Servicio de almacenamiento basado en primario/copia

Eduardo Gimeno Soriano

Sergio Álvarez Peiro

13 de enero de 2019

Introducción

En esta práctica el objetivo es diseñar e implementar el servidor de un servicio de almacenamiento tolerante a fallos basado en el sistema de vistas primario/copia. Para llevar a cabo esta práctica nos tenemos que apoyar en el servicio implementado en la anterior.

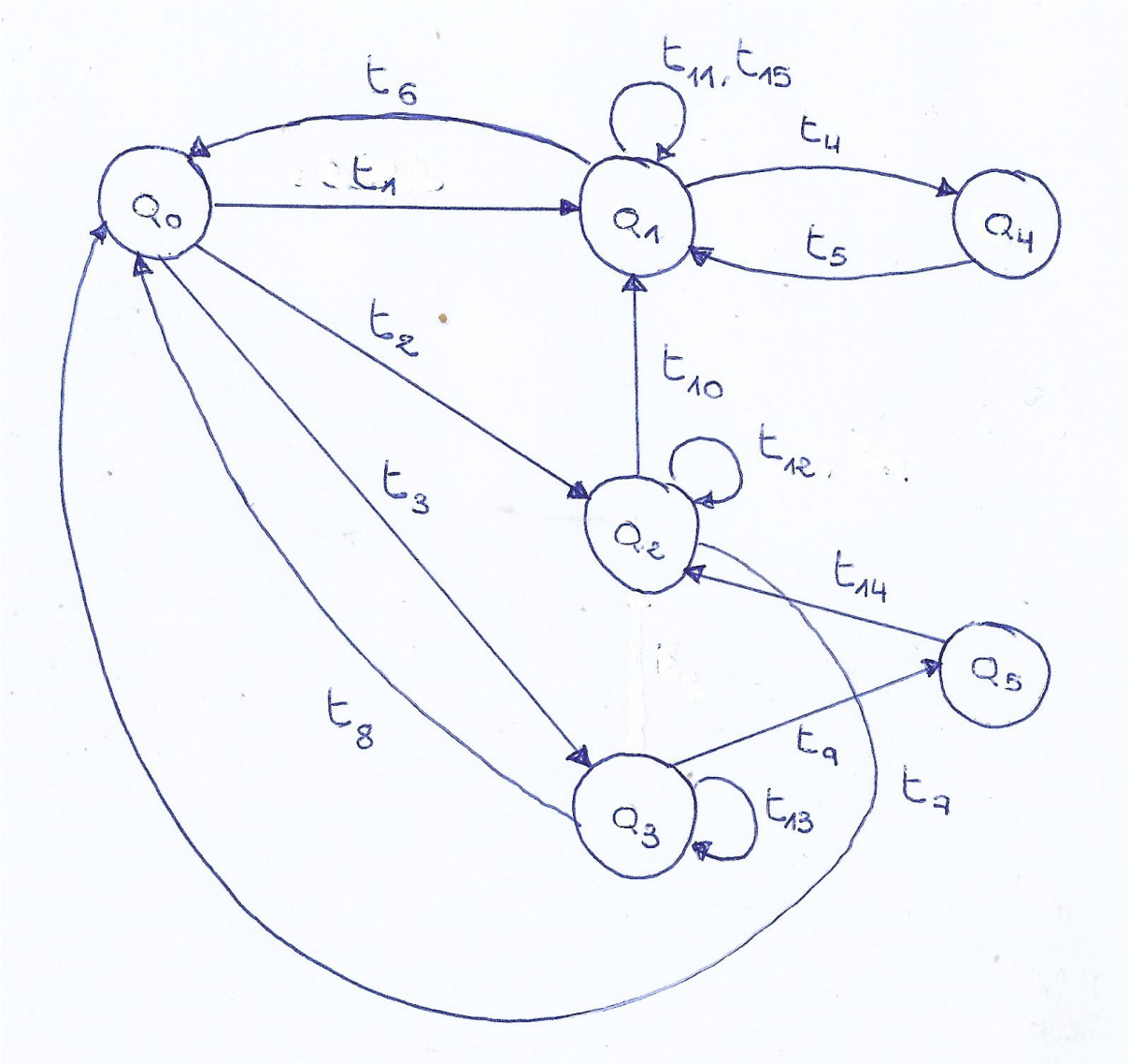
Sistema de almacenamiento basado en primario/copia

El sistema de almacenamiento distribuido funciona mediante clave/valor. Los clientes pueden enviar peticiones de lectura y escritura de manera concurrente al servidor y recibirán una respuesta con datos o confirmación. Para mantener la tolerancia a fallos, en el servidor se utiliza el sistema de gestor de vistas ya implementado, por lo que se tiene que mantener la consistencia de los datos entre copias. A continuación se explican los 4 mensajes que puede recibir el servidor:

1. Leer. Cuando se recibe una petición de lectura, se comprueba que el nodo es el primario la clave de la petición contiene datos que devolver. Si esto se cumple se envían los datos. Esto solo se hace si el nodo está en una estado válido. Si el mensaje no es un nodo primario o no es una vista válida se envía un mensaje de error.
2. Escritura genérica. Si lo recibe un nodo primario, se comprueba que es una vista válida y si lo es se escribe el valor en los datos a la vez que se envía una petición de escritura a la copia y se envía la confirmación al cliente. Cuando el mensaje lo recibe la copia y es válida escribe el dato en su base de datos y se envía confirmación al primario. En caso de que no sea ni primario ni copia o no sea válida se envía un mensaje de error.
3. Latido. Para asegurarse de que los nodos siguen vivos se utiliza el sistema de latidos como en la práctica anterior. Cada vez que se recibe un latido se actualiza el estado. Si no hay primario, se pone el número de vista a -1. Si hay primario se establece una copia y se envia petición de copia de datos.
4. Copiar datos. El nodo copia recibe los datos del nodo primario para almacenarlos en su estado.

El estado del servidor se compone del número de vista, el identificador del nodo primario, la copia y si es una vista válida. También los datos del servidor (claves y valores de las lecturas y escrituras) que se tienen que mantener consistentes entre los nodos primario y copia.

A continuación se muestra el autómata de estados:



Estados:

- Q0: Nodo indefinido.
- Q1: Nodo primario.
- Q2: Nodo copia.
- Q3: Nodo en espera.
- Q4: Nodo primario y a la espera de recibir confirmación de escritura de la copia.
- Q5: Nodo en espera y a la espera de copiar los datos del primario.

Transiciones:

- t1: El nodo recibe la vista por parte del gestor de vistas y queda como primario.
- t2: El nodo recibe la vista por parte del gestor de vistas y queda como copia.
- t3: El nodo recibe la vista por parte del gestor de vistas y queda como nodo en espera.
- t4: Recibe petición de escritura y la propaga a la copia.
- t5: Recibe confirmación de la copia. datos: {X*Y}.
- t6, t7, t8: El nodo ha caído.
- t9: Recibida vista, el nodo ha sido promocionado a copia.
- t10: Recibida vista, el nodo ha sido promocionado a primario.
- t11: Recibida vista, el nodo se mantiene como primario.
- t12: Recibida vista, el nodo se mantiene como copia.
- t13: Recibida vista, el nodo se mantiene como nodo en espera.
- t14: Recibida copia de los datos por parte del primario.
- t15: Recibe petición de lectura.

Salidas:

Estado	**num_vista**	**primario**	**copia**	**valida**	**datos**
Q0	0	indefinido	indefinido	false	{}
Q1	X	self	X	true	{X*}
Q2	X	X	self	true	{X*}
Q3	X	X	X	true	{}
Q4	X	self	X	true	{X*}
Q5	X	X	self	true	{}

Validación

Para validar el correcto funcionamiento del sistema, se realizan 4 test tanto como en local como con los servidores en máquinas distintas:

1. Arranque y parada del sistema.
2. Escritura tras caerse nodo copia.
3. Escritura concurrente y comprobación de consistencia de datos tras caída del nodo primario.
4. Escritura concurrente y comprobación de consistencia de datos tras caída del nodo primario y nodo copia.

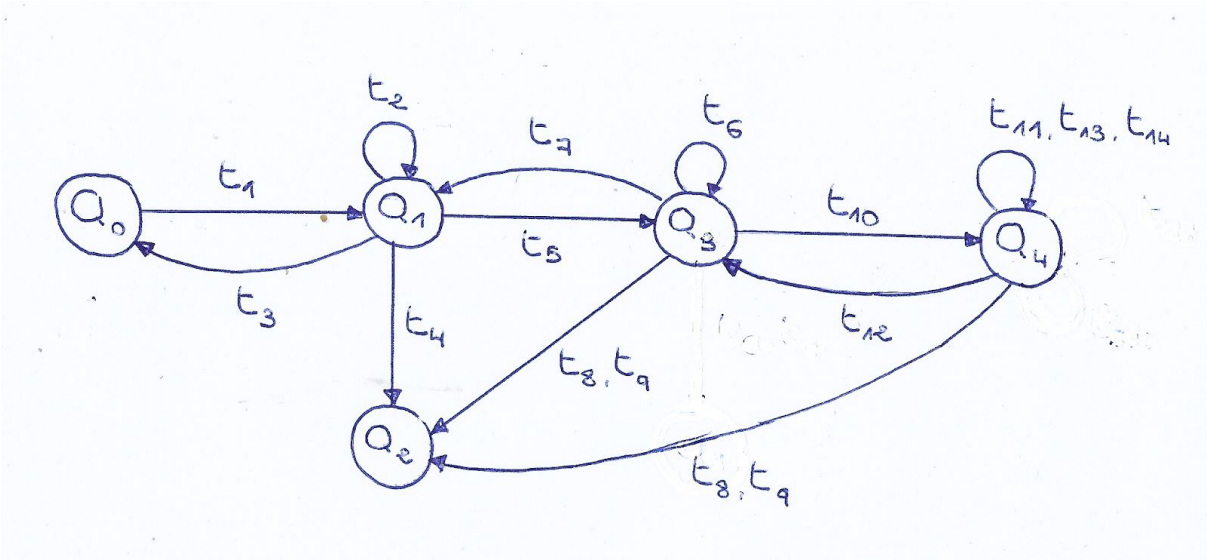
El resultado de la ejecución de validar_servicios_almacenamiento ejecuta los tests listados e informa por pantalla que se han superado todos.

Conclusiones

El sistema de almacenamiento distribuido clave/valor necesita ser tolerante a fallos si se implementara en un entorno real, ya que en esa situación se producirían fallos constantemente y sería inservible. Por eso al combinar el concepto de primario/copia podría funcionar siendo tolerante a fallos en red.

Anexo 1. Sistema de gestión de vistas

A continuación se explica el autómata de estados del sistema gestor de vistas.



Estados:

- Q0: Primario y copia indefinidos.
- Q1: Primario definido y copia indefinida.
- Q2: Se ha perdido la consistencia, sistema bloqueado.
- Q3: Primario y copia definidos pero no hay nodos en espera.
- Q4: Primario y copia definidos con al menos un nodo en espera.

Transiciones:
t1: Llega latido con número de vista 0.
Se añade como primario y se actualiza el número de vista en 1 en la vista tentativa.
Se añade a la lista de latidos.

t2: Llega latido con número de vista distinto de 0.
Se reinicia el latido para el nodo primario.
Se actualiza la vista válida con la vista tentativa.

t3: Primario ha caído.
Se actualiza el número de vista en 1 en la vista tentativa.
Se elimina como primario y de la lista de latidos.

t4: Primario ha caído sin confirmar la vista.
Se ha perdido la consistencia, primario queda como indefinido.

t5: Llega latido con número de vista 0.

Si no es el primario, se añade como copia, se actualiza el número de vista en 1 en la vista tentativa y se añade a la lista de latidos.
Si es el primario, se actualiza el número de vista en 1 en la vista tentativa y se actualiza la lista de latidos.

t6: Llega un latido con número de vista distinto de 0.
Se reinicia el latido para el nodo que lo ha enviado.
Si es el primario, se actualiza la vista válida con la vista tentativa.

t7: Copia o primario han caído.
Si es la copia, se actualiza el número de vista en 1 en la vista tentativa, se elimina como copia y de la lista de latidos.
Si es el primario, se promociona la copia, se actualiza el número de vista en 1 en la vista tentativa, se elimina como primario y de la lista de latidos.

t8: Primario ha caído sin confirmar la vista.
Se ha perdido la consistencia, primario queda como indefinido.

t9: Primario y copia han caído.
Se ha perdido la consistencia, primario y copia quedan como indefinidos.

t10: Llega latido con número de vista 0.
Si es el primario, se promociona la copia, se actualiza el número de vista en 1 en la vista tentativa y se actualiza en la lista de latidos.
Si es la copia, se actualiza el número de vista en 1 en la vista tentativa y se actualiza en la lista de latidos.
Si es un nodo nuevo, se añade como nodo en espera, se actualiza el número de vista en 1 en la vista tentativa y se añade a la lista de latidos.

t11: Llega un latido con número de vista distinto de 0.
Se reinicia el latido para el nodo que lo ha enviado.
Si es el primario, se actualiza la vista válida con la vista tentativa.

t12: Primario, copia o el único nodo en espera que había han caído (solo un nodo en espera)
Si es el primario, se promociona la copia, se actualiza el número de vista en 1 en la vista tentativa, se elimina como primario y de la lista de latidos.
Si es la copia, se promociona nodo en espera a copia, se actualiza el número de vista en 1 en la vista tentativa, se elimina como copia y de la lista de latidos.
Si es el nodo en espera, se elimina de la lista de latidos.

t13: Primario, copia o nodo en espera que había han caído (varios nodos en espera)
Si es el primario, se promociona la copia, se actualiza el número de vista en 1 en la vista tentativa, se elimina como primario y de la lista de latidos.
Si es la copia, se promociona nodo en espera a copia, se actualiza el número de vista en 1 en la vista tentativa, se elimina como copia y de la lista de latidos.
Si es el nodo en espera, se elimina de la lista de latidos.

t14: Llega latido con número de vista 0.
Si es el primario, se promociona la copia, se añade como nodo en espera, se actualiza el número de vista en 1 en la vista tentativa y se actualiza en la lista de latidos como nodo en espera.
Si es la copia, se promociona el primer nodo en espera, se añade como nodo en espera, se actualiza el número de vista en 1 en la vista tentativa y se actualiza en la lista de nodos como nodo en espera.
Si es un nodo nuevo, se añade como nodo en espera, se actualiza el número de vista en 1 en la vista tentativa y se añade a la lista de latidos.

Bibliografía

[1] Material de la asignatura de Sistemas Distribuidos del grado de Ingeniería Informática de UNIZAR.

[2] Documentación del lenguaje Elixir https://elixir-lang.org/docs.html