

# **M a n u a l**

---

## **Reconversión monetaria del Sistema Sigefirrh 2018**

---

---

### **Procedimiento para la implementación de la reconversión monetaria en el SIGEFIRRH**

## Objetivo principal:

Proveer a los órganos/ entes de la Administración Pública de una herramienta capaz de realizar el proceso de la reconversión monetaria del sistema SIGEFIRRH, basado en instrucciones SQL y comandos Debian Linux, sencillos amigables y fáciles de implementar, actualizando los datos de los campos relacionados a montos en Bolívares Soberanos de las tablas de la base de datos, lo cual implica un mínimo impacto.

## Realizar respaldo de la base de datos de Producción

### Pasos a seguir:

1. Conectarse al servidor de producción de la base de datos del SIGEFIRRH:

```
usuario: root  
password:*****
```

2. Conectarse al manejador de base de datos Postgresql con el usuarios postgres:

```
su - postgres
```

3. Realizar el respaldo de la base datos de producción:

```
pg_dump -d nombre_base_datos -f nombre_archivo `date '+%Y-%m-%d-%H%M-%S'` .sql
```

**NOTA:**

El formato `date '+%Y-%m-%d-%H%M-%S'` permite colocar la fecha de ejecución del respaldo; muestra: año-mes-día-horaminutos-segundos =>2018-05-04-1616\_53

**Ejemplo:**

```
pg_dump -d SIGEFIRRH -f bck_sigefirrh `date '+%Y-%m-%d-%H%M-%S'` .sql
```

Resultado que se obtiene:

```
bck_sigefirrh_2018-05-04-1056-53.sql
```

Se puede realizar el proceso de compresión de la información, con la siguiente instrucción:

```
tar cvf nombre_archivo_destino.tar nombre_archivo_fuente.sql
```

**Ejemplo:**

```
tar cvf bck_sigefirrh_2018-05-04-1056-53.tar bck_sigefirrh_2018-05-04-1056-53.sql
```

Resultado que se obtiene: **bck\_sigefirrh\_2018-05-04-1056-53.tar**

---

## **Pasar el respaldo de la base de datos de Producción a otro medio de almacenamiento**

### **Pasos a seguir:**

Es recomendable pasar el respaldo de la base de datos de producción a otros medios de almacenamiento, a su vez se debe dejar una copia en el servidor de producción y otras copias en CD, Pendrive, Cinta, servidor y/o computador, para evitar contratiempos a la hora de algún problema, a su vez se debe cumplir con las políticas de seguridad de cada órganos/ entes, creación de servidor espejo a los fines de realizar los respectivos paralelos.

*\*Como sugerencia se debe restaurar el resultado del respaldo (backup) en otro equipo con Postgresql, y verificar que el proceso se hizo correctamente, a los fines de prever cualquier inconveniente existente.*

---

## **Pasar al servidor de prueba donde está la base de datos la herramienta (script) de reconversión monetaria (script).**

### **Que se debe hacer:**

Se debe copiar el archivo de reconversión monetaria (**generar\_query.sql**), suministrado por Digesafsp al servidor de base de datos de prueba.

Se debe tener en cuenta que en el directorio donde te encuentres ejecutando este “**script**” este genera otro archivo (**reconversion\_monetaria.sql**), el cual ejecutaras posteriormente.

---

## **Ejecución de la herramienta tecnológica de reconversión monetaria.**

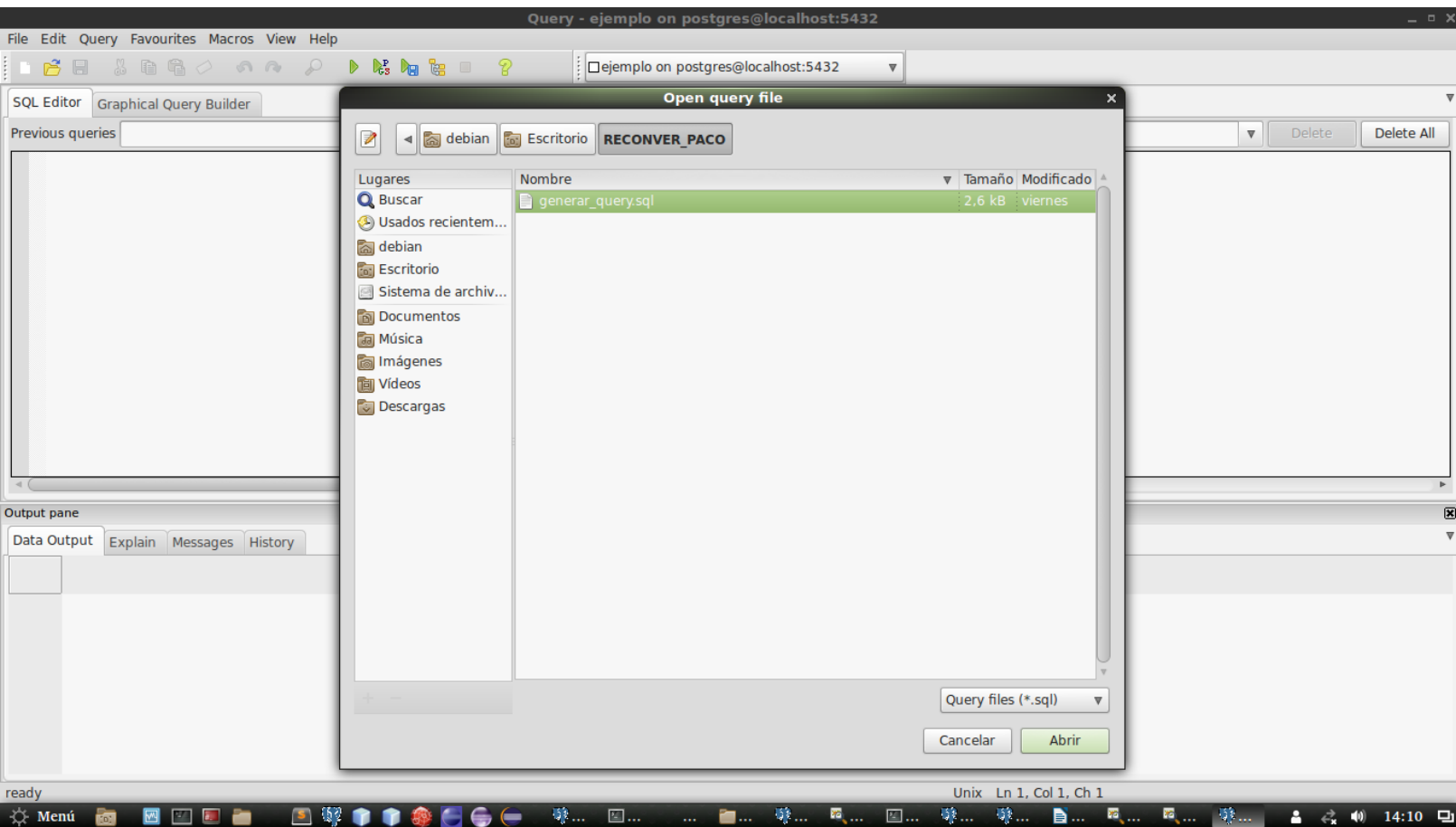
### **Ejecución del proceso “script”:**

Haciendo uso de las instrucciones SQL y comandos Debian Linux ejecutar en el servidor de prueba de base de datos, el archivo de reconversión monetaria (script) suministrado por Digesafsp.

Conectarse a PgAdmin, Base de Datos Prueba.

## PASO 1.-

Buscar el archivo (**generar\_query.sql**) enviado por Digesafsp para la realización de la Reconversión Monetaria.



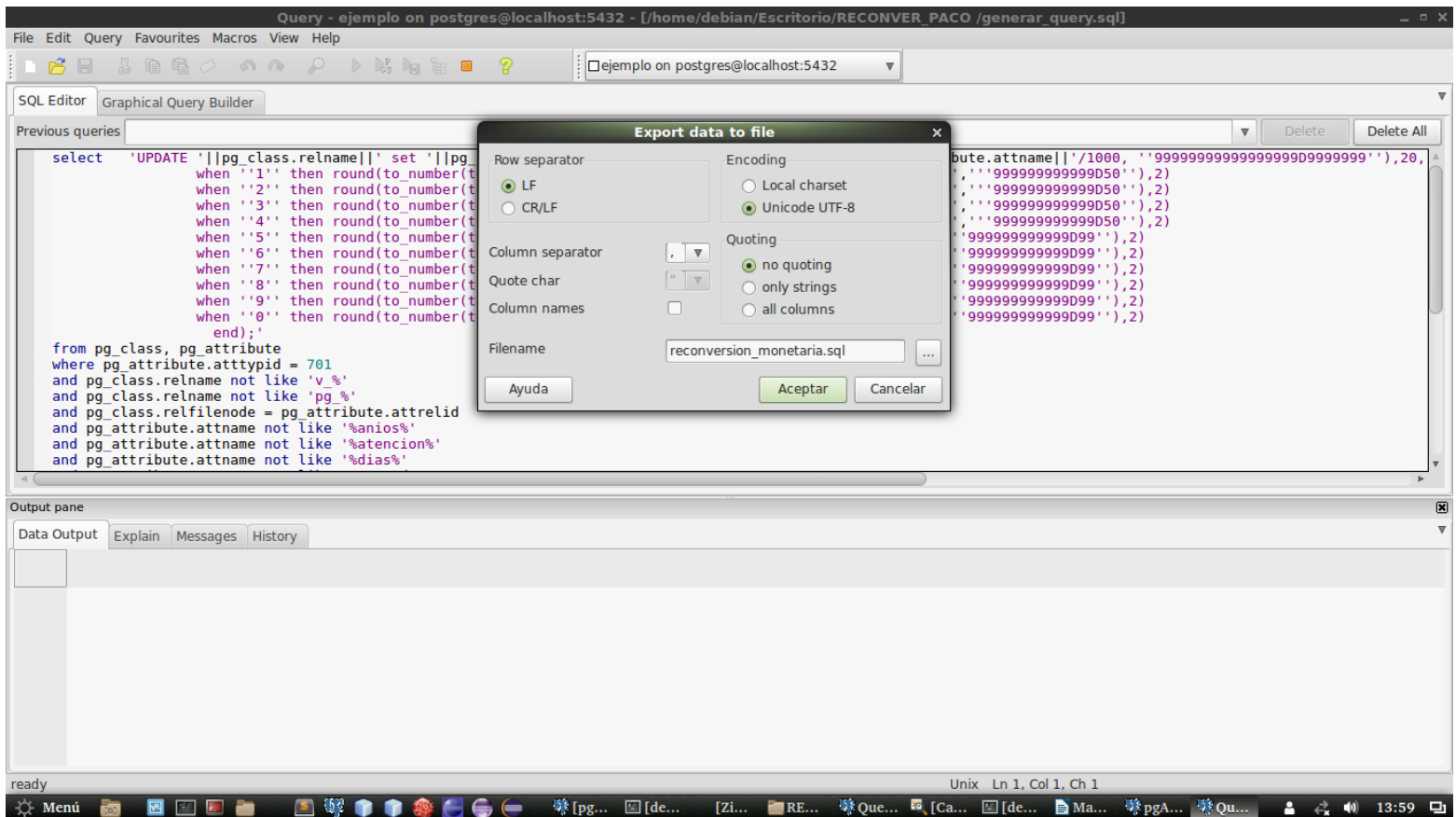
## PASO 2.-

Ejecutar el comando `generar_query.sql`

```
Query - ejemplo on postgres@localhost:5432 - [/home/debian/Escritorio/RECONVER_PACO /generar_query.sql]
File Edit Query Favourites Macros View Help
ejemplo on postgres@localhost:5432
Execute query, write result to file
SQL Editor Graphical Query Builder
Previous queries
select 'UPDATE '||pg_class.relname||' set '||pg_attribute.attname||'=(case substring(to_char('||pg_attribute.attname||'/1000, '9999999999999999D999999'),20,
when '1' then round(to_number(to_char('||pg_attribute.attname||'/1000, '999999999D50'), '9999999999999999D50'),2)
when '2' then round(to_number(to_char('||pg_attribute.attname||'/1000, '999999999D50'), '9999999999999999D50'),2)
when '3' then round(to_number(to_char('||pg_attribute.attname||'/1000, '999999999D50'), '9999999999999999D50'),2)
when '4' then round(to_number(to_char('||pg_attribute.attname||'/1000, '999999999D50'), '9999999999999999D50'),2)
when '5' then round(to_number(to_char('||pg_attribute.attname||'/1000, '999999999'), '9999999999999999D99'),2)
when '6' then round(to_number(to_char('||pg_attribute.attname||'/1000, '999999999'), '9999999999999999D99'),2)
when '7' then round(to_number(to_char('||pg_attribute.attname||'/1000, '999999999'), '9999999999999999D99'),2)
when '8' then round(to_number(to_char('||pg_attribute.attname||'/1000, '999999999'), '9999999999999999D99'),2)
when '9' then round(to_number(to_char('||pg_attribute.attname||'/1000, '999999999'), '9999999999999999D99'),2)
when '0' then round(to_number(to_char('||pg_attribute.attname||'/1000, '999999999'), '9999999999999999D99'),2)
end);'
from pg_class, pg_attribute
where pg_attribute.atttypid = 701
and pg_class.relname not like 'v_%'
and pg_class.relname not like 'pg_%'
and pg_class.relfilenode = pg_attribute.attrelid
and pg_attribute.attname not like '%anios%'
and pg_attribute.attname not like '%atencion%'
and pg_attribute.attname not like '%dias%'
Output pane
Data Output Explain Messages History
ready
Unix Ln 1, Col 1, Ch 1
Menú [p... [d... [Zi... RE... Qu... [Vi... [d... pg... Qu... Ca... 14:01
```

### PASO 3.-

Generación del archivo que contiene los comandos **Update**, según los siguientes parámetros.-

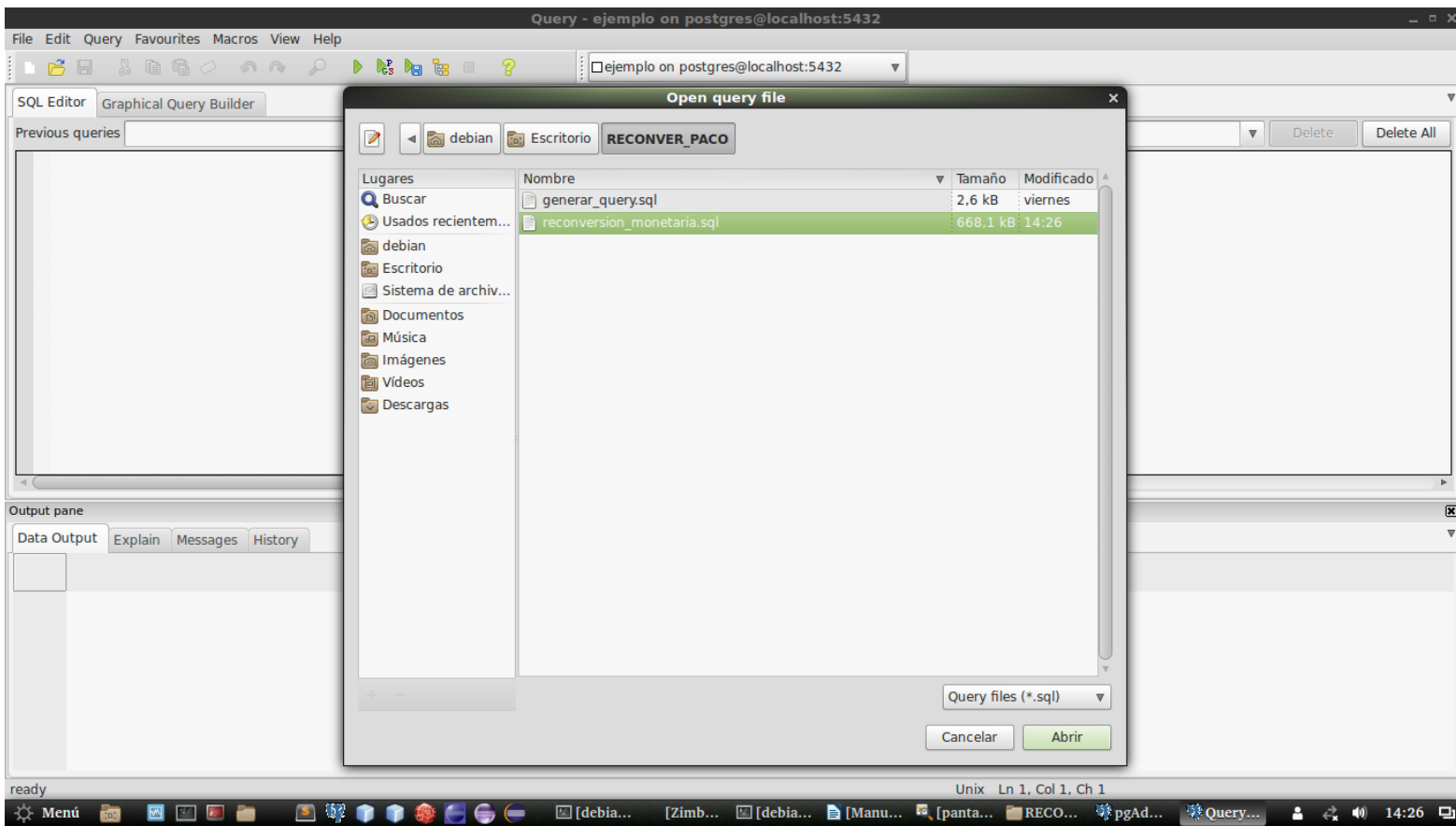


**Nota:** Esta opción genera un archivo llamado **“reconversión\_monetaria.sql”** que contiene todas las instrucciones update requeridas para la reconversión monetaria, es decir se debe ejecutar este archivo resultante de esta opción para realizar el proceso completo (paso4)

#### PASO 4 .-

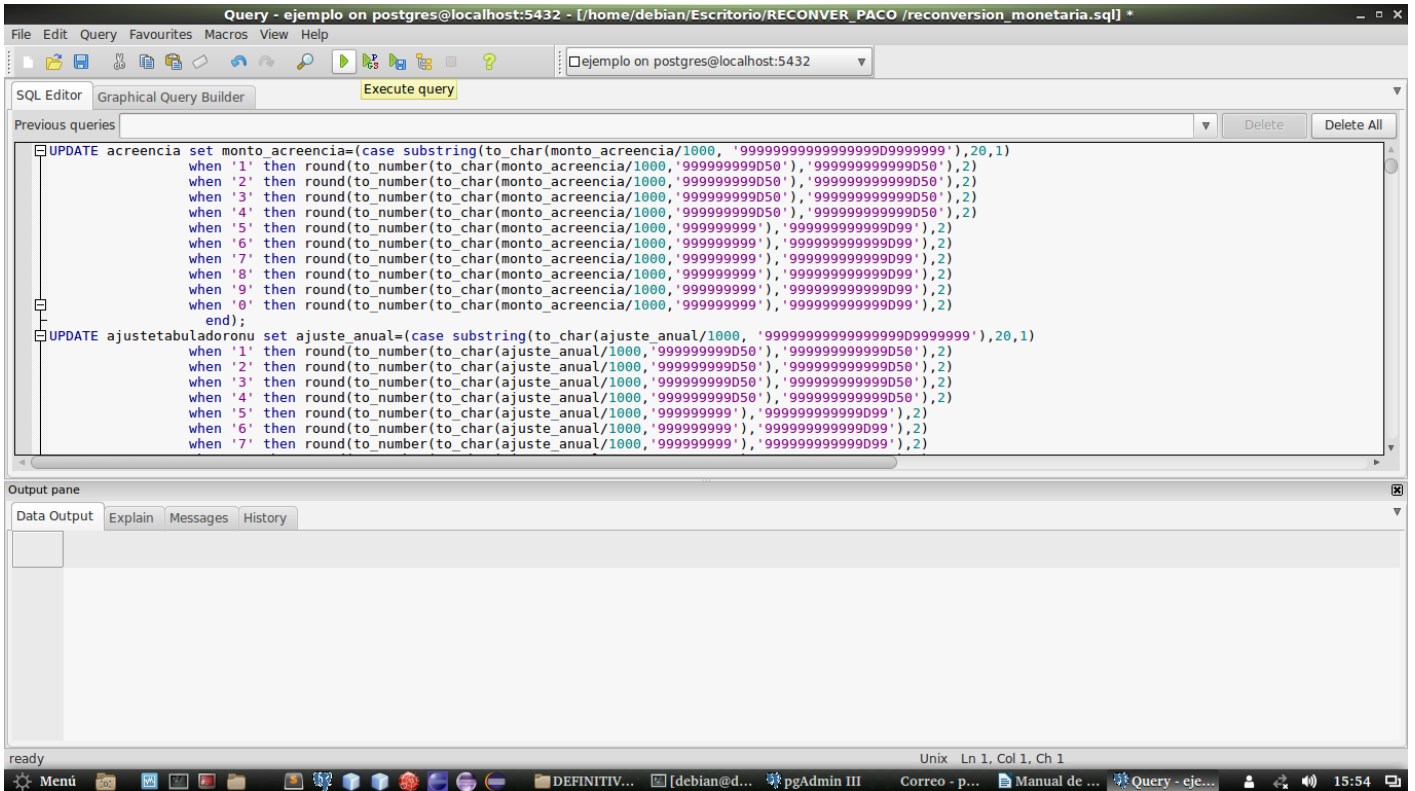
Esta opción ejecuta los comandos (**update**), necesarios para ejecutar la reconversión monetaria tomando en consideración todos los campos expresados en monto decimales para convertirlos en Bs. Soberanos (monto/1000), a su vez realizar el redondeo ordenado por el Banco Central de Venezuela BCV.

El resultado de la ejecución de este query (**reconversion\_monetaria.sql**), se debe realizar conectados a la base de Datos prueba con la herramienta PgAdmin, una vez ejecutado sin inconvenientes, el personal de Talento Humano de la Institución deberá realizar un chequeo para verificar el éxito del proceso de Reconversión Monetaria.



## PASO 5.-

Ejecutar por PgAdmin, los Update de todos los campos de las tablas involucradas.



The screenshot shows the PgAdmin SQL Editor interface. The title bar indicates the connection is to 'ejemplo on postgres@localhost:5432'. The SQL Editor contains two SQL queries:

```
UPDATE acreencia set monto_acreencia=(case substring(to_char(monto_acreencia/1000, '99999999999999999999'),20,1)
when '1' then round(to_number(to_char(monto_acreencia/1000, '99999999999999999999'),2)
when '2' then round(to_number(to_char(monto_acreencia/1000, '99999999999999999999'),2)
when '3' then round(to_number(to_char(monto_acreencia/1000, '99999999999999999999'),2)
when '4' then round(to_number(to_char(monto_acreencia/1000, '99999999999999999999'),2)
when '5' then round(to_number(to_char(monto_acreencia/1000, '99999999999999999999'),2)
when '6' then round(to_number(to_char(monto_acreencia/1000, '99999999999999999999'),2)
when '7' then round(to_number(to_char(monto_acreencia/1000, '99999999999999999999'),2)
when '8' then round(to_number(to_char(monto_acreencia/1000, '99999999999999999999'),2)
when '9' then round(to_number(to_char(monto_acreencia/1000, '99999999999999999999'),2)
when '0' then round(to_number(to_char(monto_acreencia/1000, '99999999999999999999'),2)
end);
UPDATE ajustetabuladoronu set ajuste_anual=(case substring(to_char(ajuste_anual/1000, '99999999999999999999'),20,1)
when '1' then round(to_number(to_char(ajuste_anual/1000, '99999999999999999999'),2)
when '2' then round(to_number(to_char(ajuste_anual/1000, '99999999999999999999'),2)
when '3' then round(to_number(to_char(ajuste_anual/1000, '99999999999999999999'),2)
when '4' then round(to_number(to_char(ajuste_anual/1000, '99999999999999999999'),2)
when '5' then round(to_number(to_char(ajuste_anual/1000, '99999999999999999999'),2)
when '6' then round(to_number(to_char(ajuste_anual/1000, '99999999999999999999'),2)
when '7' then round(to_number(to_char(ajuste_anual/1000, '99999999999999999999'),2)
```

The Output pane at the bottom is empty, showing tabs for 'Data Output', 'Explain', 'Messages', and 'History'. The system tray at the bottom shows the time as 15:54.

**Nota:** Esta opción se debe ejecutar una sola vez.



## **Notas importantes:**

- ✓ En el caso de las tablas con alto volumen de información como: **historicoquincena, historiconomina, historicosemana, sueldopromedio y trabajador** se recomienda ejecutar la reconversión monetaria por separados, para evitar inconvenientes al ejecutar el query por espacio y memoria del servidor.
  
- ✓ Realizar el respaldo de la Base de Datos en producción y verificación del mismo antes de la ejecución de los script de reconversión monetaria.
  
- ✓ Este proceso de Reconversión se debe generar una sola vez por tabla en la Base de datos del órgano/ente.

---

## **Verificación de los datos convertidos:**

Proceso de chequeo de datos:

Realizar la debida verificación de los datos convertidos a través de la aplicación Sigefirrh. Se recomienda ampliamente realizar casos de estudios típicos antes y después de la ejecución de la actualización.

---

## **Establecer plan de contingencia:**

1. Estar claro y entender completamente el proceso de redondeo establecido por el BCV.
2. Establecer las fechas de cierre de nómina con la nueva actualización.
3. Definir las acciones a tomar en el caso de presentarse algún problema con información de la base de datos y con la aplicación.
4. Tener a disposición los usuarios y password de los servidores de capa web y capa de datos
5. Asegurar el correcto funcionamiento de la red y de los servicios conexos.
6. Disponer de las instrucciones necesarias para subir y bajar los servicios de la web y de la base de datos.
7. Familiarizarse con la herramienta de reconversión monetaria.

---

## **Resistencia al cambio:**

Manejar y conocer los casos típicos de posibles problemas a presentarse debido al proceso de redondeo para proveer de una respuesta acorde al problema.

---

## **Servidor de respaldo:**

**Por las experiencias vividas se hace importante contar con un servidor de respaldo de la base de datos SIGEFIRRH antes de la reconversión y en la medida de lo posible contar con otro servidor espejo u otro medio de almacenamiento magnético para evitar inconvenientes en el futuro.**

---

## Limitaciones:

1. Conectividad permanente con la aplicación y la base de datos.
  2. Soporte al usuario disminuido por ausencia de personal.
- 

## Participantes del proyecto:

Nombre y Apellido	Cargo	Teléfono	e-mail
Yorglee Alvarez	Consultor Funcional	802-53-43	yalvarez@mppef.gob.ve
José Caicedo	Consultor Informática	802-5318	jcaicedo@mppef.gob.ve
Paco Cabrera	Consultor Informática	802-53-40	pacabrera@mppef.gob.ve

## **Anexo:**

Herramienta tecnológica capaz de realizar el proceso de la reconversión monetaria del sistema SIGEFIRRH (script)

### **generar\_query.sql**

Query que genera el archivo (**reconversion\_monetaria.sql**) con las instrucciones (update) requeridas para la realización de la reconversión monetaria y redondeo establecido por el Banco Central de Venezuela.

### **reconversion\_monetaria.sql**

Resultado de la generación del query (**generar\_query.sql**), que contiene las instrucciones sql (update), para la realización de la reconversión monetaria definitiva de todos los campos numéricos con decimales (16,2), en la base de datos del órgano/ente.

---