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

-- Author: A. Gierana

-- Last Modified:

-- Description: MMD all New DataBase Requirements as Tables, Triggers, Sequence and Pakages

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

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

-- Create XXREF\_RELAY\_COMP\_SET\_POINTS Table <<<<

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

create table XREF\_RELAY\_COMP\_SET\_POINTS

(

XREF\_RELAY\_COMP\_SP\_ID NUMBER (10,0),

COMPONENT\_ID NUMBER (10,0) NOT NULL ENABLE,

SET\_POINT\_VALUE NUMBER (10,3),

ACTION VARCHAR2(25 CHAR),

UNIT\_ID NUMBER (10,0) NOT NULL ENABLE,

COMP\_AS\_LEFT NUMBER (1,0) DEFAULT 0 NOT NULL ENABLE,

COMP\_AS\_FOUND NUMBER (1,0) DEFAULT 0 NOT NULL ENABLE,

COMP\_LOW\_TOL NUMBER (10,3),

COMP\_HIGHT\_TOL NUMBER (10,3),

COMP\_DRIFT\_LIMIT NUMBER (10,3),

SPAN NUMBER (10,3),

DESCRIPTION VARCHAR2(4000 CHAR),

DELETE\_IND NUMBER(1,0) DEFAULT 0 NOT NULL ENABLE,

POSITION NUMBER(2,0) NOT NULL ENABLE,

LAST\_MODIFIED\_BY\_ID NUMBER(10,0),

LAST\_MODIFIED\_DATE DATE,

CONSTRAINT FK\_XREF\_RELA\_C\_SP\_USER\_ID FOREIGN KEY (LAST\_MODIFIED\_BY\_ID) REFERENCES TBL\_USERS (USER\_ID) ENABLE,

CONSTRAINT XREF\_RELAY\_COMP\_SP\_PK PRIMARY KEY (XREF\_RELAY\_COMP\_SP\_ID),

CONSTRAINT FK\_XREF\_RELAY\_COMP\_SP\_ID FOREIGN KEY (COMPONENT\_ID) REFERENCES TBL\_COMPONENTS (COMPONENT\_ID) ENABLE

);

/

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

-- Create LU\_PROCEDURE\_FREQUENCY <<<<

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

CREATE TABLE "LU\_PROCEDURE\_FREQUENCY"

(LU\_FREQUENCY\_ID NUMBER(10,0) NOT NULL ENABLE,

PROCEDURE\_ID NUMBER(10,0),

FREQUENCY\_DESC VARCHAR2(25 CHAR),

CONSTRAINT "LU\_PROCEDURE\_FREQUENCY\_PK" PRIMARY KEY ("LU\_FREQUENCY\_ID"),

CONSTRAINT "FK\_PROCEDURE\_ID" FOREIGN KEY ("PROCEDURE\_ID") REFERENCES "TBL\_PROCEDURES" ("PROCEDURE\_ID") ENABLE

);

/

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

-- Create LU\_RELAY\_ACTION <<<<

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

CREATE TABLE "LU\_RELAY\_ACTION"

(LU\_RELAY\_ACTION\_ID NUMBER(10,0) NOT NULL ENABLE,

SITE\_ID NUMBER(10,0) NOT NULL ENABLE,

UNITS\_ID NUMBER(10,0) NOT NULL ENABLE,

ACTION\_DESCRIPTION VARCHAR2(50 CHAR) NOT NULL ENABLE,

DELETED NUMBER(1,0),

CONSTRAINT "LU\_RELAY\_ACTION\_PK" PRIMARY KEY ("LU\_RELAY\_ACTION\_ID"),

CONSTRAINT "FK\_SITE\_ID" FOREIGN KEY ("SITE\_ID")

REFERENCES "LU\_SITES" ("SITE\_ID") ENABLE,

CONSTRAINT "FK\_UNIT\_ID" FOREIGN KEY ("UNITS\_ID")

REFERENCES "TBL\_UNITS" ("UNIT\_ID") ENABLE

);

/

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

-- Create TBL\_RELAY\_IDENTIFIER <<<<

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

CREATE TABLE "TBL\_RELAY\_IDENTIFIER"

(RELAY\_IDENTIFIER\_ID NUMBER(10,0) NOT NULL ENABLE,

COMPONENT\_TYPE\_SETPOINT\_ID NUMBER(10,0) NOT NULL ENABLE, RELAY\_IDENTIFIER\_DESC VARCHAR2(50 CHAR),

CONSTRAINT "TBL\_RELAY\_IDENTIFIER\_PK" PRIMARY KEY ("RELAY\_IDENTIFIER\_ID"),

CONSTRAINT "FK\_COMPONENT\_TYPE\_SETPOINT\_ID" FOREIGN KEY ("COMPONENT\_TYPE\_SETPOINT\_ID")

REFERENCES "TBL\_COMPONENT\_TYPE\_SETPOINT" ("COMPONENT\_TYPE\_SETPOINT\_ID") ENABLE

);

/

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

-- Create XREF\_PROC\_COMP\_SET\_POINT <<<<

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

CREATE TABLE “XREF\_PROC\_COMP\_SET\_POINT"

( XREF\_PROC\_COMP\_SET\_POINT\_ID NUMBER(10,0) NOT NULL ENABLE,

XREF\_COMPONENT\_SET\_POINT\_ID NUMBER(10,0) NOT NULL ENABLE,

XREF\_PROCEDURE\_COMPONENT\_ID NUMBER(10,0),

DELETE\_IND NUMBER(1,0),

CONSTRAINT "XREF\_PROC\_COMP\_SET\_POINT\_PK" PRIMARY KEY ("XREF\_PROC\_COMP\_SET\_POINT\_ID"),

CONSTRAINT "FK\_XREF\_COMPNENT\_SET\_POINT\_ID" FOREIGN KEY ("XREF\_COMPONENT\_SET\_POINT\_ID")

REFERENCES XREF\_COMPONENT\_SET\_POINTS" ("XREF\_COMPONENT\_SET\_POINTS\_ID") ENABLE,

CONSTRAINT "FK\_XREF\_PROCEDURE\_COMPONENT\_ID" FOREIGN KEY ("XREF\_PROCEDURE\_COMPONENT\_ID")

REFERENCES

“XREF\_PROCEDURE\_COMPONENT" ("XREF\_PROCEDURE\_COMPONENT\_ID") ENABLE

); /

/\* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \*/

/\* create sequences and triggers \*/

/\* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \*/

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

-- Create SEQ\_LU\_PROCEDURE\_FREQUENCY\_ID Sequence <<<<

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

CREATE SEQUENCE SEQ\_LU\_PROCEDURE\_FREQUENCY\_ID

INCREMENT BY 1

MAXVALUE 9999999999999999999999999999

MINVALUE 1

CYCLE

CACHE 2

ORDER;

NOCACHE;

/

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

-- Create LU\_PROCEDURE\_FREQUENCY Trigger <<<<

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

CREATE OR REPLACE TRIGGER "LU\_PROCEDURE\_FREQUENCY"

BEFORE INSERT ON LU\_PROCEDURE\_FREQUENCY REFERENCING

NEW AS NEW OLD AS OLD

FOR EACH row BEGIN

IF (:new.LU\_FREQUENCY\_ID IS NULL) THEN

SELECT SEQ\_TBL\_RELAY\_IDENTIFIER\_ID.NEXTVAL INTO :new.LU\_FREQUENCY\_ID FROM dual;

END IF;

END;

/

ALTER TRIGGER "LU\_PROCEDURE\_FREQUENCY" ENABLE;

/

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

-- Create SEQ\_LU\_RELAY\_ACTION\_ID Sequence <<<<

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

CREATE SEQUENCE SEQ\_LU\_RELAY\_ACTION\_ID

INCREMENT BY 1

MAXVALUE 999999999

MINVALUE 1

CYCLE

CACHE 20

ORDER;

/

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

-- Create TR\_LU\_RELAY\_ACTION\_ID Trigger <<<<

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

CREATE OR REPLACE TRIGGER "TR\_LU\_RELAY\_ACTION\_ID"

BEFORE INSERT ON LU\_RELAY\_ACTION REFERENCING

NEW AS NEW OLD AS OLD

FOR EACH row BEGIN

IF (:new.LU\_RELAY\_ACTION\_ID IS NULL) THEN

SELECT SEQ\_LU\_RELAY\_ACTION\_ID.NEXTVAL INTO :new.LU\_RELAY\_ACTION\_ID FROM dual;

END IF;

END;

/

ALTER TRIGGER "TR\_LU\_RELAY\_ACTION\_ID" ENABLE;

/

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

-- Create SEQ\_TBL\_RELAY\_IDENTIFIER\_ID Sequence <<<<

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

CREATE SEQUENCE SEQ\_TBL\_RELAY\_IDENTIFIER\_ID

MINVALUE 1 MAXVALUE 9999999999999999999999999999

INCREMENT BY 1

START WITH 21

CACHE 20

ORDER

CYCLE ;

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

-- Create TR\_TBL\_RELAY\_IDENTIFIER\_ID Trigger <<<<

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

CREATE OR REPLACE TRIGGER "TR\_TBL\_RELAY\_IDENTIFIER\_ID"

BEFORE INSERT ON TBL\_RELAY\_IDENTIFIER REFERENCING

NEW AS NEW OLD AS OLD

FOR EACH row BEGIN

IF (:new.RELAY\_IDENTIFIER\_ID IS NULL) THEN

SELECT SEQ\_TBL\_RELAY\_IDENTIFIER\_ID.NEXTVAL INTO :new.RELAY\_IDENTIFIER\_ID FROM dual;

END IF;

END;

/

ALTER TRIGGER "TR\_TBL\_RELAY\_IDENTIFIER\_ID" ENABLE;

/

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

-- Create SEQ\_XREF\_RELAY\_COMP\_SP\_ID Sequence <<<<

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

CREATE SEQUENCE SEQ\_XREF\_RELAY\_COMP\_SP\_ID

MINVALUE 1

MAXVALUE 999999999

INCREMENT BY 1

START WITH 1

NOCACHE

ORDER

CYCLE ;

/

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

-- Create TR\_XREF\_RELAY\_COMP\_SP\_ID Trigger <<<<

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

CREATE OR REPLACE TRIGGER "TR\_XREF\_RELAY\_COMP\_SP\_ID"

BEFORE INSERT ON XREF\_RELAY\_COMP\_SET\_POINTS REFERENCING

NEW AS NEW OLD AS OLD

FOR EACH row BEGIN

IF (:new.XREF\_RELAY\_COMP\_SP\_ID IS NULL) THEN

SELECT SEQ\_XREF\_RELAY\_COMP\_SP\_ID.NEXTVAL INTO :new.XREF\_RELAY\_COMP\_SP\_ID FROM dual;

END IF;

END;

/

ALTER TRIGGER "TR\_XREF\_RELAY\_COMP\_SP\_ID" ENABLE;

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

-- Create TR\_TBL\_RELAY\_IDENTIFIER\_ID Trigger <<<<

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

CREATE OR REPLACE TRIGGER "TR\_XREF\_PROC\_COMP\_SET\_POINT\_ID"

BEFORE INSERT ON XREF\_PROC\_COMP\_SET\_POINT REFERENCING

NEW AS NEW OLD AS OLD

FOR EACH row BEGIN

IF (:new.XREF\_PROC\_COMP\_SET\_POINT\_ID IS NULL) THEN

SELECT SEQ\_XREF\_PROC\_COMP\_SET\_POINT.NEXTVAL INTO :new.XREF\_PROC\_COMP\_SET\_POINT\_ID FROM dual;

END IF;

END;

/

ALTER TRIGGER "TR\_XREF\_PROC\_COMP\_SET\_POINT\_ID" ENABLE;

/

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

-- Create TR\_TBL\_RELAY\_IDENTIFIER\_ID Trigger <<<<

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

CREATE OR REPLACE TRIGGER "TR\_XREF\_PROC\_COMP\_SET\_POINT\_ID"

BEFORE INSERT ON XREF\_PROC\_COMP\_SET\_POINT REFERENCING

NEW AS NEW OLD AS OLD

FOR EACH row BEGIN

IF (:new.XREF\_PROC\_COMP\_SET\_POINT\_ID IS NULL) THEN

SELECT SEQ\_XREF\_PROC\_COMP\_SET\_POINT.NEXTVAL INTO :new.XREF\_PROC\_COMP\_SET\_POINT\_ID FROM dual;

END IF;

END;

/

ALTER TRIGGER "TR\_XREF\_PROC\_COMP\_SET\_POINT\_ID" ENABLE;

/

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

-- Create SEQ\_XREF\_PROC\_COMP\_SET\_POINT Sequence <<<<

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

MINVALUE 1

MAXVALUE 999999999

INCREMENT BY 1

START WITH 1

NOCACHE

ORDER

CYCLE ;