

Instituto Tecnológico de Costa Rica

IC4302 Bases de Datos II – GR 20

Prof. Alberto Shum Chan

Práctica de Replicación en PostgreSQL

Jonathan Quesada Salas

Alberto Zumbado Abarca

II Semestre

2021

3. Evalúe las tres herramientas teniendo en cuenta los requerimientos del proyecto.

Para la evaluación llene la siguiente tabla:

Nombre de la herramienta	Herramienta 1 Replicación en cascada	Herramienta 2 Slony-I	Herramienta 3 Replicación Hot Standby / de transmisión
Desarrollada por:	PostgreSQL Global Development Group	PostgreSQL Global Development Group	NTT OS Center para PostgreSQL
Documentación disponible en:	https://www.postgresql.org/docs/13/warm-standby.html#CASCADING-REPLICATION	https://www.slony.info	https://www.postgresql.org/docs/13/warm-standby.html#STREAMING-REPLICATION
¿Es software libre? / Licencia de uso	Al ser una herramienta incluida en PostgreSQL desde la versión 9.2, la misma está regida bajo la licencia de "PostgreSQL License" la cual es open Source. La licencia de uso está incluida en el paquete de instalación.	Slony es una herramienta distribuida bajo los términos de la licencia PostgreSQL License y la licencia de la Universidad de California. Se permite el uso para todo propósito sin cobro.	Al ser una herramienta incluida en PostgreSQL desde la versión 9, la misma está regida bajo la licencia de "PostgreSQL License" la cual es open Source. La licencia de uso está incluida en el paquete de instalación.
Tipo de replicación que implementa	La replicación en cascada es asincrónica.	La replicación de Slony es asincrónica de tipo Master hacia multiples servidores standby, se puede esperar que esté entre 1 y 20 segundos por detrás del maestro.	La replicación de transmisión es asíncrona de forma predeterminada. Permite replicación sincrónica desde la versión 9.1.
Forma de transmisión de los datos	Forma de transmisión continua en sentido descendente mientras haya nuevos registros WAL disponibles. Disponer réplicas a varios servidores primarios para hacer la replicación a servidores esclavos.	Funciona como un método de replicación basado en triggers que es una solución de "maestro a varios esclavos". Slony opera instalando triggers en cada tabla que se va a replicar, tanto en el maestro como en los esclavos, y cada vez que la tabla obtiene un INSERT, UPDATE o DELETE, registra qué registro se cambia y cuál es el cambio.	El servidor standby se conecta al primario, el cual transmite los registros WAL al standby mientras estos mismos son generados, sin esperar a que el archivo se haya terminado de llenar. Hay un pequeño retraso entre la confirmación de una transacción en el primario y los cambios que se hacen visibles en el modo de espera. Sin embargo, este retraso es mucho menor que con el envío de registros basado en archivos.
¿Permite que el esclavo realice	Sí, porque en el caso que el servidor esclavo está realizando una nueva sincronización,	Pueden realizar solamente consultas de lectura.	Los servidores standby solamente pueden realizar consultas de lectura, debido a la

consultas a la base de datos?:	se puede usar la información de datos anterior para proporcionar consultas		configuración de Hot Standby.
Complejidad de instalación	Complejidad media, ya que se deben sincronizar instancias como la red, necesidades, para que de esta manera desarrollar las previstas de un plan de continuidad.	Complejidad media baja, ya que la mayoría de los componentes a instalar, se deben de hacer mediante comandos en terminal, por lo que puede ser difícil para un usuario sin experiencia.	Complejidad media ya que este tipo de replicación necesita una configuración específica en los diferentes archivos de la base de datos con lo que respecta a las transacciones de datos.
Complejidad de administración	Complejidad media ya que se debe de enmarcar muy bien las conexiones de replicación y transmitirlos a los registros WAL.	Complejidad media baja, ya que consta de una interfaz visual que permite realizar las demás configuraciones de una forma intuitiva.	Complejidad media, ya que la autenticación del cliente para la replicación se controla mediante una configuración específica. Aparte el nombre del host y el número del puerto primario, nombre y contraseña del usuario tiene una especificación delimitada.
Ventajas	Se puede ver recudido la cantidad de conexiones al servidor principal y se llega a reducir los gastos generales del ancho de banda. Actúa como receptor y remitente de modo que se puede apreciar un comportamiento en cascada entre servidores. No impone límites en el número de servidores descendentes.	Interfaz visual Independiente de la plataforma de los nodos. Se puede realizar en enlaces de red ocultos. Soporta diferentes versiones de postgres. Permite seleccionar que bases de datos y que tablas de la base de datos se replicaran.	Es sencillo de implementar. Todo lo que haga el servidor master se replica en el standby. Puede usarse para aligerar cargas. Si utiliza la replicación de transmisión por secuencias sin un archivado continuo basado en archivos, el servidor puede reciclar los segmentos WAL antiguos antes de que los haya recibido el modo de espera. Si esto ocurre, será necesario reiniciar el modo de espera desde una nueva copia de seguridad base.
Limitaciones	Existe un modo de espera en este tipo de replicación envía esta restricción a los registros WAL recibidos del maestro, sino también a los restaurados desde el archivo.	No permite replicar automáticamente: Cambios por consultas de DDL, cambios realizados por usuarios y cambios en los BLOBs.	No permita especificar base de datos o tablas a replicar. No se puede cambiar el esquema del servidor standby. Ambos dispositivos deben tener versiones de Postgre similares.

			Asegurarse de que los segmentos WAL no se reciclen demasiado pronto o configurando una ranura de replicación para el modo de espera. Si configura un archivo WAL al que se puede acceder desde el modo de espera, estas soluciones no son necesarias, ya que el modo de espera siempre puede usar el archivo para ponerse al día siempre que conserve suficientes segmentos.
¿Existe un grupo de desarrolladores dando soporte a la herramienta?	Grupo de desarrollo global de PostgreSQL.	Grupo de desarrollo global de PostgreSQL.	Grupo de desarrollo global de PostgreSQL.
Existe documentación	Existe documentación del equipo de desarrollo de la replicación y también en páginas diversas en la web.	Existe documentación del equipo de desarrollo de la replicación y también en páginas diversas en la web.	Existe documentación del equipo de desarrollo de la replicación y también en páginas diversas en la web.
Costo de la herramienta	Es gratis ya que forma parte de la instalación de la base de datos.	Se permite el uso para todo propósito sin cobro.	Es gratis ya que forma parte de la instalación de la base de datos.

4. Seleccione la herramienta que, de acuerdo a lo investigado, solventa las necesidades del proyecto. Explique por qué recomienda la herramienta.

Herramienta 1: Por la característica de no imponer límites del número de servidores, se puede aprovechar esto para poder repartir subfunciones, o bien repartir el proyecto en general entre varios servidores para que de esta manera poder tener una mejor disponibilidad y el factor de que se puede comportar como un receptor y remitente da paso a tener un mejor flujo de información con lo que respecta a la base transaccional que se pide en la especificación por medio de está replicación

Herramienta 2: Mediante este método pueden encontrarse muchas ventajas, como la optimización del nodo maestro, posee amplio soporte de parte de la comunidad y los equipos de desarrollo, al igual que las distintas versiones de postgres, sin embargo, falla un poco a la hora de detectar fallas en los nodos de algún cluster, por lo que para conexiones grandes esto puede llegar a ser perjudicial, y en ciertos dispositivos requiere instalación de software adicional. Además, a pesar de no ser tan complejo de utilizar el mínimo error puede echar a perder toda la conexión entre los distintos servidores.

Herramienta 3: El hecho que la replicación se base en secuencias puede traer un mayor orden en cuanto al tipo de replicación que se quisiera manejar, siendo igual de

esta manera con que el servidor puede reciclar archivos WAL, puede dar una salvada a archivos que uno haya dejado de lado por equivocación, aprovechando esto no tanto para tener una mejor disponibilidad sino una mejor resolución del manejo de archivos, o bien de datos de transaccionales. El hecho que los nodos esclavos lleguen a ser una parte más de la conexión a la base de datos para la obtención de los registros WAL, permite que el gasto sea bajo, y que la latencia de replicación sea muy baja (esto en forma asincrónica, el modo sincrónico no hay pérdida de datos, pero es más propenso a fallos).

Debido a las necesidades del proyecto con respecto a implementar un datamart, esto a largo plazo supondría tener una gran cantidad de datos debido a las características del datamart de ser un subconjunto de datos, optamos que la mejor herramienta a replicar seria la de **Streaming replication**, ya que el hecho de simplemente configurar la base de datos junto al WAL Shipping se nos hace una tarea sencilla debido a la cantidad de información que existe. De igual forma la implementación de esta podría suponer a largo plazo el uso de mas de un servidor standby para mejorar la replicación, la latencia que este genera es corta por lo que no va a ver en caso de fallas una pérdida de datos significativa. A pesar de sonar como limitante el hecho de no poder seleccionar bases de datos y tablas especificas a replicar, por las características del datamart sentimos que al final toda la información que puede llegar a ser implementada en la base de datos es de importancia, por lo que la necesidad de poder replicar todo es grande, y de igual forma los servidores el hecho que los standby puedan hacer consultas nos parece una herramienta provechosa. Al final se tomarán las medidas necesarias para que no existan inconvenientes con respecto a las versiones de Postgre.