

Reconversión Monetaria 2021

Introducción

Para el año 2021 en la historia de Venezuela se llevaría a cabo la tercera reconversión implementada bajo decreto No. 4.553, mediante el cual el Ejecutivo Nacional decretó la nueva expresión monetaria. Dicho decreto fue publicado en la Gaceta Oficial No. 42.185 del 6 de agosto de 2021.

El siguiente procedimiento intenta describir de manera general los pasos a seguir para lograr aplicar una función matemática y eliminar seis (6) ceros a campos numéricos en tablas de dos tipos de bases de datos distintos, una de ellas PostgreSQL y la otra Microsoft SQL Server. Cabe destacar que a través del uso del lenguaje de programación Python, fue la estrategia ideal a implementar, para aplicar una función matemática a ambas bases de datos.

Este procedimiento más el documento fue escrito y desarrollado por mi persona, quien estuvo a cargo además del desarrollo del programa en python para actualizar los valores en campos numéricos de ambas bases de datos SQL Server y PostgreSQL.

Procedimiento para reconvertir las bases de datos de SQL Server

Base de Datos PostgreSQL

Paso 0. (Preparación)

- Crear copia de seguridad de la base de datos
- Crear un duplicado de la base de datos sanos : `CREATE DATABASE sanos_bs WITH TEMPLATE sanos OWNER postgres;` (Opcional)
- Dar permiso al usuario `reconversion_user` en `pg_hba.conf`
- Ejecutar el programa de reconversión (python) con privilegios root
- Editar y colocar el nombre de la base de datos en el programa python

Paso 1. Crear función `sp_sanosDiccionarios` en la base de datos PostgreSQL

La función analiza la metadata, tablas y campos de la base de datos, realizando inicialmente sobre las tablas un conteo del número de registros, excluyendo las tablas sin registro, luego hará un análisis sobre los campos de tipo numérico, creando una tabla con estos campos, llamada `<nombre_tabla>_diccionario`, este store procedure tiene la función de excluir tablas y campos.

Reconversión Monetaria 2021

```
CREATE OR REPLACE function sp_sanosDiccionarios (valor INT)
returns void
language plpgsql AS
$$
DECLARE
registros INT = 0;
entidad VARCHAR(100) := NULL;
comando VARCHAR(1000) := NULL;
tupla RECORD;
columna RECORD;
-- DECLARE tabla CURSOR FOR
-- SELECT table_schema,table_name
-- FROM information_schema.tables
-- WHERE table_schema NOT IN ('pg_catalog','information_schema') and table_type = 'BASE TABLE'
-- AND table_name NOT IN ('inventario_articulo_ambulatorio', 'aps_cliente_especialidad')
-- ORDER BY table_schema,table_name;
-- BEGIN
DROP TABLE IF EXISTS sanos_diccionario;
CREATE TABLE sanos_diccionario (esquema varchar(50), nombretabla varchar(50), nombrecampo varchar(50));
END
```

Paso 2. Ejecutar la función : SELECT sp_sanosDiccionarios (1);

Paso 3. Verificar la creación de la tabla :

```
SELECT * FROM sanos_diccionario;
```

Paso 4. Otorgar permisos al usuario **reconversion_user** sobre toda la base de datos

Paso 5. Desactivar triggers

```
ALTER TABLE presupuesto.baremo_detalle DISABLE TRIGGER trg_biu_baremo_detalle_hm;
ALTER TABLE inventario_detalle_movimiento DISABLE TRIGGER trg_bidu_invdetmovimiento;
```

Paso 6. Ejecutar el programa **reconvertirSanos.py** para iniciar el proceso de reconversión. Desde un terminal de linux y usuario con privilegio root.

```
Archivo Editar Ver Buscar Terminal Ayuda
root@dbalx:/home/dba/Documentos/personal/laboratorio/python/reconversion/sanos# ./reconvertirSanos.py
```

Paso 7. Al culminar el proceso, se debe habilitar los triggers

```
ALTER TABLE presupuesto.baremo_detalle ENABLE TRIGGER trg_biu_baremo_detalle_hm;
ALTER TABLE inventario_detalle_movimiento ENABLE TRIGGER trg_bidu_invdetmovimiento;
```

Reconversión Monetaria 2021

Programa reconvertirSanos.py

```
1  #!/usr/bin/python3
2  import pandas as pd
3  from datetime import datetime
4
5  import psycopg2 #Postgres
6
7  #Definiciones
8  driver = 'psql'
9  usuario = 'reconversion_user'
10 credenciales = 'xxxxxxxxxx'
11 server = '34.206.176.180'
12 puerto = '5432'
13 dbname = db = 'sanos_test'
14
15
16 formula = 'ROUND({}/1000000,2)'
17 scriptsSQL = []
18
19 ruta = '/home/dba/xxxxxxxxxxxxxxxxxxxxxx/reconversion/sanos/'
20 diccionariocsv = ruta + 'diccionariosanos.csv'
21 updatescsv = ruta + 'updatessanos.csv'
22
23 select_distinct_diccionario = 'SELECT DISTINCT nombretabla,esquema FROM sanos_diccionario ORDER BY 2,1;'
24 select_campos_tablas = 'SELECT nombrecampo FROM sanos_diccionario WHERE nombretabla = \'{}\'' ORDER BY 1;'
25
26 connection = psycopg2.connect(user = usuario, password = credenciales, host = server, port = puerto, database = dbname)
27
28 def leer_tabla_diccionario (cursor):
29     #Ejecutar store procedure para crear la tabla Diccionario con los campos nombretabla, nombrecampo, espk
30     #cursor.execute(execute_storeprocedure)
31     #time.sleep(360)
32     #input('Teclee cualquier tecla para continuar ...')
33     cursor.execute(select_distinct_diccionario)
34     data = cursor.fetchall()
35     tablas = []
```

Elaborado : Ing. Jesús Parra
Barquisimeto, Venezuela

Reconversión Monetaria 2021

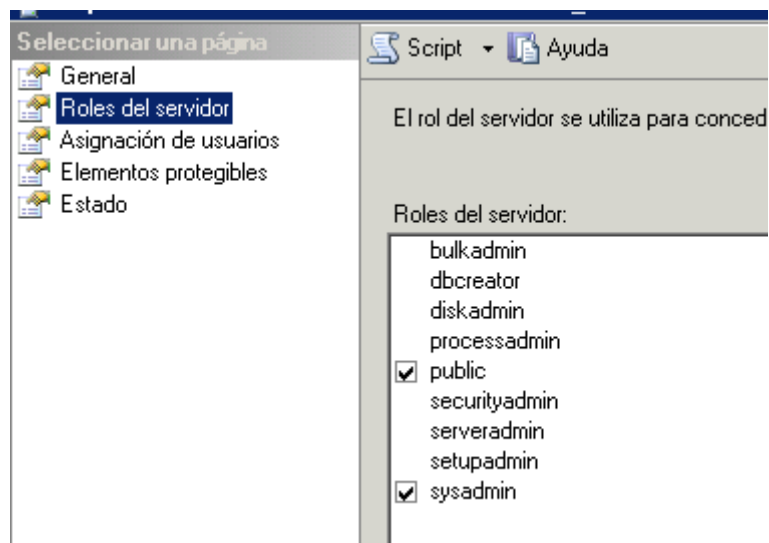
Base de Datos Microsoft SQL Server

Paso 0. (Preparación)

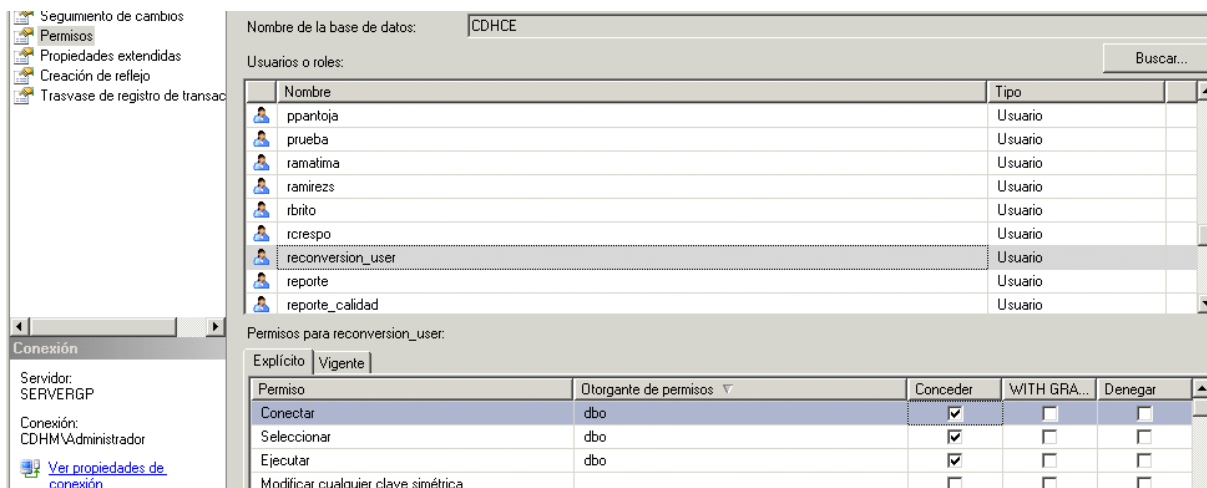
- Verificar que no existan las tablas **BDNDiccionario** y **CDHDiccionario** en la **tempdb** (Opcional)
- Crear copia de seguridad de la base de datos
- Dar permiso al usuario **reconversion_user** en **BDNBS | CDHBS**
- **Ejecutar el programa de reconversión (python) con privilegios root**
- **Editar y colocar el nombre de la base de datos en el programa python**

Asignación del usuario y roles de base de la base de datos

Asignación de roles de servidor



Asignación de permisos



Elaborado : Ing. Jesús Parra
Barquisimeto, Venezuela

Reconversión Monetaria 2021

Paso 1. Crear un procedimiento almacenado en la base de datos a reconvertir

El procedimiento almacenado analiza los objetos, tablas y campos de la base de datos, realizando inicialmente sobre las tablas un conteo del número de registros, excluyendo las tablas sin registro, luego hará un análisis sobre los campos de tipo numérico, creando una tabla solo con estos campos, llamada **BDNDiccionario** o **CDHDiccionario** en la base de datos **tempdb**, el procedimiento almacenado tiene la facilidad de excluir tablas y campos.

Paso 2. Ejecución del procedimiento almacenado **spBDNDiccionario**

```
DECLARE @t int = 0;  
EXECUTE spBDNDiccionario @t output;  
SELECT @t;
```

Paso 3. Ejecutar el programa python **reconvertirHigea.py** para iniciar el proceso de reconversión. Desde un terminal de linux y usuario con privilegio root.

```
dba@dba1x:~/Documentos/personal/laboratorio/python/reconversion/gp$ ./reconvertirFundacion.py
```

Elaborado : Ing. Jesús Parra
Barquisimeto, Venezuela