

Tipo de artículo: Artículo original
Temática: Soluciones informáticas
Recibido: 10/12/16 | Aceptado: 10/03/17 | Publicado: 27/03/17

Configuración de la herramienta SymmetricDS para el proyecto Sistema Integral de Perforación de Pozos (SIPP)

Conguration of SymmetricDS's tool for Integral System of Drilling Petroleum (SIPP)

Dayana Leticia López Chávez^{1*}, *Leanni Rodríguez Noblet*¹

1 Universidad de las Ciencias Informáticas. Centro de Informática Industrial, Carretera a San Antonio de los Baños, km 2 1/2, Torrens, Boyeros, La Habana, Cuba. CP.: 19370, Telf. (837) 8246, Email: {dlchavez, lnoblet}@uci.cu

* Autor para correspondencia: dlchavez@uci.cu

Resumen

La Universidad de las Ciencias Informáticas, específicamente el Centro de Desarrollo Industrial perteneciente a la Facultad 4, conjuntamente con la empresa Cuba-Petróleo, desarrolló el Sistema Integral de Perforación de Pozos. La investigación consiste en configurar la herramienta SymmetricDS para la replicación de los datos, eliminando redundancia, errores y desactualización de la información en la base de datos del sistema desplegado en la Dirección de Intervención y Perforación de Pozos y en cada uno de los pozos petroleros. El principal resultado consiste en obtener una base de datos actualizada en todas las entidades donde se encuentra el sistema desplegado y así lograr una correcta gestión, control y flujo de la información.

Palabras Clave: base de datos, información, pozos petroleros, replicación de los datos.

Abstract

Information Science University (UCI), specifically the Industrial Development Center (CEDIN) belonging to the Faculty 4 together with the direction of CUPET, develop a system for Integral System of Drilling Petroleum (SIPP). The research aims to configure the SymmetricDS tool for data replication to eliminate redundancy, errors and

outdated information in the Intervention and Drilling Direction entity (DIPP) and in each of the Oil wells that SIPP system is deployed. The result is a database updated with all necessary information for the correct function of the management process, control and information flow.

Keywords: *database, data replication, information, oil wells.*

Introducción

Desde que el hombre comenzó a dejar pruebas de su paso a través de los años, la información se volvió imprescindible para su vida, más ahora en el siglo XXI, con el establecimiento de una sociedad donde la información está presente en todos los ámbitos en que se desarrolla, especialmente en los centros informáticos que tienen que manejar grandes cantidades de información. Por tales motivos, surge la necesidad de utilizar bases de datos para gestionar este valioso recurso de forma eficiente y efectiva.

Según (Gillermo Aramburu, 2010) se le denomina Base de Datos a los bancos de información que contienen datos relativos a diversas temáticas y categorizados de distinta manera, pero que comparten entre sí algún tipo de vínculo o relación que busca ordenarlos y clasificarlos en conjunto.

Con la evolución de las computadoras, las bases de datos fueron progresando paralelamente con el desarrollo de la tecnología, haciéndose cada vez más útiles en el desarrollo de la sociedad.

Con la necesidad de lograr la gestión de las bases de datos surgen los Sistemas Gestores de Bases de Datos (SGBD), que según (Menéndez, 2015) se definen en “Un tipo de software muy específico, dedicado a servir de interfaz entre las bases de datos y los sistemas que la utilizan. Sus funciones principales son la creación, mantenimiento, eliminación de bases de datos, el control de accesos y la manipulación de información de acuerdo a las necesidades del usuario”.

Para poder garantizar que ciertos datos de la base de datos sean almacenados en más de un sitio y así aumentar la disponibilidad de la información y mejorar el funcionamiento de las consultas globales a la base de datos. Se hace necesario el uso de la replicación de datos.

Actualmente la empresa Cuba-Petróleo (CUPET) se encuentra inmersa en un proceso de expansión de sus principales procesos productivos, las expectativas que se plantean sobre las posibilidades reales del país de la exploración y

perforación de petróleo han propiciado un ambiente de trabajo favorable para la realización de convenios de trabajos con la Universidad de las Ciencias Informáticas (UCI), debido a la imperiosa necesidad de informatizar sus principales procesos de producción, surge el Sistema Integral de Perforación de Pozos (SIPP) que se encuentra en el Centro de Desarrollo Informático Industrial (CEDIN) de la Facultad 4.

El Sistema Integral de Perforación de Pozos (SIPP) se encuentra desplegado en la Dirección de Intervención y Perforación de Pozos (DIPP, encargada de controlar todas las operaciones y actividades que se realizan en los pozos en perforación) y en cada uno de los pozos petroleros que tienen convenio con esta entidad.

Actualmente debido a la distancia que existe entre la DIPP y los pozos petroleros, la comunicación entre ellos se ha afectado, trayendo consigo que el proceso de gestión, control y flujo de la información, no se realice correctamente ocasionando redundancia, errores y desactualización en el almacenamiento de los datos.

En tal sentido, se decide realizar un estudio y análisis de las herramientas de replicación de datos para dar solución inmediata a los problemas antes mencionados.

Materiales y métodos o Metodología computacional

Para alcanzar los objetivos propuestos se utilizan los siguientes métodos científicos:

Métodos Teóricos

- **Histórico-Lógico:** Se investigó las características y elementos fundamentales de las herramientas de replicación existentes.
- **Analítico-sintético:** Con el análisis de las teorías y documentos se determinaron los elementos fundamentales de las herramientas de replicación.

Métodos Empíricos

- **Análisis de documentos:** Permitió realizar el estudio de variada documentación referente a las herramientas de replicación con el objetivo de obtener la experiencia y sugerencias que pudieran ser incorporadas a través de esta investigación.

Entornos de réplica

A continuación, se detallan los diferentes tipos de entornos de réplica que se pueden utilizar para realizar una correcta replicación de datos, como son:

- **Maestro-Eslavo (*master-slave*):** Es un entorno de replicación de datos que permite que los datos sean almacenados por un grupo donde un solo miembro del grupo es designado como el "maestro" de una pieza dada de datos y es el único nodo que puede modificar ese elemento de datos, si otros miembros desean modificar el primer elemento de datos deben comunicarse con el nodo maestro, permitiendo lograr la coherencia entre los miembros del grupo. La principal desventaja es que si ocurre un cambio en alguno de los miembros del grupo no se podrá actualizar ningún otro nodo (Cherencio, 2013).

La siguiente figura muestra un ejemplo de entorno maestro-esclavo donde se puede apreciar que la réplica se realiza en un solo sentido.

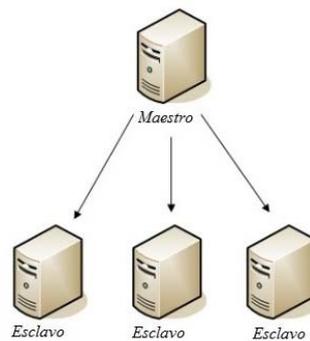


Figura 1: Entorno maestro-esclavo (Luaces, 2014)

- **Maestro-Maestro (*multi-master*):** Consiste en varias bases de datos que interactúan como pares iguales para participar en un modelo de actualización en cualquier lugar. Las actualizaciones realizadas en un sitio maestro individual se propagan a otro sitio maestro participante. Según (Keating, 2014) es un entorno de replicación de datos que permite que los datos sean almacenados por un grupo de computadores y actualizado por cualquier miembro del grupo. El sistema de replicación *multi-master* es responsable de propagar las modificaciones de datos realizadas por cada miembro con el resto del grupo. Entre las ventajas fundamentales se encuentra que cada miembro del grupo puede estar ubicado en cualquier lugar y permite leer o escribir

consultas que serán enviadas a varias computadoras replicadas, aumentando el rendimiento mediante la sincronización de los cambios entre servidores.

La siguiente figura muestra gráficamente como quedaría la réplica donde se utilice el entorno maestro-maestro.

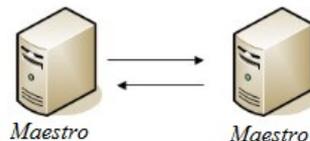


Figura 2: Entorno maestro-maestro. (Luaces, 2014)

Modelos de distribución de datos

Los Sistemas de Bases de Datos Distribuida (SBDD) tienen dos modelos de distribución que se aplican a cada uno de los entornos de replicación vistos, los cuales se basan en la medida de latencia para llevar a cabo la sincronización de bases de datos. La medida de latencia es la cantidad de tiempo que una réplica puede estar inconsistente hasta llegar a estar consistente con la fuente primaria designada. La sincronización es la actualización de las bases de datos en un ambiente de replicación mediante el intercambio de la información actualizada de cada nodo.

Los modelos de distribución son:

- **Asincrónica:** es llamada de consistencia débil respecto a los datos almacenados, ya que existe una latencia, es decir una desactualización de la copia replicada respecto a la original. La replicación asincrónica permite un alto nivel de autonomía en los sitios ya que un usuario puede trabajar sin estar conectado a la red, estas modificaciones se guardan en una bitácora local y una vez conectados a la red se actualizarán en el siguiente período de replicación.
- **Sincrónica:** también llamada replicación en tiempo real por su rapidez con que actualiza los cambios, la desactualización es casi nula y el tiempo de latencia casi es cero. La replicación sincrónica genera un alto nivel de sobrecarga en la red y no permite que los usuarios puedan trabajar desconectados. Un sistema que trabaje con este modelo solo soporta clientes tradicionales¹, ya que no permite manejar clientes móviles². (Oracle America, 2010)

¹ Clientes tradicionales: Computadores que mantienen conectividad de forma permanente.

² Clientes móviles: Computadores que no mantienen conectividad de forma permanente, entre los que se pueden encontrar las laptops.

Análisis de las herramientas de replicación de datos existentes

Al enunciar los conceptos fundamentales en la presente investigación, es necesario abordar sobre las herramientas de replicación de datos a nivel internacional que son privativas y libres:

Entre las **herramientas privativas** se encontraron:

EMC MirrorView

- Replicación remota basada en arreglos.
- Espejeado de datos independiente del host.
- Síncrono y Asíncrono.
- Administración centralizada. (Luján, 2015)

RecoverPoint

- Protección continua de datos.
- Replicación remota continua.
- Protección de datos local y remota simultánea.
- Reducción del ancho de banda.
- Registro diario de los cambios en los datos a nivel de bloques. (EMC Corporation, 2013)

EMC *MirrorView* y *RecoverPoint* a pesar de permitir la replicación asincrónica y proteger los datos, están implementadas sobre software propietario, esto no cumple con las necesidades del sistema de utilizarse sobre software libre, por lo que es necesario buscar herramientas que no sean propietarias.

Entre las **herramientas libres** se encontraron:

Pyreplica

Es una herramienta desarrollada en el lenguaje Python, permite réplicas maestro-esclavo y multi-maestro limitado, funciona de forma asincrónica y especialmente para el gestor PostgreSQL. Se caracteriza por ser fácil de usar, no se necesita aprender nuevos comandos para su administración, es muy fácil de adaptar manualmente, es multiplataforma y además permite:

- La replicación condicional.
- La detección de conflictos.

- El monitoreo de las conexiones.
- Notificaciones vía correo.
- La replicación de valores devueltos por funciones de fecha, aleatorias y secuencias. (Reingart, 2010)

PgCluster

Es un sistema de replicación sincrónico multi-maestro para PostgreSQL. Contiene tres tipos de servidores para realizar la replicación, un servidor para balance de carga, un clúster de base de datos y un servidor de réplica. Es compatible con los sistemas operativos Linux, Solaris y FreeBSD, pero no lo es con Windows. Este tiene dos funciones principales:

- Función de compartición de carga: la carga de la sesión de las demandas es distribuida. Es efectivo en aplicaciones web donde existe gran demanda por el número de peticiones.
- Función de alta disponibilidad: cuando ocurre un fallo en el clúster de base de datos, el servidor de balance de carga y el de replicación separan el fallo del sistema, y continúa el servicio.

PgCluster tiene una compleja instalación y configuración y puede requerir configuraciones avanzadas de hardware, realiza la replicación a nivel de sentencias, por lo que en algunos casos no se replican correctamente. El clúster de base de datos cuando es reparado puede restaurarse dinámicamente a un sistema, sin detener el servicio. Los datos son copiados automáticamente a la Base de Datos (BD) restaurada o añadidos desde otra. (Casaseca, 2015)

SymmetricDS

Es un software desarrollado en Java que permite la replicación de datos de forma asincrónica multimaestro y sincronización bidireccional la cual se define a nivel de tablas. El software está diseñado para replicar una gran cantidad de bases de datos, trabajar con conexiones de bajo ancho de banda y resistir a períodos de inoperatividad de la red. *SymmetricDS* soporta la sincronización entre diferentes plataformas de base de datos (MySQL, Oracle, SQL Server, PostgreSQL y Firebird) y puede ser utilizado tanto en el sistema operativo Linux como en Windows. Su uso está basado en HTTP o HTTPS, por lo que resulta ligero y fácil de manejar. Esta herramienta cuenta con una aplicación del lado del servidor y otra del lado del cliente, las cuales interactúan entre sí para realizar la réplica. Está liberado bajo los términos de software libre³. (Eric Long, 2013)

Dentro de las herramientas libres encontradas, se desecha Pyreplica ya que soporta réplicas maestro-esclavo y maestro-maestro limitado, y es imprescindible una herramienta que soporte réplicas multimaestro sin restricciones,

³ Software libre: se refiere a la libertad que tienen los usuarios para ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el software.

PgCluster no se escoge ya que soporta como modelo de distribución de datos el sincrónico. Por todo lo explicado anteriormente se selecciona la herramienta *SymmetricDS* porque soporta el entorno de replicación maestro-maestro asincrónico, es multiplataforma y funciona con un gran número de gestores de BD incluyendo PostgreSQL.

Configuración básica de la herramienta *SymmetricDS*

La herramienta *SymmetricDS* cuenta con varias tablas de configuración, en las que se definen los detalles, las más importantes serán explicadas de manera que se pueda realizar una correcta configuración para su entorno de réplica.

Las tablas de configuración de *SymmetricDS* serán creadas inicialmente vacías y la configuración se puede agregar en cada una de las tablas correspondientes para esta operación, además se puede configurar a través de un script o archivo nombrado *insert_sample.sql*, en este caso *sipp.sql*.

Una configuración básica describe lo siguiente.

- **Node** - No son más que los servidores de base de datos, donde se ejecutarán instancias de la herramienta *SymmetricDS*. Es importante y necesario organizar los nodos pues constituyen un factor primordial para la sincronización de los datos, clasificándolos según la información que manejarán y de esta manera definir los grupos de nodos, los cuales son los encargados de encaminar los datos. Aunque los datos que se deben sincronizar son prácticamente los mismos, cada institución maneja particularidades diferentes, la ruta para garantizar la sincronización de los datos es configurada mediante el grupo de nodo.
- **Node Group** - Cada nodo pertenece a un grupo. Una categoría de nodos que sincroniza datos con uno o más Grupos de Nodos. Un uso común de Grupos de Nodos es para describir un nivel en la jerarquía de la sincronización de datos.
- **Node Group Link** - Dos Grupos de Nodos son enlazados a través de la sincronización. Donde se especifica el esquema de notificación (*Push- P, Pull – W Wait for Pull* o personalizado por el usuario) a utilizar para enviar datos modificados en un grupo de nodo a otro grupo de nodos. Para establecer la sincronización entre Nodos, dos Grupos de Nodos deben ser enlazados entre ellos. La dirección de la sincronización es determinada por la especificación de un Grupo de Nodos fuente y uno destino. Si la sincronización ocurre en ambas direcciones, entonces dos enlaces son creados en direcciones opuestas. El Grupo de Nodos destino recibe los datos modificados por un método *Push* o *Pull*. Un método *Push* causa que el Grupo de Nodos fuente se conecte al destino, mientras que un *Pull* causa una espera hasta que el destino se conecte.

- **Channel** – Constituyen el medio por el cual viaja la información. En este se definen las prioridades de sincronización de los datos y pueden ser habilitados o deshabilitados.
- **Trigger** – Los disparadores se utilizan para registrar y capturar los cambios. Cada disparador se define teniendo en cuenta una tabla, el canal de transmisión y además se puede especificar los cambios definidos como: actualizar, eliminar o simples inserciones. Se definen tantos disparadores como tabla a replicar. (Lic. Vivian Romero Buchillón, 2010)

Durante la puesta en marcha, los *trigger* son verificados por cada una de las tablas de la base de datos, y son instalados en las tablas que requieren que los cambios sean capturados.

Después de conocer algunos de los conceptos fundamentales de SymmetricDS, a continuación, se hace un análisis más profundo de cada uno de los archivos que deben ser cambiados para asegurar una configuración exitosa. Es importante recordar que este proceso que a continuación se describe hay que hacerlo en todos los servidores de base de datos tanto en la Dirección de Intervención de Pozos (DIPP) como en los pozos donde se vaya a instalar el sistema (Pozo).

En primer lugar, se deben cambiar los archivos:

- *symmetric/samples/corp-000.properties*: que no es más que su nodo raíz. Solo se configura en el servidor central.
- *symmetric/samples/store-001.properties*: identifica a cada uno de los nodos asociados. Este archivo se modifica en cada una de las instituciones asociadas.

La configuración de los ficheros *corp-000.properties* y *store-001.properties* es de vital importancia puesto que cuando el SymmetricDS se inicia, lee estas configuraciones, así como el estado de las bases de datos.

Estos archivos se copian para la ruta *symmetric/engines*, y se le cambian el nombre por *dipp.properties* y *pozo.properties*. La configuración de estos archivos implica varias **propiedades** a tener en cuenta como:

- **engine.name**=dipp o pozo (Nombre de la Base de Datos).
- **db.driver**=org.postgresql.Driver, (Nombre de la clase del driver para JDBC).
- **db.user**=usuario, **db.password**=contraseña (El usuario y contraseña encriptada son para conectarse a la base de datos).
- **registration.url** (El URL donde el Nodo puede conectarse para registrarse y recibir su configuración.)

Ejemplos:

registration.url=http://localhost:8080/syn/dipp (en el caso de la dipp).

registration.url=http://localhost:8080/sync (en el caso del pozo)

- **group.id**=dippNode o pozoNode (El Grupo al cual pertenece el Nodo. La sincronización se realiza entre Grupos de Nodos.)
- **external.id**=0 (en el caso de la dipp) y **external.id**=00001 (en el caso del pozo)
(El ID externo del nodo, tiene un significado para el usuario y ofrece una integración en el sistema en que esté desplegado. El identificador externo se puede utilizar en expresiones condicionales. Cada nodo tiene un número de secuencia única para el seguimiento de eventos de sincronización. Es posible asignar el mismo identificador externo a varios nodos, si se desea.) (JumpMind, Inc, 2007-2015)

Configuración del Gestor de Base de Datos PostgreSQL versión 9.3 para la interacción con SymmetricDS 3.7.

PostgreSQL es un SGBD objeto-relacional, bajo licencia BSD⁴. Además de soportar todas las características de una base de datos profesional, es altamente adaptable a las necesidades del cliente y cuenta con un soporte nativo para los lenguajes más populares como: PHP, C, C++, Perl, Python, etc. (Martinez, 2013). Algunas de sus características son su alta concurrencia: mediante el sistema MVCC⁵ permite que un proceso escriba en una tabla mientras otros accedan a la misma sin necesidad de bloqueos y también que es una herramienta gráfica de diseño y administración de base de datos.

Para la configuración del Gestor de Base de Datos PostgreSQL es necesario agregar la variable de replicación “symmetric”. Esto se puede hacer por dos vías diferentes, la primera es haciéndolo directamente en la carpeta de archivos propio del Gestor de Base de Datos o dentro del servidor accediendo al archivo de configuración y agregando la variable. A continuación, se explican las dos formas de hacerlo.

Vía #1

- Acceder a la carpeta de archivo del Gestor de Base de Datos “C:\Archivos de programa\PostgreSQL\9.3\data”. Aquí se selecciona el archivo de configuración del Gestor de Base de Datos nombrado “postgresql.conf”.

⁴ Berkeley Software Distribution: Distribución de Software Berkeley

⁵ Multiversion Concurrency Control : Acceso Concurrente Multiversión

- Luego se abre el archivo con un editor de texto y al final del archivo en la opción “CUSTOMIZED OPTIONS” se agrega la siguiente línea:

```
custom_variable_classes = 'symmetric' # replication var
```

Vía #2

- Dentro del Servidor de Base de Datos, en la barra de Menú presionar la opción Archivo y seleccionar el sub menú “Abrir postgresql.conf”.
- Luego acceder a la carpeta donde se encuentran los archivos del Gestor de Base de Datos y abrir el archivo de configuración del gestor.
- En la opción del archivo “custom_variable_classes” hacer doble clic, habilitar el checkbox “Habilitado”, agregar en el campo de texto Valor la variable “symmetric” y en el campo de texto Comentario “replication var”. Para completar la acción se selecciona el botón OK. Para que funcione la nueva configuración se debe reiniciar el servidor de base de datos.

Después de haber configurado la variable de la herramienta de replicación faltaría configurar las bases de datos que van a estar involucradas en la replicación, para ello la herramienta de replicación define tres funciones de validación e instalación del lenguaje procedimental plpgsql. A continuación, se explica cuáles son las funciones a incluir y la forma correcta de insertarlas.

Ya después de insertada la base de datos de SIPP en el Gestor de Base de Datos, se debe desplegar el esquema “Público” y dirigirse a la opción “Funciones”, donde se selecciona el “Ejecutor de consultas SQL”, que muestra una ventana para ejecutar las siguientes funciones:

```
CREATE FUNCTION plpgsql_call_handler() RETURNS language_handler AS  
'$libdir/plpgsql' LANGUAGE C;  
  
CREATE FUNCTION plpgsql_validator(oid) RETURNS void AS  
'$libdir/plpgsql' LANGUAGE C;  
  
CREATE TRUSTED PROCEDURAL LANGUAGE plpgsql  
HANDLER plpgsql_call_handler  
VALIDATOR plpgsql_validator;
```

Advertencia:

Existen casos en donde la tercera función al ejecutarla ocurre un error en tiempo de ejecución porque el Gestor ya tiene implícito el lenguaje, en este caso solo se ejecutarán las restantes funciones.

Resultados y discusión

Funcionamiento del SymmetricDS

Para la replicación de los datos se debe haber configurado los archivos: sipp.sql, dipp.properties, pozo.properties y el Gestor de Base de Datos como se explicó con anterioridad. Se hace necesario ejecutar un conjunto de comandos que permitirán la correcta sincronización de las bases de datos perteneciente a los servidores de la entidad DIPP y de cada uno de los Pozos. A continuación, los siguientes pasos permitirán el correcto funcionamiento del SymmetricDS:

Paso 1:

Se necesita crear las tablas de SymmetricDS en la base de datos que guardarán la configuración de la sincronización. Se debe ejecutar por consola **cd symmetric/bin** y luego ejecutar los siguientes comandos:

En el servidor de la DIPP

```
symadmin --engine dipp create-sym-tables
```

En el servidor del Pozo

```
symadmin --engine pozo create-sym-tables
```

Al ejecutar estos comandos se crean todas las tablas necesarias del SymmetricDS en las bases de datos de los servidores.

Paso 2:

Se debe importar la configuración realizada en el script de base de datos y ejecutar los siguientes comandos:

En el servidor de la DIPP

```
dbimport --engine dipp sipp.sql
```

En el servidor del Pozo

```
dbimport --engine pozo sipp.sql
```

Paso 3:

Para iniciar los puertos y lograr la comunicación de los servidores se deben ejecutar los siguientes comandos:

En el servidor de la DIPP

```
sym –engine dipp –port 8080
```

En el servidor del Pozo

```
sym –engine pozo –port 9090
```

Se puede inicializar por otros puertos.

Paso 4:

El nodo central tiene varios nodos asociados, que deben ser registrados para que puedan obtener su carga inicial y de esta forma lograr recibir y enviar datos al nodo central. Para recargar la institución dipp se debe ejecutar la siguiente instrucción.

```
symadmin –engine dipp reload-node 1
```

Y de esta manera para cada uno de las instituciones asociadas al nodo central, especificando el grupo de nodo y el identificador externo. (JumpMind, Inc, 2007-2015)

Beneficios que aporta:

La replicación de los datos que maneja el Sistema Integral de Perforación de Pozos (SIPP) mediante la herramienta SymmetricDS garantiza:

- Centralizar la información generada en la DIPP y en los Pozos, facilitando la comunicación y actualización de la información generada.

Conclusiones

Como resultado del trabajo desarrollado se arribaron a las siguientes conclusiones:

De las soluciones similares existentes estudiadas se determinó que SymmetricDS cumple con el entorno de réplica y el modelo de distribución de datos necesarios para darle solución al problema, por lo que se escogió como herramienta que realizará la réplica de los datos.

La configuración del fichero sipp.sql permitió definir los servidores de base de datos, donde se ejecutarán las instancias de la herramienta SymmetricDS (Node), establecer la sincronización entre dos Grupos de Nodos (Node Group Link) y clasificar todas las tablas de la base de datos para lograr un orden de procesamiento.

Mediante los comandos explicados en el presente trabajo se podrán crear las tablas de SymmetricDS en la base de datos, importar las configuraciones del SQL, definir los puertos por lo que se va a comunicar cada servidor y registrar cada una de las instituciones implicadas en la replicación.

A partir de la integración de SymmetricDS con el Sistema Integral de Perforación de Pozos (SIPP) se logrará replicar los datos del sistema, permitiendo activar o desactivar la réplica, ver el estado de los datos replicados y manejar los posibles datos a replicar.

Referencias

- BUCHILLÓN, LIC.VIVIAN ROMERO, I. D. (2010).** Recuperado el 25 de octubre de 2016, de <http://www.monografias.com/trabajos83/symmetricds-herramienta-replicacion-codigo-abierto/symmetricds-herramienta-replicacion-codigo-abierto.shtml>
- CASASECA, G. H. (1 de agosto de 2015).** *Replicación de Datos en PostgreSQL*. Recuperado el 24 de octubre de 2016, de <http://es.slideshare.net/jockbrera/replicacion-postgresql>
- CHERENCIO, M. G. (2013).** *Replicación de Datos*. Recuperado el 24 de octubre de 2016, de <http://www.grch.com.ar/docs/bdd/apuntes/unidad.iv/Replicacion%20Slonik.pdf>
- EMC CORPORATION. (2013).** *Familia EMC RecoverPoint*. Recuperado el 24 de octubre de 2016, de <http://www.ipm.es/pdf/1259/>
- ERIC LONG, C. H. (2013).** Recuperado el 25 de octubre de 2016, de <http://www.symmetricds.org/doc/3.5/html-single/user-guide.html>
- GILLERMO ARAMBURU, M. L. (Agosto de 2010).** *Protección de los datos*. Obtenido de Rafael Pascual, Jose Genoud: <http://servicios.infoleg.gov.ar/infolegInternet/anexos/60000-64999/64790/norma.htm>
- JUMPMIND, INC. (2007-2015).** *SymmetricDS 3.7 Tutorials*. Recuperado el 26 de octubre de 2016, de <http://www.symmetricds.org/doc/3.7/html/tutorials.html>
- JUMPMIND, INC. (2007-2015).** *SymmetricDS 3.7 User Guide*. Obtenido de <http://www.symmetricds.org/doc/3.7/html/user-guide.html>
- KEATING, B. (2014).** *Database Specialists*. Recuperado el 24 de octubre de 2016, de Providing Oracle Database Administration & Consulting Services: <http://www.dbspecialists.com/author/bkeating/>

- KRONOS, J. T. (2010).** *Introducción a la Documática*. Obtenido de Los sistemas de bases de datos y los SGBD: <http://eprints.rclis.org/24735/1/Documatica.pdf>
- LONG, E. (31 de octubre de 2012).** *Sitio oficial SymmetricDS*. Obtenido de <http://www.symmetricds.org/docs/how-to/connect-to-database>
- LUACES, M. R. (2014).** *Bases de Datos III*. Recuperado el 25 de octubre de 2016, de <http://docplayer.es/1704305-Bases-de-datos-iii-curso-2013-2014.html>
- LUJÁN, U. N. (noviembre de 2015).** *Herramienta de Replicación EMC MirrorView*. Recuperado el 24 de octubre de 2016, de 4. <https://www.emc.com/collateral/white-papers/h12079-vnx-replication-technologies-overview-wp.pdf>
- MARTINEZ, R. (2013).** *PostgreSQL-es*. Recuperado el 24 de octubre de 2016, de Portal en español sobre PostgreSQL: http://www.postgresql.org.es/sobre_postgresql
- MENÉNDEZ, J. A. (2015).** *Utilización de las bases de datos relacionales y la gestión y almacenamiento de datos*. (S. Paraninfo, Ed.) Recuperado el 24 de octubre de 2016, de <https://books.google.com/cu/books?id=pbUbCgAAQBAJ&pg=PA153&lpg=PA153&dq=%E2%80%99CU+tipo+de+software+muy+espec%C3%ADfico,+dedicado+a+servir+de+interfaz+entre+las+bases+de+datos+y+los+sistemas+que+la+utilizan&source=bl&ots=qI2JNuiDTW&sig=PagAsLa9gGwJFLpEnE>
- ORACLE AMERICA. (septiembre de 2010).** *ORACLE*. Recuperado el 24 de octubre de 2016, de Guía de administración del sistema de Oracle Solaris Cluster: https://docs.oracle.com/cd/E22734_01/html/821-2812/gbbsi.html
- REINGART, M. (2010).** *PyReplica*. Recuperado el 24 de octubre de 2016, de Sistema de replicación simple para PostgreSQL programado en Python: www.postgresql.org.ar/trac/raw-attachment/wiki/PgDayUnnoba/PDF_PyReplica.pdf
- The IDG Network. (2002).** *Network World*. Recuperado el 24 de octubre de 2016, de Replicación de datos asíncrona: <http://www.networkworld.es/archive/replicacion-de-datos-asincrona>