

RC: Actividades Tema 2

ACTIVIDAD 1

Antes de que se popularizaran los servicios de pertenencia descritos en la primera sección de este tema, algunos algoritmos describieron ciertos mecanismos de detección de fallos. Por ejemplo, las primeras propuestas de servicios replicados utilizaron el modelo de replicación pasivo. En ese modelo existe una réplica primaria que gestiona directamente las peticiones originadas por los clientes. Cuando han sido procesadas, difunde las modificaciones resultantes al resto de réplicas (réplicas secundarias). Para detectar la caída de la réplica primaria, cada réplica secundaria emitía periódicamente un mensaje a la primaria y éste lo respondía. Se fijaba un tiempo límite para esa respuesta y si la respuesta no llegaba dentro de ese plazo se asumía que la réplica primaria había dejado de funcionar.

1. ¿Qué ventajas o inconvenientes plantea este mecanismo de detección de fallos cuando es comparado con un servicio de pertenencia?

Ventajas que puede tener el modelo de replicación pasivo es que es muy fácil de implementar. Tiene un coste muy bajo en comunicaciones y es muy útil para el caso de tener solo 2 nodos.

Inconvenientes es que harían falta detectores perfectos para poder detectar bien las sospechas del nodo principal y si no se dispone un detector perfecto se dificultará la elección de un nodo primario nuevo.

2. ¿Qué grado de completitud y qué grado de precisión tendría este mecanismo si fuera considerado un detector de fallos? Justifíquelo, complementando la descripción dada en este enunciado (en caso de que sea necesario) para adaptarla al grado que Vd. considere más cercano.

Dependiendo del detectores de fallos, pero en el peor caso el grado de completitud es débil porque en el caso en el que el nodo principal fallara, algunas de las replicas lo detectarían y sospecharán de él. A la vez cabe la posibilidad de que la replica secundaria tendría algún fallo y esta no detecte el fallo de la replica principal.

La precisión también depende de la red, dado que si se tiene un sistema puramente síncrono con detectores perfectos el grado que tiene es fuerte dado que se está monitorizando una solo nodo primario y hasta que este no responda no será sospechado.

3. ¿Qué ocurriría si cada secundario estuviese ubicado en una subred diferente y fallara una de las subredes en lugar de fallar el primario?

En el caso en el que fallara una de las subredes, al estar aisladas, esta se configuraría para funcionar como un nodo principal ya que no tiene comunicación con las demás subredes para comprobar su sospecha. Además como el nodo principal tampoco revisa el estado de su replicas podría haber un momento en el que hay dos nodos principales si no se realiza alguna comprobación para impedir este caso.

4. En este ejemplo, ¿qué ocurriría si se complementara el mecanismo de detección descrito con la ejecución subsiguiente de un algoritmo de elección de líder para seleccionar una nueva réplica primaria? ¿Sería equivalente a un servicio de pertenencia? ¿Qué diferencias habría?

En el caso de que se utilice un algoritmo de elección de líder para seleccionar una nueva réplica primaria, los nodos secundarios deben de ponerse de acuerdo mediante un consenso. En el caso en el que se sospecha del nodo principal, se debería de confirmar por todos y no solo por una minoría, que es el caso. Por ello no se puede considerar esta mejora un equivalente a un servicio de pertenencia porque que no se conocen los nodos secundarios que funcionan de manera adecuada y cuales no. En este caso, si un nodo sospecha del nodo principal, solo se lanzaría una petición de puesta en común de un nuevo líder.

ACTIVIDAD 2

Justificar por qué un servicio de pertenencia no puede implantarse de manera precisa en un sistema distribuido asíncrono. Describir cómo se han resuelto esas dificultades a la hora de implantar los detectores de fallos. ¿Se puede implantar un detector de fallos perfecto en un sistema asíncrono?

Un servicio no puede implementarse de manera precisa un servicio de pertenencia en un sistema distribuido asíncrono dado que al ser asíncronos, no se puede diferenciar de manera exacta si un nodo ha fallado o tiene un retardo.

A la hora de detectar los fallos se crea un sistema de pertenencia a grupos donde se crea también una vista con los nodos que están funcionando actualmente. Si se añade un nuevo nodo o un nodo falla, se crea una nueva vista y se informa a todos los nodos restantes de la existencia de esta nueva vista. Para ello también se tiene un algoritmo de consenso para detectar de manera más exacta posible si un nodo ha fallado o no.

No es posible implementar un detector de fallos perfecto, lo que se puede hacer es establecer un tiempo límite de respuesta que sea suficientemente grande para que el nodo responda. Si el nodo no ha contestado antes del tiempo límite establecido, se considera que el nodo a fallado y si hay un consenso por el resto de nodos entonces se crearía una vista nueva con los nodos restantes. El problema viene dado al estar en un entorno asíncrono.

ACTIVIDAD 3

En la hoja 17 de la presentación se muestra un algoritmo que es capaz de transformar un detector con completitud débil en otro con completitud fuerte. Responda a las siguientes cuestiones relacionadas con ese algoritmo de reducción:

1. ¿Permite este algoritmo implantar detectores de fallos perfectos?

No puede implementarse un detector de fallos perfecto, dado que si uno de los nodos sospechan de un nodo, este difunde sus sospecha a los demás nodos y lo aceptan sin comprobar si es un falso positivo.

2. Cuando la comunicación es fiable (esto es, los mensajes no se pierden ni se corrompen y siempre llegan a entregarse en un plazo aceptable aunque desconocido a priori), ¿qué propiedad puede garantizarse más fácilmente, la completitud o la precisión?

De completitud, ya que los procesos que han fallado se considerarán sospechosos. En el caso de que se da un falso positivo, entonces se sospechará de un proceso correcto y por lo tanto no se cumple la propiedad de precisión.

3. En un sistema distribuido asincrónico... ¿Podrá haber precisión fuerte?
¿Y completitud fuerte? ¿Por qué?

No puede haber fuerte dado que es muy difícil diferenciar en un sistema distribuido asíncrono si un proceso ha fallado o ha tardado más tiempo en contestar. Si se da este caso, se llegaría a sospechar de un sistema correcto y la propiedad de la precisión fuerte es que ningún proceso es sospechado antes de fallar.

Completitud fuerte si que puede haber dado que con el paso del tiempo el proceso fallido será detectado por todos los procesos correctos. Todos los procesos correctos pueden comprobar la información que tienen del proceso fallido y compararla con la información recibida por los demás procesos y llegar a un consenso.

ACTIVIDAD 4

Encontrar al menos tres referencias bibliográficas que propongan detectores de fallos que no aparezcan directamente en la clasificación de Chandra y Toueg. Proporcione la referencia bibliográfica, el nombre de la clase de detector propuesto y un párrafo que relacione ese nuevo detector con la taxonomía de Chandra y Toueg.

"Implementation and performance evaluation of an adaptable failure detector", por Marin Bertier, Olivier Marin y Pierre Sens.

Se explica como crean una variante de la implementación basado en el modelo de diamante de Chandra y Toueg que se puede adaptar y escalar de manera más optima.

"Fault-tolerant broadcast in anonymous distributed systems with fair lossy channels" por J. Tang, M. Larrea, S. Arévalo, E. Jiménez.

El identificador que se utiliza en el artículo para la identificación de procesos se basa en el de Chandra y Toueg, aunque se explica como realizan una ampliación.

"Byzantine Fault Detectors for Solving Consensus" por K. P. Kihlstrom, L. E. Moser, P. M. Melliar-Smith.

Se explica como amplían el trabajo de Chandra y Toueg para que los detectores de fallos no fiables pueden resolver el consenso en sistemas asíncronos distribuidos sujetos a fallos bizantinos.