusive\_flag, public\_only\_flag, created\_by, created\_date, modified\_by, modified\_date,  asgn\_rqd\_read\_prv\_exmpt\_flag, asgn\_rqd\_updt\_prv\_exmpt\_flag, asgn\_rqd\_read\_pub\_exmpt\_flag, asgn\_rqd\_updt\_pub\_exmpt\_flag)  values (security\_profile\_id\_seq.nextval, (select role\_id from role where role\_code = 'STATE\_LICENSING\_SPECIALIST'),  (select business\_function\_id from business\_function where business\_function\_code = 'RM04a'),  'Waiver/Variance - State Licensing Specialist', 'Reviews/Approves the Waiver/Variance requests', 0, 0,  100, sysdate, 100, sysdate, 0, 0, 0, 0)  **NOTIFICATION\_TEMPLATE**  insert into NOTIFICATION\_TEMPLATE (NOTIFICATION\_TEMPLATE\_ID, NOTIFICATION\_CODE, NOTIFICATION\_NAME, SUBJECT\_TEXT, MESSAGE\_TEXT, PRIORITY\_CODE,  RECIPIENTS\_EMAIL, RECIPIENTS\_FLAG, SYSTEM\_SUPPLIED\_FLAG, REASON\_FOR\_CHANGE, CREATED\_BY, CREATED\_DATE, MODIFIED\_BY, MODIFIED\_DATE)  values (NOTIFICATION\_TEMPLATE\_ID\_SEQ.nextval, 'PROVIDER\_WAIVER\_APPROVAL', 'Waiver Approval', 'Waiver Approval',  'A Waiver for provider ID has been Approved/Denied.', 'Normal', null, 0, 1, 'SACWIS Initial', 100, sysdate, 100, sysdate)  **PROFILE\_RESOURCE**  **ROLE**  **BUSINESS\_FUNCTION**  **MENU**  **REPORT\_TEMPLATE**  **Changes:**  **REF\_DATA**  UPDATE ref\_data rd SET rd.short\_desc = 'Annual Court Review (ACR)', rd.long\_desc = 'Annual Court Review (ACR)'  WHERE rd.domain\_code = 'LegalActionTypes' AND rd.short\_desc = 'ACR';  update ref\_data set sort\_value = 10 where domain\_code='HearingStatus' and short\_desc='Cancelled';  update ref\_data set short\_desc = 'Continued/Rescheduled/Postponed', long\_desc= 'Continued/Rescheduled/Postponed', sort\_value = 20 where domain\_code='HearingStatus' and short\_desc='Continued (Rescheduled)';  update ref\_data set sort\_value = 30 where domain\_code='HearingStatus' and short\_desc='Mapping Default';  update ref\_data set sort\_value = 40 where domain\_code='HearingStatus' and short\_desc='Dismissed';  update ref\_data set sort\_value = 50 where domain\_code='HearingStatus' and short\_desc='Held';  update ref\_data set sort\_value = 60 where domain\_code='HearingStatus' and short\_desc='Pending';  update ref\_data set PARENT\_REF\_DATA\_CODE = 'TRADITIONAL' where domain\_code='VisitationPlanStatus' and REF\_DATA\_CODE = 'LINKEDTOCASE';  update REF\_DATA set PARENT\_REF\_DATA\_CODE = 'TRADITIONAL' where DOMAIN\_CODE='VisitationPlanStatus' and REF\_DATA\_CODE = 'COMPLETED';  update REF\_DATA set PARENT\_REF\_DATA\_CODE = 'TRADITIONAL' where DOMAIN\_CODE='VisitationPlanStatus' and REF\_DATA\_CODE = 'DRAFT';  update ref\_data set PARENT\_REF\_DATA\_CODE = 'TRADITIONAL' where domain\_code='VisitationPlanStatus' and REF\_DATA\_CODE = 'CONVERT\*\*';  UPDATE ref\_data SET PARENT\_REF\_DATA\_CODE = 'RECOMMENDED' WHERE domain\_code='CPPlacementType' AND ref\_data\_code IN (  'ADOPTIVEPLACEMENT', 'CERTIFIEDCRC', 'CERTIFIEDESC', 'CERTIFIEDFOSTERHOME', 'CERTIFIEDGH', 'DET',  'CERTIFIEDAPPROVEDRELATIVECAR', 'CERTIFIEDAPPROVEDNONRELATIVECA', 'INDEPENDENTLIVING', 'RPF','OWNHOME')  UPDATE ref\_data SET short\_desc = 'Party Sick/Emergency Situation',  long\_desc = 'Party Sick/Emergency Situation' WHERE domain\_code = 'ReasonHearingNotHeld' AND ref\_data\_code = 'PARTYEMERGENCY'  **BUSINESS\_FUNCTION**  **MENU**  **REPORT\_TEMPLATE**  **MENU**  **GROUP\_CATEGORY**  **ROLE / SECURITY\_PROFILE**  **UC\_ACTION**  **ACTION\_ROLE**  **SECURITY\_PROFILE**  **PROFILE\_RESOURCE**  **GROUP\_CAT\_CEIL\_TYP\_LINK**  **TICKLER\_TEMPLATE** |
| Batch Ref Data / Seed Data Update |  |
| Materialized View |  |
| Sequence | **Added 0 New Sequence(s) Overall:**  **Modified 0 Sequence(s) Overall:** |

|  |  |
| --- | --- |
| Index | **Add 2 New Index(es) Overall:**  CREATE INDEX intake\_part\_acv\_cust\_f01 ON intake\_part\_acv\_cust (intake\_participant\_id)  CREATE INDEX intake\_part\_acv\_cust\_f02 ON intake\_part\_acv\_cust (intake\_participant\_avc\_id)  **Modified 0 Index(es) Overall:**  **Dropped 0 Index(es) Overall:** |
| Constraint | **Added 0 New and Modified Existing Constraint(s):**  **Modified 107 Constraint(s) Overall:**    ALTER TABLE intake\_participant\_history  MODIFY unknown\_flag CONSTRAINT intake\_parthist\_unk\_flag\_nn NOT NULL;  ALTER TABLE intake\_participant\_history  MODIFY unknown\_flag CONSTRAINT intake\_parthist\_unk\_flag\_01 CHECK (unknown\_flag = ANY (0,1));  ALTER TABLE intake\_participant\_history  MODIFY worker\_hazard\_flag CONSTRAINT intake\_parthist\_haz\_flag\_nn NOT NULL;  ALTER TABLE intake\_participant\_history  MODIFY worker\_hazard\_flag CONSTRAINT intake\_parthist\_haz\_flag\_01 CHECK (worker\_hazard\_flag = ANY (0,1));  ALTER TABLE intake\_participant\_history  MODIFY lep\_flag CONSTRAINT intake\_parthist\_lep\_flag\_nn NOT NULL;  ALTER TABLE intake\_participant\_history  MODIFY lep\_flag CONSTRAINT intake\_parthist\_lep\_flag\_01 CHECK (lep\_flag = ANY (0,1));  ALTER TABLE intake  MODIFY completed\_flag CONSTRAINT intake\_complete\_flag\_nn NOT NULL;  commit;  ALTER TABLE intake  MODIFY completed\_flag CONSTRAINT intake\_complete\_flag\_01 CHECK (completed\_flag = ANY (0,1));  commit;  ALTER TABLE intake  MODIFY third\_party\_involved\_flag CONSTRAINT intake\_third\_party\_inv\_flag\_nn NOT NULL;  commit;  ALTER TABLE intake  MODIFY third\_party\_involved\_flag CONSTRAINT intake\_third\_party\_inv\_flag\_01 CHECK (third\_party\_involved\_flag = ANY (0,1));  commit;  ALTER TABLE intake  MODIFY law\_enforcement\_involve\_flag CONSTRAINT intake\_law\_enf\_inv\_flag\_nn NOT NULL;  commit;  ALTER TABLE intake  MODIFY law\_enforcement\_involve\_flag CONSTRAINT intake\_law\_enf\_inv\_flag\_01 CHECK (law\_enforcement\_involve\_flag = ANY (0,1));  commit;  ALTER TABLE intake  MODIFY disabled\_infant\_nutrition\_flag CONSTRAINT intake\_dis\_inf\_nut\_flag\_nn NOT NULL;  commit;  ALTER TABLE intake  MODIFY disabled\_infant\_nutrition\_flag CONSTRAINT intake\_dis\_inf\_nut\_flag\_01 CHECK (disabled\_infant\_nutrition\_flag = ANY (0,1));  commit;  ALTER TABLE intake  MODIFY disabled\_infant\_hydration\_flag CONSTRAINT intake\_dis\_inf\_hyd\_flag\_nn NOT NULL;  commit;  ALTER TABLE intake  MODIFY disabled\_infant\_hydration\_flag CONSTRAINT intake\_dis\_inf\_hyd\_flag\_01 CHECK (disabled\_infant\_hydration\_flag = ANY (0,1));  commit;  ALTER TABLE intake  MODIFY disabled\_infant\_med\_flag CONSTRAINT intake\_dis\_inf\_med\_flag\_nn NOT NULL;  commit;  ALTER TABLE intake  MODIFY disabled\_infant\_med\_flag CONSTRAINT intake\_dis\_inf\_med\_flag\_01 CHECK (disabled\_infant\_med\_flag = ANY (0,1));  commit;  ALTER TABLE intake  MODIFY disabled\_infant\_medtreat\_flag CONSTRAINT intake\_dis\_inf\_medtrt\_flag\_nn NOT NULL;  commit;  ALTER TABLE intake  MODIFY disabled\_infant\_medtreat\_flag CONSTRAINT intake\_dis\_inf\_medtrt\_flag\_01 CHECK (disabled\_infant\_medtreat\_flag = ANY (0,1));  commit;  ALTER TABLE intake  MODIFY is\_disp\_complete CONSTRAINT intake\_disp\_complete\_flag\_nn NOT NULL;  commit;  ALTER TABLE intake  MODIFY is\_disp\_complete CONSTRAINT intake\_disp\_complete\_flag\_01 CHECK (is\_disp\_complete = ANY (0,1));  commit;  ALTER TABLE intake  MODIFY restricted\_flag CONSTRAINT intake\_restricted\_flag\_nn NOT NULL;  commit;  ALTER TABLE intake  MODIFY restricted\_flag CONSTRAINT intake\_restricted\_flag\_01 CHECK (restricted\_flag = ANY (0,1));  commit;  ALTER TABLE reporter  MODIFY mandated\_reporter\_flag CONSTRAINT reporter\_mandated\_flag\_nn NOT NULL;  commit;  ALTER TABLE reporter  MODIFY mandated\_reporter\_flag CONSTRAINT reporter\_mandated\_flag\_01 CHECK (mandated\_reporter\_flag = ANY (0,1));  commit;  ALTER TABLE reporter  MODIFY mandated\_reporter\_inform\_flag CONSTRAINT reporter\_mand\_inf\_flag\_nn NOT NULL;  commit;  ALTER TABLE reporter  MODIFY mandated\_reporter\_inform\_flag CONSTRAINT reporter\_mand\_inf\_flag\_01 CHECK (mandated\_reporter\_inform\_flag = ANY (0,1));  commit;  ALTER TABLE reporter  MODIFY mandated\_reporter\_rqst\_flag CONSTRAINT reporter\_mand\_rqst\_flag\_nn NOT NULL;  commit;  ALTER TABLE reporter  MODIFY mandated\_reporter\_rqst\_flag CONSTRAINT reporter\_mand\_rqst\_flag\_01 CHECK (mandated\_reporter\_rqst\_flag = ANY (0,1));  commit;  ALTER TABLE reporter  MODIFY anonymous\_reporter\_flag CONSTRAINT reporter\_anonymous\_flag\_nn NOT NULL;  commit;  ALTER TABLE reporter  MODIFY anonymous\_reporter\_flag CONSTRAINT reporter\_anonymous\_flag\_01 CHECK (anonymous\_reporter\_flag = ANY (0,1));  commit;  ALTER TABLE reporter  MODIFY non\_mandated\_reporter\_flag CONSTRAINT reporter\_nonmandated\_flag\_nn NOT NULL;  commit;  ALTER TABLE reporter  MODIFY non\_mandated\_reporter\_flag CONSTRAINT reporter\_nonmandated\_flag\_01 CHECK (non\_mandated\_reporter\_flag = ANY (0,1));  commit;  ALTER TABLE reporter  MODIFY mandated\_reporter\_solo\_flag CONSTRAINT reporter\_mand\_solo\_flag\_nn NOT NULL;  commit;  ALTER TABLE reporter  MODIFY mandated\_reporter\_solo\_flag CONSTRAINT reporter\_mand\_solo\_flag\_01 CHECK (mandated\_reporter\_solo\_flag = ANY (0,1));  commit;  ALTER TABLE reporter\_contact  MODIFY src\_of\_knldg\_observed\_flag CONSTRAINT rpt\_cont\_observed\_flag\_nn NOT NULL;  commit;  ALTER TABLE reporter\_contact  MODIFY src\_of\_knldg\_observed\_flag CONSTRAINT rpt\_cont\_observed\_flag\_01 CHECK (src\_of\_knldg\_observed\_flag = ANY (0,1));  commit;  ALTER TABLE reporter\_contact  MODIFY src\_of\_knldg\_other\_party\_flag CONSTRAINT rpt\_cont\_other\_party\_flag\_nn NOT NULL;  commit;  ALTER TABLE reporter\_contact  MODIFY src\_of\_knldg\_other\_party\_flag CONSTRAINT rpt\_cont\_other\_party\_flag\_01 CHECK (src\_of\_knldg\_other\_party\_flag = ANY (0,1));  commit;  ALTER TABLE reporter\_contact  MODIFY addrd\_con\_parent\_ct\_flag CONSTRAINT rpt\_cont\_addressed\_flag\_nn NOT NULL;  commit;  ALTER TABLE reporter\_contact  MODIFY addrd\_con\_parent\_ct\_flag CONSTRAINT rpt\_cont\_addressed\_flag\_01 CHECK (addrd\_con\_parent\_ct\_flag = ANY (0,1));  commit;  ALTER TABLE intake\_participant  MODIFY unknown\_flag CONSTRAINT intake\_part\_unknown\_flag\_01 CHECK (unknown\_flag = ANY (0,1));  commit;  ALTER TABLE sacwis.intake\_participant  MODIFY worker\_hazard\_flag CONSTRAINT intake\_part\_haz\_flag\_nn NOT NULL;  commit;  ALTER TABLE intake\_participant  MODIFY worker\_hazard\_flag CONSTRAINT intake\_part\_haz\_flag\_01 CHECK (worker\_hazard\_flag = ANY (0,1));  commit;  ALTER TABLE intake\_participant  MODIFY lep\_flag CONSTRAINT intake\_part\_lep\_flag\_nn NOT NULL;  commit;  ALTER TABLE intake\_participant  MODIFY lep\_flag CONSTRAINT intake\_part\_lep\_flag\_01 CHECK (lep\_flag = ANY (0,1));  commit;  ALTER TABLE intake\_participant\_acv  MODIFY acv\_pend\_court\_hear\_flag CONSTRAINT intake\_partacv\_hearing\_flag\_nn NOT NULL;  commit;  ALTER TABLE intake\_participant\_acv  MODIFY acv\_pend\_court\_hear\_flag CONSTRAINT intake\_partacv\_hearing\_flag\_01 CHECK (acv\_pend\_court\_hear\_flag = ANY (0,1));  commit;  ALTER TABLE psa\_participant  MODIFY is\_matched\_flag CONSTRAINT psa\_part\_match\_flag\_nn NOT NULL;  commit;  ALTER TABLE psa\_participant  MODIFY is\_matched\_flag CONSTRAINT psa\_part\_match\_flag\_01 CHECK (is\_matched\_flag = ANY (0,1));  commit;  ALTER TABLE psa\_participant  MODIFY is\_reference\_name\_flag CONSTRAINT psa\_part\_reference\_flag\_nn NOT NULL;  commit;  ALTER TABLE psa\_participant  MODIFY is\_reference\_name\_flag CONSTRAINT psa\_part\_reference\_flag\_01 CHECK (is\_reference\_name\_flag = ANY (0,1));  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_if\_found\_safetycheck\_flag CONSTRAINT psa\_found\_safety\_flag\_nn NOT NULL;  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_if\_found\_safetycheck\_flag CONSTRAINT psa\_found\_safety\_flag\_01 CHECK (is\_if\_found\_safetycheck\_flag = ANY (0,1));  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_if\_found\_notify\_flag CONSTRAINT psa\_found\_notify\_flag\_nn NOT NULL;  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_if\_found\_notify\_flag CONSTRAINT psa\_found\_notify\_flag\_01 CHECK (is\_if\_found\_notify\_flag = ANY (0,1));  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_if\_found\_other\_flag CONSTRAINT psa\_found\_other\_flag\_nn NOT NULL;  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_if\_found\_other\_flag CONSTRAINT psa\_found\_other\_flag\_01 CHECK (is\_if\_found\_other\_flag = ANY (0,1));  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_notified\_police\_flag CONSTRAINT psa\_notified\_police\_flag\_nn NOT NULL;  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_notified\_police\_flag CONSTRAINT psa\_notified\_police\_flag\_01 CHECK (is\_notified\_police\_flag = ANY (0,1));  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_obtained\_courtorder\_flag CONSTRAINT psa\_courtorder\_flag\_nn NOT NULL;  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_obtained\_courtorder\_flag CONSTRAINT psa\_courtorder\_flag\_01 CHECK (is\_obtained\_courtorder\_flag = ANY (0,1));  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_conducted\_usps\_search\_flag CONSTRAINT psa\_usps\_search\_flag\_nn NOT NULL;  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_conducted\_usps\_search\_flag CONSTRAINT psa\_usps\_search\_flag\_01 CHECK (is\_conducted\_usps\_search\_flag = ANY (0,1));  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_cdjfs\_addr\_checked\_flag CONSTRAINT psa\_addr\_checked\_flag\_nn NOT NULL;  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_cdjfs\_addr\_checked\_flag CONSTRAINT psa\_addr\_checked\_flag\_01 CHECK (is\_cdjfs\_addr\_checked\_flag = ANY (0,1));  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_contacted\_utilities\_flag CONSTRAINT psa\_cont\_utilities\_flag\_nn NOT NULL;  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_contacted\_utilities\_flag CONSTRAINT psa\_cont\_utilities\_flag\_01 CHECK (is\_contacted\_utilities\_flag = ANY (0,1));  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_contacted\_schools\_flag CONSTRAINT psa\_cont\_schools\_flag\_nn NOT NULL;  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_contacted\_schools\_flag CONSTRAINT psa\_cont\_schools\_flag\_01 CHECK (is\_contacted\_schools\_flag = ANY (0,1));  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_contacted\_services\_flag CONSTRAINT psa\_cont\_services\_flag\_nn NOT NULL;  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_contacted\_services\_flag CONSTRAINT psa\_cont\_services\_flag\_01 CHECK (is\_contacted\_services\_flag = ANY (0,1));  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_contacted\_relatives\_flag CONSTRAINT psa\_cont\_relatives\_flag\_nn NOT NULL;  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_contacted\_relatives\_flag CONSTRAINT psa\_cont\_relatives\_flag\_01 CHECK (is\_contacted\_relatives\_flag = ANY (0,1));  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_other\_actions\_flag CONSTRAINT psa\_other\_actions\_flag\_nn NOT NULL;  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_other\_actions\_flag CONSTRAINT psa\_other\_actions\_flag\_01 CHECK (is\_other\_actions\_flag = ANY (0,1));  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_canai\_inprocess\_flag CONSTRAINT psa\_canai\_inprocess\_flag\_nn NOT NULL;  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_canai\_inprocess\_flag CONSTRAINT psa\_canai\_inprocess\_flag\_01 CHECK (is\_canai\_inprocess\_flag = ANY (0,1));  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_supportive\_svcs\_flag CONSTRAINT psa\_supportive\_svcs\_flag\_nn NOT NULL;  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_supportive\_svcs\_flag CONSTRAINT psa\_supportive\_svcs\_flag\_01 CHECK (is\_supportive\_svcs\_flag = ANY (0,1));  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_prot\_sprvsn\_ordered\_flag CONSTRAINT psa\_sprvsn\_ordered\_flag\_nn NOT NULL;  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_prot\_sprvsn\_ordered\_flag CONSTRAINT psa\_sprvsn\_ordered\_flag\_01 CHECK (is\_prot\_sprvsn\_ordered\_flag = ANY (0,1));  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_custody\_obtained\_flag CONSTRAINT psa\_custody\_obtained\_flag\_nn NOT NULL;  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_custody\_obtained\_flag CONSTRAINT psa\_custody\_obtained\_flag\_01 CHECK (is\_custody\_obtained\_flag = ANY (0,1));  ALTER TABLE protective\_service\_alert  MODIFY is\_childtaken\_violation\_flag CONSTRAINT psa\_childtaken\_viol\_flag\_nn NOT NULL;  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_childtaken\_violation\_flag CONSTRAINT psa\_childtaken\_viol\_flag\_01 CHECK (is\_childtaken\_violation\_flag = ANY (0,1));  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_othercase\_situation\_flag CONSTRAINT psa\_othercase\_sit\_flag\_nn NOT NULL;  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_othercase\_situation\_flag CONSTRAINT psa\_othercase\_sit\_flag\_01 CHECK (is\_othercase\_situation\_flag = ANY (0,1));  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_docs\_none\_flag CONSTRAINT psa\_docs\_none\_flag\_nn NOT NULL;  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_docs\_none\_flag CONSTRAINT psa\_docs\_none\_flag\_01 CHECK (is\_docs\_none\_flag = ANY (0,1));  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_docs\_court\_flag CONSTRAINT psa\_docs\_court\_flag\_nn NOT NULL;  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_docs\_court\_flag CONSTRAINT psa\_docs\_court\_flag\_01 CHECK (is\_docs\_court\_flag = ANY (0,1));  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_docs\_warrant\_flag CONSTRAINT psa\_docs\_warrant\_flag\_nn NOT NULL;  ALTER TABLE protective\_service\_alert  MODIFY is\_docs\_warrant\_flag CONSTRAINT psa\_docs\_warrant\_flag\_01 CHECK (is\_docs\_warrant\_flag = ANY (0,1));  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_docs\_med\_flag CONSTRAINT psa\_docs\_med\_flag\_nn NOT NULL;  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_docs\_med\_flag CONSTRAINT psa\_docs\_med\_flag\_01 CHECK (is\_docs\_med\_flag = ANY (0,1));  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_docs\_photos\_flag CONSTRAINT psa\_docs\_photos\_flag\_nn NOT NULL;  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_docs\_photos\_flag CONSTRAINT psa\_docs\_photos\_flag\_01 CHECK (is\_docs\_photos\_flag = ANY (0,1));  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_docs\_additional\_flag CONSTRAINT psa\_docs\_additional\_flag\_nn NOT NULL;  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_docs\_additional\_flag CONSTRAINT psa\_docs\_additional\_flag\_01 CHECK (is\_docs\_additional\_flag = ANY (0,1));  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_docs\_sent\_mail\_flag CONSTRAINT psa\_docs\_sent\_mail\_flag\_nn NOT NULL;  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_docs\_sent\_mail\_flag CONSTRAINT psa\_docs\_sent\_mail\_flag\_01 CHECK (is\_docs\_sent\_mail\_flag = ANY (0,1));  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_docs\_sent\_fax\_flag CONSTRAINT psa\_docs\_sent\_fax\_flag\_nn NOT NULL;  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_docs\_sent\_fax\_flag CONSTRAINT psa\_docs\_sent\_fax\_flag\_01 CHECK (is\_docs\_sent\_fax\_flag = ANY (0,1));  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_docs\_sent\_email\_flag CONSTRAINT psa\_docs\_sent\_email\_flag\_nn NOT NULL;  commit;  ALTER TABLE protective\_service\_alert  MODIFY is\_docs\_sent\_email\_flag CONSTRAINT psa\_docs\_sent\_email\_flag\_01 CHECK (is\_docs\_sent\_email\_flag = ANY (0,1));  **Dropped 0 Constraint(s) Overall:** |
|  |  |
| Function Updated | **Added 0 New & Modified 2 Existing Function(s) & Dropped 0 Function(s) Overall:**  **New Function(s):**      **Updated 2 Existing Function(s):**   1. GET\_LEGAL\_PARTICIPANTS 2. GETCASETRIGGERACTIVITYINFO   **Dropped 0 Unused Function(s) :** |
| Package(s) Updated | **Added 0 New & Modified 6 Package(s) & Dropped 0 Packages Overall:**  **0 New Package(s):**  **Updated 6 Existing Package(s):**   1. COPY\_CASE\_DATA\_TO\_ADPT\_CASE 2. PKCASETOOLS 3. PKG\_NON\_ODJFS\_PRVDR\_MERGE 4. PKG\_NON\_ODJFS\_PRVDR\_MERGE\_CNST 5. FM25a\_MITS\_HMO\_ERRORS 6. PKG\_PRVDR\_PERSON\_CHAR\_MAP   **Dropped 0 Unused Package(s) :** |
| Procedure to be compiled | **Added 0 New & Modified Existing Procedure(s) & Dropped 0 Existing Overall:**  **Updated Existing Procedure:**  **Dropped 0 Unused Procedure(s) :** |
|  |  |

|  |  |
| --- | --- |
| Triggers to be updated and enabled | **Added 0 New and Modified 1 Existing Trigger(s) Overall:**  **Added 0 New Trigger(s):**  **Modified 1 Existing Trigger(s):**   1. INTAKE\_PARTICIPANT\_AUDR |
| Script Updated | UPDATE case\_plan SET recommended\_flag = 0;  UPDATE intake\_participant\_acv ipacv  SET acv\_custodian\_participant\_id = NULL  WHERE acv\_custodian\_participant\_id IS NOT NULL  AND NOT EXISTS (SELECT 'True'  FROM intake\_participant ip  WHERE ip.intake\_participant\_id = ipacv.acv\_custodian\_participant\_id);  commit;  INSERT INTO intake\_part\_acv\_cust (intake\_part\_acv\_cust\_id,  intake\_participant\_id,  intake\_participant\_avc\_id,  created\_by,  created\_date,  modified\_by,  modified\_date)  SELECT intake\_part\_acv\_cust\_id\_seq.NEXTVAL,  acv\_custodian\_participant\_id,  intake\_participant\_acv\_id,  created\_by,  created\_date,  modified\_by,  modified\_date  FROM intake\_participant\_acv  WHERE acv\_custodian\_participant\_id IS NOT NULL;  commit;  /\*  +----------------------------------------------------------------+  | Update the status of the associated person. |  +----------------------------------------------------------------+  UPDATE intake\_participant\_acv  SET acv\_custodian\_participant\_id = NULL  WHERE acv\_custodian\_participant\_id IS NOT NULL;  \*/  UPDATE intake  SET completed\_flag = NVL(completed\_flag, 0),  third\_party\_involved\_flag = NVL(third\_party\_involved\_flag, 0),  law\_enforcement\_involve\_flag = NVL(law\_enforcement\_involve\_flag, 0),  disabled\_infant\_nutrition\_flag = NVL(disabled\_infant\_nutrition\_flag, 0),  disabled\_infant\_hydration\_flag = NVL(disabled\_infant\_hydration\_flag, 0),  disabled\_infant\_med\_flag = NVL(disabled\_infant\_med\_flag, 0),  disabled\_infant\_medtreat\_flag = NVL(disabled\_infant\_medtreat\_flag, 0),  is\_disp\_complete = NVL(is\_disp\_complete, 0),  restricted\_flag = NVL(restricted\_flag, 0)  WHERE completed\_flag IS NULL  OR third\_party\_involved\_flag IS NULL  OR law\_enforcement\_involve\_flag IS NULL  OR disabled\_infant\_nutrition\_flag IS NULL  OR disabled\_infant\_hydration\_flag IS NULL  OR disabled\_infant\_med\_flag IS NULL  OR disabled\_infant\_medtreat\_flag IS NULL  OR is\_disp\_complete IS NULL  OR restricted\_flag IS NULL;  UPDATE reporter  SET mandated\_reporter\_flag = NVL(mandated\_reporter\_flag, 0),  mandated\_reporter\_inform\_flag = NVL(mandated\_reporter\_inform\_flag, 0),  mandated\_reporter\_rqst\_flag = NVL(mandated\_reporter\_rqst\_flag, 0),  anonymous\_reporter\_flag = NVL(anonymous\_reporter\_flag, 0),  non\_mandated\_reporter\_flag = NVL(non\_mandated\_reporter\_flag, 0),  mandated\_reporter\_solo\_flag = NVL(mandated\_reporter\_solo\_flag, 0)  WHERE mandated\_reporter\_flag IS NULL  OR mandated\_reporter\_inform\_flag IS NULL  OR mandated\_reporter\_rqst\_flag IS NULL  OR anonymous\_reporter\_flag IS NULL  OR non\_mandated\_reporter\_flag IS NULL  OR mandated\_reporter\_solo\_flag IS NULL;  UPDATE reporter\_contact  SET src\_of\_knldg\_observed\_flag = NVL(src\_of\_knldg\_observed\_flag, 0),  src\_of\_knldg\_other\_party\_flag = NVL(src\_of\_knldg\_other\_party\_flag, 0),  addrd\_con\_parent\_ct\_flag = NVL(addrd\_con\_parent\_ct\_flag, 0)  WHERE src\_of\_knldg\_observed\_flag IS NULL  OR src\_of\_knldg\_other\_party\_flag IS NULL  OR addrd\_con\_parent\_ct\_flag IS NULL;  UPDATE intake\_participant  SET worker\_hazard\_flag = NVL(worker\_hazard\_flag, 0),  lep\_flag = NVL(lep\_flag, 0)  WHERE worker\_hazard\_flag IS NULL  OR lep\_flag IS NULL;  UPDATE intake\_participant\_acv  SET acv\_pend\_court\_hear\_flag = 0  WHERE acv\_pend\_court\_hear\_flag IS NULL;  UPDATE intake\_participant\_history  SET unknown\_flag = NVL(unknown\_flag, 0),  worker\_hazard\_flag = NVL(worker\_hazard\_flag, 0),  lep\_flag = NVL(lep\_flag, 0)  WHERE unknown\_flag IS NULL  OR worker\_hazard\_flag IS NULL  OR lep\_flag IS NULL;  UPDATE psa\_participant  SET is\_matched\_flag = NVL(is\_matched\_flag, 0),  is\_reference\_name\_flag = NVL(is\_reference\_name\_flag, 0)  WHERE is\_matched\_flag IS NULL  OR is\_reference\_name\_flag IS NULL;  UPDATE protective\_service\_alert  SET is\_if\_found\_safetycheck\_flag = NVL(is\_if\_found\_safetycheck\_flag, 0),  is\_if\_found\_notify\_flag = NVL(is\_if\_found\_notify\_flag, 0),  is\_if\_found\_other\_flag = NVL(is\_if\_found\_other\_flag, 0),  is\_notified\_police\_flag = NVL(is\_notified\_police\_flag, 0),  is\_obtained\_courtorder\_flag = NVL(is\_obtained\_courtorder\_flag, 0),  is\_conducted\_usps\_search\_flag = NVL(is\_conducted\_usps\_search\_flag, 0),  is\_cdjfs\_addr\_checked\_flag = NVL(is\_cdjfs\_addr\_checked\_flag, 0),  is\_contacted\_utilities\_flag = NVL(is\_contacted\_utilities\_flag, 0),  is\_contacted\_schools\_flag = NVL(is\_contacted\_schools\_flag, 0),  is\_contacted\_services\_flag = NVL(is\_contacted\_services\_flag, 0),  is\_contacted\_relatives\_flag = NVL(is\_contacted\_relatives\_flag, 0),  is\_other\_actions\_flag = NVL(is\_other\_actions\_flag, 0),  is\_canai\_inprocess\_flag = NVL(is\_canai\_inprocess\_flag, 0),  is\_supportive\_svcs\_flag = NVL(is\_supportive\_svcs\_flag, 0),  is\_prot\_sprvsn\_ordered\_flag = NVL(is\_prot\_sprvsn\_ordered\_flag, 0),  is\_custody\_obtained\_flag = NVL(is\_custody\_obtained\_flag, 0),  is\_childtaken\_violation\_flag = NVL(is\_childtaken\_violation\_flag, 0),  is\_othercase\_situation\_flag = NVL(is\_othercase\_situation\_flag, 0),  is\_docs\_none\_flag = NVL(is\_docs\_none\_flag, 0),  is\_docs\_court\_flag = NVL(is\_docs\_court\_flag, 0),  is\_docs\_warrant\_flag = NVL(is\_docs\_warrant\_flag, 0),  is\_docs\_med\_flag = NVL(is\_docs\_med\_flag, 0),  is\_docs\_photos\_flag = NVL(is\_docs\_photos\_flag, 0),  is\_docs\_additional\_flag = NVL(is\_docs\_additional\_flag, 0),  is\_docs\_sent\_mail\_flag = NVL(is\_docs\_sent\_mail\_flag, 0),  is\_docs\_sent\_fax\_flag = NVL(is\_docs\_sent\_fax\_flag, 0),  is\_docs\_sent\_email\_flag = NVL(is\_docs\_sent\_email\_flag, 0)  WHERE is\_if\_found\_safetycheck\_flag IS NULL  OR is\_if\_found\_notify\_flag IS NULL  OR is\_if\_found\_other\_flag IS NULL  OR is\_notified\_police\_flag IS NULL  OR is\_obtained\_courtorder\_flag IS NULL  OR is\_conducted\_usps\_search\_flag IS NULL  OR is\_cdjfs\_addr\_checked\_flag IS NULL  OR is\_contacted\_utilities\_flag IS NULL  OR is\_contacted\_schools\_flag IS NULL  OR is\_contacted\_services\_flag IS NULL  OR is\_contacted\_relatives\_flag IS NULL  OR is\_other\_actions\_flag IS NULL  OR is\_canai\_inprocess\_flag IS NULL  OR is\_supportive\_svcs\_flag IS NULL  OR is\_prot\_sprvsn\_ordered\_flag IS NULL  OR is\_custody\_obtained\_flag IS NULL  OR is\_childtaken\_violation\_flag IS NULL  OR is\_othercase\_situation\_flag IS NULL  OR is\_docs\_none\_flag IS NULL  OR is\_docs\_court\_flag IS NULL  OR is\_docs\_warrant\_flag IS NULL  OR is\_docs\_med\_flag IS NULL  OR is\_docs\_photos\_flag IS NULL  OR is\_docs\_additional\_flag IS NULL  OR is\_docs\_sent\_mail\_flag IS NULL  OR is\_docs\_sent\_fax\_flag IS NULL  OR is\_docs\_sent\_email\_flag IS NULL;  DROP TABLE sacwis\_fix.defect\_11610;  commit;  CREATE TABLE sacwis\_fix.defect\_11610 (  TABLENAME VARCHAR2(30),  ID NUMBER);        DECLARE    securityProfileId NUMBER;  businessFunctionId NUMBER;  roleId NUMBER;      BEGIN    select r.ROLE\_ID into roleId from role r where r.ROLE\_NAME = 'Supervisor';  select business\_function\_id\_seq.nextval into businessFunctionId from dual;    insert into sacwis\_fix.defect\_11610 values ('BUSINESS\_FUNCTION', businessFunctionId);  insert into business\_function values (businessFunctionId, 'Service Authorization Final Approval', 'FM05\_SA\_FINAL\_APPROVAL',  'Service Authorization Final Approval', 'Financial', 0, 0, 0, 0, 'ACTIVE', 100, SYSDATE, 100, SYSDATE);      select security\_profile\_id\_seq.nextval into securityProfileId from dual;      insert into sacwis\_fix.defect\_11610 values ('SECURITY\_PROFILE', securityProfileId);  insert into security\_profile spr values (securityProfileId, roleId, businessFunctionId, 'Service Authorization Final Approval',  'Service Authorization Final Approval', 0, 0, 100, SYSDATE, 100, SYSDATE,  0, 0, 0, 0);    COMMIT;  END;  /  create table temp\_datafix\_11061 as select \* from ref\_data where domain\_code = 'WaiverVarianceRequestType' and ref\_data\_code = 'VARIANCE';  update ref\_data set inactive\_flag = 1, modified\_date = sysdate where domain\_code = 'WaiverVarianceRequestType' and ref\_data\_code = 'VARIANCE';  create table temp\_datafix\_11061\_1 as select \* from ref\_data where domain\_code = 'CertificationRecommendTransact'  and ref\_data\_code in ('WAIVERONLY', 'VARIANCEONLY');  update ref\_data set inactive\_flag = 1, modified\_date = sysdate where domain\_code = 'CertificationRecommendTransact'  and ref\_data\_code in ('WAIVERONLY', 'VARIANCEONLY');  create table temp\_datafix\_11061 as select distinct rq.provider\_id, pn.NAME\_TEXT  from PRVDR\_WAIVER\_VARIANCE\_REQUEST rq, PRVDR\_WAIVER\_VARIANCE\_RESPONSE rs, provider\_name pn  where rq.PRVDR\_WAIVER\_VAR\_REQ\_ID = rs.PRVDR\_WAIVER\_VAR\_REQ\_ID and rs.DECISION\_CODE <> 'PENDING'  and rq.PROVIDER\_ID = pn.PROVIDER\_ID and pn.END\_DATE is null  and not exists (select workload\_item\_id from workload\_item where work\_item\_id = rq.provider\_id and work\_item\_type\_code = 'PROVIDER');  insert into workload\_item select WORKLOAD\_ITEM\_ID\_SEQ.nextval,  provider\_id, 'PROVIDER', NAME\_TEXT, 999, sysdate, 999, sysdate from temp\_datafix\_11061;  insert into routed\_task\_history select routed\_task\_history\_id\_seq.nextval, 63713,  (select workload\_item\_id from workload\_item where work\_item\_id = rq.provider\_id and work\_item\_type\_code = 'PROVIDER'),  rs.EFFECTIVE\_DATE, 'A', rq.PRVDR\_WAIVER\_VAR\_REQ\_ID, 'WAIVER', null, 1, 999, sysdate, 999, sysdate, 999, null, 'AF'  from PRVDR\_WAIVER\_VARIANCE\_REQUEST rq, PRVDR\_WAIVER\_VARIANCE\_RESPONSE rs  where rq.PRVDR\_WAIVER\_VAR\_REQ\_ID = rs.PRVDR\_WAIVER\_VAR\_REQ\_ID and rs.DECISION\_CODE = 'ACCEPTED';  insert into routed\_task\_history select routed\_task\_history\_id\_seq.nextval, 63713,  (select workload\_item\_id from workload\_item where work\_item\_id = rq.provider\_id and work\_item\_type\_code = 'PROVIDER'),  rs.EFFECTIVE\_DATE, 'D', rq.PRVDR\_WAIVER\_VAR\_REQ\_ID, 'WAIVER', null, 1, 999, sysdate, 999, sysdate, 999, null, 'D'  from PRVDR\_WAIVER\_VARIANCE\_REQUEST rq, PRVDR\_WAIVER\_VARIANCE\_RESPONSE rs  where rq.PRVDR\_WAIVER\_VAR\_REQ\_ID = rs.PRVDR\_WAIVER\_VAR\_REQ\_ID and rs.DECISION\_CODE = 'REJECTED';  create table temp\_datafix\_11478 as select \* from tickler where tickler\_template\_code = 'RM04\_PROV\_ANNUAL\_REVIEW' and tickler\_status\_code = 'Open';  update tickler set completion\_date = to\_date(sysdate, 'MM/DD/YYYY'), tickler\_status\_code = 'Disposed', modified\_by = 999, modified\_date = sysdate  where tickler\_template\_code = 'RM04\_PROV\_ANNUAL\_REVIEW' and tickler\_status\_code = 'Open';  create table temp\_datafix\_14123\_pta as select \* from provider\_type where type\_code = 'ADOPTIVECARE' and current\_status\_code = 'CERTIFIED';  create table temp\_datafix\_14123\_ptsa as select \* from provider\_type\_status where provider\_type\_id in (select provider\_type\_id  from provider\_type where type\_code = 'ADOPTIVECARE') and type\_status\_code = 'CERTIFIED';  create table temp\_datafix\_14123\_ptf as select \* from provider\_type where type\_code = 'FOSTERCARE' and current\_status\_code = 'APPROVED';  create table temp\_datafix\_14123\_ptsf as select \* from provider\_type\_status where provider\_type\_id in (select provider\_type\_id  from provider\_type where type\_code = 'FOSTERCARE') and type\_status\_code = 'APPROVED';  update provider\_type set current\_status\_code = 'APPROVED', modified\_by = 999, modified\_date = sysdate  where type\_code = 'ADOPTIVECARE' and current\_status\_code = 'CERTIFIED';  update provider\_type\_status set type\_status\_code = 'APPROVED', modified\_by = 999, modified\_date = sysdate  where provider\_type\_id in (select provider\_type\_id from provider\_type where type\_code = 'ADOPTIVECARE') and type\_status\_code = 'CERTIFIED';  update provider\_type set current\_status\_code = 'CERTIFIED', modified\_by = 999, modified\_date = sysdate  where type\_code = 'FOSTERCARE' and current\_status\_code = 'APPROVED';  update provider\_type\_status set type\_status\_code = 'CERTIFIED', modified\_by = 999, modified\_date = sysdate  where provider\_type\_id in (select provider\_type\_id from provider\_type where type\_code = 'FOSTERCARE') and type\_status\_code = 'APPROVED';  DROP TABLE sacwis\_fix.defect\_13212\_fix;  commit;  CREATE TABLE sacwis\_fix.defect\_13212\_fix as (select sp.STATE\_PAYMENT\_ID,sp.REIMBURSED\_UNITS\_AMT, pr.PAID\_UNITS\_AMT,  sp.MODIFIED\_BY, sp.MODIFIED\_DATE  from payment\_request pr, state\_payment sp  where pr.service\_cat\_code = 'Foster Care Miscellaneous'  and pr.created\_date > '12/01/2010'  and sp.payment\_request\_id = pr.payment\_request\_id  and sp.reimbursed\_units\_amt > pr.paid\_units\_amt  and nvl(pr.action\_type\_code, '1') != 'Overpayment');      DECLARE CURSOR c is select \* from sacwis\_fix.defect\_13212\_fix;    counter NUMBER := 0;    BEGIN  FOR r in c  LOOP  update STATE\_PAYMENT sp set sp.REIMBURSED\_UNITS\_AMT = r.paid\_units\_amt, sp.MODIFIED\_BY = 999, sp.MODIFIED\_DATE = SYSDATE  where sp.STATE\_PAYMENT\_ID = r.STATE\_PAYMENT\_ID;    counter := counter + 1;    END LOOP;    COMMIT;    DBMS\_OUTPUT.PUT\_LINE ( 'Number of records updated: = ' || counter );    END;  /  DROP TABLE sacwis\_fix.defect\_14643\_fix;  commit;  CREATE TABLE sacwis\_fix.defect\_14643\_fix as (select pr.PAYMENT\_REQUEST\_ID, pr.REIMBURSEABILITY\_CHANGE\_FLAG, pr.MODIFIED\_BY, pr.MODIFIED\_DATE  from payment\_request pr  inner join state\_payment sp on sp.PAYMENT\_REQUEST\_ID = pr.PAYMENT\_REQUEST\_ID  where pr.TRANSACTION\_CODE = 'CC'  and trunc(paid\_units\_amt) <> paid\_units\_amt );  DECLARE    CURSOR c is select \* from sacwis\_fix.defect\_14643\_fix;    counter NUMBER := 0;  BEGIN  FOR r in c  LOOP  update payment\_request pr set pr.REIMBURSEABILITY\_CHANGE\_FLAG = 1, pr.MODIFIED\_BY = 999, pr.MODIFIED\_DATE = SYSDATE  where pr.PAYMENT\_REQUEST\_ID = r.payment\_request\_id;    counter := counter + 1;    END LOOP;  COMMIT;  DBMS\_OUTPUT.PUT\_LINE ( 'counter = ' || counter );  END;  /  DROP TABLE sacwis\_fix.defect\_11958\_fix;  commit;  CREATE TABLE sacwis\_fix.defect\_11958\_fix as (select \*  from ROUTED\_TASK\_HISTORY rth  where rth.TASK\_TYPE\_CODE = 'PR'  and NOT EXISTS (select pro.ROSTER\_ID  from payment\_roster pro  where pro.ROSTER\_ID = rth.TASK\_ID));  DECLARE    CURSOR c is select \* from sacwis\_fix.defect\_11958\_fix;      counter NUMBER := 0;    BEGIN  FOR r in c  LOOP    delete from ROUTED\_TASK\_HISTORY rth where rth.ROUTED\_TASK\_HISTORY\_ID = r.ROUTED\_TASK\_HISTORY\_ID;    counter := counter + 1;    END LOOP;  COMMIT;    DBMS\_OUTPUT.PUT\_LINE ( 'Number of ROUTED TASK HISTORY records deleted = ' || counter );    END;  /  create table temp\_datafix\_11061\_plr as select \* from provider\_licensing\_request where TRANSACTION\_TYPE\_CODE in ('WAIVERONLY', 'VARIANCEONLY')  and PRVDR\_LICENSING\_REQUEST\_ID not in (select task\_id from routed\_task\_history where TASK\_TYPE\_CODE = 'PLR' and CURRENT\_STATUS\_FLAG = 1  and TASK\_ACTION\_CODE in ('AF', 'D')) and PRVDR\_LICENSING\_REQUEST\_ID not in (select PRVDR\_LICENSING\_REQUEST\_ID from PROVIDER\_LICENSING\_WV\_LINK);  create table temp\_datafix\_11061\_rth as select \* from routed\_task\_history where TASK\_TYPE\_CODE = 'PLR'  and task\_id in (select PRVDR\_LICENSING\_REQUEST\_ID from temp\_datafix\_11061\_plr);  delete from routed\_task\_history where TASK\_TYPE\_CODE = 'PLR' and task\_id in (select PRVDR\_LICENSING\_REQUEST\_ID from temp\_datafix\_11061\_plr);  delete from provider\_licensing\_request where PRVDR\_LICENSING\_REQUEST\_ID in (select PRVDR\_LICENSING\_REQUEST\_ID from temp\_datafix\_11061\_plr);  create table temp\_datafix\_11061\_nt1 as select \* from NOTIFICATION\_TEMPLATE where NOTIFICATION\_CODE = 'PROVIDER\_WAIVER\_APPROVAL';  update NOTIFICATION\_TEMPLATE set SUBJECT\_TEXT = 'Waiver Approval/Denial', MESSAGE\_TEXT = ' ' where NOTIFICATION\_CODE = 'PROVIDER\_WAIVER\_APPROVAL'; |
| Audit Tables Updated | New Tables and Modified Tables  Reference Above for Table and Column Adds and Modifications |
| List of reports deployed : | Reference Bill Ennis |
| List of reports deactivated : | Reference Bill Ennis |

**Section 3 – Testing Plan/Results**

3.1 Test Plan

|  |  |
| --- | --- |
| Identify the database used to test the software/SQL: | Flair , system test |
| How many days removed from production is the data in this database? | n/a |
| Name of person(s) completing the test in this database: | Stephen Fibelkorn |
| Identify the database used for user acceptance test: | UAT |
| How many days removed from production is the data in this database? | Start of build Cycle (32 Days) |
| Name of person(s) completing the user acceptance test in this database: | Iroabuchi Arum |

**Section 4 – Production Execution**

* 1. Execution Plan

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Identify whether a full or partial backup of the production database is needed, or indicate that no backup is needed. | Full | **Successful RMAN backup Needed** | Partial |  |
| Explanation: | | | |
| Identify when the backup will be taken: | Date: 12/15/2011  Time: 9.00 pm | | | |
| Identify who will apply the software/SQL: | ODJFS DBA | | | |

* 1. Execution Results

|  |  |
| --- | --- |
| Location of results documentation: | Results to be e-mailed to the SACWIS team. |