

Introducción

En este documento se describe la ejercitación a realizar durante el taller de Backup de postgres. Los ejercicios están basados o requieren el uso de una máquina virtual con Linux, donde se encuentra pre-instalado Postgres. Dicha máquina virtual se encuentra instalada en los equipos del laboratorio.

Usuarios y Claves

Usuario demo, con clave demo. Sirve para conectarse a Linux

Usuario postgres: hacer un “sudo su postgres” desde el usuario demo

Usuario root: hacer un “sudo su “ desde el usuario demo

Usuario de Guarani virasol con clave 123456

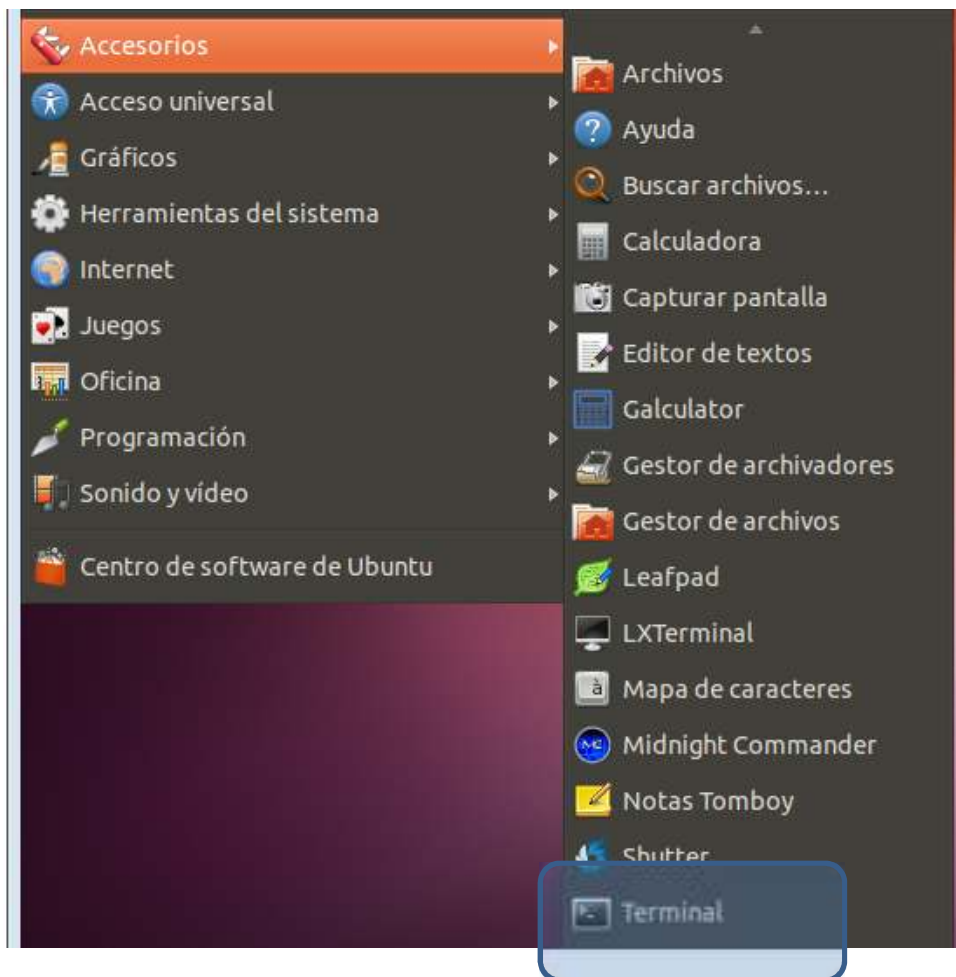
Usuario de Guarani pepin con clave 123456

Otros usuario de Guarani: palazzo, Gabriela.acosta, canedo. Todos con la misma passwd

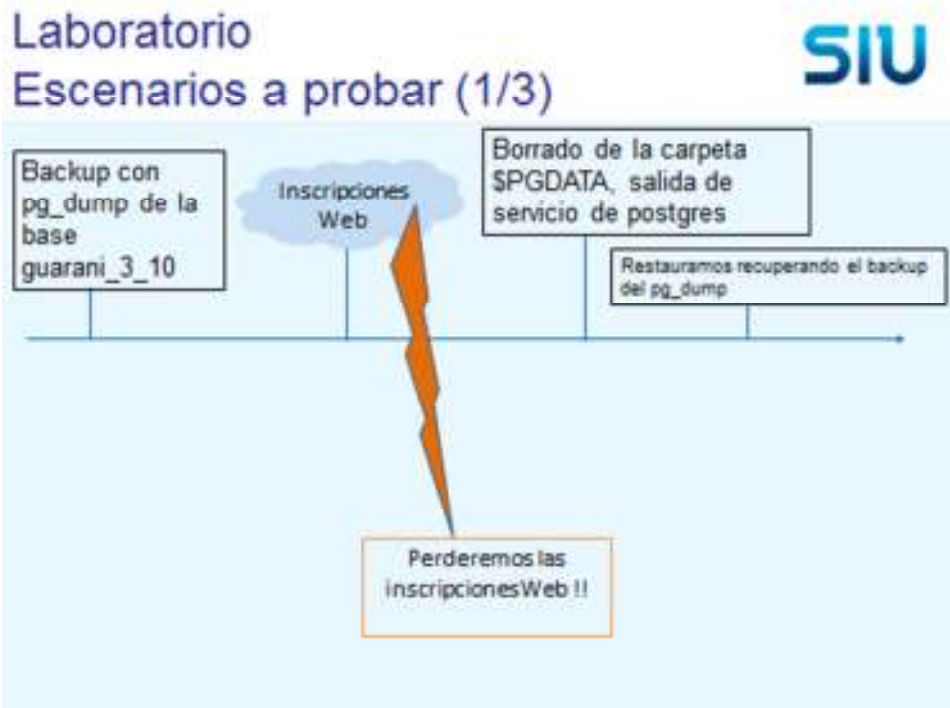
Como abrir una terminal en Linux

Logearse a Linux como usuario demo con clave demo

Luego seleccionar: Aplicaciones + Accesorios + Terminal como se puede ver a continuación:



ESCENARIO 1



1.1 Backup con pg_dump

La idea de este punto es hacer un backup con pg_dump, que no tendrá las inscripciones, luego cuando se haga el restore, las inscripciones no van a ser restauradas y se perderán.

En los siguientes comandos vamos a hacer un pg_dump de la base guaraní_3_10 en el postgres versión 9.3 que esta escuchando en el puerto 5434

```
sudo su postgres
```

```
cd /home/taller_guarani
```

```
./entorno.sh
```

```
pg_dump -p 5434 guarani_3_10 > guaraní_3_10.sql
```

Los mismos comandos los vemos en la siguiente print-screen

```
demo@g3w2-demo: ~
Archivo Editar Ver Buscar Terminal Ayuda
demo@g3w2-demo:~$ id
uid=1000(demo) gid=1000(demo) grupos=1000(demo),4(adm),20(dialout),24(cdrom),33(
www-data),46(plugdev),104(fuse),112(lpadmin),120(admin),122(sambashare)
demo@g3w2-demo:~$ sudo su postgres
[sudo] password for demo:
postgres@g3w2-demo:/home/demo$ cd ..
postgres@g3w2-demo:/home$ cd taller_guarani/
postgres@g3w2-demo:/home/taller_guarani$ ./entorno.sh
postgres@g3w2-demo:/home/taller_guarani$ pg_dump -p 5434 guarani_3_10 > guarani_
pg_dump.sql
postgres@g3w2-demo:/home/taller_guarani$
```

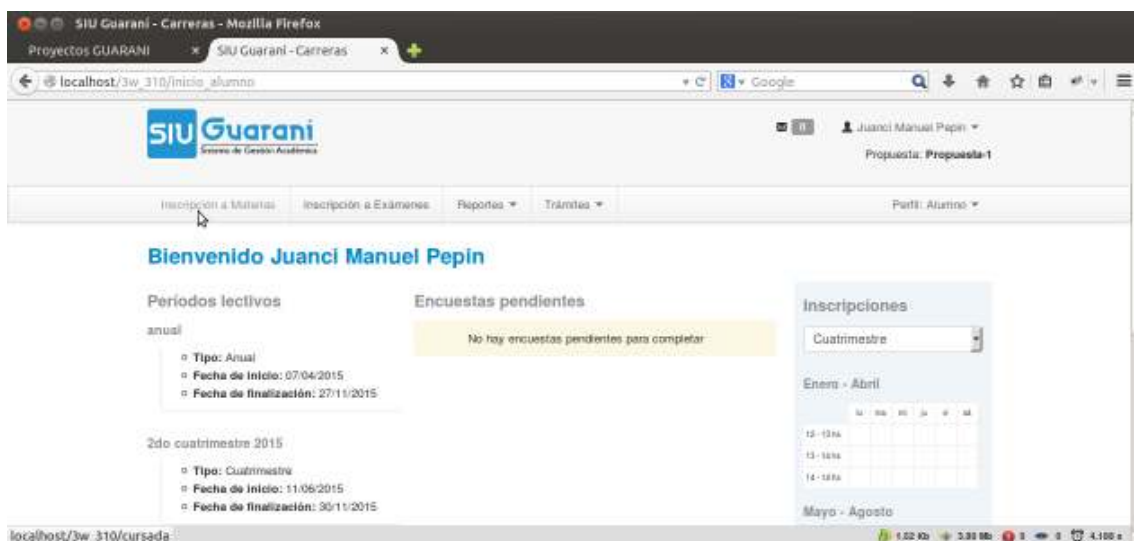
1.2 Inscripcion a Materias

Abrimos Firefox y ponemos localhost o 127.0.0.1 como url

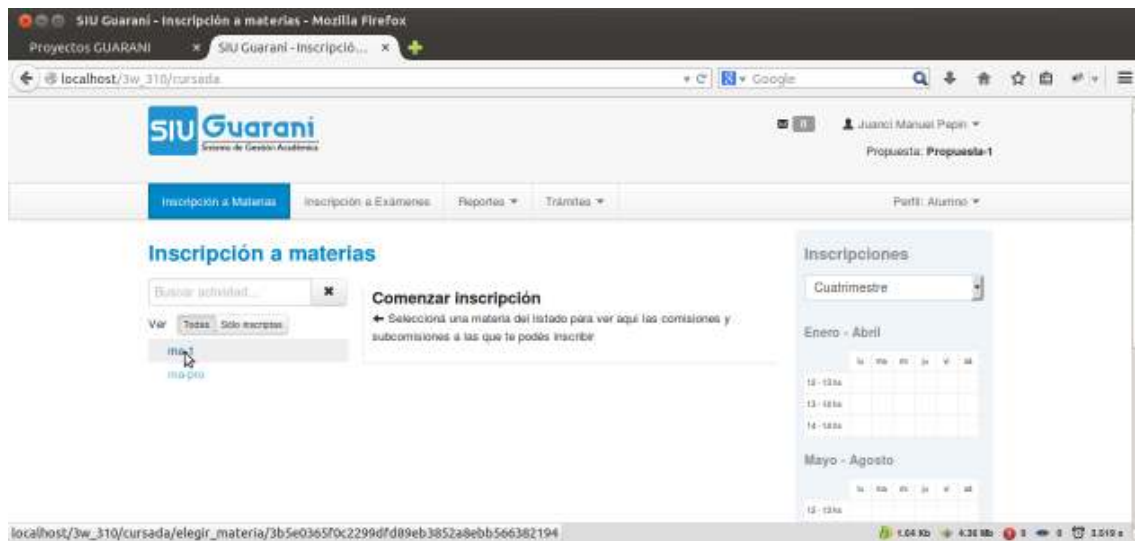


Seleccionamos Autogestion 3.10

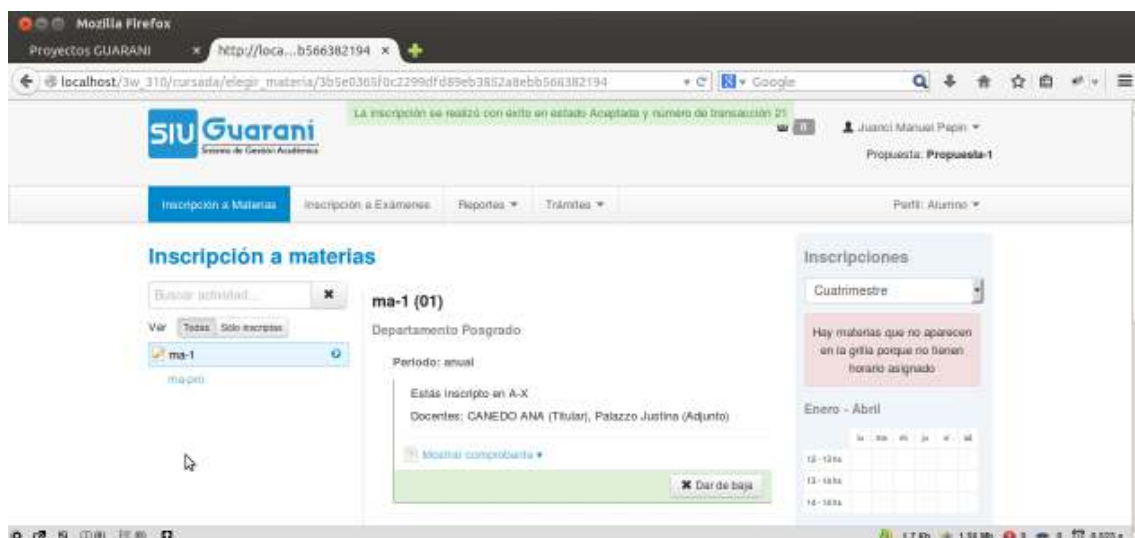
Entramos como usuario pepin (ver password al principio del documento)



Seleccionamos inscripción a materias



Seleccionamos **ma1** y luego presionamos inscribirse



Luego repetimos lo mismo para la otra materia (**ma-pro**)

Cerramos sesión como usuario pepin y nos conectamos como usuario virasol y nos inscribimos en **ma-1**. Cerrar sesión como virasol, y cerrar firefox

1.3 Verificar Inscripciones

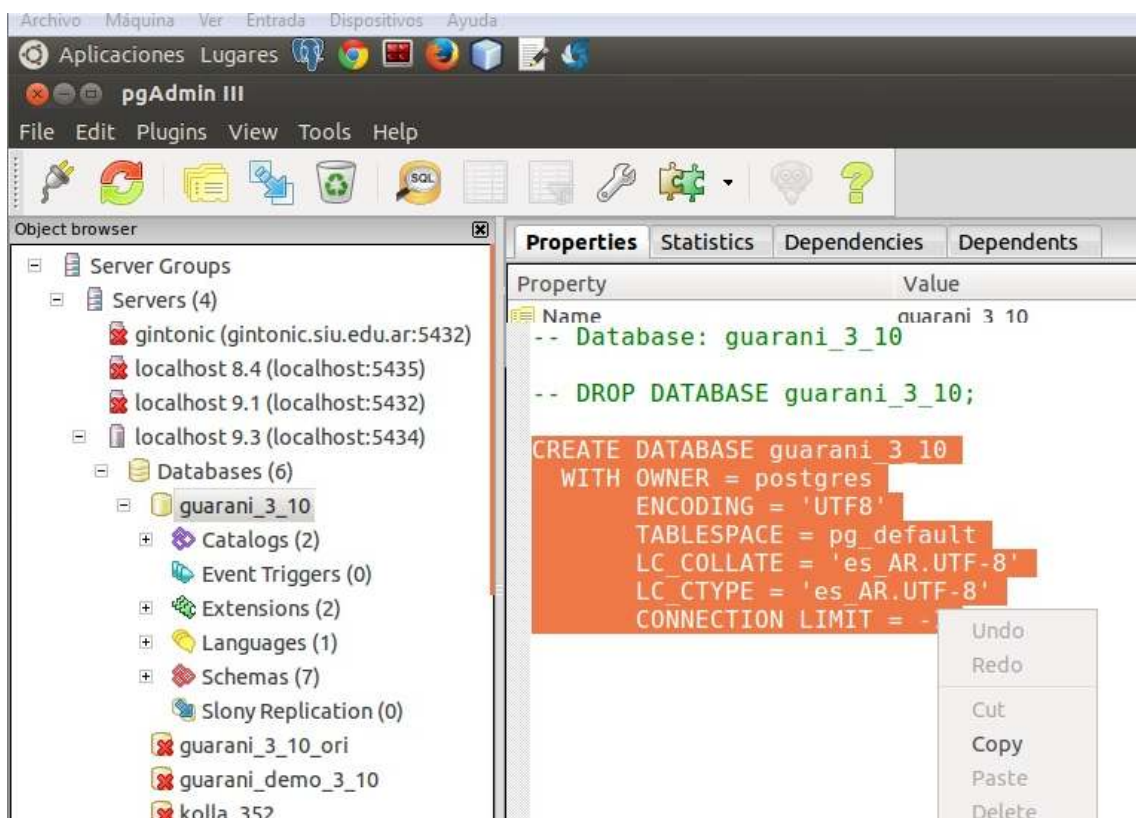
Abrir sesión como usuario pepin y verificar que esté inscripto en las materias ma-1 y ma-pro deberíamos ver una pantalla como la siguiente que tiene los lapicitos al lado de las materias indicando que esta iinscripto.



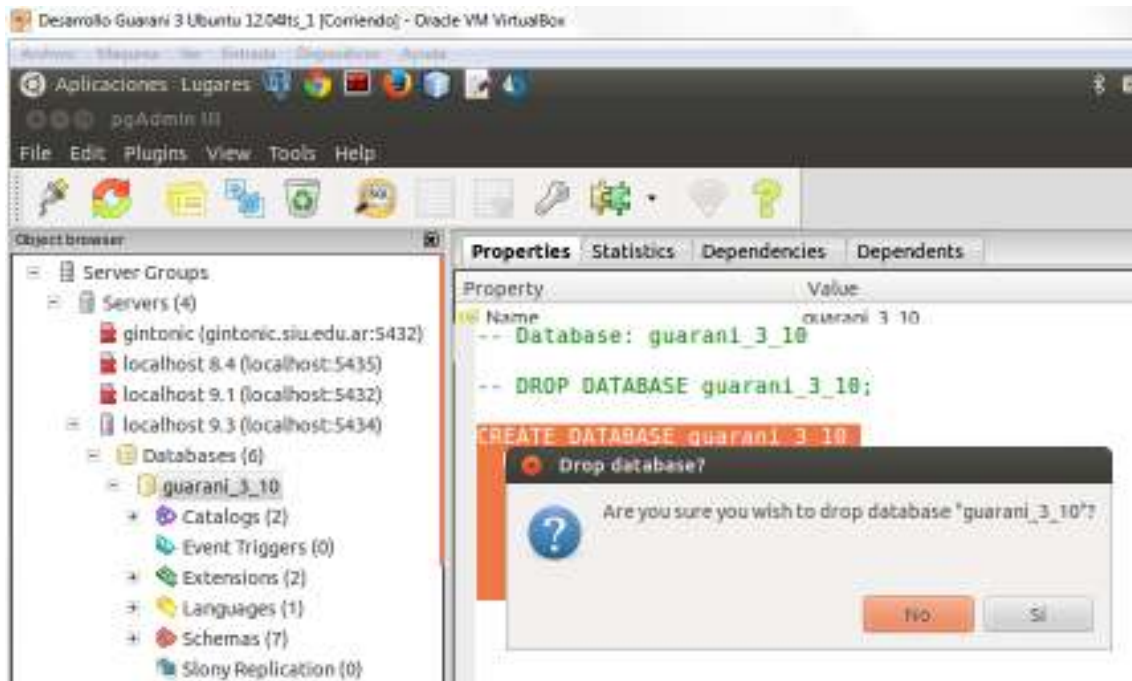
No modificamos nada, cerramos sesión y salir del firefox

1.4 Restore del backup realizado con pg_dump

Abrir pgadminIII y conectarse al localhost 9.3, luego seleccionar la base de datos guarani3_10, se despliega a la derecha el SQL para crear la base de datos. Seleccionarlo y hacer control+copy, como se ve a continuación.



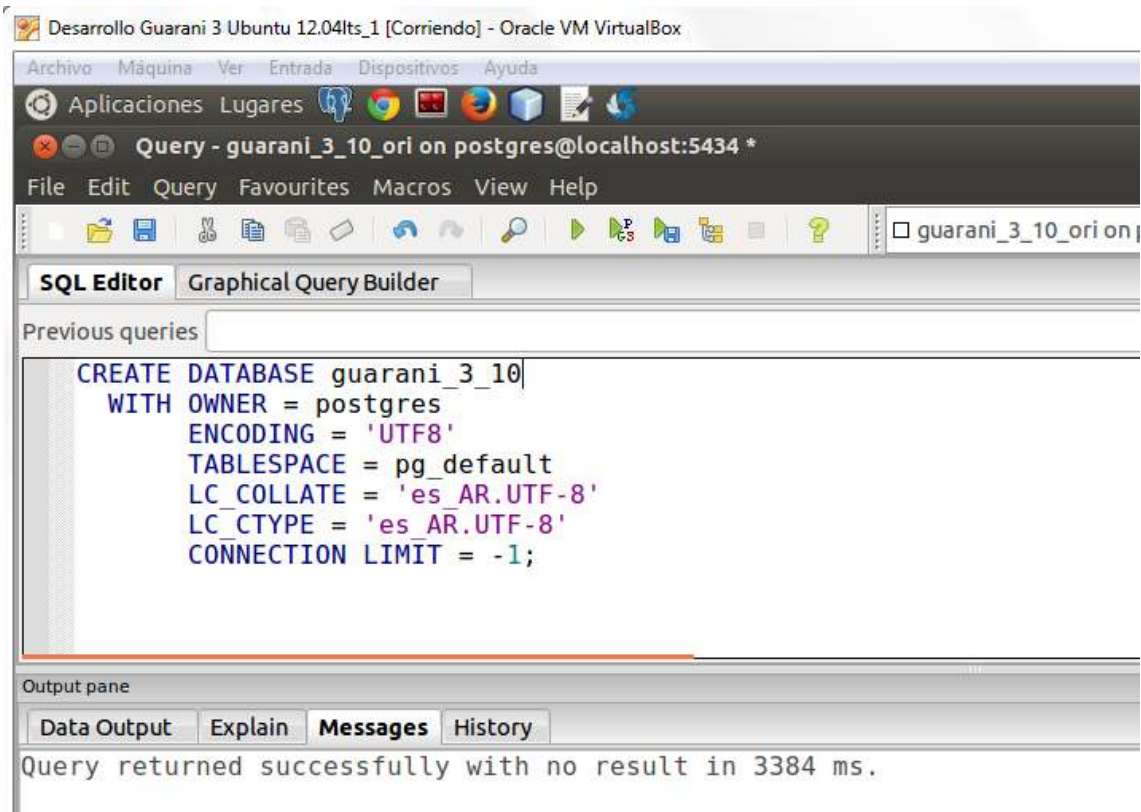
Luego ponemos el mouse en la base guaraní_3_10, le damos botón derecho y seleccionamos drop/delete para borrar la base de datos como se ve a continuación



En este punto una buena prueba es intentar entrar al G3, deberíamos ver una pantalla indicando que la base guaraní_3_10 no existe como la siguiente:



Luego seleccionamos la base guaraní_3_10_ori y accedemos al SQL , y pegamos y ejecutamos el **CREATE DATABASE guaraní_3_10** como se ve a continuación.



Una vez que creamos la base guaraní_3_10 vacía, vamos a importar el pg_dumo generado en el paso 1.1 con la instrucción psql que se ve a continuación:

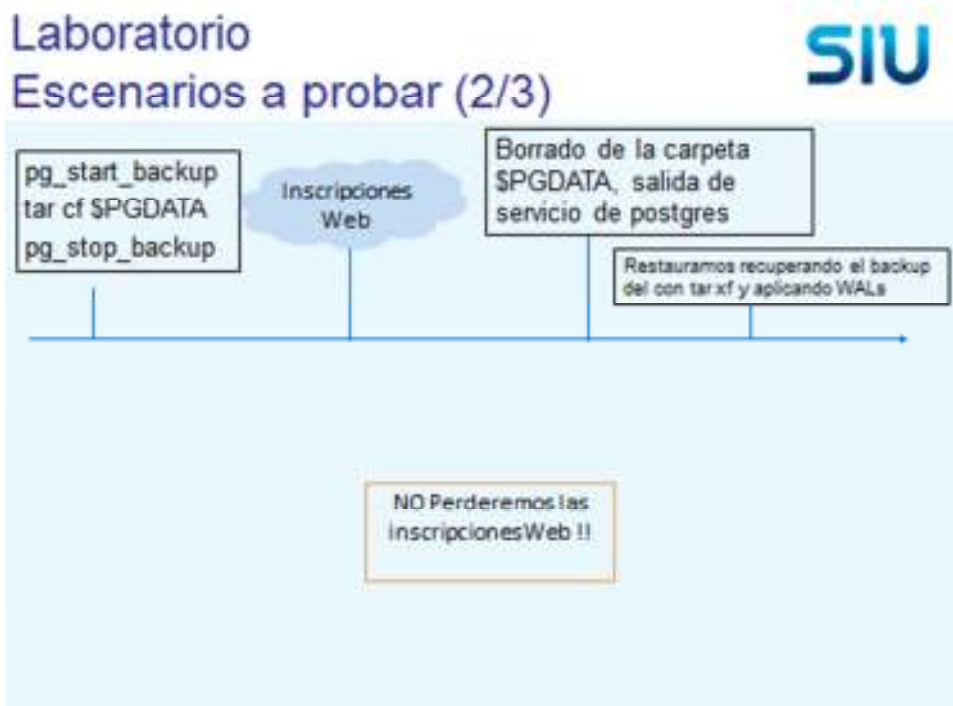
```
postgres@pg3w2-deno:/home/taller_guarani$ psql -p 5434 guarani_3_10 -f guarani_3_10.sql
```

Al finalizar el import la pantalla que vemos es la siguiente:

Poner aca la P8

Luego de finalizar el import de los datos, volvemos a entrar a G3 y no deberíamos ver las inscripciones que realizamos, es decir que perdimos los datos de esas inscripciones debido a que restauramos un pg_dump tomado previamente a las inscripciones

ESCENARIO 2



NOTA: Este ejercicio se puede hacer sin haber realizado el ejercicio anterior (Escenario 1), o habiéndolo realizado en forma completa. Sin embargo si empezaste el ejercicio anterior y no lo terminaste correctamente, antes de comenzar con el escenario 2 deberías des-inscribir de las materias a los alumnos que inscribiste en el paso 1.2 del ejercicio anterior.

2.1 Configuración de backup de WAL

Tal como ya vimos para activar el backup de los segmentos WAL de postgres, tenemos que hacer algunos cambios en la configuración de Postgres, para lo cual vamos a editar el archivo `/etc/postgresql.conf` y modificar los siguientes parámetros

Parámetro	Valor original	Nuevo Valor
wal_level	minimal	archive
Archive_mode	off	On
Archive_command	' '	cp %p /usr/local/carpetadebackupdelwals/%f

Para editar el archivo de configuración seguimos los pasos que están en la siguiente print-screen (usar nano si lo prefieren)

```
demo@g3w2-demo: ~
Archivo Editar Ver Buscar Terminal Ayuda
postgres@g3w2-demo:/home/taller_guarani$ id
uid=115(postgres) gid=124(postgres) grupos=124(postgres),113(ssl-cert)
postgres@g3w2-demo:/home/taller_guarani$ vi /etc/postgresql/9.3/main/postgresql.conf
```

```
demo@g3w2-demo: ~
Archivo Editar Ver Buscar Terminal Ayuda

#wal_writer_delay = 200ms                # (change requires restart)
                                           # 1-10000 milliseconds

#commit_delay = 0                        # range 0-100000, in microseconds
#commit_siblings = 5                    # range 1-1000

# - Checkpoints -

#checkpoint_segments = 3                 # in logfile segments, min 1, 16MB each
#checkpoint_timeout = 5min                # range 30s-1h
#checkpoint_completion_target = 0.5      # checkpoint target duration, 0.0 - 1.0
#checkpoint_warning = 30s                # 0 disables

# - Archiving -

archive_mode = off                       # allows archiving to be done
#archive_mode = on                       # allows archiving to be done
                                           # (change requires restart)
archive_command = ''                     # command to use to archive a logfile segment
#archive_command = 'cp %p /usr/local/carpetadebackupdewals/%f' # command to use
to archive a logfile segment
```

La carpeta destino de los WAL ya está creada y los cambios en el archivo de configuración son simples (solo se debe quitar un # en la línea que corresponde y agregarlo a la línea anterior), ya que los nuevos valores están comentados como se puede ver en la print-screen anterior

Luego se debe hacer un restart (bajar y subir) de postgres como se ve a continuación:

```
postgres@g3w2-demo:/home/taller_guarani$ /etc/init.d/postgresql restart
* Restarting PostgreSQL 8.4 database server [ OK ]
* Restarting PostgreSQL 9.1 database server [ OK ]
* Restarting PostgreSQL 9.3 database server [ OK ]
postgres@g3w2-demo:/home/taller_guarani$
```

Es conveniente verificar que postgres ya esté copiando archivos en la carpeta de respaldo de WALs llamada /usr/local/carpetadebackupdewals. Hacerlo mediante los comandos

/etc/init.d/postgresql restart

ls -ltr /usr/local/carpetadebackupdewals

Y fijarse de que fecha es el último archivo WAL que aparece en el ls, debería ser una fecha/hora muy reciente como se ve en la siguiente print screen que el archivo 000062 es de fecha sep 2 18:35 y la hora del server es sep 2 18:42.

PONER AQUÍ P7

2.2 Realizar un backup completo (tipo nivel 0 en informix) teniendo el WAL activado.

Entrar como usuario demo en 2 terminales (term1 y term2)

En term1 hacer :

- sudo su postgres
- cd /home/taller_guarani
- . ./entorno.sh

En term2 hacer :

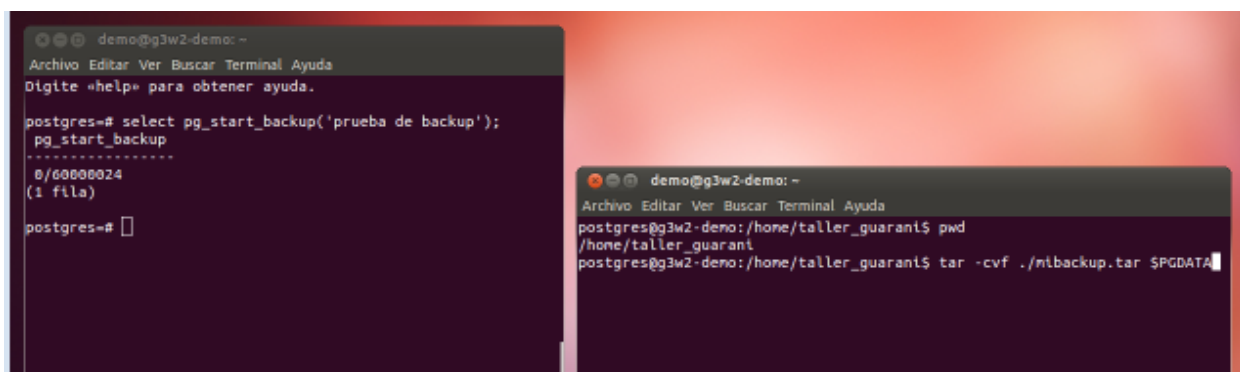
- sudo su postgres
- cd /home/taller_guarani
- . ./entorno.sh

En term1 hacer:

- Entrar a **psql -p5434** y ejecutar un **select pg_start_backup('test backup');**
- Tomar nota del WAL vigente con el pg_start_backup _____

En term2 hacer:

- cd /home/taller_guarani
- **tar -cvf ./mibackup.tar \$PGDATA** (tarda unos minutos)



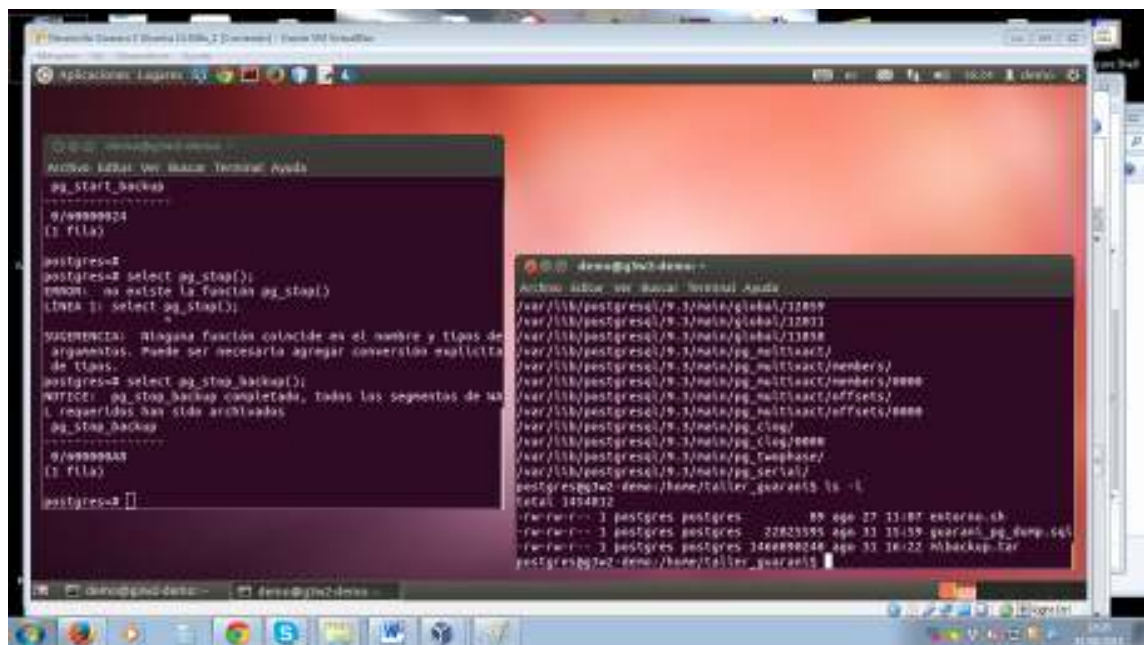
The image shows two terminal windows side-by-side. The left window shows a PostgreSQL prompt where the command `select pg_start_backup('prueba de backup');` is entered, followed by `pg_start_backup` and the output `0/60000024 (1 fila)`. The right window shows a shell prompt where the command `tar -cvf ./mibackup.tar $PGDATA` is entered.

Nota Importante: El archivo resultante del punto anterior **mibackup.tar** contiene el backup completo del cluster postgres del puerto 5434, y deberá resguardarse por los medios que cada Organización considere

conveniente. Este taller es para explicar cómo generar el backup y como restaurarlo. Cada Organización debe definir cómo administrar los backups y sus distintas versiones.

En term 1 hacer:

- En la misma **psql -p 5434** de antes ejecutar un **select pg_stop_backup();**
- Tomar nota del WAL vigente con el **pg_stop _____**



```

pg_start_backup
pg_stop_backup
select pg_stop_backup();
NOTICE: pg_stop_backup completed. All the WAL segments
        requested have been archived.
pg_stop_backup

```

2.3 Inscripciones

Nuevamente vamos a hacer inscripciones como lo hicimos en el escenario anterior (recordemos que esas inscripciones las perdimos) en esta oportunidad vamos a tomar la hora en que se hace cada inscripción

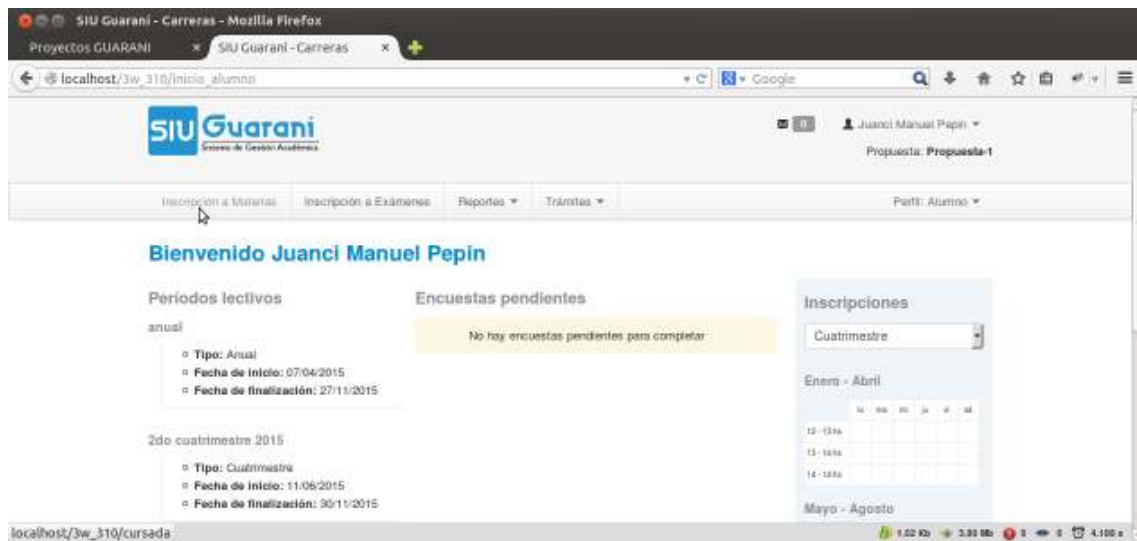
Abrimos Firefox y ponemos localhost o 127.0.0.1 como url



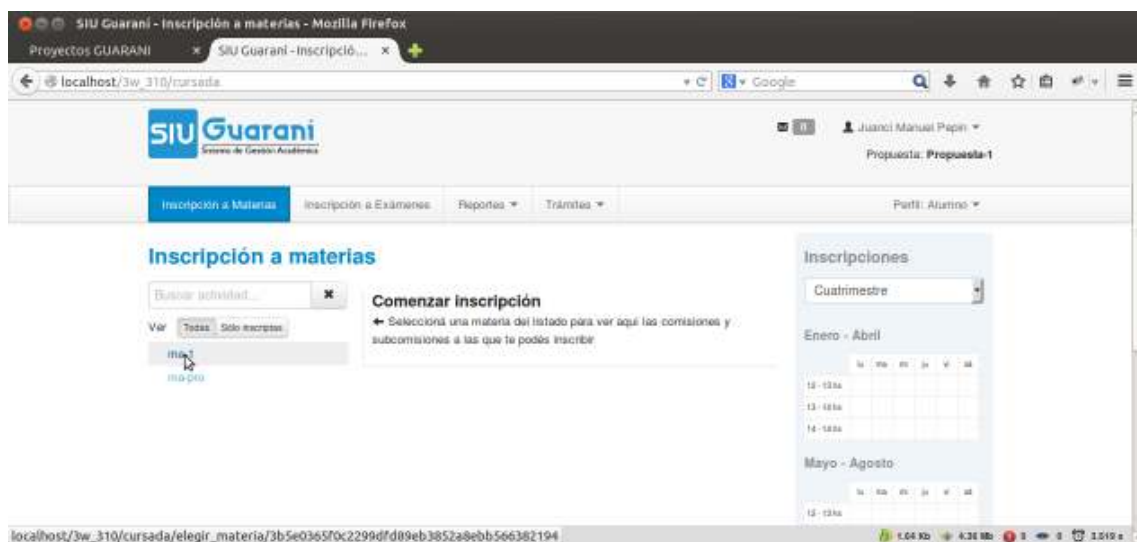
Seleccionamos Autogestion 3.10

Entramos como usuario pepin (ver password al principio del documento)

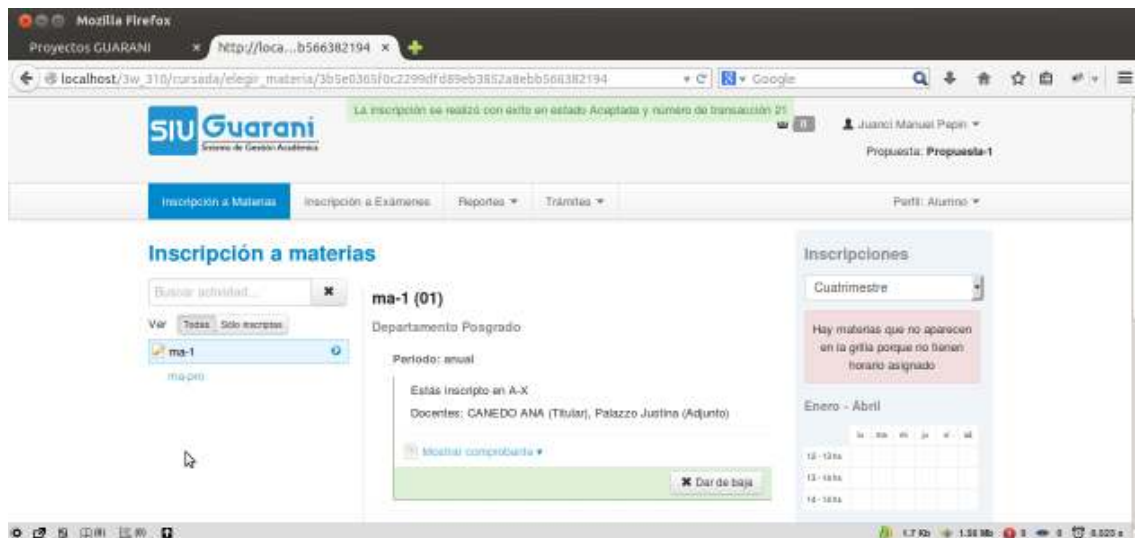




Seleccionamos inscripción a materias



Seleccionamos **ma1** y luego presionamos inscribirse



Luego repetimos lo mismo para la otra materia (**ma-pro**)

- Inscripción de pepin a ma-1 Hora: _____
- Inscripción de pepin a ma-pro Hora: _____

Cerramos sesión como usuario pepin y nos conectamos como usuario virasol y nos inscribimos en **ma-1**.

- Inscripción de Virasol a ma-1 Hora: _____

Para tener otra transacción más que recuperar, entra a Trámites + Mis Datos Personales + Domicilio e ingresa una calle, numero ciudad, y código postal y presiona Guardar

- Tomar nota de la hora al guardar los cambios del domicilio: _____
- Tomar nota del WAL vigente al hacer la ultima inscripción: _____

2.4 Vamos a simular una rotura en el disco

2.4.1 Bajar postgres con **/etc/init.d/postgresql stop** (es IMPORTANTE bajar postgres antes de borrar el \$PGDATA) asi copiamos los WAL

(Nota: tener en cuenta que el postgres no baja bien cuando el PGDATA esta borrado)

```
postgres@3w2-demo:/$ /etc/init.d/postgresql stop
* Stopping PostgreSQL 8.4 database server [ OK ]
* Stopping PostgreSQL 9.1 database server [ OK ]
* Stopping PostgreSQL 9.3 database server [ OK ]
```

Verificar que haya copiado en carpeta /usr/local/carpetadebackupdewals el último WAL completado en \$PGDATA/pg_xlog. Hacer un ls -ltr de ambas carpetas se debería ver algo así:

ls -l /usr/local/carpetadebackupdewals

ls -l \$PGDATA/pg_xlog

```
-rw-r----- 1 postgres postgres 16777216 ago 31 16:32 00000001000000000000000000000000
-rw-r----- 1 postgres postgres 16777216 ago 31 16:35 00000001000000000000000000000049
-rw-r----- 1 postgres postgres 16777216 ago 31 16:35 00000001000000000000000000000061
postgres@g3w2-demo:/home/taller_guarani$
```

2.4.2 Borrar todo el \$PGDATA, como usuario postgres hacer:
rm -rf \$PGDATA

```
postgres@g3w2-demo:/home/taller_guarani$ rm -rf $PGDATA
postgres@g3w2-demo:/home/taller_guarani$
```

Intentar acceder a G3 o levantar postgres, no se va a poder, pues todos los archivos del cluster de postgres han sido borrados

```
postgres@g3w2-demo:/$ /etc/init.d/postgresql stop
* Stopping PostgreSQL 8.4 database server [ OK ]
* Stopping PostgreSQL 9.1 database server [ OK ]
* Stopping PostgreSQL 9.3 database server [ OK ]
postgres@g3w2-demo:/$ rm -rf $PGDATA
postgres@g3w2-demo:/$ /etc/init.d/postgresql start
* Starting PostgreSQL 8.4 database server [ OK ]
* Starting PostgreSQL 9.1 database server [ OK ]
* Starting PostgreSQL 9.3 database server
* Error: /var/lib/postgresql/9.3/main is not accessible or does not exist [ FATAL ]
```

En la pantalla anterior se ve como el start de postgres falla para el cluster del puerto 5434, ya que solo ese cluster fue borrado.

Probar de entrar a G3 como usuario virasol/123456 y tenemos



También es posible intentar conectar con el cluster del puerto 5434 desde pgadminIII, lo cual dará un error al conectar

2.5 Restore del backup continuo

2.5.1 Postgres debería estar bajo, si no lo está hay que bajarlo

2.5.2 Para hacer el restore hay que pararse en / como usuario postgres y ejecutar un

cd /

tar -xf /home/taller_guarani/mi_backup.tar (tarda unos minutos)

```
postgresgw2-demo:/$ tar -xf /home/taller_guarani/mi_backup.tar
postgresgw2-demo:/$ /etc/init.d/postgresql start
* Starting PostgreSQL 8.4 database server [ OK ]
* Starting PostgreSQL 9.1 database server [ OK ]
* Starting PostgreSQL 9.3 database server [ OK ]
```

2.5.3 Verificar que se encuentre el recovery.conf en la carpeta \$PGDATA. Si el archivo no existe y existe el recovery.done, entonces renombrar recovery.done a recovery.conf, el renombre se hace con el comando:

mv \$PGDATA/recovery.done \$PGDATA/recovery.conf

2.5.4 En otra terminal es posible poner un tail -f como root del /var/log/syslog para ver que pasa cuando levanta el postgres, Como user demo hacer un **sudo tail -f /var/log/syslog**

2.5.5 Levantar el postgres con un /etc/init.d/postgresql start

Poner p9 para analizar el log de la levantada

A continuacion hay una pantalla que no tiene comandos para ejecutar, es solo para mostrar una secuencia de como borrar el PGDATA, intentar levantar postgres fallidamente, luego bajar postgres (los otros cluster tambien) y por ultimo hacer el restore del tar, y finalmente si levantar postgres en forma exitosa.

```
postgres@3w2-demo:/$ rm -rf SPGDATA
postgres@3w2-demo:/$ /etc/init.d/postgresql start
* Starting PostgreSQL 8.4 database server [ OK ]
* Starting PostgreSQL 9.1 database server [ OK ]
* Starting PostgreSQL 9.3 database server
Error: /var/lib/postgresql/9.3/main is not accessible or does not exist [FAIL]

postgres@3w2-demo:/$ /etc/init.d/postgresql stop
* Stopping PostgreSQL 8.4 database server [ OK ]
* Stopping PostgreSQL 9.1 database server [ OK ]
* Stopping PostgreSQL 9.3 database server
Error: /var/lib/postgresql/9.3/main is not accessible or does not exist [FAIL]

postgres@3w2-demo:/$ tar -xvf /home/taller_guarani/nt_backup.tar
postgres@3w2-demo:/$ /etc/init.d/postgresql start
* Starting PostgreSQL 8.4 database server [ OK ]
* Starting PostgreSQL 9.1 database server [ OK ]
* Starting PostgreSQL 9.3 database server [ OK ]
postgres@3w2-demo:/$
```

2.6

Abrir G3 y fijarse si están registradas inscripciones recién realizadas y los datos del domicilio que modificaste.

Virasol ma 1: _____

Pepin ma 1: _____

Pepin ma pro: _____

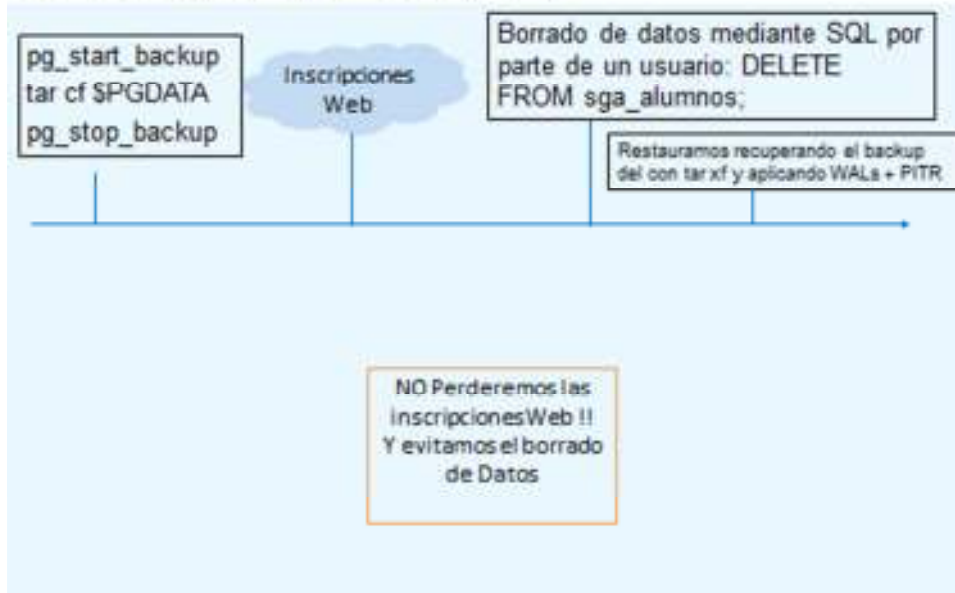
Deberíamos ver las inscripciones como se ven el siguiente pantalla:



Si viste las inscripciones y los cambios al domicilio, entonces hiciste un restore de un backup continuo aplicando WALs que fue exitoso!!!

ESCENARIO 3

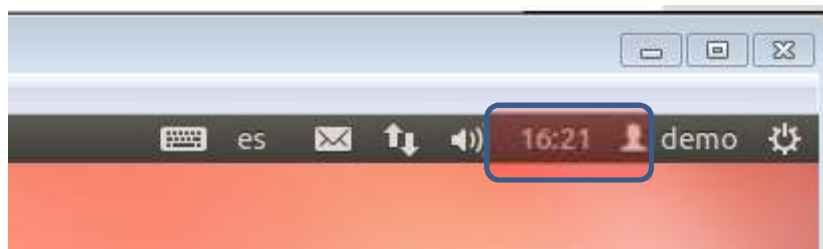
Laboratorio Escenarios a probar (3/3)



En este ejercicio vamos a utilizar el backup total realizado en el escenario 2 y los WAL backupeados (que contienen las inscripciones). Adicionalmente vamos a hacer un DELETE de una tabla importante y luego vamos a restaurar todo, pero haciendo un PITR hasta 1 minuto antes de que se ejecute el DELETE

Situacion inicial: Tenemos el backup de nivel 0 realizado en el escenario 2, junto con todos los WAL posteriores backupeados. En esos WAL figuran las inscripciones y el cambio de datos personales. Los pasos a realizar en este ejercicio son los siguientes:

- 1- Tomamos nota de la hora del sistema: _____



- 2- Hacemos un UPDATE de la tabla negocio.mdp_personas cambiándole el apellido a todas personas....*&^@\$#^%#@#%^\$%#@#@#\$

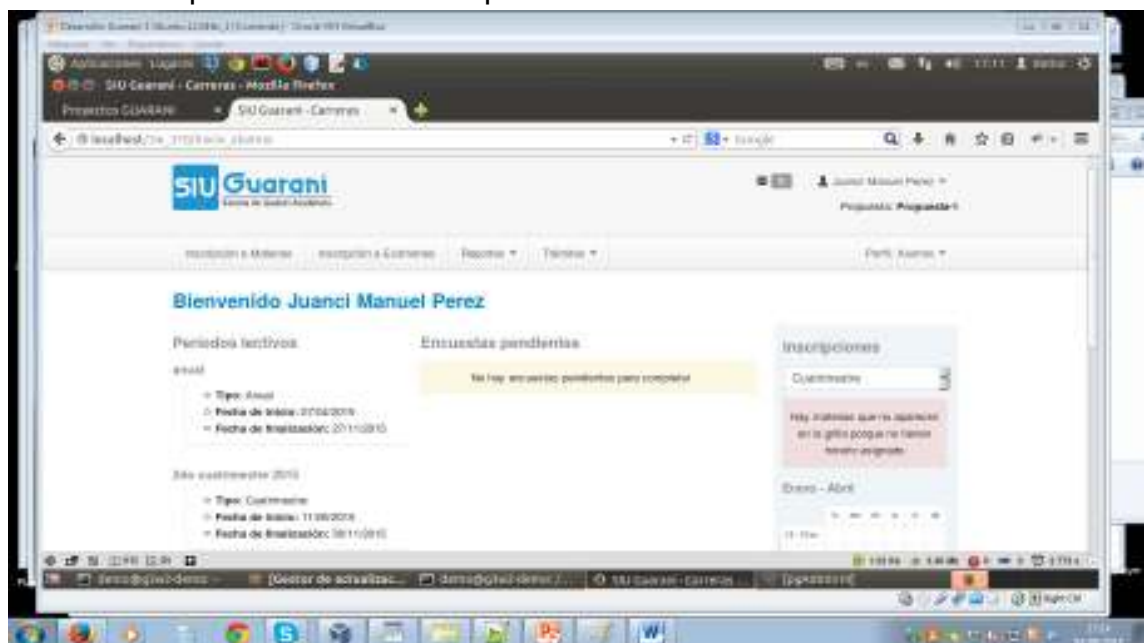

```
demo@g3w2-demo: ~
Archivo Editar Ver Buscar Terminal Ayuda
ERROR: no existe la relación «negocio.sga_detalle_acta»
LÍNEA 1: select * from negocio.sga_detalle_acta;
      ^
guarani_3_10=# select * from negocio.sga_actas_detalle;
guarani_3_10=# select * from negocio.sga_actas;
guarani_3_10=# DELETE from negocio.sga_actas;
ERROR: update o delete en «sga_actas» viola la llave foránea «fk_sga_actas_instancias_sga_actas» en la tabla «sga_actas_instancias»
DETALLE: La llave (id_acta)=(1) todavía es referida desde la tabla «sga_actas_instancias».
guarani_3_10=# select * from negocio.sga_actas;
guarani_3_10=# \q
postgres@g3w2-demo:/$ psql -p 5434 guarani_3_10
psql (9.3.5)
Digite «help» para obtener ayuda.

guarani_3_10=# select * from negocio.mdp_personas;
guarani_3_10=# UPDATE negocio.mdp_personas SET apellido = 'Perez' -- WHERE nro_legajo = 11111
guarani_3_10=# ;
UPDATE 9
guarani_3_10=#
```

Según el log el UPDATE se ejecutó a las 17:07

```
Sep  4 17:06:13 g3w2-demo postgres[10031]: [4-1] 2015-09-04 17:06:13 ART LOG: duración: 15.168 ms sentenc
select * from negocio.mdp_personas;
Sep  4 17:07:22 g3w2-demo postgres[10031]: [5-1] 2015-09-04 17:07:22 ART LOG: duración: 135.889 ms senten
UPDATE negocio.mdp_personas SET apellido = 'Perez'
Sep  4 17:07:22 g3w2-demo postgres[10031]: [5-2] #011:
```

- 3- Intentamos entrar al G3 como cualquier usuario (pepin), debería mostrarnos que el usuario tiene apellido Perez



- 4- Bajamos postgres y hacemos un restore del tar con el backup full

```
postgres@g3w2-demo:/$ psql -p 5434 guarani_3_10
psql (9.3.5)
Digite «help» para obtener ayuda.

guarani_3_10=# select * from negocio.mdp_personas;
guarani_3_10=# UPDATE negocio.mdp_personas SET apellido = 'Perez' -- WHERE nro_l
egajo = 11111
guarani_3_10-# ;
UPDATE 9
guarani_3_10=# \q
postgres@g3w2-demo:/$ /etc/init.d/postgresql stop
* Stopping PostgreSQL 8.4 database server [ OK ]
* Stopping PostgreSQL 9.1 database server [ OK ]
* Stopping PostgreSQL 9.3 database server [ OK ]
postgres@g3w2-demo:/$ cd /
postgres@g3w2-demo:/$ tar -xf /home/taller_guarani/mibackup.tar
```

- 5- Renombramos el recovery.done a recovery.conf y editamos el recovery.conf poniéndole al parámetro **recovery_target_time** la hora un minuto antes del UPDATE

```
postgres@g3w2-demo:/$ tar -xf /home/taller_guarani/mibackup.tar
postgres@g3w2-demo:/$ mv $PGDATA/recovery.done $PGDATA/recovery.conf
postgres@g3w2-demo:/$ vi $PGDATA/recovery.conf
```

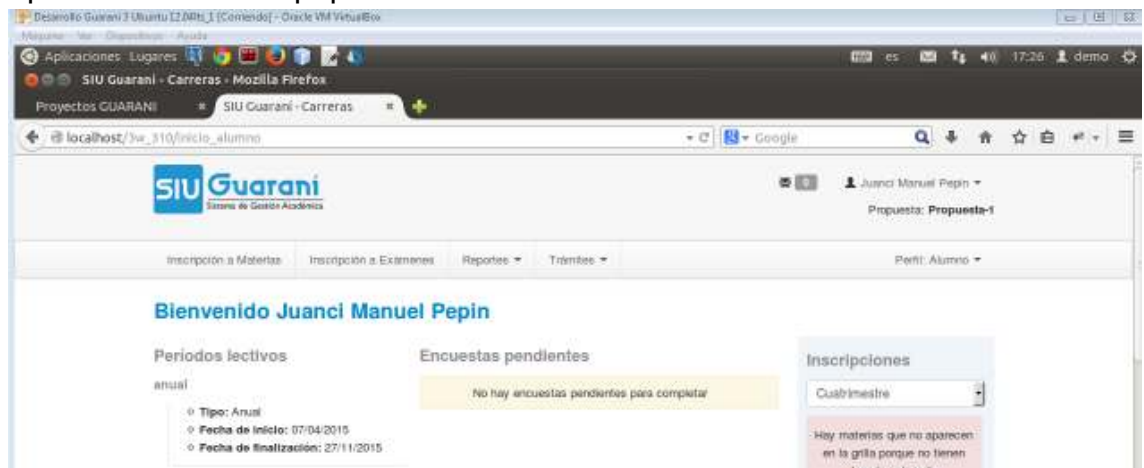
```
demo@g3w2-demo: ~
Archivo Editar Ver Buscar Terminal Ayuda
# just after or just before the given target, respectively).
#
#
#recovery_target_name = '          # e.g. 'daily backup 2011-01-26'
#
#recovery_target_time = '          # e.g. '2004-07-14 22:39:00 EST'
recovery_target_time = '2015-09-04 17:05:00'
#
#recovery_target_xid = '
#
#recovery_target_inclusive = true
#
#
# If you want to recover into a timeline other than the "main line" shown in
# pg_control, specify the timeline number here, or write 'latest' to get
# the latest branch for which there's a history file.
#
#recovery_target_timeline = 'latest'
#
#
# If pause_at_recovery_target is enabled, recovery will pause when
:wq
```

- 6- Levantamos el postgres

```
demo@g3w2-demo: /home/taller_guarani
Archivo Editar Ver Buscar Terminal Ayuda
Sep  4 17:21:12 g3w2-demo postgres[10290]: [2-2] 2015-09-04 17:21:12 ART HINT: La salida futura del registro s
erá enviada al destino de log «syslog».
Sep  4 17:21:13 g3w2-demo postgres[10299]: [3-1] 2015-09-04 17:21:13 ART LOG: el sistema de bases de datos fue
Interrumpido; última vez en funcionamiento en 2015-08-31 16:17:24 ART
Sep  4 17:21:13 g3w2-demo postgres[10300]: [3-1] 2015-09-04 17:21:13 ART LOG: el paquete de inicio está incomp
let
Sep  4 17:21:13 g3w2-demo postgres[10299]: [4-1] 2015-09-04 17:21:13 ART LOG: comenzando el proceso de recuper
ación hasta 2015-09-04 17:05:00-03.
Sep  4 17:21:13 g3w2-demo postgres[10299]: [5-1] 2015-09-04 17:21:13 ART LOG: se ha restaurado el archivo «000
00003.history» desde el área de archivado
Sep  4 17:21:13 g3w2-demo postgres[10299]: [6-1] 2015-09-04 17:21:13 ART LOG: se ha restaurado el archivo «000
000030000000000000000» desde el área de archivado
Sep  4 17:21:13 g3w2-demo postgres[10307]: [3-1] 2015-09-04 17:21:13 ART FATAL: el sistema de base de datos es
tá iniciándose
```

Notar que dice que indica que la restauración es hasta las 17:05

Vemos si efectivamente se arregló logeándonos al G3 y verificando el apellido del usuario pepin



Si aun tiene apellido Perez, algo anduvo mal, pero si ves el apellido Pepin que tenia originalmente, entonces hiciste un restore de un backup continuo con PITR (point in time recovery) !!!!!!!!!