

Procedimiento para reconvertir bases de datos de Centro Salud Higea – Fundación

Base de Datos Sanos (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 una función en la base de datos a reconvertir (PostgreSQL)

La función, 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 numerico, creando una tabla solo con estos campos, llamada sanos_diccionario, el procedimiento almacenado tiene la facilidad de excluir tablas y campos.

```
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';
```

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

```
DO $$
DECLARE usuarios VARCHAR(100)[] := array ['reconversion_user'];
DECLARE usuario VARCHAR(100) := NULL;
DECLARE entidad_esquema VARCHAR(100) := NULL;
DECLARE nombre_esquema RECORD;
DECLARE esquema CURSOR FOR SELECT DISTINCT table_schema
FROM information schema.tables
WHERE table_schema NOT IN ('pg_catalog', 'information_schema') and table_type = 'BASE TABLE';
BEGIN
FOR login IN array_lower(usuarios, 1)..array_upper(usuarios, 1) LOOP
```

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;
```

Elaborado: Ing. Jesús Parra

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. Habilitar triggers

```
ALTER TABLE presupuesto.baremo_detalle ENABLE TRIGGER trg_biu_baremo_detalle hm;
ALTER TABLE inventario_detalle_movimiento ENABLE TRIGGER trg_bidu_invdetmovimiento;
```

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 = 'xxxxxxxxxxx'
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

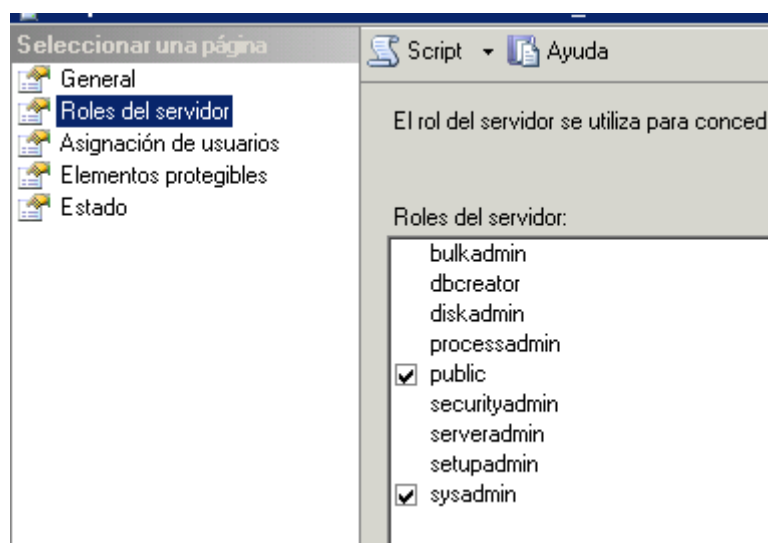
Base de Datos BDNBS | CDHBS (SQL Server)

Paso 0. (Preparación)

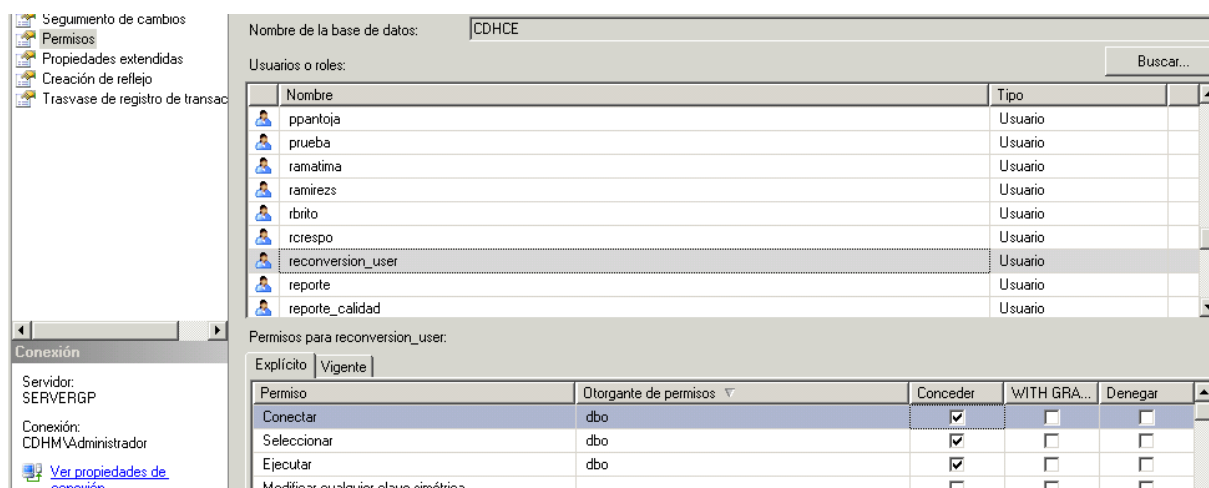
- Verificar que no existan las tablas **BDNDiccionario** y **CDHDiccionario** en la **tempdb** (Opcional)
- Crear copia de seguridad de la base de datos
- Dar permiso al usuario **reconversion_user** en **BDNBS | CDHBS**
- **Ejecutar el programa de reconversión (python) con privilegios root**
- **Editar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

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 numeric, 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.

```
USE [BDNCE]
GO
/***** Object: StoredProcedure [dbo].[spBDNDiccionario]    Script Date: 13/09/2021 16:27:51 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
ALTER PROCEDURE [dbo].[spBDNDiccionario] @atributos INT OUTPUT
AS
BEGIN
DECLARE @query nvarchar(100),
@parametros nvarchar(100),
@tabla AS nvarchar(50),
@campo AS nvarchar(50),
@sql AS nvarchar(4000),
@returnAsSelect int = 0,
@registros AS INT;
IF OBJECT_ID(N'tempdb..BDNDiccionario',N'U') IS NOT NULL
BEGIN
DROP TABLE tempdb..BDNDiccionario;
CREATE TABLE tempdb..BDNDiccionario (id int IDENTITY (1,1) NOT NULL,nombretabla varchar(50),nombrecampo varchar(50),espk int,
CONSTRAINT PK_BDNHDiccionario_id PRIMARY KEY CLUSTERED (id));
END;
END;
```

Paso 2. Ejecución del procedimiento almacenado spBDNDiccionario

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
```

Paso 3. Repetir los pasos para la siguiente base de datos a reconvertir

Programa reconvertirFundacion.py

