

# Appendix F

## Database Trigger Definitions

---

### Oracle Designer

Report : DATABASE TRIGGER

Filename :

c:\tmp\rpa\_cktrgger.html

Run by

CMA

Report Date

:

10 February 2004

Total Pages :

71

Parameter Values

Workarea :

GLOBAL SHARED WORKAREA

Container :

II\_RPA

Container Version :

Recurse Sub-Containers :

N

Table/View :

%

Trigger Name :

%

Database Triggers Created

On/After :

On/Before :

10 February 2004

and

Database Triggers Changed

On/After :

On/Before :

10 February 2004

Database Trigger Definition

```

Container:
    II_RPA
    Version :
Table/View Name
:
    LAND_UNITS (table)
Trigger Name
:
    II_LU_ADR_TRG
                                Create?
                                Yes

PL/SQL Module
Name :
    II_LU_ADR_TRG
                                Enabled?
                                Yes

PL/SQL File :
Purpose :
    Keep deleting conciled with InfraCentral
Trigger Time:
    AFTER
                                Level:
                                ROW
                                Insert?
                                No
                                Delete?
                                Yes

New Alias
                                Update?
                                No

Old Alias :
Trigger Logic
DECLARE
vThisInstanceId ii_db_instances.instance_id%type;
BEGIN
SELECT instance_id INTO vThisInstanceId
FROM this_db_instance;

ii_rpa_remote.deleteLandUnit@ic_remote(
vThisInstanceId
,old.cn
,USER
,SYSDATE);
END;

```

Oracle Designer Report : cktrgger  
Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

LAND\_UNITS (table)

Trigger Name

II\_TD\_LU\_TRG

Create?

Yes

PL/SQL Module

Name :

II\_TD\_LU\_TRG

Enabled?

Yes

PL/SQL File :

Purpose :

Trigger Time:

BEFORE

Level:

ROW

Insert?

No

Delete?

Yes

New Alias

NEW

Update?

No

Old Alias :

OLD

Trigger Logic

DECLARE

vThisInstanceId ii\_db\_instances.instance\_id%type;

BEGIN

SELECT instance\_id INTO vThisInstanceId

FROM this\_db\_instance;

ii\_rpa\_remote.deleteLandUnit@ic\_remote(

vThisInstanceId

,:old.cn

,USER

,SYSDATE);

END;

```
DECLARE
BEGIN
BEGIN
IF not ii_chk_role_assign('INFRA_SVC_MGR') and
ii_cent_mgd_lu_obj(:old.obj_name) then
raise_application_error(-20301,'Insufficient privileges to ||
'delete ||initcap(replace(:old.obj_name,'_' '))||' Land Units');
END IF;
--
-- Added by RAdams on 6/24/2002 for v5.0:
--
-- Record old/new ID and CN in PL/SQL table for purposes of
-- writing row to range permits audit table (ii_range_pmt_li_audit)
when the deleted
-- Land Unit ID is an RMU, State, Proclaimed Unit, or County
referenced in a
-- grazing permit. This action takes place in the statement-level
-- after-update/delete trigger that first queries the range permit
line items table
-- to check if this land unit is an RMU, State, Proclaimed Unit, or
County referenced in
-- a grazing permit. If this is the case, a re-query on the land_units
table is necessary
-- to get the remaining RMU, State, Proclaimed Unit, and County IDs to
be inserted
-- into the audit table. To avoid the "mutating table"
Oracle Designer Report : cktrgger
```

## Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

LAND\_UNITS (table)

error, the re-query of the land\_units

-- table needed to be moved to a statement-level after update/delete

trigger, passing "old" and "new"

-- values in the package PL/SQL table referenced in the code below.

--

if nvl(:new.id,'@') != nvl(:old.id,'@') then

ii\_range\_permits\_audit.vNumEntries :=

ii\_range\_permits\_audit.vNumEntries + 1;

ii\_range\_permits\_audit.vPRAudit(ii\_range\_permits\_audit.vNumEntries).ne

w\_id := :new.id;

ii\_range\_permits\_audit.vPRAudit(ii\_range\_permits\_audit.vNumEntries).ol

d\_id := :old.id;

ii\_range\_permits\_audit.vPRAudit(ii\_range\_permits\_audit.vNumEntries).ne

w\_cn := :new.cn;

ii\_range\_permits\_audit.vPRAudit(ii\_range\_permits\_audit.vNumEntries).ol

d\_cn := :old.cn;

end if;

--

END;

END;

Oracle Designer Report : cktrgger

Database Trigger Definition

```

Container:
  II_RPA
  Version :
Table/View Name
:
  II_CONTACTS (table)
Trigger Name
:
  II_CONT_RBD_TRG
                                Create?
                                Yes
PL/SQL Module
Name :
  II_CONT_RBD_TRG
                                Enabled?
                                Yes
PL/SQL File :
Purpose :
  Row-level before-delete trigger
Trigger Time:
  BEFORE
                                Level:
                                ROW
                                Insert?
                                No
                                Delete?
                                Yes
New Alias
                                Update?
                                No
Old Alias :
Trigger Logic
Begin
--
-- Added by RAdams on 6/24/2002 for v5.0:
--
-- Record old/new ID and CN in PL/SQL table for purposes of
-- writing row to range permits audit table (ii_range_permit_audits)
when the deleted

```

```
-- Contact ID is a permittee or managing organization for a
-- grazing permit. This action takes place in the statement-level
-- after-update/delete trigger that first queries the range permits
view
-- to check if this contact is a permittee or managing organization
for
-- a grazing permit. If this is the case, a re-query on the
ii_contacts table is necessary
-- to get the remaining Permittee or Managing Org ID to be inserted
-- into the audit table. To avoid the "mutating table" error, the
re-query of the ii_contacts
-- table needed to be moved to a statement-level after update/delete
trigger, passing "old" and "new"
-- values in the package PL/SQL table referenced in the code below.
--
if nvl(:new.id,'@') != nvl(:old.id,'@') then
ii_range_permits_audit.vNumEntries :=
ii_range_permits_audit.vNumEntries + 1;
ii_range_permits_audit.vPRAudit(ii_range_permits_audit.vNumEntries).ne
w_id := :new.id;
ii_range_permits_audit.vPRAudit(ii_range_permits_audit.vNumEntries).ol
d_id := :old.id;
ii_range_permits_audit.vPRAudit(ii_range_permits_audit.vNumEntries).ne
w_cn := :new.cn;
ii_range_permits_audit.vPRAudit(ii_range_permits_audit.vNumEntries).ol
d_cn := :old.cn;
end if;
--
End;
```

Oracle Designer Report : cktrgger

Database Trigger Definition

```

Container:
  II_RPA
  Version :
Table/View Name
:
  II_FEATURES (table)
Trigger Name
:
  II_FEA_ADR_TRG
                                     Create?
                                     Yes

PL/SQL Module
Name :
  II_FEA_ADR_TRG
                                     Enabled?
                                     Yes

PL/SQL File :
Purpose :
  Keep deleting conciled with InfraCentral.
Trigger Time:
  AFTER
                                     Level:
                                     ROW
                                     Insert?
                                     No
                                     Delete?
                                     Yes

New Alias
                                     Update?
                                     No

Old Alias :
Trigger Logic
DECLARE
vThisInstanceId ii_db_instances.instance_id%type;
BEGIN
SELECT instance_id INTO vThisInstanceId
FROM this_db_instance;

ii_rpa_remote.deleteFeature@ic_remote(
vThisInstanceId
,old.cn
,USER
,SYSDATE);
END;

```



Oracle Designer Report : cktrgger

10 February 2004

Page  
7  
of  
71

Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

II\_RPA\_SUBLEDGERS (table)

Trigger Name

II\_RPA\_SUB\_ADR

Create?

Yes

PL/SQL Module

Name :

II\_RPA\_SUB\_ADR

Enabled?

Yes

PL/SQL File :

Purpose :

Trigger Time:

AFTER

Level:

ROW

Insert?

No

Delete?

Yes

New Alias

Update?

No

Old Alias :

Trigger Logic

-- Refresh\_v\_tab

BEGIN

UPDATE ii\_rpa\_assets\_v\_tab vtab

SET subledger\_flag = 'N'

,subledger\_cn = NULL -- Added by C.Ma, 04/14/2003

WHERE (vtab.lu\_cn = :old.lu\_cn

OR vtab.fea\_cn = :old.fea\_cn

OR vtab.pld\_cn = :old.pld\_cn);

END;

```
-- Audit subledger
DECLARE
vOldSub ii_rpa_subledgers%rowtype;
BEGIN
vOldSub.DEPR_END_DATE := :old.DEPR_END_DATE;
vOldSub.CN := :old.CN;
vOldSub.SUBLEDGER_TYPE := :old.SUBLEDGER_TYPE;
vOldSub.PROJECT_GROUP := :old.PROJECT_GROUP;
vOldSub.CAPITAL_PROJECT_NO := :old.CAPITAL_PROJECT_NO;
vOldSub.MEETS_THRESHOLD_IND := :old.MEETS_THRESHOLD_IND;
vOldSub.LU_CN := :old.LU_CN;
vOldSub.FEA_CN := :old.FEA_CN;
vOldSub.PLD_CN := :old.PLD_CN;
vOldSub.ADJ_CN := :old.ADJ_CN;
vOldSub.USAGE_CODE := :old.USAGE_CODE;
vOldSub.ORG := :old.ORG;
vOldSub.ADMIN_ORG := :old.ADMIN_ORG;
vOldSub.CAPITAL_PROJECT_NO_SAV := :old.CAPITAL_PROJECT_NO_SAV;
vOldSub.REMARKS := :old.REMARKS;
vOldSub.CREATED_BY := :old.CREATED_BY;
vOldSub.CREATED_DATE := :old.CREATED_DATE;
vOldSub.CREATED_IN_INSTANCE := :old.CREATED_IN_INSTANCE;
vOldSub.MODIFIED_BY := :old.MODIFIED_BY;
vOldSub.MODIFIED_DATE := :old.MODIFIED_DATE;
vOldSub.MODIFIED_IN_INSTANCE := :old.MODIFIED_IN_INSTANCE;
vOldSub.REC_STATUS := :old.REC_STATUS;

-- Added on 01/05/2004
vOldSub.agency_code := :old.agency_code;
vOldSub.location_id := :old.location_id;
vOldSub.asset_id := :old.asset_id;
```

Oracle Designer Report : cktrgger

Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

II\_RPA\_SUBLEDGERS (table)

ii\_rpa\_auditing.audit\_sub('D', USER, SYSDATE, vOldSub, NULL);

END;

Oracle Designer Report : cktrgger





```

--
-- FETCH DESC FOR REC TYPE.
CURSOR GSA_REC_TYPE_CUR(
search_rec_type ii_record_types.rec_type%type )
IS
SELECT description
FROM ii_record_types
WHERE rec_type = search_rec_type;
--
-- FETCH DESC FOR GSA USE CODE.
CURSOR GSA_USE_CODE_CUR(
search_use_code ii_use_codes.use_code%type )
IS
SELECT description
FROM ii_use_codes
WHERE use_code = search_use_code;
BEGIN
BEGIN
IF :new.use_code IS NOT NULL and
:new.rec_type IS NOT NULL THEN
:new.usage_code := :new.rec_type || :new.use_code;
open gsa_use_code_cur (:new.use_code);
fetch gsa_use_code_cur into tmp_use_code_desc;
close gsa_use_code_cur;
open gsa_rec_type_cur (:new.rec_type);
fetch gsa_rec_type_cur into tmp_rec_type_desc;
close gsa_rec_type_cur;
:new.name :=
substr( tmp_use_code_desc||'-'|| tmp_rec_type_desc, 1,120);
END IF;
END;
END;

```

Oracle Designer Report : cktrgger

Database Trigger Definition

```

Container:
  II_RPA
  Version :
Table/View Name
:
  LAND_UNITS (table)
Trigger Name
:
  II_LU_ASUD_TRG
                                     Create?
                                     Yes

PL/SQL Module
Name :
  II_LU_ASUD_TRG
                                     Enabled?
                                     Yes

PL/SQL File :
Purpose :
  After statement update/delete.
Trigger Time:
  AFTER
                                     Level:
                                     STATEMENT
                                     Insert?
                                     No
                                     Delete?
                                     Yes

New Alias
                                     Update?
                                     Yes

Old Alias :
Trigger Logic
DECLARE
--
--
tmp_rec_type_desc VARCHAR2(60);
--
--

```

```

tmp_use_code_desc VARCHAR2(60);
--
-- FETCH DESC FOR REC TYPE.
CURSOR GSA_REC_TYPE_CUR(
search_rec_type ii_record_types.rec_type%type )
IS
SELECT description
FROM ii_record_types
WHERE rec_type = search_rec_type;
--
-- FETCH DESC FOR GSA USE CODE.
CURSOR GSA_USE_CODE_CUR(
search_use_code ii_use_codes.use_code%type )
IS
SELECT description
FROM ii_use_codes
WHERE use_code = search_use_code;
BEGIN
BEGIN
IF :new.use_code IS NOT NULL and
:new.rec_type IS NOT NULL THEN
:new.usage_code := :new.rec_type || :new.use_code;
open gsa_use_code_cur (:new.use_code);
fetch gsa_use_code_cur into tmp_use_code_desc;
close gsa_use_code_cur;
open gsa_rec_type_cur (:new.rec_type);
fetch gsa_rec_type_cur into tmp_rec_type_desc;
close gsa_rec_type_cur;
:new.name :=
substr( tmp_use_code_desc||'-'|| tmp_rec_type_desc, 1,120);
END IF;
END;
END;
Declare
--
vCN land_units.cn%TYPE := null;
Oracle Designer Report : cktrgger

```



## Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

LAND\_UNITS (table)

--

-- Cursor to check for FK to Grazing Permit line item RMUs

cursor cRMU is

select 'x'

from ii\_range\_permit\_line\_items

where rmu\_lu\_cn = vCN; -- :old.cn;

--

-- Cursor to check for FK to Grazing Permit line item State IDs

cursor cState is

select 'x'

from ii\_range\_permit\_line\_items

where lu\_state\_cn = vCN; -- :old.cn;

--

-- Cursor to check for FK to Grazing Permit line item Proclaimed  
Unit/County IDs

cursor cPUCnty is

select 'x'

from ii\_range\_permit\_line\_items

where lu\_proc\_unit\_cn = vCN; -- :old.cn;

--

vDummy varchar2(1);

--

Begin

--

for vIndex in 1..ii\_range\_permits\_audit.vNumEntries loop

--

vCN := ii\_range\_permits\_audit.vPRAudit(vIndex).old\_cn;

--

```
-- Check for FK to Grazing Permit line item RMUs.
-- If found then write audit record. If not, then continue
open cRMU;
fetch cRMU into vDummy;
if cRMU%found then
close cRMU;
ii_range_permits_audit.Write_Grz_Permit_LI_LU_Audits (
ii_range_permits_audit.vPRAudit(vIndex).old_cn,
ii_range_permits_audit.vPRAudit(vIndex).old_id,
ii_range_permits_audit.vPRAudit(vIndex).new_id,
'RPLI_RMU_FK' );
else
close cRMU;
end if;
--
-- Check for FK to Grazing Permit line item RMUs.
-- If found then write audit record. If not, then continue
open cState;
fetch cState into vDummy;
if cState%found then
close cState;
ii_range_permits_audit.Write_Grz_Permit_LI_LU_Audits (
ii_range_permits_audit.vPRAudit(vIndex).old_cn,
ii_range_permits_audit.vPRAudit(vIndex).old_id,
ii_range_permits_audit.vPRAudit(vIndex).new_id,
```

Oracle Designer Report : cktrgger

## Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

LAND\_UNITS (table)

'RPLI\_LU\_STATE\_FK');

else

close cState;

end if;

--

-- Check for FK to Grazing Permit line item Proclaimed Unit/County IDs.

-- If found then write audit record. If not, then continue

open cPUCnty;

fetch cPUCnty into vDummy;

if cPUCnty%found then

close cPUCnty;

ii\_range\_permits\_audit.Write\_Grz\_Permit\_LI\_LU\_Audits (

ii\_range\_permits\_audit.vPRAudit(vIndex).old\_cn,

ii\_range\_permits\_audit.vPRAudit(vIndex).old\_id,

ii\_range\_permits\_audit.vPRAudit(vIndex).new\_id,

'RPLI\_LU\_PROC\_UNIT\_FK');

else

close cPUCnty;

end if;

--

end loop;

--

-- Re-set counter so that next update statement in this session works correctly.

ii\_range\_permits\_audit.vNumEntries := 0;

--

End;

Oracle Designer Report : cktrgger

Database Trigger Definition

```

Container:
  II_RPA
  Version :
Table/View Name
:
  II_CONTACTS (table)
Trigger Name
:
  II_CONT_ASUD_TRG
                                Create?
                                Yes
PL/SQL Module
Name :
  II_CONT_ASUD_TRG
                                Enabled?
                                Yes
PL/SQL File :
Purpose :
  After update/delete statement-level trigger.
Trigger Time:
  AFTER
                                Level:
                                STATEMENT
                                Insert?
                                No
                                Delete?
                                Yes
New Alias
                                Update?
                                Yes
Old Alias :
Trigger Logic
Declare
--
vCN land_units.cn%TYPE := null;
--
-- Cursor to check for FK to Grazing Permit line item RMUs
cursor cRMU is
select 'x'
from ii_range_permit_line_items
where rmu_lu_cn = vCN; -- :old.cn;

```

```

--
-- Cursor to check for FK to Grazing Permit line item State IDs
cursor cState is
select 'x'
from ii_range_permit_line_items
where lu_state_cn = vCN; -- :old.cn;
--
-- Cursor to check for FK to Grazing Permit line item Proclaimed
Unit/County IDs
cursor cPUCnty is
select 'x'
from ii_range_permit_line_items
where lu_proc_unit_cn = vCN; -- :old.cn;
--
vDummy varchar2(1);
--
Begin
--
for vIndex in 1..ii_range_permits_audit.vNumEntries loop
--
vCN := ii_range_permits_audit.vPRAudit(vIndex).old_cn;
--
-- Check for FK to Grazing Permit line item RMUs.
-- If found then write audit record. If not, then continue
open cRMU;
fetch cRMU into vDummy;
if cRMU%found then
close cRMU;
ii_range_permits_audit.Write_Grz_Permit_LI_LU_Audits (
ii_range_permits_audit.vPRAudit(vIndex).old_cn,
ii_range_permits_audit.vPRAudit(vIndex).old_id,

```

Oracle Designer Report : cktrgger

## Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

II\_CONTACTS (table)

```

ii_range_permits_audit.vPRAudit(vIndex).new_id,
'RPLI_RMU_FK' );
else
close cRMU;
end if;
--
-- Check for FK to Grazing Permit line item RMUs.
-- If found then write audit record. If not, then continue
open cState;
fetch cState into vDummy;
if cState%found then
close cState;
ii_range_permits_audit.Write_Grz_Permit_LI_LU_Audits (
ii_range_permits_audit.vPRAudit(vIndex).old_cn,
ii_range_permits_audit.vPRAudit(vIndex).old_id,
ii_range_permits_audit.vPRAudit(vIndex).new_id,
'RPLI_LU_STATE_FK' );
else
close cState;
end if;
--
-- Check for FK to Grazing Permit line item Proclaimed Unit/County
IDs.
-- If found then write audit record. If not, then continue
open cPUCnty;
fetch cPUCnty into vDummy;
if cPUCnty%found then
close cPUCnty;
ii_range_permits_audit.Write_Grz_Permit_LI_LU_Audits (
ii_range_permits_audit.vPRAudit(vIndex).old_cn,
ii_range_permits_audit.vPRAudit(vIndex).old_id,
ii_range_permits_audit.vPRAudit(vIndex).new_id,
'RPLI_LU_PROC_UNIT_FK' );
else
close cPUCnty;
end if;

```

```
--  
end loop;  
--  
-- Re-set counter so that next update statement in this session works  
correctly.  
ii_range_permits_audit.vNumEntries := 0;  
--  
End;
```

Declare

```
--  
vCN ii_contacts.cn%TYPE := null;  
--  
-- Cursor to check for FK to Grazing Permit permittees  
cursor cPmtee is  
  
Oracle Designer Report : cktrgger
```

## Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

II\_CONTACTS (table)

```

select 'x'
from ii_range_permits
where permittee_cn = vCN; -- :old.cn;
--
-- Cursor to check for FK to Grazing Permit managing organizations
cursor cMOrg is
select 'x'
from accomplishment_instruments
where managing_cont_cn = vCN and -- :old.cn and
obj_name = 'RANGE_PERMIT';
--
vDummy varchar2(1);
--
Begin
--
for vIndex in 1..ii_range_permits_audit.vNumEntries loop
--
vCN := ii_range_permits_audit.vPRAudit(vIndex).old_cn;
--
-- Check for FK to Grazing Permit permittees.
-- If found then write audit record. If not, then continue
open cPmtee;
fetch cPmtee into vDummy;
if cPmtee%found then
close cPmtee;
ii_range_permits_audit.Write_Grz_Pmt_Contact_Audits (
ii_range_permits_audit.vPRAudit(vIndex).old_cn,
ii_range_permits_audit.vPRAudit(vIndex).old_id,
ii_range_permits_audit.vPRAudit(vIndex).new_id,
'RNGPMT_CONT_CN' );
else
close cPmtee;
end if;

```



```
--  
-- Check for FK to Grazing Permit managing orgs.  
-- If found then write audit record. If not, then continue  
open cMOrg;  
fetch cMOrg into vDummy;  
if cMOrg%found then  
close cMOrg;  
ii_range_permits_audit.Write_Grz_Pmt_Contact_Audits (  
ii_range_permits_audit.vPRAudit(vIndex).old_cn,  
ii_range_permits_audit.vPRAudit(vIndex).old_id,  
ii_range_permits_audit.vPRAudit(vIndex).new_id,  
'ACCINST_CONT_FK' );  
else  
close cMOrg;  
end if;  
--  
end loop;  
--  
-- Re-set counter so that next update statement in this session works  
correctly.
```

Oracle Designer Report : cktrgger

#### Database Trigger Definition

```
Container:  
    II_RPA  
    Version :  
Table/View Name  
:  
    II_CONTACTS (table)  
ii_range_permits_audit.vNumEntries := 0;  
--  
End;
```

Oracle Designer Report : cktrgger

10 February 2004

Page  
18  
of  
71

Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

LAND\_UNITS (table)

Trigger Name

:

LAND\_UNITS\_TRG

Create?

Yes

PL/SQL Module

Name :

LAND\_UNITS\_TRG

Enabled?

Yes

PL/SQL File :

Purpose :

Trigger Time:

BEFORE

Level:

ROW

Insert?

Yes

Delete?

No

New Alias

NEW

Update?

Yes

Old Alias :

OLD

Trigger Logic

Declare

--

vCN ii\_contacts.cn%TYPE := null;

```

--
-- Cursor to check for FK to Grazing Permit permittees
cursor cPmtee is
select 'x'
from ii_range_permits
where permittee_cn = vCN; -- :old.cn;
--
-- Cursor to check for FK to Grazing Permit managing organizations
cursor cMOrg is
select 'x'
from accomplishment_instruments
where managing_cont_cn = vCN and -- :old.cn and
obj_name = 'RANGE_PERMIT';
--
vDummy varchar2(1);
--
Begin
--
for vIndex in 1..ii_range_permits_audit.vNumEntries loop
--
vCN := ii_range_permits_audit.vPRAudit(vIndex).old_cn;
--
-- Check for FK to Grazing Permit permittees.
-- If found then write audit record. If not, then continue
open cPmtee;
fetch cPmtee into vDummy;
if cPmtee%found then
close cPmtee;
ii_range_permits_audit.Write_Grz_Pmt_Contact_Audits (
ii_range_permits_audit.vPRAudit(vIndex).old_cn,
ii_range_permits_audit.vPRAudit(vIndex).old_id,
ii_range_permits_audit.vPRAudit(vIndex).new_id,
'RNGPMT_CONT_CN' );
else
close cPmtee;
end if;

```

Oracle Designer Report : cktrgger

## Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

LAND\_UNITS (table)

--

-- Check for FK to Grazing Permit managing orgs.

-- If found then write audit record. If not, then continue

open cMOrg;

fetch cMOrg into vDummy;

if cMOrg%found then

close cMOrg;

ii\_range\_permits\_audit.Write\_Grz\_Pmt\_Contact\_Audits (

ii\_range\_permits\_audit.vPRAudit(vIndex).old\_cn,

ii\_range\_permits\_audit.vPRAudit(vIndex).old\_id,

ii\_range\_permits\_audit.vPRAudit(vIndex).new\_id,

'ACCINST\_CONT\_FK' );

else

close cMOrg;

end if;

--

end loop;

--

-- Re-set counter so that next update statement in this session works  
correctly.

ii\_range\_permits\_audit.vNumEntries := 0;

--

End;

Declare

--

vCN ii\_contacts.cn%TYPE := null;

--

-- Cursor to check for FK to Grazing Permit permittees

cursor cPmtee is

select 'x'

from ii\_range\_permits

where permittee\_cn = vCN; -- :old.cn;

```
--  
-- Cursor to check for FK to Grazing Permit managing organizations  
cursor cMOrg is  
select 'x'  
from accomplishment_instruments  
where managing_cont_cn = vCN and -- :old.cn and  
obj_name = 'RANGE_PERMIT';  
--  
vDummy varchar2(1);  
--  
Begin  
--  
for vIndex in 1..ii_range_permits_audit.vNumEntries loop  
--  
vCN := ii_range_permits_audit.vPRAudit(vIndex).old_cn;  
--  
-- Check for FK to Grazing Permit permittees.  
-- If found then write audit record. If not, then continue  
open cPmtee;  
fetch cPmtee into vDummy;  
if cPmtee%found then  
close cPmtee;
```

Oracle Designer Report : cktrgger

## Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

LAND\_UNITS (table)

```

ii_range_permits_audit.Write_Grz_Pmt_Contact_Audits (
ii_range_permits_audit.vPRAudit(vIndex).old_cn,
ii_range_permits_audit.vPRAudit(vIndex).old_id,
ii_range_permits_audit.vPRAudit(vIndex).new_id,
'RNGPMT_CONT_CN' );
else
close cPmtee;
end if;
--
-- Check for FK to Grazing Permit managing orgs.
-- If found then write audit record. If not, then continue
open cMOrg;
fetch cMOrg into vDummy;
if cMOrg%found then
close cMOrg;
ii_range_permits_audit.Write_Grz_Pmt_Contact_Audits (
ii_range_permits_audit.vPRAudit(vIndex).old_cn,
ii_range_permits_audit.vPRAudit(vIndex).old_id,
ii_range_permits_audit.vPRAudit(vIndex).new_id,
'ACCINST_CONT_FK' );
else
close cMOrg;
end if;
--
end loop;
--
-- Re-set counter so that next update statement in this session works
correctly.
ii_range_permits_audit.vNumEntries := 0;
--
End;
```

```

DECLARE
BEGIN
DECLARE
cursor lu_cur is
select root_cn
from land_units
where cn = :new.parent_cn;
l_root_cn varchar2(34);
BEGIN
IF INSERTING THEN
IF not ii_chk_role_assign('INFRA_SVC_MGR') and
ii_cent_mgd_lu_obj(:new.obj_name) then
raise_application_error(-20304,'Insufficient privileges to ||
'insert ||initcap(replace(:new.obj_name,'_',' '))||' Land Units');
END IF;
--
-- Added by RHA for v4.7
--

```

Oracle Designer Report : cktgrr

#### Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

LAND\_UNITS (table)

```

IF :new.cn IS NULL THEN
:new.cn := db_instance.generate_global_id('LU_CN_SEQ');
END IF;
db_instance.insert_audit_columns
(:new.created_by, :new.created_in_instance, :new.created_date);
--
--
IF :new.parent_cn IS NULL THEN
:new.root_cn := :new.cn;
ELSE
IF :new.root_cn IS NULL THEN
open lu_cur;
fetch lu_cur into l_root_cn;
close lu_cur;
:new.root_cn := l_root_cn;
END IF;
END IF;
ELSE

```

```

IF not ii_chk_role_assign('INFRA_SVC_MGR') and
ii_cent_mgd_lu_obj(:new.obj_name) and
:new.obj_name != :old.obj_name THEN
raise_application_error(-20303,'Insufficient privileges to ||
'change a Land Unit from ||
initcap(replace(:old.obj_name,'_' ' '))||' to ||
initcap(replace(:new.obj_name,'_' ' '));
END IF;
IF not ii_chk_role_assign('INFRA_SVC_MGR') and
ii_cent_mgd_lu_obj(:old.obj_name) then
IF :old.obj_name = 'WILDERNESS' then
IF (:old.obj_tech != :new.obj_tech) or
(:old.obj_class != :new.obj_class) or
(:old.obj_name != :new.obj_name) or
(nvl(:old.id,'@') != nvl(:new.id,'@')) or
(nvl(:old.name,'@') != nvl(:new.name,'@')) then
raise_application_error(-20302,'Insufficient privileges to ||
'update || initcap(replace(:old.obj_name,'_' ' '))||
' Land Unit columns');
END IF;
ELSE
raise_application_error(-20301,'Insufficient privileges to ||
'update || initcap(replace(:old.obj_name,'_' ' '))||
' Land Units');
END IF;
END IF;
db_instance.update_audit_columns
(:new.modified_by, :new.modified_in_instance, :new.modified_date);
--
-- Added by RAdams on 6/24/2002 for v5.0:
--
-- Record old/new ID and CN in PL/SQL table for purposes of
-- writing row to range permits audit table (ii_range_pmt_li_audit)
when the updated

```

Oracle Designer Report : cktrgger



## Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

LAND\_UNITS (table)

```

-- Land Unit ID is an RMU, State, Proclaimed Unit, or County
referenced in a
-- grazing permit. This action takes place in the statement-level
-- after-update/delete trigger that first queries the range permit
line items table
-- to check if this land unit is an RMU, State, Proclaimed Unit, or
County referenced in
-- a grazing permit. If this is the case, a re-query on the land_units
table is necessary
-- to get the remaining RMU, State, Proclaimed Unit, and County IDs to
be inserted
-- into the audit table. To avoid the "mutating table" error, the
re-query of the land_units
-- table needed to be moved to a statement-level after update/delete
trigger, passing "old" and "new"
-- values in the package PL/SQL table referenced in the code below.
--
if nvl(:new.id,'@') != nvl(:old.id,'@') then
ii_range_permits_audit.vNumEntries :=
ii_range_permits_audit.vNumEntries + 1;
ii_range_permits_audit.vPRAudit(ii_range_permits_audit.vNumEntries).ne
w_id := :new.id;
ii_range_permits_audit.vPRAudit(ii_range_permits_audit.vNumEntries).ol
d_id := :old.id;
ii_range_permits_audit.vPRAudit(ii_range_permits_audit.vNumEntries).ne
w_cn := :new.cn;
ii_range_permits_audit.vPRAudit(ii_range_permits_audit.vNumEntries).ol
d_cn := :old.cn;
end if;
--
-- Need to add root_cn update logic. (RHA 3/4/98)
END IF;
END;
END;
```

Oracle Designer Report : cktrgger



```

-- Check Avg. Elevation Uom
IF (:new.AVG_ELEVATION_UOM IS NOT NULL
AND NOT chk_iicode('ELEVATION_UOM', :new.AVG_ELEVATION_UOM))
THEN
raise_application_error(-20302,
'Invalid value "||:new.AVG_ELEVATION_UOM||" for Elevation Unit');
END IF;
-- Check Predominant Landform
IF (:new.PREDOMINANT_LANDFORM IS NOT NULL
AND NOT chk_iicode('LU_LANDFORM', :new.PREDOMINANT_LANDFORM))
THEN
raise_application_error(-20303,
'Invalid value "||:new.PREDOMINANT_LANDFORM||" for Predominant ||
'Landform');
END IF;
-- Check Predominant Surface Substrate
IF (:new.PREDOMNT_SFC_SUBSTRATE IS NOT NULL
AND NOT chk_iicode('LU_SURFACE_SUBSTRATE',
:new.PREDOMNT_SFC_SUBSTRATE))
THEN
raise_application_error(-20304,
'Invalid value "||:new.PREDOMNT_SFC_SUBSTRATE||" for Predominant ||
'Surface Substrate');
END IF;
-- Check LU Subtype
IF (:new.LU_SUBTYPE IS NOT NULL
AND NOT chk_iicode(:new.OBJ_NAME||'_SUBTYPE',upper( :new.LU_SUBTYPE)))
THEN
raise_application_error(-20305,
Oracle Designer Report : cktrgger

```

## Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

LAND\_UNITS (table)

```

'Invalid value "||:new.LU_SUBTYPE||" for ||
initcap(replace(:new.OBJ_NAME,'_',' '))||
' Sub-Type');
END IF;

```



```

-- Check Avg. Elevation Uom
IF (:new.AVG_ELEVATION_UOM IS NOT NULL
AND NOT chk_iicode('ELEVATION_UOM', :new.AVG_ELEVATION_UOM))
THEN
raise_application_error(-20302,
'Invalid value "||:new.AVG_ELEVATION_UOM||" for Elevation Unit');
END IF;
-- Check Predominant Landform
IF (:new.PREDOMINANT_LANDFORM IS NOT NULL
AND NOT chk_iicode('LU_LANDFORM', :new.PREDOMINANT_LANDFORM))
THEN
raise_application_error(-20303,
'Invalid value "||:new.PREDOMINANT_LANDFORM||" for Predominant ||
'Landform');
END IF;
-- Check Predominant Surface Substrate
IF (:new.PREDOMNT_SFC_SUBSTRATE IS NOT NULL
AND NOT chk_iicode('LU_SURFACE_SUBSTRATE',
:new.PREDOMNT_SFC_SUBSTRATE))
THEN
raise_application_error(-20304,
'Invalid value "||:new.PREDOMNT_SFC_SUBSTRATE||" for Predominant ||
'Surface Substrate');
END IF;
-- Check LU Subtype
IF (:new.LU_SUBTYPE IS NOT NULL
AND NOT chk_iicode(:new.OBJ_NAME||'_SUBTYPE',upper( :new.LU_SUBTYPE)))
THEN

```

Oracle Designer Report : cktrgger

Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

LAND\_UNITS (table)

```

raise_application_error(-20305,
'Invalid value "||:new.LU_SUBTYPE||"' for '||
initcap(replace(:new.OBJ_NAME, '_', ' '))||
' Sub-Type');
END IF;
END;
DECLARE
BEGIN
-- Check Area UoM
IF (:new.AREA_UOM IS NOT NULL
AND NOT chk_icode('AREA_UOM', :new.AREA_UOM))
THEN
raise_application_error(-20301,
'Invalid value "||:new.AREA_UOM||"' for Area Unit');
END IF;
-- Check Avg. Elevation Uom
IF (:new.AVG_ELEVATION_UOM IS NOT NULL
AND NOT chk_icode('ELEVATION_UOM', :new.AVG_ELEVATION_UOM))
THEN
raise_application_error(-20302,
'Invalid value "||:new.AVG_ELEVATION_UOM||"' for Elevation Unit');
END IF;
-- Check Predominant Landform
IF (:new.PREDOMINANT_LANDFORM IS NOT NULL
AND NOT chk_icode('LU_LANDFORM', :new.PREDOMINANT_LANDFORM))
THEN
raise_application_error(-20303,
'Invalid value "||:new.PREDOMINANT_LANDFORM||"' for Predominant '||
'Landform');
END IF;
-- Check Predominant Surface Substrate
IF (:new.PREDOMNT_SFC_SUBSTRATE IS NOT NULL
AND NOT chk_icode('LU_SURFACE_SUBSTRATE',
:new.PREDOMNT_SFC_SUBSTRATE))
THEN
raise_application_error(-20304,
'Invalid value "||:new.PREDOMNT_SFC_SUBSTRATE||"' for Predominant '||
'Surface Substrate');
END IF;
-- Check LU Subtype
IF (:new.LU_SUBTYPE IS NOT NULL
AND NOT chk_icode(:new.OBJ_NAME||'_SUBTYPE',upper( :new.LU_SUBTYPE)))

```

```

THEN
raise_application_error(-20305,
'Invalid value "||:new.LU_SUBTYPE||"' for '||
initcap(replace(:new.OBJ_NAME,'_',' '))||
' Sub-Type');
END IF;
END;
DECLARE
BEGIN
-- Check Area UoM
IF (:new.AREA_UOM IS NOT NULL
AND NOT chk_iicode('AREA_UOM', :new.AREA_UOM))
Oracle Designer Report : cktrgger

```

## Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

LAND\_UNITS (table)

```

THEN
raise_application_error(-20301,
'Invalid value "||:new.AREA_UOM||"' for Area Unit');
END IF;
-- Check Avg. Elevation Uom
IF (:new.AVG_ELEVATION_UOM IS NOT NULL
AND NOT chk_iicode('ELEVATION_UOM', :new.AVG_ELEVATION_UOM))
THEN
raise_application_error(-20302,
'Invalid value "||:new.AVG_ELEVATION_UOM||"' for Elevation Unit');
END IF;
-- Check Predominant Landform
IF (:new.PREDOMINANT_LANDFORM IS NOT NULL
AND NOT chk_iicode('LU_LANDFORM', :new.PREDOMINANT_LANDFORM))
THEN
raise_application_error(-20303,
'Invalid value "||:new.PREDOMINANT_LANDFORM||"' for Predominant '||
'Landform');
END IF;
-- Check Predominant Surface Substrate
IF (:new.PREDOMNT_SFC_SUBSTRATE IS NOT NULL
AND NOT chk_iicode('LU_SURFACE_SUBSTRATE',
:new.PREDOMNT_SFC_SUBSTRATE))

```

```

THEN
raise_application_error(-20304,
'Invalid value "'||:new.PREDOMNT_SFC_SUBSTRATE||'" for Predominant ||
'Surface Substrate');
END IF;
-- Check LU Subtype
IF (:new.LU_SUBTYPE IS NOT NULL
AND NOT chk_iicode(:new.OBJ_NAME||'_SUBTYPE',upper( :new.LU_SUBTYPE)))

THEN
raise_application_error(-20305,
'Invalid value "'||:new.LU_SUBTYPE||'" for ||
initcap(replace(:new.OBJ_NAME,'_',' '))||
' Sub-Type');
END IF;
END;
DECLARE
vModuleName CONSTANT VARCHAR2(30) := 'II_LU_RPA_AR_TRG'; -- Added on
01/07/2003
BEGIN
-- DELETING
IF DELETING THEN
-- DELETE from ii_rpa_assets_v_tab
DELETE FROM ii_rpa_assets_v_tab
WHERE lu_cn = :old.cn;

-- INSERTING
ELSIF INSERTING THEN
-- Insert into table ii_rpa_assets_v_tab
INSERT INTO ii_rpa_assets_v_tab (
FEA_CN
,LU_CN

```

Oracle Designer Report : cktrgger

#### Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name



```

:
      LAND_UNITS (table)
,PLD_CN
,OBJECT_NAME
,OBJECT_CLASS
,ID
,NAME
,SUB_TYPE
,OWNERSHIP
,ADMIN_ORG
,DEVELOPMENT_STATUS
,CREATED_DATE
,CREATED_BY
,SUBLEDGER_FLAG
,subledger_cn
,agency_code
) VALUES (
NULL
, :new.cn
, NULL
, :new.OBJ_NAME
, :new.OBJ_CLASS
, :new.ID
, :new.NAME
, NULL
, NULL -- :new.OWNERSHIP
, :new.ADMIN_ORG
, NULL -- :new.DEVELOPMENT_STATUS
, :new.CREATED_DATE
, :new.CREATED_BY
, 'N'
, null
, :NEW.agency_code -- Added on 10/28/2003 by C.Ma
);

-- UPDATING
ELSIF UPDATING THEN
-- Update table ii_rpa_assets_v_tab
BEGIN
UPDATE ii_rpa_assets_v_tab tab
SET (
OBJECT_NAME
,OBJECT_CLASS

```

```

,ID
,NAME
,SUB_TYPE
--,OWNERSHIP
,ADMIN_ORG
,DEVELOPMENT_STATUS
,CREATED_DATE
,CREATED_BY
--,SUBLEDGER_FLAG
) = (SELECT
:new.OBJ_NAME OBJECT_NAME
,:new.OBJ_CLASS OBJECT_CLASS
,:new.ID
,:new.NAME
,:new.LU_SUBTYPE
--, :new.OWNERSHIP
,:new.ADMIN_ORG
, NULL--, :new.DEVELOPMENT_STATUS
Oracle Designer Report : cktgrger

```

## Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

LAND\_UNITS (table)

,:new.CREATED\_DATE

,:new.CREATED\_BY

--, 'N'

FROM DUAL)

WHERE tab.lu\_cn = :new.cn;

EXCEPTION WHEN OTHERS THEN

ii\_rpa\_errors.save\_log(vModuleName, 'II\_RPA\_ASSETS\_V\_TAB', :new.cn,

null, null, SQLERRM);

END;

-- Update table ii\_rpa\_subledgers

-- The corresponding Admin\_org in II\_RPA\_SUBLEDGERS should be changed  
to-- There is a log table via triggers to keep track of all significant  
changes

-- occurred to subledger-ed assests.

```

BEGIN
UPDATE ii_rpa_subledgers sub
SET admin_org = :new.admin_org
WHERE sub.lu_cn = :new.cn
AND ii_sql_func.super_equal(admin_org, :new.admin_org) <> 'TRUE';

EXCEPTION WHEN OTHERS THEN
--ii_rpa_errors.log('LAND_UNITS', :new.cn, SQLERRM);
-- Modified on 01/07/2004
ii_rpa_errors.save_log(vModuleName, 'LAND_UNITS', :new.cn, null, null,
SQLERRM);
RAISE;
END;

-----
-- Audit
-----

DECLARE
vOldLu land_units%rowtype;
vNewLu land_units%rowtype;
BEGIN
vOldLu.CN := :old.CN;
vOldLu.OBJ_TECH := :old.OBJ_TECH;
vOldLu.OBJ_NAME := :old.OBJ_NAME;
vOldLu.OBJ_CLASS := :old.OBJ_CLASS;
vOldLu.ID := :old.ID;
vOldLu.NAME := :old.NAME;
vOldLu.LU_SUBTYPE := :old.LU_SUBTYPE;
vOldLu.ADMIN_ORG := :old.ADMIN_ORG;
vOldLu.MASTER_SITE := :old.MASTER_SITE;
vOldLu.AVG_ELEVATION := :old.AVG_ELEVATION;
vOldLu.PREDOMINANT_LANDFORM := :old.PREDOMINANT_LANDFORM;
vOldLu.STATUS := :old.STATUS;
vOldLu.AREA_UOM := :old.AREA_UOM;
vOldLu.TRANS_ID := :old.TRANS_ID;
vOldLu.AREA := :old.AREA;
vOldLu.PREDOMNT_SFC_SUBSTRATE := :old.PREDOMNT_SFC_SUBSTRATE;
vOldLu.AVG_ELEVATION_UOM := :old.AVG_ELEVATION_UOM;
vOldLu.REMARKS := :old.REMARKS;
vOldLu.ORG_JUR_CONTACT_CN := :old.ORG_JUR_CONTACT_CN;
vOldLu.ROOT_CN := :old.ROOT_CN;

```

Oracle Designer Report : cktrgger

## Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

LAND\_UNITS (table)

```

vOldLu.PARENT_CN := :old.PARENT_CN;
vOldLu.CREATED_BY := :old.CREATED_BY;
vOldLu.CREATED_DATE := :old.CREATED_DATE;
vOldLu.CREATED_IN_INSTANCE := :old.CREATED_IN_INSTANCE;
vOldLu.MODIFIED_BY := :old.MODIFIED_BY;
vOldLu.MODIFIED_DATE := :old.MODIFIED_DATE;
vOldLu.MODIFIED_IN_INSTANCE := :old.MODIFIED_IN_INSTANCE;
vOldLu.OWNERSHIP := :old.OWNERSHIP;
--vOldLu.GSA_INSTALLATION_CN := :old.GSA_INSTALLATION_CN;-- commented
out on 10/28/2003 by C.Ma
vNewLu.CN := :new.CN;
vNewLu.OBJ_TECH := :new.OBJ_TECH;
vNewLu.OBJ_NAME := :new.OBJ_NAME;
vNewLu.OBJ_CLASS := :new.OBJ_CLASS;
vNewLu.ID := :new.ID;
vNewLu.NAME := :new.NAME;
vNewLu.LU_SUBTYPE := :new.LU_SUBTYPE;
vNewLu.ADMIN_ORG := :new.ADMIN_ORG;
vNewLu.MASTER_SITE := :new.MASTER_SITE;
vNewLu.AVG_ELEVATION := :new.AVG_ELEVATION;
vNewLu.PREDOMINANT_LANDFORM := :new.PREDOMINANT_LANDFORM;
vNewLu.STATUS := :new.STATUS;
vNewLu.AREA_UOM := :new.AREA_UOM;
vNewLu.TRANS_ID := :new.TRANS_ID;
vNewLu.AREA := :new.AREA;
vNewLu.PREDOMNT_SFC_SUBSTRATE := :new.PREDOMNT_SFC_SUBSTRATE;
vNewLu.AVG_ELEVATION_UOM := :new.AVG_ELEVATION_UOM;
vNewLu.REMARKS := :new.REMARKS;
vNewLu.ORG_JUR_CONTACT_CN := :new.ORG_JUR_CONTACT_CN;
vNewLu.ROOT_CN := :new.ROOT_CN;
vNewLu.PARENT_CN := :new.PARENT_CN;
vNewLu.CREATED_BY := :new.CREATED_BY;
vNewLu.CREATED_DATE := :new.CREATED_DATE;
vNewLu.CREATED_IN_INSTANCE := :new.CREATED_IN_INSTANCE;
vNewLu.MODIFIED_BY := :new.MODIFIED_BY;
vNewLu.MODIFIED_DATE := :new.MODIFIED_DATE;
vNewLu.MODIFIED_IN_INSTANCE := :new.MODIFIED_IN_INSTANCE;
vNewLu.OWNERSHIP := :new.OWNERSHIP;
--vNewLu.GSA_INSTALLATION_CN := :new.GSA_INSTALLATION_CN;-- commented

```

Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

II\_CONTACTS (table)

Trigger Name

:

CONTACTS\_TRG

Create?

Yes

PL/SQL Module

Name :

CONTACTS\_TRG

Enabled?

Yes

PL/SQL File :

Purpose :

.

Trigger Time:

BEFORE

Level:

ROW

Insert?

Yes

Delete?

No

New Alias

NEW

Update?

Yes

Old Alias :

OLD

Trigger Logic

DECLARE

vModuleName CONSTANT VARCHAR2(30) := 'II\_LU\_RPA\_AR\_TRG'; -- Added on  
01/07/2003

BEGIN

-- DELETING

IF DELETING THEN

-- DELETE from ii\_rpa\_assets\_v\_tab

DELETE FROM ii\_rpa\_assets\_v\_tab

WHERE lu\_cn = :old.cn;

```
-- INSERTING
ELSIF INSERTING THEN
-- Insert into table ii_rpa_assets_v_tab
INSERT INTO ii_rpa_assets_v_tab (
FEA_CN
,LU_CN
,PLD_CN
,OBJECT_NAME
,OBJECT_CLASS
,ID
,NAME
,SUB_TYPE
,OWNERSHIP
,ADMIN_ORG
,DEVELOPMENT_STATUS
,CREATED_DATE
,CREATED_BY
,SUBLEDGER_FLAG
,subledger_cn
,agency_code
) VALUES (
NULL
, :new.cn
, NULL
, :new.OBJ_NAME
, :new.OBJ_CLASS
, :new.ID
, :new.NAME
, NULL
, NULL -- :new.OWNERSHIP
, :new.ADMIN_ORG
, NULL -- :new.DEVELOPMENT_STATUS
Oracle Designer Report : cktrgger
```

## Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

II\_CONTACTS (table)

```

,:new.CREATED_DATE
,:new.CREATED_BY
,'N'
,null
,:NEW.agency_code -- Added on 10/28/2003 by C.Ma
);

```

```

-- UPDATING
ELSIF UPDATING THEN
-- Update table ii_rpa_assets_v_tab
BEGIN
UPDATE ii_rpa_assets_v_tab tab
SET (
OBJECT_NAME
,OBJECT_CLASS
,ID
,NAME
,SUB_TYPE
--,OWNERSHIP
,ADMIN_ORG
,DEVELOPMENT_STATUS
,CREATED_DATE
,CREATED_BY
--,SUBLEDGER_FLAG
) = (SELECT
:new.OBJ_NAME OBJECT_NAME
,:new.OBJ_CLASS OBJECT_CLASS
,:new.ID
,:new.NAME
,:new.LU_SUBTYPE
--, :new.OWNERSHIP
,:new.ADMIN_ORG
,NULL--, :new.DEVELOPMENT_STATUS
,:new.CREATED_DATE
,:new.CREATED_BY
--, 'N'
FROM DUAL)
WHERE tab.lu_cn = :new.cn;

```

```

-- Update table ii_rpa_subledgers
-- The corresponding Admin_org in II_RPA_SUBLEDGERS should be changed
too
-- There is a log table via triggers to keep track of all significant
changes
-- occurred to subledger-ed assests.
BEGIN
UPDATE ii_rpa_subledgers sub
SET admin_org = :new.admin_org
WHERE sub.lu_cn = :new.cn
AND ii_sql_func.super_equal(admin_org, :new.admin_org) <> 'TRUE';

EXCEPTION WHEN OTHERS THEN
--ii_rpa_errors.log('LAND_UNITS', :new.cn, SQLERRM);
                Oracle Designer Report : cktrgger

```

#### Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

II\_CONTACTS (table)

-- Modified on 01/07/2004

```

ii_rpa_errors.save_log(vModuleName, 'LAND_UNITS', :new.cn, null, null,
SQLERRM);

```

RAISE;

END;

-----

-- Audit

-----

DECLARE

vOldLu land\_units%rowtype;

vNewLu land\_units%rowtype;

BEGIN

vOldLu.CN := :old.CN;

vOldLu.OBJ\_TECH := :old.OBJ\_TECH;

vOldLu.OBJ\_NAME := :old.OBJ\_NAME;

vOldLu.OBJ\_CLASS := :old.OBJ\_CLASS;

vOldLu.ID := :old.ID;

vOldLu.NAME := :old.NAME;



```

vOldLu.LU_SUBTYPE := :old.LU_SUBTYPE;
vOldLu.ADMIN_ORG := :old.ADMIN_ORG;
vOldLu.MASTER_SITE := :old.MASTER_SITE;
vOldLu.AVG_ELEVATION := :old.AVG_ELEVATION;
vOldLu.PREDOMINANT_LANDFORM := :old.PREDOMINANT_LANDFORM;
vOldLu.STATUS := :old.STATUS;
vOldLu.AREA_UOM := :old.AREA_UOM;
vOldLu.TRANS_ID := :old.TRANS_ID;
vOldLu.AREA := :old.AREA;
vOldLu.PREDOMNT_SFC_SUBSTRATE := :old.PREDOMNT_SFC_SUBSTRATE;
vOldLu.AVG_ELEVATION_UOM := :old.AVG_ELEVATION_UOM;
vOldLu.REMARKS := :old.REMARKS;
vOldLu.ORG_JUR_CONTACT_CN := :old.ORG_JUR_CONTACT_CN;
vOldLu.ROOT_CN := :old.ROOT_CN;
vOldLu.PARENT_CN := :old.PARENT_CN;
vOldLu.CREATED_BY := :old.CREATED_BY;
vOldLu.CREATED_DATE := :old.CREATED_DATE;
vOldLu.CREATED_IN_INSTANCE := :old.CREATED_IN_INSTANCE;
vOldLu.MODIFIED_BY := :old.MODIFIED_BY;
vOldLu.MODIFIED_DATE := :old.MODIFIED_DATE;
vOldLu.MODIFIED_IN_INSTANCE := :old.MODIFIED_IN_INSTANCE;
vOldLu.OWNERSHIP := :old.OWNERSHIP;
--vOldLu.GSA_INSTALLATION_CN := :old.GSA_INSTALLATION_CN;-- commented
out on 10/28/2003 by C.Ma

```

```

vNewLu.CN := :new.CN;
vNewLu.OBJ_TECH := :new.OBJ_TECH;
vNewLu.OBJ_NAME := :new.OBJ_NAME;
vNewLu.OBJ_CLASS := :new.OBJ_CLASS;
vNewLu.ID := :new.ID;
vNewLu.NAME := :new.NAME;
vNewLu.LU_SUBTYPE := :new.LU_SUBTYPE;
vNewLu.ADMIN_ORG := :new.ADMIN_ORG;
vNewLu.MASTER_SITE := :new.MASTER_SITE;
vNewLu.AVG_ELEVATION := :new.AVG_ELEVATION;
vNewLu.PREDOMINANT_LANDFORM := :new.PREDOMINANT_LANDFORM;
vNewLu.STATUS := :new.STATUS;
vNewLu.AREA_UOM := :new.AREA_UOM;
vNewLu.TRANS_ID := :new.TRANS_ID;
vNewLu.AREA := :new.AREA;

```

Oracle Designer Report : cktrgger

## Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

II\_CONTACTS (table)

```

vNewLu.PREDOMNT_SFC_SUBSTRATE := :new.PREDOMNT_SFC_SUBSTRATE;
vNewLu.AVG_ELEVATION_UOM := :new.AVG_ELEVATION_UOM;
vNewLu.REMARKS := :new.REMARKS;
vNewLu.ORG_JUR_CONTACT_CN := :new.ORG_JUR_CONTACT_CN;
vNewLu.ROOT_CN := :new.ROOT_CN;
vNewLu.PARENT_CN := :new.PARENT_CN;
vNewLu.CREATED_BY := :new.CREATED_BY;
vNewLu.CREATED_DATE := :new.CREATED_DATE;
vNewLu.CREATED_IN_INSTANCE := :new.CREATED_IN_INSTANCE;
vNewLu.MODIFIED_BY := :new.MODIFIED_BY;
vNewLu.MODIFIED_DATE := :new.MODIFIED_DATE;
vNewLu.MODIFIED_IN_INSTANCE := :new.MODIFIED_IN_INSTANCE;
vNewLu.OWNERSHIP := :new.OWNERSHIP;
--vNewLu.GSA_INSTALLATION_CN := :new.GSA_INSTALLATION_CN;-- commented
out on 10/28/2003 by C.Ma

```

```

ii_rpa_auditing.audit_lu('U', :new.modified_by, :new.modified_date,
vOldLu, vNewLu);
END;

```

END IF;

END;

DECLARE

BEGIN

DECLARE

```

vContCN ii_contacts.cn%TYPE := null;
vAgency ii_contacts.agency_code%TYPE := null;

```

cursor cFSOrg is

select 'x'

from ii\_organizations

where org\_type = 'FS ORGANIZATION' and

cont\_cn = vContCN;

vDummy char(1);

```

cursor user_agency is
select agency_code
from ii_rpa_fiscal_users
where user_name = USER;
BEGIN
IF INSERTING THEN
IF :new.cn IS NULL THEN
:new.cn := db_instance.generate_global_id('CONT_CN_SEQ');
END IF;
db_instance.insert_audit_columns
(:new.created_by, :new.created_in_instance, :new.created_date);

IF :new.agency_code IS NULL THEN
open user_agency;
fetch user_agency into vAgency;
close user_agency;
:new.agency_code := vAgency;
END IF;

ELSE -- Updating

```

Oracle Designer Report : cktrgger

#### Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

II\_CONTACTS (table)

```

-- Check role assignment for FS Org updates.
if not ii_chk_role_assign('INFRA_SVC_MGR') then
-- Make sure user isn't trying to update an FS Org.
vContCN := :old.cn;
open cFSOrg;
fetch cFSOrg into vDummy;
if not cFSOrg%notfound then
close cFSOrg;
raise_application_error(-20300,'Insufficient privileges to ||
'update an FS Organization');
else
close cFSOrg;
end if;
end if;

```

```

--
-- Set audit columns
db_instance.update_audit_columns
(:new.modified_by, :new.modified_in_instance, :new.modified_date);
--
-- Added by RAdams on 6/24/2002 for v5.0:
--
-- Record old/new ID and CN in PL/SQL table for purposes of
-- writing row to range permits audit table (ii_range_permit_audits)
when the updated
-- Contact ID is a permittee or managing organization for a
-- grazing permit. This action takes place in the statement- level
-- after-update/delete trigger that first queries the range permits
view
-- to check if this contact is a permittee or managing organization
for
-- a grazing permit. If this is the case, a re-query on the
ii_contacts table is necessary
-- to get the remaining Permittee or Managing Org ID to be inserted
-- into the audit table. To avoid the "mutating table" error, the
re-query of the ii_contacts
-- table needed to be moved to a statement-level after update/delete
trigger, passing "old" and "new"
-- values in the package PL/SQL table referenced in the code below.
--
if nvl(:new.id,'@') != nvl(:old.id,'@') then
ii_range_permits_audit.vNumEntries :=
ii_range_permits_audit.vNumEntries + 1;
ii_range_permits_audit.vPRAudit(ii_range_permits_audit.vNumEntries).ne
w_id := :new.id;
ii_range_permits_audit.vPRAudit(ii_range_permits_audit.vNumEntries).ol
d_id := :old.id;
ii_range_permits_audit.vPRAudit(ii_range_permits_audit.vNumEntries).ne
w_cn := :new.cn;
ii_range_permits_audit.vPRAudit(ii_range_permits_audit.vNumEntries).ol
d_cn := :old.cn;

```

Oracle Designer Report : cktrgger

Database Trigger Definition

```
Container:
  II_RPA
  Version :
Table/View Name
:
  II_CONTACTS (table)
end if;
--
END IF;
END;
END;
```

Oracle Designer Report : cktrgger

Database Trigger Definition

```
Container:
  II_RPA
  Version :
Table/View Name
:
  II_FEATURES (table)
Trigger Name
:
  II_RBIU_FEATURES
Create?
  Yes
PL/SQL Module
Name :
  II_RBIU_FEATURES_1
Enabled?
  Yes
PL/SQL File :
Purpose :
.
Trigger Time:
  BEFORE
Level:
  ROW
Insert?
  Yes
Delete?
  No
New Alias
```

```

Old Alias :
Trigger Logic
DECLARE
BEGIN
DECLARE
vContCN ii_contacts.cn%TYPE := null;
vAgency ii_contacts.agency_code%TYPE := null;

cursor cFSOrg is
select 'x'
from ii_organizations
where org_type = 'FS ORGANIZATION' and
cont_cn = vContCN;
vDummy char(1);

cursor user_agency is
select agency_code
from ii_rpa_fiscal_users
where user_name = USER;
BEGIN
IF INSERTING THEN
IF :new.cn IS NULL THEN
:new.cn := db_instance.generate_global_id('CONT_CN_SEQ');
END IF;
db_instance.insert_audit_columns
(:new.created_by, :new.created_in_instance, :new.created_date);

IF :new.agency_code IS NULL THEN
open user_agency;
fetch user_agency into vAgency;
close user_agency;
:new.agency_code := vAgency;
END IF;

ELSE -- Updating
-- Check role assignment for FS Org updates.
if not ii_chk_role_assign('INFRA_SVC_MGR') then
-- Make sure user isn't trying to update an FS Org.
vContCN := :old.cn;
open cFSOrg;
fetch cFSOrg into vDummy;
if not cFSOrg%notfound then
close cFSOrg;

```

Oracle Designer Report : cktrgger

## Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

II\_FEATURES (table)

```
raise_application_error(-20300,'Insufficient privileges to '||
'update an FS Organization');
```

```
else
```

```
close cFSOrg;
```

```
end if;
```

```
end if;
```

```
--
```

```
-- Set audit columns
```

```
db_instance.update_audit_columns
```

```
(:new.modified_by, :new.modified_in_instance, :new.modified_date);
```

```
--
```

```
-- Added by RAdams on 6/24/2002 for v5.0:
```

```
--
```

```
-- Record old/new ID and CN in PL/SQL table for purposes of
-- writing row to range permits audit table (ii_range_permit_audits)
when the updated
```

```
-- Contact ID is a permittee or managing organization for a
-- grazing permit. This action takes place in the statement- level
-- after-update/delete trigger that first queries the range permits
view
```

```
-- to check if this contact is a permittee or managing organization
for
```

```
-- a grazing permit. If this is the case, a re-query on the
ii_contacts table is necessary
```

```
-- to get the remaining Permittee or Managing Org ID to be inserted
-- into the audit table. To avoid the "mutating table" error, the
re-query of the ii_contacts
```

```
-- table needed to be moved to a statement-level after update/delete
trigger, passing "old" and "new"
```

```
-- values in the package PL/SQL table referenced in the code below.
```

```
--
```

```
if nvl(:new.id,'@') != nvl(:old.id,'@') then
```

```
ii_range_permits_audit.vNumEntries :=
```

```
ii_range_permits_audit.vNumEntries + 1;
```

```
ii_range_permits_audit.vPRAudit(ii_range_permits_audit.vNumEntries).ne
```

```
w_id := :new.id;
```

```
ii_range_permits_audit.vPRAudit(ii_range_permits_audit.vNumEntries).ol
```

```
d_id := :old.id;
```

```

ii_range_permits_audit.vPRAudit(ii_range_permits_audit.vNumEntries).ne
w_cn := :new.cn;
ii_range_permits_audit.vPRAudit(ii_range_permits_audit.vNumEntries).ol
d_cn := :old.cn;
end if;
--
END IF;
END;
END;

```

DECLARE

Oracle Designer Report : cktrgger

#### Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

II\_FEATURES (table)

BEGIN

DECLARE

vContCN ii\_contacts.cn%TYPE := null;

vAgency ii\_contacts.agency\_code%TYPE := null;

cursor cFSOrg is

select 'x'

from ii\_organizations

where org\_type = 'FS ORGANIZATION' and

cont\_cn = vContCN;

vDummy char(1);

cursor user\_agency is

select agency\_code

from ii\_rpa\_fiscal\_users

where user\_name = USER;

BEGIN

IF INSERTING THEN

IF :new.cn IS NULL THEN

:new.cn := db\_instance.generate\_global\_id('CONT\_CN\_SEQ');

END IF;

db\_instance.insert\_audit\_columns

(:new.created\_by, :new.created\_in\_instance, :new.created\_date);



```

IF :new.agency_code IS NULL THEN
open user_agency;
fetch user_agency into vAgency;
close user_agency;
:new.agency_code := vAgency;
END IF;

ELSE -- Updating
-- Check role assignment for FS Org updates.
if not ii_chk_role_assign('INFRA_SVC_MGR') then
-- Make sure user isn't trying to update an FS Org.
vContCN := :old.cn;
open cFSOrg;
fetch cFSOrg into vDummy;
if not cFSOrg%notfound then
close cFSOrg;
raise_application_error(-20300,'Insufficient privileges to ||
'update an FS Organization');
else
close cFSOrg;
end if;
end if;
--
-- Set audit columns
db_instance.update_audit_columns
(:new.modified_by, :new.modified_in_instance, :new.modified_date);
--
-- Added by RAdams on 6/24/2002 for v5.0:
--
-- Record old/new ID and CN in PL/SQL table for purposes of
-- writing row to range permits audit table (ii_range_permit_audits)
when the updated

```

Oracle Designer Report : cktrgger

#### Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

II\_FEATURES (table)

```

-- Contact ID is a permittee or managing organization for a
-- grazing permit. This action takes place in the statement- level
-- after-update/delete trigger that first queries the range permits

```

```

-- to check if this contact is a permittee or managing organization
for
-- a grazing permit. If this is the case, a re-query on the
ii_contacts table is necessary
-- to get the remaining Permittee or Managing Org ID to be inserted
-- into the audit table. To avoid the "mutating table" error, the
re-query of the ii_contacts
-- table needed to be moved to a statement-level after update/delete
trigger, passing "old" and "new"
-- values in the package PL/SQL table referenced in the code below.
--
if nvl(:new.id,'@') != nvl(:old.id,'@') then
ii_range_permits_audit.vNumEntries :=
ii_range_permits_audit.vNumEntries + 1;
ii_range_permits_audit.vPRAudit(ii_range_permits_audit.vNumEntries).ne
w_id := :new.id;
ii_range_permits_audit.vPRAudit(ii_range_permits_audit.vNumEntries).ol
d_id := :old.id;
ii_range_permits_audit.vPRAudit(ii_range_permits_audit.vNumEntries).ne
w_cn := :new.cn;
ii_range_permits_audit.vPRAudit(ii_range_permits_audit.vNumEntries).ol
d_cn := :old.cn;
end if;
--
END IF;
END;
END;

DECLARE
BEGIN
IF :new.OWNERSHIP is not null and
:new.OWNERSHIP <> nvl(:old.OWNERSHIP,'@') and
NOT Chk_IiCode( 'FEAT_OWNERSHIP', :new.OWNERSHIP )
THEN
Raise_Application_Error( -20301,
'Invalid value "'||:new.OWNERSHIP||'" '||
'for II_Features.OWNERSHIP.' );
END IF;
--
IF :new.MAINTENANCE_LEVEL is not null and
:new.MAINTENANCE_LEVEL <> nvl(:old.MAINTENANCE_LEVEL,'@') and
NOT Chk_IiCode( 'FEAT_MAINTENANCE_LEVEL', :new.MAINTENANCE_LEVEL )
THEN
Raise_Application_Error( -20302,
'Invalid value "'||:new.MAINTENANCE_LEVEL||'" '||
'for II_Features.MAINTENANCE_LEVEL.' );

```

## Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

II\_FEATURES (table)

END IF;

--

IF :new.MAINTENANCE\_RESPONSIBILITY is not null and

:new.MAINTENANCE\_RESPONSIBILITY &lt;&gt;

nvl(:old.MAINTENANCE\_RESPONSIBILITY,'@') and

NOT Chk\_IiCode( 'MAINTENANCE\_RESPONSIBILITY',

:new.MAINTENANCE\_RESPONSIBILITY )

THEN

Raise\_Application\_Error( -20303,

'Invalid value ""||:new.MAINTENANCE\_RESPONSIBILITY||"' ||

'for II\_Features.MAINTENANCE\_RESPONSIBILITY.' );

END IF;

--

IF :new.CONDITION\_RATING is not null and

:new.CONDITION\_RATING &lt;&gt; nvl(:old.CONDITION\_RATING,'@') and

NOT Chk\_IiCode( 'FEAT\_CONDITION\_RATING', :new.CONDITION\_RATING )

THEN

Raise\_Application\_Error( -20304,

'Invalid value ""||:new.CONDITION\_RATING||"' ||

'for II\_Features.CONDITION\_RATING.' );

END IF;

--

IF :new.APPRAISAL\_RATING is not null and

:new.APPRAISAL\_RATING &lt;&gt; nvl(:old.APPRAISAL\_RATING,'@') and

NOT Chk\_IiCode( 'FEAT\_APPRAISAL\_RATING', :new.APPRAISAL\_RATING )

THEN

Raise\_Application\_Error( -20305,

'Invalid value ""||:new.APPRAISAL\_RATING||"' ||

'for II\_Features.APPRAISAL\_RATING.' );

END IF;

--

IF :new.DEVELOPMENT\_STATUS is not null and

:new.DEVELOPMENT\_STATUS &lt;&gt; nvl(:old.DEVELOPMENT\_STATUS,'@') and

NOT Chk\_IiCode( 'FEAT\_DEVELOPMENT\_STATUS', :new.DEVELOPMENT\_STATUS

)

```

THEN
Raise_Application_Error( -20306,
'Invalid value "||:new.DEVELOPMENT_STATUS||"' ||
'for II_Features.DEVELOPMENT_STATUS. ');
END IF;
--
IF :new.HISTORICAL_STATUS is not null and
:new.HISTORICAL_STATUS <> nvl(:old.HISTORICAL_STATUS,'@') and
NOT Chk_IiCode( 'FEAT_HISTORICAL_STATUS', :new.HISTORICAL_STATUS )
THEN
Raise_Application_Error( -20307,
'Invalid value "||:new.HISTORICAL_STATUS||"' ||
'for II_Features.HISTORICAL_STATUS. ');
END IF;
--
IF :new.LOCATION is not null and
:new.LOCATION <> nvl(:old.LOCATION,'@') and
NOT Chk_IiCode( 'FEAT_LOCATION', :new.LOCATION )
THEN
Raise_Application_Error( -20308,
'Invalid value "||:new.LOCATION||"' ||
'for II_Features.LOCATION. ');
END IF;

```

Oracle Designer Report : cktrgger

#### Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

II\_FEATURES (table)

```

--
IF :new.LOCATION_UOM is not null and
:new.LOCATION_UOM <> nvl(:old.LOCATION_UOM,'@') and
NOT Chk_IiCode( 'FEAT_UOM', :new.LOCATION_UOM )
THEN
Raise_Application_Error( -20309,
'Invalid value "||:new.LOCATION_UOM||"' ||
'for II_Features.LOCATION_UOM. ');
END IF;
--

```

```

IF :new.ACCESSIBILITY_STATUS is not null and
:new.ACCESSIBILITY_STATUS <> nvl(:old.ACCESSIBILITY_STATUS,@') and
NOT Chk_IiCode( 'ACCESSIBILITY_STATUS', :new.ACCESSIBILITY_STATUS )
THEN
Raise_Application_Error( -203010,
'Invalid value ""||:new.ACCESSIBILITY_STATUS||"" '||
'for II_Features.ACCESSIBILITY_STATUS.' );
END IF;
--
IF :new.ACQUISITION_METHOD is not null and
:new.ACQUISITION_METHOD <> nvl(:old.ACQUISITION_METHOD,@') and
NOT Chk_IiCode( 'ACQUISITION_METHOD', :new.ACQUISITION_METHOD )
THEN
Raise_Application_Error( -203011,
'Invalid value ""||:new.ACQUISITION_METHOD||"" '||
'for II_Features.ACQUISITION_METHOD.' );
END IF;
--
IF :new.OPERATIONAL_STATUS is not null and
:new.OPERATIONAL_STATUS <> nvl(:old.OPERATIONAL_STATUS,@') and
NOT Chk_IiCode( 'OPERATIONAL_STATUS', :new.OPERATIONAL_STATUS )
THEN
Raise_Application_Error( -203012,
'Invalid value ""||:new.OPERATIONAL_STATUS||"" '||
'for II_Features.OPERATIONAL_STATUS.' );
END IF;
--
IF Inserting
THEN
-- Supply CN if it is null.
IF :new.CN is null
THEN
:new.CN := Db_Instance.GENERATE_GLOBAL_ID( 'FEAT_CN_SEQ' );
END IF;
--
-- Set the CREATED... audit columns on row insert if not supplied.
IF :new.CREATED_BY is null or
:new.CREATED_DATE is null or
:new.CREATED_IN_INSTANCE is null
THEN
Db_Instance.INSERT_AUDIT_COLUMNS( :new.CREATED_BY
, :new.CREATED_IN_INSTANCE
, :new.CREATED_DATE );
END IF;

```

---

ELSIF Updating

Oracle Designer Report : cktrgger

Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

II\_FEATURES (table)

THEN

-- Set the MODIFIED... audit columns on row update.

Db\_Instance.UPDATE\_AUDIT\_COLUMNS( :new.MODIFIED\_BY

, :new.MODIFIED\_IN\_INSTANCE

, :new.MODIFIED\_DATE );

---

END IF;

END;

Oracle Designer Report : cktrgger

10 February 2004

Page

44

of

71

Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

II\_FEATURES (table)

Trigger Name

:

II\_FEA\_RPA\_AR\_TRG

Create?

Yes

PL/SQL Module

Name :

II\_FEA\_RPA\_AR\_TRG\_1

Enabled?

Yes

PL/SQL File :

Purpose :

Maintain Ins/Upd/Del after row for Real Property  
Accounting

Trigger Time:

AFTER

Level:

ROW

Insert?

Yes

Delete?

Yes

New Alias

Update?

Yes

Old Alias :

Trigger Logic

DECLARE

BEGIN

IF :new.OWNERSHIP is not null and

:new.OWNERSHIP <> nvl(:old.OWNERSHIP,'@') and

NOT Chk\_IiCode( 'FEAT\_OWNERSHIP', :new.OWNERSHIP )

THEN

Raise\_Application\_Error( -20301,

'Invalid value ""||:new.OWNERSHIP||"" '||

'for II\_Features.OWNERSHIP.' );

END IF;

--

IF :new.MAINTENANCE\_LEVEL is not null and

:new.MAINTENANCE\_LEVEL <> nvl(:old.MAINTENANCE\_LEVEL,'@') and

NOT Chk\_IiCode( 'FEAT\_MAINTENANCE\_LEVEL', :new.MAINTENANCE\_LEVEL )

THEN

Raise\_Application\_Error( -20302,

'Invalid value ""||:new.MAINTENANCE\_LEVEL||"" '||

'for II\_Features.MAINTENANCE\_LEVEL.' );

END IF;

--

IF :new.MAINTENANCE\_RESPONSIBILITY is not null and

:new.MAINTENANCE\_RESPONSIBILITY <>

nvl(:old.MAINTENANCE\_RESPONSIBILITY,'@') and

NOT Chk\_IiCode( 'MAINTENANCE\_RESPONSIBILITY',

:new.MAINTENANCE\_RESPONSIBILITY )

THEN

```

Raise_Application_Error( -20303,
'Invalid value ""||:new.MAINTENANCE_RESPONSIBILITY||"" '||
'for II_Features.MAINTENANCE_RESPONSIBILITY.' );
END IF;
--
IF :new.CONDITION_RATING is not null and
:new.CONDITION_RATING <> nvl(:old.CONDITION_RATING,'@') and
NOT Chk_IiCode( 'FEAT_CONDITION_RATING', :new.CONDITION_RATING )
THEN
Raise_Application_Error( -20304,
'Invalid value ""||:new.CONDITION_RATING||"" '||
'for II_Features.CONDITION_RATING.' );
END IF;
--

```

Oracle Designer Report : cktrgger

#### Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

II\_FEATURES (table)

```

IF :new.APPRAISAL_RATING is not null and
:new.APPRAISAL_RATING <> nvl(:old.APPRAISAL_RATING,'@') and
NOT Chk_IiCode( 'FEAT_APPRAISAL_RATING', :new.APPRAISAL_RATING )
THEN
Raise_Application_Error( -20305,
'Invalid value ""||:new.APPRAISAL_RATING||"" '||
'for II_Features.APPRAISAL_RATING.' );
END IF;
--
IF :new.DEVELOPMENT_STATUS is not null and
:new.DEVELOPMENT_STATUS <> nvl(:old.DEVELOPMENT_STATUS,'@') and
NOT Chk_IiCode( 'FEAT_DEVELOPMENT_STATUS', :new.DEVELOPMENT_STATUS
)
THEN
Raise_Application_Error( -20306,
'Invalid value ""||:new.DEVELOPMENT_STATUS||"" '||
'for II_Features.DEVELOPMENT_STATUS.' );
END IF;
--

```



```

IF :new.HISTORICAL_STATUS is not null and
:new.HISTORICAL_STATUS <> nvl(:old.HISTORICAL_STATUS,'@') and
NOT Chk_IiCode( 'FEAT_HISTORICAL_STATUS', :new.HISTORICAL_STATUS )
THEN
Raise_Application_Error( -20307,
'Invalid value ""||:new.HISTORICAL_STATUS||"" '||
'for II_Features.HISTORICAL_STATUS.' );
END IF;
--
IF :new.LOCATION is not null and
:new.LOCATION <> nvl(:old.LOCATION,'@') and
NOT Chk_IiCode( 'FEAT_LOCATION', :new.LOCATION )
THEN
Raise_Application_Error( -20308,
'Invalid value ""||:new.LOCATION||"" '||
'for II_Features.LOCATION.' );
END IF;
--
IF :new.LOCATION_UOM is not null and
:new.LOCATION_UOM <> nvl(:old.LOCATION_UOM,'@') and
NOT Chk_IiCode( 'FEAT_UOM', :new.LOCATION_UOM )
THEN
Raise_Application_Error( -20309,
'Invalid value ""||:new.LOCATION_UOM||"" '||
'for II_Features.LOCATION_UOM.' );
END IF;
--
IF :new.ACCESSIBILITY_STATUS is not null and
:new.ACCESSIBILITY_STATUS <> nvl(:old.ACCESSIBILITY_STATUS,'@') and
NOT Chk_IiCode( 'ACCESSIBILITY_STATUS', :new.ACCESSIBILITY_STATUS )
THEN
Raise_Application_Error( -203010,
'Invalid value ""||:new.ACCESSIBILITY_STATUS||"" '||
'for II_Features.ACCESSIBILITY_STATUS.' );
END IF;
--
IF :new.ACQUISITION_METHOD is not null and

```

Oracle Designer Report : cktgrger

## Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

II\_FEATURES (table)

```
:new.ACQUISITION_METHOD <> nvl(:old.ACQUISITION_METHOD,'@') and
NOT Chk_IiCode( 'ACQUISITION_METHOD', :new.ACQUISITION_METHOD )
THEN
```

```
Raise_Application_Error( -203011,
'Invalid value "||:new.ACQUISITION_METHOD||"' ||
'for II_Features.ACQUISITION_METHOD.' );
END IF;
```

--

```
IF :new.OPERATIONAL_STATUS is not null and
:new.OPERATIONAL_STATUS <> nvl(:old.OPERATIONAL_STATUS,'@') and
NOT Chk_IiCode( 'OPERATIONAL_STATUS', :new.OPERATIONAL_STATUS )
THEN
```

```
Raise_Application_Error( -203012,
'Invalid value "||:new.OPERATIONAL_STATUS||"' ||
'for II_Features.OPERATIONAL_STATUS.' );
END IF;
```

--

```
IF Inserting
THEN
```

```
-- Supply CN if it is null.
```

```
IF :new.CN is null
```

```
THEN
```

```
:new.CN := Db_Instance.GENERATE_GLOBAL_ID( 'FEAT_CN_SEQ' );
END IF;
```

--

```
-- Set the CREATED... audit columns on row insert if not supplied.
```

```
IF :new.CREATED_BY is null or
:new.CREATED_DATE is null or
:new.CREATED_IN_INSTANCE is null
THEN
```

```
Db_Instance.INSERT_AUDIT_COLUMNS( :new.CREATED_BY
, :new.CREATED_IN_INSTANCE
, :new.CREATED_DATE );
END IF;
```

---

```
ELSIF Updating
```

```
THEN
```

```
-- Set the MODIFIED... audit columns on row update.
```

```

, :new.MODIFIED_IN_INSTANCE
, :new.MODIFIED_DATE );
---
END IF;
END;
DECLARE
BEGIN
IF :new.OWNERSHIP is not null and
:new.OWNERSHIP <> nvl(:old.OWNERSHIP,'@') and
NOT Chk_liCode( 'FEAT_OWNERSHIP', :new.OWNERSHIP )
THEN
Raise_Application_Error( -20301,
'Invalid value ""||:new.OWNERSHIP||"" '||
'for II_Features.OWNERSHIP.' );
END IF;
--
IF :new.MAINTENANCE_LEVEL is not null and
:new.MAINTENANCE_LEVEL <> nvl(:old.MAINTENANCE_LEVEL,'@') and

```

Oracle Designer Report : cktgrger

#### Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

II\_FEATURES (table)

```

NOT Chk_liCode( 'FEAT_MAINTENANCE_LEVEL', :new.MAINTENANCE_LEVEL )
THEN

```

```

Raise_Application_Error( -20302,
'Invalid value ""||:new.MAINTENANCE_LEVEL||"" '||
'for II_Features.MAINTENANCE_LEVEL.' );
END IF;

```

--

```

IF :new.MAINTENANCE_RESPONSIBILITY is not null and
:new.MAINTENANCE_RESPONSIBILITY <>
nvl(:old.MAINTENANCE_RESPONSIBILITY,'@') and
NOT Chk_liCode( 'MAINTENANCE_RESPONSIBILITY',
:new.MAINTENANCE_RESPONSIBILITY )
THEN
Raise_Application_Error( -20303,
'Invalid value ""||:new.MAINTENANCE_RESPONSIBILITY||"" '||
'for II_Features.MAINTENANCE_RESPONSIBILITY.' );
END IF;

```

```

IF :new.CONDITION_RATING is not null and
:new.CONDITION_RATING <> nvl(:old.CONDITION_RATING,'@') and
NOT Chk_liCode( 'FEAT_CONDITION_RATING', :new.CONDITION_RATING )
THEN
Raise_Application_Error( -20304,
'Invalid value "'||:new.CONDITION_RATING||"' ||
'for II_Features.CONDITION_RATING.' );
END IF;
--
IF :new.APPRAISAL_RATING is not null and
:new.APPRAISAL_RATING <> nvl(:old.APPRAISAL_RATING,'@') and
NOT Chk_liCode( 'FEAT_APPRAISAL_RATING', :new.APPRAISAL_RATING )
THEN
Raise_Application_Error( -20305,
'Invalid value "'||:new.APPRAISAL_RATING||"' ||
'for II_Features.APPRAISAL_RATING.' );
END IF;
--
IF :new.DEVELOPMENT_STATUS is not null and
:new.DEVELOPMENT_STATUS <> nvl(:old.DEVELOPMENT_STATUS,'@') and
NOT Chk_liCode( 'FEAT_DEVELOPMENT_STATUS', :new.DEVELOPMENT_STATUS
)
THEN
Raise_Application_Error( -20306,
'Invalid value "'||:new.DEVELOPMENT_STATUS||"' ||
'for II_Features.DEVELOPMENT_STATUS.' );
END IF;
--
IF :new.HISTORICAL_STATUS is not null and
:new.HISTORICAL_STATUS <> nvl(:old.HISTORICAL_STATUS,'@') and
NOT Chk_liCode( 'FEAT_HISTORICAL_STATUS', :new.HISTORICAL_STATUS )
THEN
Raise_Application_Error( -20307,
'Invalid value "'||:new.HISTORICAL_STATUS||"' ||
'for II_Features.HISTORICAL_STATUS.' );
END IF;
--
IF :new.LOCATION is not null and
:new.LOCATION <> nvl(:old.LOCATION,'@') and

```

Oracle Designer Report : cktrgger

## Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

II\_FEATURES (table)

NOT Chk\_IiCode( 'FEAT\_LOCATION', :new.LOCATION )

THEN

Raise\_Application\_Error( -20308,

'Invalid value "'||:new.LOCATION||"' ||

'for II\_Features.LOCATION.' );

END IF;

--

IF :new.LOCATION\_UOM is not null and

:new.LOCATION\_UOM &lt;&gt; nvl(:old.LOCATION\_UOM,'@') and

NOT Chk\_IiCode( 'FEAT\_UOM', :new.LOCATION\_UOM )

THEN

Raise\_Application\_Error( -20309,

'Invalid value "'||:new.LOCATION\_UOM||"' ||

'for II\_Features.LOCATION\_UOM.' );

END IF;

--

IF :new.ACCESSIBILITY\_STATUS is not null and

:new.ACCESSIBILITY\_STATUS &lt;&gt; nvl(:old.ACCESSIBILITY\_STATUS,'@') and

NOT Chk\_IiCode( 'ACCESSIBILITY\_STATUS', :new.ACCESSIBILITY\_STATUS )

THEN

Raise\_Application\_Error( -203010,

'Invalid value "'||:new.ACCESSIBILITY\_STATUS||"' ||

'for II\_Features.ACCESSIBILITY\_STATUS.' );

END IF;

--

IF :new.ACQUISITION\_METHOD is not null and

:new.ACQUISITION\_METHOD &lt;&gt; nvl(:old.ACQUISITION\_METHOD,'@') and

NOT Chk\_IiCode( 'ACQUISITION\_METHOD', :new.ACQUISITION\_METHOD )

THEN

Raise\_Application\_Error( -203011,

'Invalid value "'||:new.ACQUISITION\_METHOD||"' ||

'for II\_Features.ACQUISITION\_METHOD.' );

END IF;

--

IF :new.OPERATIONAL\_STATUS is not null and

:new.OPERATIONAL\_STATUS &lt;&gt; nvl(:old.OPERATIONAL\_STATUS,'@') and

NOT Chk\_IiCode( 'OPERATIONAL\_STATUS', :new.OPERATIONAL\_STATUS )

THEN

```

Raise_Application_Error( -203012,
'Invalid value ""||:new.OPERATIONAL_STATUS||"" '||
'for II_Features.OPERATIONAL_STATUS.' );
END IF;
--
IF Inserting
THEN
-- Supply CN if it is null.
IF :new.CN is null
THEN
:new.CN := Db_Instance.GENERATE_GLOBAL_ID( 'FEAT_CN_SEQ' );
END IF;
--
-- Set the CREATED... audit columns on row insert if not supplied.
IF :new.CREATED_BY is null or
:new.CREATED_DATE is null or
:new.CREATED_IN_INSTANCE is null
THEN

```

Oracle Designer Report : cktrgger

#### Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

II\_FEATURES (table)

```

Db_Instance.INSERT_AUDIT_COLUMNS( :new.CREATED_BY
, :new.CREATED_IN_INSTANCE
, :new.CREATED_DATE );
END IF;

```

---

ELSIF Updating

THEN

-- Set the MODIFIED... audit columns on row update.

```

Db_Instance.UPDATE_AUDIT_COLUMNS( :new.MODIFIED_BY
, :new.MODIFIED_IN_INSTANCE
, :new.MODIFIED_DATE );

```

---

END IF;

END;

DECLARE

BEGIN

```

IF :new.OWNERSHIP is not null and
:new.OWNERSHIP <> nvl(:old.OWNERSHIP,'@') and
NOT Chk_IiCode( 'FEAT_OWNERSHIP', :new.OWNERSHIP )
THEN
Raise_Application_Error( -20301,
'Invalid value ""||:new.OWNERSHIP||"" '||
'for II_Features.OWNERSHIP.' );
END IF;
--
IF :new.MAINTENANCE_LEVEL is not null and
:new.MAINTENANCE_LEVEL <> nvl(:old.MAINTENANCE_LEVEL,'@') and
NOT Chk_IiCode( 'FEAT_MAINTENANCE_LEVEL', :new.MAINTENANCE_LEVEL )
THEN
Raise_Application_Error( -20302,
'Invalid value ""||:new.MAINTENANCE_LEVEL||"" '||
'for II_Features.MAINTENANCE_LEVEL.' );
END IF;
--
IF :new.MAINTENANCE_RESPONSIBILITY is not null and
:new.MAINTENANCE_RESPONSIBILITY <>
nvl(:old.MAINTENANCE_RESPONSIBILITY,'@') and
NOT Chk_IiCode( 'MAINTENANCE_RESPONSIBILITY',
:new.MAINTENANCE_RESPONSIBILITY )
THEN
Raise_Application_Error( -20303,
'Invalid value ""||:new.MAINTENANCE_RESPONSIBILITY||"" '||
'for II_Features.MAINTENANCE_RESPONSIBILITY.' );
END IF;
--
IF :new.CONDITION_RATING is not null and
:new.CONDITION_RATING <> nvl(:old.CONDITION_RATING,'@') and
NOT Chk_IiCode( 'FEAT_CONDITION_RATING', :new.CONDITION_RATING )
THEN
Raise_Application_Error( -20304,
'Invalid value ""||:new.CONDITION_RATING||"" '||
'for II_Features.CONDITION_RATING.' );
END IF;
--
IF :new.APPRAISAL_RATING is not null and
:new.APPRAISAL_RATING <> nvl(:old.APPRAISAL_RATING,'@') and
NOT Chk_IiCode( 'FEAT_APPRAISAL_RATING', :new.APPRAISAL_RATING )

```

Oracle Designer Report : cktggr

## Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

II\_FEATURES (table)

THEN

Raise\_Application\_Error( -20305,

'Invalid value "||:new.APPRAISAL\_RATING||" '||

'for II\_Features.APPRAISAL\_RATING.');

END IF;

--

IF :new.DEVELOPMENT\_STATUS is not null and

:new.DEVELOPMENT\_STATUS &lt;&gt; nvl(:old.DEVELOPMENT\_STATUS,'@') and

NOT Chk\_IiCode( 'FEAT\_DEVELOPMENT\_STATUS', :new.DEVELOPMENT\_STATUS

)

THEN

Raise\_Application\_Error( -20306,

'Invalid value "||:new.DEVELOPMENT\_STATUS||" '||

'for II\_Features.DEVELOPMENT\_STATUS.');

END IF;

--

IF :new.HISTORICAL\_STATUS is not null and

:new.HISTORICAL\_STATUS &lt;&gt; nvl(:old.HISTORICAL\_STATUS,'@') and

NOT Chk\_IiCode( 'FEAT\_HISTORICAL\_STATUS', :new.HISTORICAL\_STATUS )

THEN

Raise\_Application\_Error( -20307,

'Invalid value "||:new.HISTORICAL\_STATUS||" '||

'for II\_Features.HISTORICAL\_STATUS.');

END IF;

--

IF :new.LOCATION is not null and

:new.LOCATION &lt;&gt; nvl(:old.LOCATION,'@') and

NOT Chk\_IiCode( 'FEAT\_LOCATION', :new.LOCATION )

THEN

Raise\_Application\_Error( -20308,

'Invalid value "||:new.LOCATION||" '||

'for II\_Features.LOCATION.');

END IF;

--

IF :new.LOCATION\_UOM is not null and

:new.LOCATION\_UOM &lt;&gt; nvl(:old.LOCATION\_UOM,'@') and

NOT Chk\_IiCode( 'FEAT\_UOM', :new.LOCATION\_UOM )

THEN



```

Raise_Application_Error( -20309,
'Invalid value "||:new.LOCATION_UOM||"' ||
'for II_Features.LOCATION_UOM.' );
END IF;
--
IF :new.ACCESSIBILITY_STATUS is not null and
:new.ACCESSIBILITY_STATUS <> nvl(:old.ACCESSIBILITY_STATUS,'@') and
NOT Chk_IiCode( 'ACCESSIBILITY_STATUS', :new.ACCESSIBILITY_STATUS )
THEN
Raise_Application_Error( -203010,
'Invalid value "||:new.ACCESSIBILITY_STATUS||"' ||
'for II_Features.ACCESSIBILITY_STATUS.' );
END IF;
--
IF :new.ACQUISITION_METHOD is not null and
:new.ACQUISITION_METHOD <> nvl(:old.ACQUISITION_METHOD,'@') and
NOT Chk_IiCode( 'ACQUISITION_METHOD', :new.ACQUISITION_METHOD )
THEN

```

Oracle Designer Report : cktrgger

#### Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

II\_FEATURES (table)

```

Raise_Application_Error( -203011,
'Invalid value "||:new.ACQUISITION_METHOD||"' ||
'for II_Features.ACQUISITION_METHOD.' );
END IF;
--
IF :new.OPERATIONAL_STATUS is not null and
:new.OPERATIONAL_STATUS <> nvl(:old.OPERATIONAL_STATUS,'@') and
NOT Chk_IiCode( 'OPERATIONAL_STATUS', :new.OPERATIONAL_STATUS )
THEN
Raise_Application_Error( -203012,
'Invalid value "||:new.OPERATIONAL_STATUS||"' ||
'for II_Features.OPERATIONAL_STATUS.' );
END IF;
--
IF Inserting
THEN
-- Supply CN if it is null.

```

```

IF :new.CN is null
THEN
:new.CN := Db_Instance.GENERATE_GLOBAL_ID( 'FEAT_CN_SEQ' );
END IF;
--
-- Set the CREATED... audit columns on row insert if not supplied.
IF :new.CREATED_BY is null or
:new.CREATED_DATE is null or
:new.CREATED_IN_INSTANCE is null
THEN
Db_Instance.INSERT_AUDIT_COLUMNS( :new.CREATED_BY
, :new.CREATED_IN_INSTANCE
, :new.CREATED_DATE );
END IF;
---
ELSIF Updating
THEN
-- Set the MODIFIED... audit columns on row update.
Db_Instance.UPDATE_AUDIT_COLUMNS( :new.MODIFIED_BY
, :new.MODIFIED_IN_INSTANCE
, :new.MODIFIED_DATE );
---
END IF;
END;
DECLARE
vModuleName CONSTANT VARCHAR2(30) := 'II_FEA_RPA_AR_TRG'; -- Added on
01/07/2003
BEGIN
-- DELETING
IF DELETING THEN
-- DELETE from ii_rpa_assets_v_tab
DELETE FROM ii_rpa_assets_v_tab
WHERE fea_cn = :old.cn;

-- INSERTING
ELSIF INSERTING THEN
-- Insert into table ii_rpa_assets_v_tab
BEGIN
INSERT INTO ii_rpa_assets_v_tab (
FEA_CN

```

Oracle Designer Report : cktrgger

## Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

II\_FEATURES (table)

```

,LU_CN
,PLD_CN
,OBJECT_NAME
,OBJECT_CLASS
,ID
,NAME
,SUB_TYPE
,OWNERSHIP
,ADMIN_ORG
,DEVELOPMENT_STATUS
,CREATED_DATE
,CREATED_BY
,SUBLEDGER_FLAG
,subledger_cn
,agency_code
) VALUES (
:new.CN
,NULL
,NULL
,:new.OBJ_NAME
,:new.OBJ_CLASS
,:new.ID
,:new.NAME
,NULL
,:new.OWNERSHIP
,:new.ADMIN_ORG
,:new.DEVELOPMENT_STATUS
,:new.CREATED_DATE
,:new.CREATED_BY
,'N'
,null
,:NEW.agency_code -- Added on 10/28/2003 by C.Ma
);

```

EXCEPTION WHEN OTHERS THEN

```

ii_rpa_errors.save_log(vModuleName, 'II_RPA_ASSETS_V_TAB', :new.cn,
null, null, SQLERRM);

```

END;

```

-- UPDATING
ELSIF UPDATING THEN
-- Update table ii_rpa_assets_v_tab
UPDATE ii_rpa_assets_v_tab tab
SET (
OBJECT_NAME
,OBJECT_CLASS
,ID
,NAME
,SUB_TYPE
,OWNERSHIP
,ADMIN_ORG
,DEVELOPMENT_STATUS
,CREATED_DATE
,CREATED_BY
) = (SELECT
:new.OBJ_NAME OBJECT_NAME
,:new.OBJ_CLASS OBJECT_CLASS
,:new.ID
,:new.NAME

```

Oracle Designer Report : cktrgger

#### Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

II\_FEATURES (table)

```

,NULL
,:new.OWNERSHIP
,:new.ADMIN_ORG
,:new.DEVELOPMENT_STATUS
,:new.CREATED_DATE
,:new.CREATED_BY
FROM DUAL)
WHERE tab.fea_cn = :new.cn;

```

```

-- Update table ii_rpa_subledgers
UPDATE ii_rpa_subledgers sub
SET admin_org = :new.admin_org
WHERE sub.fea_cn = :new.cn
AND ii_sql_func.super_equal(admin_org, :new.admin_org) <> 'TRUE';

```

```

-----
-- Audit
-----
DECLARE
vNewFea ii_features%rowtype;
vOldFea ii_features%rowtype;
BEGIN
--vOldFea.GSA_INSTALLATION_CN := :old.GSA_INSTALLATION_CN;-- commented
out on 10/28/2003 by C.Ma
vOldFea.GROSS_SQFT := :old.GROSS_SQFT;
vOldFea.CN := :old.CN;
vOldFea.OBJ_TECH := :old.OBJ_TECH;
vOldFea.OBJ_NAME := :old.OBJ_NAME;
vOldFea.OBJ_CLASS := :old.OBJ_CLASS;
vOldFea.ID := :old.ID;
vOldFea.NAME := :old.NAME;
vOldFea.SUB_TYPE := :old.SUB_TYPE;
vOldFea.OWNERSHIP := :old.OWNERSHIP;
vOldFea.ADMIN_ORG := :old.ADMIN_ORG;
vOldFea.MASTER_SITE := :old.MASTER_SITE;
vOldFea.MANAGING_CONT_CN := :old.MANAGING_CONT_CN;
vOldFea.FUNDING_RESPONSIBILITY := :old.FUNDING_RESPONSIBILITY;
vOldFea.MAINTENANCE_LEVEL := :old.MAINTENANCE_LEVEL;
vOldFea.MAINTENANCE_RESPONSIBILITY :=
:old.MAINTENANCE_RESPONSIBILITY;

vOldFea.PRINCIPAL_MATERIAL := :old.PRINCIPAL_MATERIAL;
vOldFea.FEATURE_SIZE := :old.FEATURE_SIZE;
vOldFea.ACQUISITION_METHOD := :old.ACQUISITION_METHOD;
vOldFea.YEAR_CONSTRUCTED := :old.YEAR_CONSTRUCTED;
vOldFea.INVENTORIED := :old.INVENTORIED;
vOldFea.DATE_LAST_INSPECTED := :old.DATE_LAST_INSPECTED;
vOldFea.CONDITION_RATING := :old.CONDITION_RATING;
vOldFea.APPRAISAL_RATING := :old.APPRAISAL_RATING;
vOldFea.DEVELOPMENT_STATUS := :old.DEVELOPMENT_STATUS;
vOldFea.OPERATIONAL_STATUS := :old.OPERATIONAL_STATUS;
vOldFea.HISTORICAL_STATUS := :old.HISTORICAL_STATUS;
vOldFea.ACCESSIBILITY_STATUS := :old.ACCESSIBILITY_STATUS;
vOldFea.ACCESSIBILITY_REMARKS := :old.ACCESSIBILITY_REMARKS;
vOldFea.REMARKS := :old.REMARKS;
vOldFea.CHECKED_OUT_BY := :old.CHECKED_OUT_BY;
vOldFea.CHECKED_OUT_DATE := :old.CHECKED_OUT_DATE;
vOldFea.MOST_RECENT_YR_RECONSTR := :old.MOST_RECENT_YR_RECONSTR;
Oracle Designer Report : cktrgger

```

## Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

II\_FEATURES (table)

```

vOldFea.CREATED_BY := :old.CREATED_BY;
vOldFea.CREATED_DATE := :old.CREATED_DATE;
vOldFea.CREATED_IN_INSTANCE := :old.CREATED_IN_INSTANCE;
vOldFea.MODIFIED_BY := :old.MODIFIED_BY;
vOldFea.MODIFIED_DATE := :old.MODIFIED_DATE;
vOldFea.MODIFIED_IN_INSTANCE := :old.MODIFIED_IN_INSTANCE;
--vNewFea.GSA_INSTALLATION_CN := :new.GSA_INSTALLATION_CN;-- commented
out on 10/28/2003 by C.Ma
vNewFea.GROSS_SQFT := :new.GROSS_SQFT;
vNewFea.CN := :new.CN;
vNewFea.OBJ_TECH := :new.OBJ_TECH;
vNewFea.OBJ_NAME := :new.OBJ_NAME;
vNewFea.OBJ_CLASS := :new.OBJ_CLASS;
vNewFea.ID := :new.ID;
vNewFea.NAME := :new.NAME;
vNewFea.SUB_TYPE := :new.SUB_TYPE;
vNewFea.OWNERSHIP := :new.OWNERSHIP;
vNewFea.ADMIN_ORG := :new.ADMIN_ORG;
vNewFea.MASTER_SITE := :new.MASTER_SITE;
vNewFea.MANAGING_CONT_CN := :new.MANAGING_CONT_CN;
vNewFea.FUNDING_RESPONSIBILITY := :new.FUNDING_RESPONSIBILITY;
vNewFea.MAINTENANCE_LEVEL := :new.MAINTENANCE_LEVEL;
vNewFea.MAINTENANCE_RESPONSIBILITY :=
:new.MAINTENANCE_RESPONSIBILITY;

vNewFea.PRINCIPAL_MATERIAL := :new.PRINCIPAL_MATERIAL;
vNewFea.FEATURE_SIZE := :new.FEATURE_SIZE;
vNewFea.ACQUISITION_METHOD := :new.ACQUISITION_METHOD;
vNewFea.YEAR_CONSTRUCTED := :new.YEAR_CONSTRUCTED;
vNewFea.INVENTORIED := :new.INVENTORIED;
vNewFea.DATE_LAST_INSPECTED := :new.DATE_LAST_INSPECTED;
vNewFea.CONDITION_RATING := :new.CONDITION_RATING;
vNewFea.APPRAISAL_RATING := :new.APPRAISAL_RATING;
vNewFea.DEVELOPMENT_STATUS := :new.DEVELOPMENT_STATUS;
vNewFea.OPERATIONAL_STATUS := :new.OPERATIONAL_STATUS;
vNewFea.HISTORICAL_STATUS := :new.HISTORICAL_STATUS;
vNewFea.ACCESSIBILITY_STATUS := :new.ACCESSIBILITY_STATUS;

```

```

vNewFea.ACCESSIBILITY_REMARKS := :new.ACCESSIBILITY_REMARKS;
vNewFea.REMARKS := :new.REMARKS;
vNewFea.CHECKED_OUT_BY := :new.CHECKED_OUT_BY;
vNewFea.CHECKED_OUT_DATE := :new.CHECKED_OUT_DATE;
vNewFea.MOST_RECENT_YR_RECONSTR := :new.MOST_RECENT_YR_RECONSTR;
vNewFea.CREATED_BY := :new.CREATED_BY;
vNewFea.CREATED_DATE := :new.CREATED_DATE;
vNewFea.CREATED_IN_INSTANCE := :new.CREATED_IN_INSTANCE;
vNewFea.MODIFIED_BY := :new.MODIFIED_BY;
vNewFea.MODIFIED_DATE := :new.MODIFIED_DATE;
vNewFea.MODIFIED_IN_INSTANCE := :new.MODIFIED_IN_INSTANCE;

ii_rpa_auditing.audit_fea('U', :new.modified_by, :new.modified_date,
vOldFea, vNewFea);
END;

END IF;

END;

```

Oracle Designer Report : cktrgger

Database Trigger Definition

```

Container:
  II_RPA
  Version :
Table/View Name
:
  ACCPLISHMENT_INSTRUMENTS (table)
Trigger Name
:
  AI_BRIU_TRG
                                Create?
                                Yes
PL/SQL Module
Name :
  AI_BRIU_TRG
                                Enabled?
                                Yes
PL/SQL File :
Purpose :
  .

```

Trigger Time:  
BEFORE

Level:  
ROW

Insert?  
Yes  
Delete?  
No

New Alias  
NEW

Update?  
Yes

Old Alias :  
OLD

```

Trigger Logic
DECLARE
BEGIN
IF inserting
THEN
IF :new.cn is null
THEN :new.cn := db_instance.generate_global_id('ACCINST_CN_SEQ');
END IF;
db_instance.insert_audit_columns(:new.created_by,
:new.created_in_instance,
:new.created_date);
ELSE db_instance.update_audit_columns(:new.modified_by,
:new.modified_in_instance,
:new.modified_date);
END IF;
END;

```

Oracle Designer Report : cktrgger

#### Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

ACCPHISHMENT\_INSTRUMENTS (table)

Trigger Name

:

II\_AI\_ARIUD\_TRG

Create?



PL/SQL Module

Name :

II\_AI\_ARIUD\_TRG

Enabled?

Yes

PL/SQL File :

Purpose :

After-row I/U/D to populate range permit audit tables.

Trigger Time:

AFTER

Level:

ROW

Insert?

Yes

Delete?

Yes

New Alias

Update?

Yes

Old Alias :

Trigger Logic

DECLARE

BEGIN

IF inserting

THEN

IF :new.cn is null

THEN :new.cn := db\_instance.generate\_global\_id('ACCINST\_CN\_SEQ');

END IF;

db\_instance.insert\_audit\_columns(:new.created\_by,

:new.created\_in\_instance,

:new.created\_date);

ELSE db\_instance.update\_audit\_columns(:new.modified\_by,

:new.modified\_in\_instance,

:new.modified\_date);

END IF;

END;

DECLARE

BEGIN

IF inserting

THEN

IF :new.cn is null

THEN :new.cn := db\_instance.generate\_global\_id('ACCINST\_CN\_SEQ');

END IF;

db\_instance.insert\_audit\_columns(:new.created\_by,

:new.created\_in\_instance,

```

ELSE db_instance.update_audit_columns(:new.modified_by,
:new.modified_in_instance,
:new.modified_date);
END IF;
END;
DECLARE
BEGIN
IF inserting
THEN
IF :new.cn is null
THEN :new.cn := db_instance.generate_global_id('ACCINST_CN_SEQ');
END IF;
db_instance.insert_audit_columns(:new.created_by,
                                Oracle Designer Report : cktrgger

```

## Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

ACCOMPLISHMENT\_INSTRUMENTS (table)

```

:new.created_in_instance,
:new.created_date);
ELSE db_instance.update_audit_columns(:new.modified_by,
:new.modified_in_instance,
:new.modified_date);
END IF;
END;
Declare
--
vRNGPAUDRecord ii_range_permit_audits%rowtype;
--
Begin
--
if INSERTING then
--
if :new.obj_name = 'RANGE_PERMIT' then
--
if ii_range_permits_audit.vAuditCN is not null then -- set in
instead-of trigger of II_RANGE_PERMITS_V
null; -- Audit record already written.
else

```

```

vRNGPAUDRecord.changed_by := user;
vRNGPAUDRecord.changed_on := sysdate;
vRNGPAUDRecord.change_type := 'I';
vRNGPAUDRecord.old_id := null;
vRNGPAUDRecord.new_id := :new.ID;
vRNGPAUDRecord.old_cn := null;
vRNGPAUDRecord.new_cn := :new.CN;
vRNGPAUDRecord.old_description := null;
vRNGPAUDRecord.new_description := :new.Description;
vRNGPAUDRecord.old_managing_cont_cn := null;
vRNGPAUDRecord.new_managing_cont_cn := :new.managing_cont_cn;
vRNGPAUDRecord.old_obj_name := null;
vRNGPAUDRecord.new_obj_name := :new.obj_name;
vRNGPAUDRecord.old_permittee_cn := null;
vRNGPAUDRecord.new_permittee_cn := null;
vRNGPAUDRecord.old_supercedes_rngpmt_cn := null;
vRNGPAUDRecord.new_supercedes_rngpmt_cn := null;
vRNGPAUDRecord.old_split_bill_flag := null;
vRNGPAUDRecord.new_split_bill_flag := null;
vRNGPAUDRecord.old_split_bill_month := null;
vRNGPAUDRecord.new_split_bill_month := null;
vRNGPAUDRecord.old_status := null;
vRNGPAUDRecord.new_status := null;
vRNGPAUDRecord.old_escrow_waiver_flag := null;
vRNGPAUDRecord.new_escrow_waiver_flag := null;
vRNGPAUDRecord.old_bid_bill_flag := null;
vRNGPAUDRecord.new_bid_bill_flag := null;
vRNGPAUDRecord.change_occ_in_table := 'ACCOMPLISHMENT_INSTRUMENTS';
ii_range_permits_audit.Insert_II_RANGE_PERMIT_AUDITS ( vRNGPAUDRecord
);
end if;
--
end if;
--
elsif UPDATING then

```

Oracle Designer Report : cktrgger

#### Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

ACCOMPLISHMENT\_INSTRUMENTS (table)

```

--
if :new.obj_name = 'RANGE_PERMIT' then
if nvl(:new.id,'@') != nvl(:old.id,'@') or
nvl(:new.description,'@') != nvl(:old.description,'@') or
nvl(:new.managing_cont_cn,'@') != nvl(:old.managing_cont_cn,'@') or
nvl(:new.obj_name,'@') != nvl(:old.obj_name,'@') or
nvl(:new.cn,'@') != nvl(:old.cn,'@') or
nvl(:new.description,'@') != nvl(:old.description,'@') then
if ii_range_permits_audit.vAuditCN is not null then -- set in
instead-of trigger of II_RANGE_PERMITS_V
null; -- Audit record already written.
else
vRNGPAUDRecord.changed_by := user;
vRNGPAUDRecord.changed_on := sysdate;
vRNGPAUDRecord.change_type := 'U';
vRNGPAUDRecord.old_id := :old.ID;
vRNGPAUDRecord.new_id := :new.ID;
vRNGPAUDRecord.old_cn := :old.CN;
vRNGPAUDRecord.new_cn := :new.CN;
vRNGPAUDRecord.old_description := :old.Description;
vRNGPAUDRecord.new_description := :new.Description;
vRNGPAUDRecord.old_managing_cont_cn := :old.managing_cont_cn;
vRNGPAUDRecord.new_managing_cont_cn := :new.managing_cont_cn;
vRNGPAUDRecord.old_obj_name := :old.obj_name;
vRNGPAUDRecord.new_obj_name := :new.obj_name;
vRNGPAUDRecord.old_permittee_cn := null;
vRNGPAUDRecord.new_permittee_cn := null;
vRNGPAUDRecord.old_supercedes_rngpmt_cn := null;
vRNGPAUDRecord.new_supercedes_rngpmt_cn := null;
vRNGPAUDRecord.old_split_bill_flag := null;
vRNGPAUDRecord.new_split_bill_flag := null;
vRNGPAUDRecord.old_split_bill_month := null;
vRNGPAUDRecord.new_split_bill_month := null;
vRNGPAUDRecord.old_status := null;
vRNGPAUDRecord.new_status := null;
vRNGPAUDRecord.old_escrow_waiver_flag := null;
vRNGPAUDRecord.new_escrow_waiver_flag := null;
vRNGPAUDRecord.old_bid_bill_flag := null;
vRNGPAUDRecord.new_bid_bill_flag := null;
vRNGPAUDRecord.change_occ_in_table := 'ACCOMPLISHMENT_INSTRUMENTS';
ii_range_permits_audit.Insert_II_RANGE_PERMIT_AUDITS ( vRNGPAUDRecord
);
end if;
end if;
end if;

```

```

--
elsif DELETING then
--
if :old.obj_name = 'RANGE_PERMIT' then
if ii_range_permits_audit.vAuditCN is not null then -- set in
instead-of trigger of II_RANGE_PERMITS_V
null; -- Audit record already written.
else

```

Oracle Designer Report : cktrgger

### Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

ACCOMPLISHMENT\_INSTRUMENTS (table)

```

vRNGPAUDRecord.changed_by := user;
vRNGPAUDRecord.changed_on := sysdate;
vRNGPAUDRecord.change_type := 'D';
vRNGPAUDRecord.old_id := :old.ID;
vRNGPAUDRecord.new_id := null;
vRNGPAUDRecord.old_cn := :old.CN;
vRNGPAUDRecord.new_cn := null;
vRNGPAUDRecord.old_description := :old.Description;
vRNGPAUDRecord.new_description := null;
vRNGPAUDRecord.old_managing_cont_cn := :old.managing_cont_cn;
vRNGPAUDRecord.new_managing_cont_cn := null;
vRNGPAUDRecord.old_obj_name := :old.obj_name;
vRNGPAUDRecord.new_obj_name := null;
vRNGPAUDRecord.old_permittee_cn := null;
vRNGPAUDRecord.new_permittee_cn := null;
vRNGPAUDRecord.old_supercedes_rngpmt_cn := null;
vRNGPAUDRecord.new_supercedes_rngpmt_cn := null;
vRNGPAUDRecord.old_split_bill_flag := null;
vRNGPAUDRecord.new_split_bill_flag := null;
vRNGPAUDRecord.old_split_bill_month := null;
vRNGPAUDRecord.new_split_bill_month := null;
vRNGPAUDRecord.old_status := null;
vRNGPAUDRecord.new_status := null;
vRNGPAUDRecord.old_escrow_waiver_flag := null;
vRNGPAUDRecord.new_escrow_waiver_flag := null;
vRNGPAUDRecord.old_bid_bill_flag := null;
vRNGPAUDRecord.new_bid_bill_flag := null;

```

```
vRNGPAUDRecord.change_occ_in_table := 'ACCOMPLISHMENT_INSTRUMENTS';  
ii_range_permits_audit.Insert_II_RANGE_PERMIT_AUDITS ( vRNGPAUDRecord  
);  
end if;  
end if;  
end if;  
End;
```

Oracle Designer Report : cktrgger

Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

II\_POOLEDS (table)

Trigger Name

:

II\_POOLED\_CN\_AUDIT

Create?

Yes

PL/SQL Module

Name :

II\_POOLED\_CN\_AUDIT

Enabled?

Yes

PL/SQL File :

Purpose :

.

Trigger Time:

BEFORE

Level:

ROW

Insert?

Yes

Delete?

No

New Alias

NEW

Update?

Yes

Old Alias :

OLD

```

Trigger Logic
Declare
--
vRNGPAUDRecord ii_range_permit_audits%rowtype;
--
Begin
--
if INSERTING then
--
if :new.obj_name = 'RANGE_PERMIT' then
--
if ii_range_permits_audit.vAuditCN is not null then -- set in
instead-of trigger of II_RANGE_PERMITS_V
null; -- Audit record already written.
else
vRNGPAUDRecord.changed_by := user;
vRNGPAUDRecord.changed_on := sysdate;
vRNGPAUDRecord.change_type := 'I';
vRNGPAUDRecord.old_id := null;
vRNGPAUDRecord.new_id := :new.ID;
vRNGPAUDRecord.old_cn := null;
vRNGPAUDRecord.new_cn := :new.CN;
vRNGPAUDRecord.old_description := null;
vRNGPAUDRecord.new_description := :new.Description;
vRNGPAUDRecord.old_managing_cont_cn := null;
vRNGPAUDRecord.new_managing_cont_cn := :new.managing_cont_cn;
vRNGPAUDRecord.old_obj_name := null;
vRNGPAUDRecord.new_obj_name := :new.obj_name;
vRNGPAUDRecord.old_permittee_cn := null;
vRNGPAUDRecord.new_permittee_cn := null;
vRNGPAUDRecord.old_supercedes_rngpmt_cn := null;
vRNGPAUDRecord.new_supercedes_rngpmt_cn := null;
vRNGPAUDRecord.old_split_bill_flag := null;
vRNGPAUDRecord.new_split_bill_flag := null;
vRNGPAUDRecord.old_split_bill_month := null;
vRNGPAUDRecord.new_split_bill_month := null;
vRNGPAUDRecord.old_status := null;
vRNGPAUDRecord.new_status := null;
vRNGPAUDRecord.old_escrow_waiver_flag := null;
vRNGPAUDRecord.new_escrow_waiver_flag := null;
vRNGPAUDRecord.old_bid_bill_flag := null;
vRNGPAUDRecord.new_bid_bill_flag := null;
Oracle Designer Report : cktrgger

```

## Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

II\_POOLEDS (table)

```

vRNGPAUDRecord.change_occ_in_table := 'ACCOMPLISHMENT_INSTRUMENTS';
ii_range_permits_audit.Insert_II_RANGE_PERMIT_AUDITS ( vRNGPAUDRecord
);
end if;
--
end if;
--
elsif UPDATING then
--
if :new.obj_name = 'RANGE_PERMIT' then
if nvl(:new.id,'@') != nvl(:old.id,'@') or
nvl(:new.description,'@') != nvl(:old.description,'@') or
nvl(:new.managing_cont_cn,'@') != nvl(:old.managing_cont_cn,'@') or
nvl(:new.obj_name,'@') != nvl(:old.obj_name,'@') or
nvl(:new.cn,'@') != nvl(:old.cn,'@') or
nvl(:new.description,'@') != nvl(:old.description,'@') then
if ii_range_permits_audit.vAuditCN is not null then -- set in
instead-of trigger of II_RANGE_PERMITS_V
null; -- Audit record already written.
else
vRNGPAUDRecord.changed_by := user;
vRNGPAUDRecord.changed_on := sysdate;
vRNGPAUDRecord.change_type := 'U';
vRNGPAUDRecord.old_id := :old.ID;
vRNGPAUDRecord.new_id := :new.ID;
vRNGPAUDRecord.old_cn := :old.CN;
vRNGPAUDRecord.new_cn := :new.CN;
vRNGPAUDRecord.old_description := :old.Description;
vRNGPAUDRecord.new_description := :new.Description;
vRNGPAUDRecord.old_managing_cont_cn := :old.managing_cont_cn;
vRNGPAUDRecord.new_managing_cont_cn := :new.managing_cont_cn;
vRNGPAUDRecord.old_obj_name := :old.obj_name;
vRNGPAUDRecord.new_obj_name := :new.obj_name;
vRNGPAUDRecord.old_permittee_cn := null;
vRNGPAUDRecord.new_permittee_cn := null;
vRNGPAUDRecord.old_supercedes_rngpmt_cn := null;
vRNGPAUDRecord.new_supercedes_rngpmt_cn := null;
vRNGPAUDRecord.new_split_bill_flag := null;

```



```

vRNGPAUDRecord.old_split_bill_month := null;
vRNGPAUDRecord.new_split_bill_month := null;
vRNGPAUDRecord.old_status := null;
vRNGPAUDRecord.new_status := null;
vRNGPAUDRecord.old_escrow_waiver_flag := null;
vRNGPAUDRecord.new_escrow_waiver_flag := null;
vRNGPAUDRecord.old_bid_bill_flag := null;
vRNGPAUDRecord.new_bid_bill_flag := null;
vRNGPAUDRecord.change_occ_in_table := 'ACCOMPLISHMENT_INSTRUMENTS';
ii_range_permits_audit.Insert_II_RANGE_PERMIT_AUDITS ( vRNGPAUDRecord
);
end if;
end if;

```

Oracle Designer Report : cktrgger

#### Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

II\_POOLEDS (table)

end if;

--

elsif DELETING then

--

if :old.obj\_name = 'RANGE\_PERMIT' then

if ii\_range\_permits\_audit.vAuditCN is not null then -- set in  
instead-of trigger of II\_RANGE\_PERMITS\_V  
null; -- Audit record already written.

else

vRNGPAUDRecord.changed\_by := user;

vRNGPAUDRecord.changed\_on := sysdate;

vRNGPAUDRecord.change\_type := 'D';

vRNGPAUDRecord.old\_id := :old.ID;

vRNGPAUDRecord.new\_id := null;

vRNGPAUDRecord.old\_cn := :old.CN;

vRNGPAUDRecord.new\_cn := null;

vRNGPAUDRecord.old\_description := :old.Description;

vRNGPAUDRecord.new\_description := null;

vRNGPAUDRecord.old\_managing\_cont\_cn := :old.managing\_cont\_cn;

vRNGPAUDRecord.new\_managing\_cont\_cn := null;

vRNGPAUDRecord.old\_obj\_name := :old.obj\_name;

vRNGPAUDRecord.new\_obj\_name := null;

```

vRNGPAUDRecord.old_permittee_cn := null;
vRNGPAUDRecord.new_permittee_cn := null;
vRNGPAUDRecord.old_supercedes_rngpmt_cn := null;
vRNGPAUDRecord.new_supercedes_rngpmt_cn := null;
vRNGPAUDRecord.old_split_bill_flag := null;
vRNGPAUDRecord.new_split_bill_flag := null;
vRNGPAUDRecord.old_split_bill_month := null;
vRNGPAUDRecord.new_split_bill_month := null;
vRNGPAUDRecord.old_status := null;
vRNGPAUDRecord.new_status := null;
vRNGPAUDRecord.old_escrow_waiver_flag := null;
vRNGPAUDRecord.new_escrow_waiver_flag := null;
vRNGPAUDRecord.old_bid_bill_flag := null;
vRNGPAUDRecord.new_bid_bill_flag := null;
vRNGPAUDRecord.change_occ_in_table := 'ACCOMPLISHMENT_INSTRUMENTS';
ii_range_permits_audit.Insert_II_RANGE_PERMIT_AUDITS ( vRNGPAUDRecord
);
end if;
end if;
end if;
End;
DECLARE
--
--
v_tmp_hold CHAR(1);
--
--
v_admin_org_id II_CONTACTS.ID%TYPE;
--
--
p_id II_POOLEDS.ID%TYPE;
--
--
p_admin_org_cn II_POOLEDS.ADMIN_ORG_CN%TYPE;

```

Oracle Designer Report : cktrgger

#### Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

II\_POOLEDS (table)

```

--
--
p_obj_name II_POOLEDS.OBJ_NAME%TYPE;
--
--
admin_org_missing EXCEPTION;
PRAGMA EXCEPTION_INIT(admin_org_missing, -20004);
--
--
id_missing EXCEPTION;
PRAGMA EXCEPTION_INIT(id_missing, -20003);
--
--
obj_name_missing EXCEPTION;
PRAGMA EXCEPTION_INIT(obj_name_missing, -20002);
--
--
contact_notfound EXCEPTION;
PRAGMA EXCEPTION_INIT(contact_notfound, -20000);
--
--
duplicate_found EXCEPTION;
PRAGMA EXCEPTION_INIT(duplicate_found, -20001);
--
-- POOLED INSTANCE EXISTS - DUPLICATE
CURSOR POOLED_EXIST_CUR(
p_obj_name ii_pooled.obj_name%type ,
p_id ii_pooled.id%type ,
p_contact_id ii_contacts.id%type )
IS
SELECT '1'
FROM II_POOLEDS p, ii_contacts cnt
WHERE substr(cnt.id,1,4) = p_contact_id
AND p.admin_org_cn = cnt.cn
AND p.OBJ_NAME = P_OBJ_NAME
AND p.ID = P_ID;
--
-- FETCH CONTACT ID FOR ADMIN ORG CN.
CURSOR CONTACT_ID_CUR(
search_admin_org_cn ii_contacts.cn%type )
IS
SELECT id
FROM ii_contacts
WHERE cn = search_admin_org_cn;

```

```
BEGIN
BEGIN
IF INSERTING THEN
if :new.obj_name is null then

-- raise obj_name_missing exception;
raise_application_error(-20002,
'Pooled asset Type is mandatory.');
```

```
elseif :new.id is null then

-- raise id_missing exception;
raise_application_error(-20003,
'Pooled asset ID is mandatory.');
```

Oracle Designer Report : cktrgger

#### Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

II\_POOLEDS (table)

```
elseif :new.admin_org_cn is null then

-- raise admin_org_missing exception;
raise_application_error(-20004,
'Pooled asset Admin Org is mandatory.');
```

```
end if;
```

```
-- initialize instance values for cursors
```

```
p_id := :new.id;
p_obj_name := :new.obj_name;
p_admin_org_cn := :new.admin_org_cn;
```

```
open contact_id_cur(p_admin_org_cn);
fetch contact_id_cur into v_admin_org_id;
if contact_id_cur%notfound then
close contact_id_cur;
```

```

-- raise contact_notfound exception;
raise_application_error(-20000,
'Contact ID was not found.');
```

else

```

-- initialize
v_admin_org_id := substr(v_admin_org_id,1,4);
close contact_id_cur;
```

end if;

```

-- check for duplicates

OPEN pooled_exist_cur(p_obj_name, p_id, v_admin_org_id);
FETCH pooled_exist_cur INTO v_tmp_hold;
```

IF pooled\_exist\_cur%FOUND THEN

```

CLOSE pooled_exist_cur;
-- raise duplicate_found exception;
raise_application_error(-20001,
'Duplicate Pooled Asset Found with Type ||
p_obj_name||, ID ||p_id||, and Contact ID ||
v_admin_org_id||.');
```

ELSE

```

/* Instance does not exist */
CLOSE pooled_exist_cur;
END IF;
```

IF :new.cn IS NULL THEN

```

:new.cn := db_instance.generate_global_id('II_POOLED_CN_SEQ');
END IF;
db_instance.insert_audit_columns
(:new.created_by, :new.created_in_instance, :new.created_date);
```

Oracle Designer Report : cktrgger

#### Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

```

:
      II_POOLEDS (table)
ELSE
db_instance.update_audit_columns
(:new.modified_by, :new.modified_in_instance, :new.modified_date);
END IF;
END;
EXCEPTION
WHEN duplicate_found THEN
RAISE;
WHEN admin_org_missing THEN
RAISE;
WHEN id_missing THEN
RAISE;
WHEN obj_name_missing THEN
RAISE;
WHEN contact_notfound THEN
RAISE;

END;

```

Oracle Designer Report : cktrgger

Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

II\_POOLEDS (table)

Trigger Name

:

II\_POOLED\_RPA\_AR\_TRG

Create?

Yes

PL/SQL Module

Name :

II\_POOLED\_RPA\_AR

Enabled?

Yes

PL/SQL File :

Purpose :

Maintain RPA functions when AFTER ROW

Trigger Time:

AFTER

Level:

ROW

Insert?

Yes

Delete?

Yes

New Alias

Update?

Yes

Old Alias :

Trigger Logic

DECLARE

--

--

v\_tmp\_hold CHAR(1);

--

--

v\_admin\_org\_id II\_CONTACTS.ID%TYPE;

--

--

p\_id II\_POOLEDS.ID%TYPE;

--

--

p\_admin\_org\_cn II\_POOLEDS.ADMIN\_ORG\_CN%TYPE;

--

--

p\_obj\_name II\_POOLEDS.OBJ\_NAME%TYPE;

--

--

admin\_org\_missing EXCEPTION;

PRAGMA EXCEPTION\_INIT(admin\_org\_missing, -20004);

--

--

id\_missing EXCEPTION;

PRAGMA EXCEPTION\_INIT(id\_missing, -20003);

--

--

obj\_name\_missing EXCEPTION;

PRAGMA EXCEPTION\_INIT(obj\_name\_missing, -20002);

--

--

contact\_notfound EXCEPTION;

PRAGMA EXCEPTION\_INIT(contact\_notfound, -20000);

```

--
--
duplicate_found EXCEPTION;
PRAGMA EXCEPTION_INIT(duplicate_found, -20001);
--
-- POOLED INSTANCE EXISTS - DUPLICATE
CURSOR POOLED_EXIST_CUR(
p_obj_name ii_pooled.obj_name%type ,
p_id ii_pooled.id%type ,
p_contact_id ii_contacts.id%type )
Oracle Designer Report : cktrgger

```

## Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

II\_POOLED (table)

IS

```

SELECT '1'
FROM II_POOLED p, ii_contacts cnt
WHERE substr(cnt.id,1,4) = p_contact_id
AND p.admin_org_cn = cnt.cn
AND p.OBJ_NAME = P_OBJ_NAME
AND p.ID = P_ID;

```

--

-- FETCH CONTACT ID FOR ADMIN ORG CN.

```

CURSOR CONTACT_ID_CUR(
search_admin_org_cn ii_contacts.cn%type )

```

IS

```

SELECT id
FROM ii_contacts
WHERE cn = search_admin_org_cn;

```

BEGIN

BEGIN

IF INSERTING THEN

if :new.obj\_name is null then

```

-- raise obj_name_missing exception;
raise_application_error(-20002,
'Pooled asset Type is mandatory.');
```



```

elsif :new.id is null then

-- raise id_missing exception;
raise_application_error(-20003,
'Pooled asset ID is mandatory.');
```

```

elsif :new.admin_org_cn is null then

-- raise admin_org_missing exception;
raise_application_error(-20004,
'Pooled asset Admin Org is mandatory.');
```

```

end if;

-- initialize instance values for cursors

p_id := :new.id;
p_obj_name := :new.obj_name;
p_admin_org_cn := :new.admin_org_cn;

open contact_id_cur(p_admin_org_cn);
fetch contact_id_cur into v_admin_org_id;
if contact_id_cur%notfound then
close contact_id_cur;

-- raise contact_notfound exception;
raise_application_error(-20000,
'Contact ID was not found.');
```

```

else
-- initalize
v_admin_org_id := substr(v_admin_org_id,1,4);
close contact_id_cur;

```

Oracle Designer Report : cktrgger

#### Database Trigger Definition

```

Container:
    II_RPA
    Version :
Table/View Name
:
    II_POOLEDS (table)

```

```

end if;

-- check for duplicates

OPEN pooled_exist_cur(p_obj_name, p_id, v_admin_org_id);
FETCH pooled_exist_cur INTO v_tmp_hold;

IF pooled_exist_cur%FOUND THEN

CLOSE pooled_exist_cur;
-- raise duplicate_found exception;
raise_application_error(-20001,
'Duplicate Pooled Asset Found with Type ||
p_obj_name||, ID ||p_id||, and Contact ID ||
v_admin_org_id||');

ELSE
/* Instance does not exist */
CLOSE pooled_exist_cur;
END IF;

IF :new.cn IS NULL THEN
:new.cn := db_instance.generate_global_id('II_POOLED_CN_SEQ');
END IF;
db_instance.insert_audit_columns
(:new.created_by, :new.created_in_instance, :new.created_date);
ELSE
db_instance.update_audit_columns
(:new.modified_by, :new.modified_in_instance, :new.modified_date);
END IF;
END;
EXCEPTION
WHEN duplicate_found THEN
RAISE;
WHEN admin_org_missing THEN
RAISE;
WHEN id_missing THEN
RAISE;
WHEN obj_name_missing THEN
RAISE;
WHEN contact_notfound THEN
RAISE;

```

```

END;
-- DELETING
IF DELETING THEN
DELETE FROM ii_rpa_assets_v_tab
WHERE
pld_cn = :OLD.cn;

-- INSERTING
ELSIF INSERTING THEN
INSERT INTO ii_rpa_assets_v_tab (
Oracle Designer Report : cktrgger

```

Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

II\_POOLEDS (table)

```

FEA_CN
,LU_CN
,PLD_CN
,OBJECT_NAME
,OBJECT_CLASS
,ID
,NAME
,SUB_TYPE
,OWNERSHIP
,ADMIN_ORG
,DEVELOPMENT_STATUS
,CREATED_DATE
,CREATED_BY
,SUBLEDGER_FLAG
,subledger_cn
,agency_code
) VALUES (
NULL
,NULL
, :NEW.CN
, :NEW.OBJ_NAME
, :NEW.OBJ_CLASS
, :NEW.ID
, :NEW.NAME

```

```

,NULL
,NULL
,:NEW.ADMIN_ORG
,NULL
,:NEW.CREATED_DATE
,:NEW.CREATED_BY
,'N'
,null
,:NEW.agency_code -- Added on 10/28/2003 by C.Ma
);
-- UPDATING
ELSIF UPDATING THEN
-- Update table ii_rpa_assets_v_tab
UPDATE ii_rpa_assets_v_tab
SET OBJECT_NAME = :NEW.OBJ_NAME
,OBJECT_CLASS = :NEW.OBJ_CLASS
,ID = :NEW.ID
,NAME = :NEW.NAME
,ADMIN_ORG = :NEW.ADMIN_ORG
WHERE
pld_cn = :NEW.cn;
-- Update table ii_rpa_subledgers
UPDATE ii_rpa_subledgers sub
SET admin_org = :new.admin_org
WHERE sub.pld_cn = :new.cn
AND ii_sql_func.super_equal(admin_org, :new.admin_org) <> 'TRUE';
-----
-- Audit
-----
DECLARE
vNewPld ii_pooled%rowtype;

```

Oracle Designer Report : cktrgger

#### Database Trigger Definition

Container:

II\_RPA

Version :

Table/View Name

:

II\_POOLED (table)

vOldPld ii\_pooled%rowtype;

BEGIN

```

--vOldPld.GSA_INSTALLATION_CN := :old.GSA_INSTALLATION_CN;-- commented
out on 10/28/2003 by C.Ma

```

```

vOldPld.CN := :old.CN;
vOldPld.OBJ_CLASS := :old.OBJ_CLASS;
vOldPld.OBJ_TECH := :old.OBJ_TECH;
vOldPld.OBJ_NAME := :old.OBJ_NAME;
vOldPld.ID := :old.ID;
vOldPld.ADMIN_ORG := :old.ADMIN_ORG;
vOldPld.NAME := :old.NAME;
vOldPld.MASTER_SITE := :old.MASTER_SITE;
vOldPld.REMARKS := :old.REMARKS;
vOldPld.CREATED_BY := :old.CREATED_BY;
vOldPld.CREATED_DATE := :old.CREATED_DATE;
vOldPld.CREATED_IN_INSTANCE := :old.CREATED_IN_INSTANCE;
vOldPld.MODIFIED_BY := :old.MODIFIED_BY;
vOldPld.MODIFIED_DATE := :old.MODIFIED_DATE;
vOldPld.MODIFIED_IN_INSTANCE := :old.MODIFIED_IN_INSTANCE;

--vNewPld.GSA_INSTALLATION_CN := :new.GSA_INSTALLATION_CN;-- commented
out on 10/28/2003 by C.Ma
vNewPld.CN := :new.CN;
vNewPld.OBJ_CLASS := :new.OBJ_CLASS;
vNewPld.OBJ_TECH := :new.OBJ_TECH;
vNewPld.OBJ_NAME := :new.OBJ_NAME;
vNewPld.ID := :new.ID;
vNewPld.ADMIN_ORG := :new.ADMIN_ORG;
vNewPld.NAME := :new.NAME;
vNewPld.MASTER_SITE := :new.MASTER_SITE;
vNewPld.REMARKS := :new.REMARKS;
vNewPld.CREATED_BY := :new.CREATED_BY;
vNewPld.CREATED_DATE := :new.CREATED_DATE;
vNewPld.CREATED_IN_INSTANCE := :new.CREATED_IN_INSTANCE;
vNewPld.MODIFIED_BY := :new.MODIFIED_BY;
vNewPld.MODIFIED_DATE := :new.MODIFIED_DATE;
vNewPld.MODIFIED_IN_INSTANCE := :new.MODIFIED_IN_INSTANCE;

ii_rpa_auditing.audit_pld('U', :new.modified_by, :new.modified_date,
vOldPld, vNewPld);
END;
END IF;

```

Oracle Designer Report : cktrgger

Oracle Designer  
DATABASE TRIGGER REPORT  
End of Report