

Laura Castro	l.castro12	201423296	
Esteban Galán	e.galan	10	201425045
Margarita Gómez	mm.gomez10	201423591	
Nicolás Gómez	n.gomez22	201426109	
Jose Gabriel Tamura	jg.tamura10	201424484	
Nicolás Vásquez Murcia	n.vasquez10	201424562	

Iteración 5

Análisis de la implementación de transacciones distribuidas

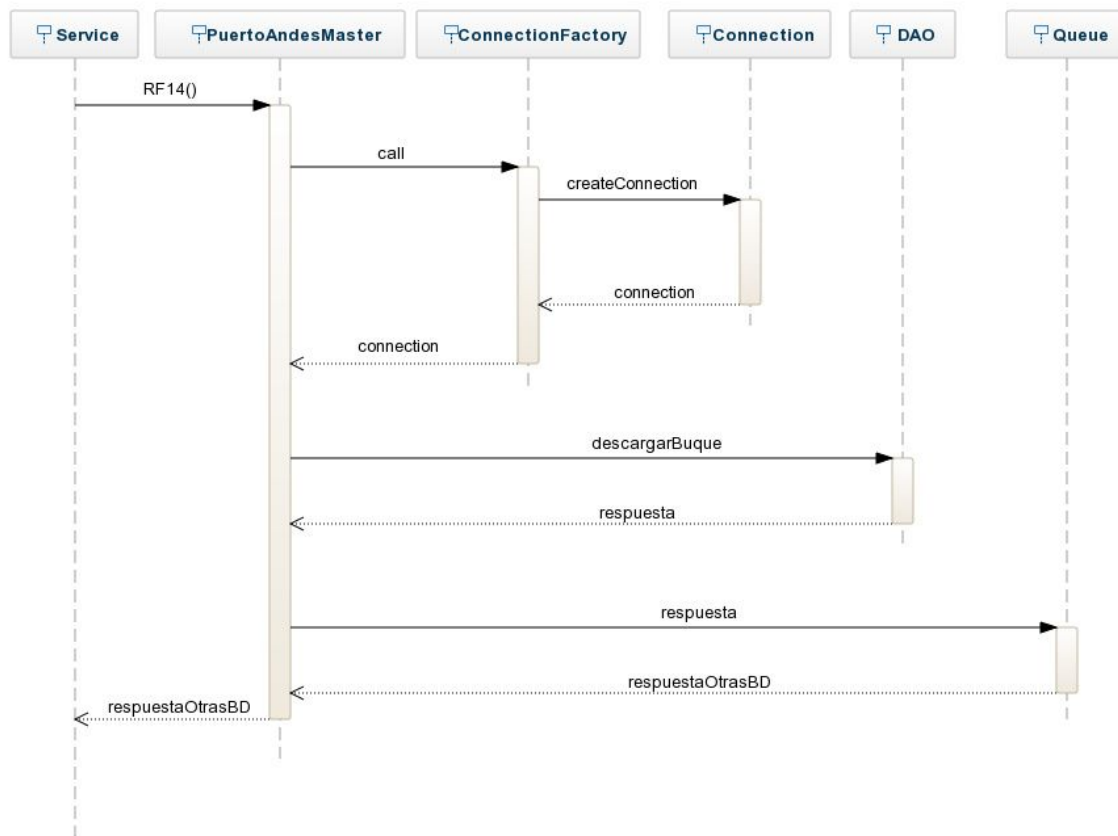
Restricciones

Las restricciones existentes para los tres sitios que se van a montar en la aplicación PuertoAndes³ son: que a la hora de implementar el RF14, como se realiza por medio de colas de mensajes, y esto se resuelve de manera sincrónica, los tres sitios disponen de la posibilidad tanto de encolar una solicitud, como de desencolar una proveniente de otro, seguido siempre de una respuesta, ya sea de éxito o de fracaso de la transacción correspondiente. Una vez se encola la respuesta, esta debe ser desencolada para así proceder de acuerdo a como dicte el protocolo. Ahora bien, para el RF15 se tienen las mismas restricciones en cuanto a su implementación con colas de mensajes pues también debe esperar una respuesta en cuanto a si se pudo o no calcular el bono respectivo y continuar de acuerdo al protocolo. En cuanto al uso de Two-Phase Commit del RF15, se hace uso del XA con el objetivo de acceder a las tres bases de datos en una misma transacción, asegurando su total consistencia. Siendo así, las restricciones de la manera como procede una base de datos depende del éxito de las subtransacciones que la acompañan. Los requerimientos de consulta obedecen las restricciones previamente mencionadas de acuerdo a la estrategia de comunicación que estas siguen.

Ajuste de la arquitectura

Para implementar los nuevos requerimientos, los cambios realizados en la arquitectura consisten en la creación de atributos, con sus respectivas inicializaciones que representan tanto las colas de mensajes, como las conexiones a las bases de datos de los tres grupos. Para aquellos requerimientos que son resueltos utilizando como estrategia de comunicación colas de mensajes, se hace uso de un Sender y un Receiver. De esta manera, cada parte de PuertoAndes³ es capaz de añadir y remover solicitudes de la cola, así como respuestas. En cuanto a los requerimientos que utilizan Two-Phase Commit, se crean las nuevas conexiones correspondientes a las nuevas bases de datos, para así, al tener acceso a ellas, recuperar los datos pertinentes de acuerdo a la consulta u operación indicada.

Diagrama de secuencia RF14



Análisis de estrategias

Analice las estrategias que deben ser desarrolladas para el cumplimiento del requerimiento solicitado, tanto para el caso de uso de colas de mensajes, como de two phase commit. Compare los resultados que se obtendrían al implantar las dos estrategias.

Al implementar el protocolo Two Phase Commit, el requerimiento no va a ser completado satisfactoriamente si alguna de las bases de datos no logra desarrollar la tarea. Entonces, para que la petición de consulta o modificación sea realizada, se necesita que cada una de las bases de datos ejecuten y devuelvan correctamente la solicitud del cliente. Para cumplir un requerimiento con este proceso se debieron seguir varios pasos. Primero, el controlador, o monitor de toda la transacción, le va a pedir a cada base de datos que cumpla con el requisito. Este va a esperar que cada una le devuelva un mensaje informando si lo logró o no. Si todos lo lograron, le va a pedir a todas que realicen commit para confirmar el cambio en todas las bases de datos. Pero si alguna informa que se encontró con un inconveniente para satisfacer el requisito, se le va a pedir a todas que hagan rollback y no se esperaría ninguna modificación en

los datos. Por esa razón, esta implementación es sincrónica, donde se ejecuta en todas las bases de datos o en ninguna.

Por otro lado, en la implementación de colas de mensajes, se realiza de manera asincrónica. Primero, se le pide a una base de datos que, por medio de la conexión con las otras, realice el requerimiento pedido por un cliente. Por lo tanto, primero se va a demandar de manera local la solicitud, si ésta no puede ser concluida, se va a enviar un mensaje a la cola de una de las conexiones que se tiene con las otras bases de datos. De esta manera, se le pide a esta que termine la tarea. Y si esta no puede, se lo va a pedir a la siguiente. Por lo tanto, lo que demandó el cliente va a ser completado y distribuido entre las tres bases de datos. Esto no significa que cada base de datos realiza la misma tarea.

La diferencia entre la implementación de estos dos procedimientos, es que para two phase commit, si no se completan las tareas en cada base de datos, lo que pidió el cliente no va a ser ejecutado, si no es realizado en su totalidad. Mientras que, para las colas de mensajes, si puede pasar que unas, o todas, de las bases de datos sufran cambios (con commit) por el requerimiento. Cada base de datos va a ir realizando las tareas que le llegan por la cola, sin tener en cuenta si las otras bases de datos lo pudieron hacer. Pero en two phase commit, la modificación para cada base de datos si depende de las otras. Por esto, los resultados van a ser distintos para cada protocolo, teniendo en cuenta las especificaciones del cliente y del requerimiento.

Especificación e implementación de transacciones distribuidas con colas de mensajes

Alistamiento de las bases de datos y de servicios

Bases de datos listas desde todas las sesiones.

Implementación RF14 - Colas de mensajes

Se implementó el método de descargue de buques con colas de mensajes a partir de un protocolo previamente establecido por los integrantes. Dicho protocolo consiste en enviar un String con partes separadas por "-". La primera parte llega como el número del requerimiento, en este caso 14. Después de esto se debe separar todo de la siguiente manera: ["tipo carga": "cuanta carga llega"]- *.

Implementación RF15 - Colas de mensajes

Se implementó el método de descuentos con colas de mensajes a partir de un protocolo previamente establecido por los integrantes. Dicho protocolo consiste en enviar un String con partes separadas por "-". La primera parte llega como el número del requerimiento, en este caso 15. En la cola de mensajes se pregunta si hay un exportador con id x. Si existe un exportador con ese id entonces se responde de forma afirmativa y si no de forma negativo.

Implementación RF15 - Two Phase Commit

Se implementó el método de descuentos con Two Phase Commit a partir de un protocolo previamente establecido por los integrantes. Dicho protocolo consiste en enviar un String con partes separadas por "-". La primera parte llega como el número del requerimiento, en este caso 15. En la cola de mensajes se pregunta si hay un exportador con id x. Si existe un exportador con ese id entonces se responde de forma afirmativa y si no de forma negativo.

Implementación RFC11 y RFC12

En los requerimientos de consulta se enfatizó en asegurar la rapidez de la transacción. Por tal motivo decidimos implementar el protocolo two face commit. Estos requerimientos de consulta funcionan y tienen un buen desempeño optimo.

Estrategia global más indicada para PuertoAndes³

Analice el impacto de las estrategias de transacciones distribuidas para cada requerimiento adicional solicitado, de forma independiente. Analice cuál es la estrategia global más indicada para PuertoAndes3 de acuerdo con sus reglas de negocio y las condiciones esperadas de operación, en el cumplimiento de todos sus requerimientos como integrador de las unidades de negocio. Justifique plenamente su análisis.

- RF14: Para este requerimiento se utilizó una estrategia de colas de mensajes. En este caso, utilizar esta estrategia presenta una ventaja a la hora de completar el requerimiento dado que se aprovecha la conexión con otras bases de datos para complementar lo que un solo puerto no es capaz de hacer. Originalmente cuando no habían almacenamientos disponibles para un cargamento que llegaba en un barco, este simplemente se ignoraba y se dejaba en el buque. Ahora, con la nueva estrategia utilizada, mediante la cola de mensajes y el protocolo definido, se puede comunicar con los otros puertos para descargar más efectivamente los puertos.
- RF15: Este requerimiento utilizó una estrategia de Two Phase Commit. En este caso, esto resultó ser la mejor opción dado que solo fue necesario realizar una consulta muy simple en cada base de datos.

Por otro lado, también se utilizaron colas de mensajes para cumplir el requerimiento. Este se basa en pedir de manera local la consulta, y luego de ser ejecutada, le manda un mensaje a las otras bases de datos que busquen si se cumple para estas también. Si esto sucede, se aplica el bono, de lo contrario no.